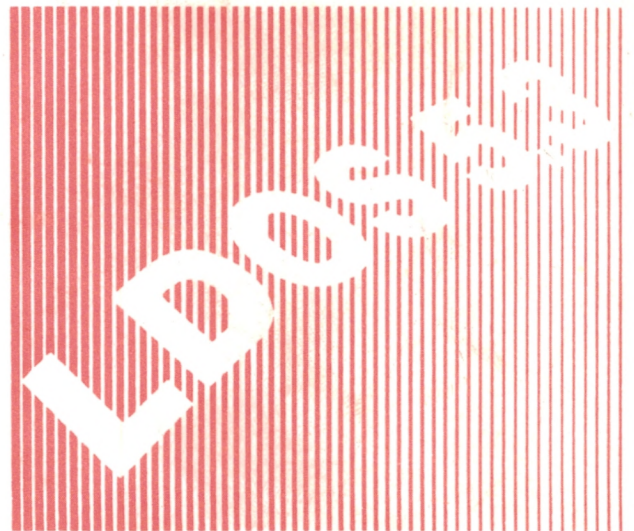
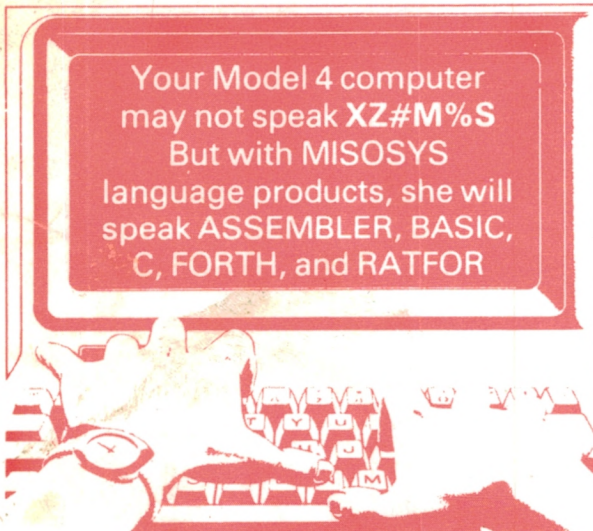


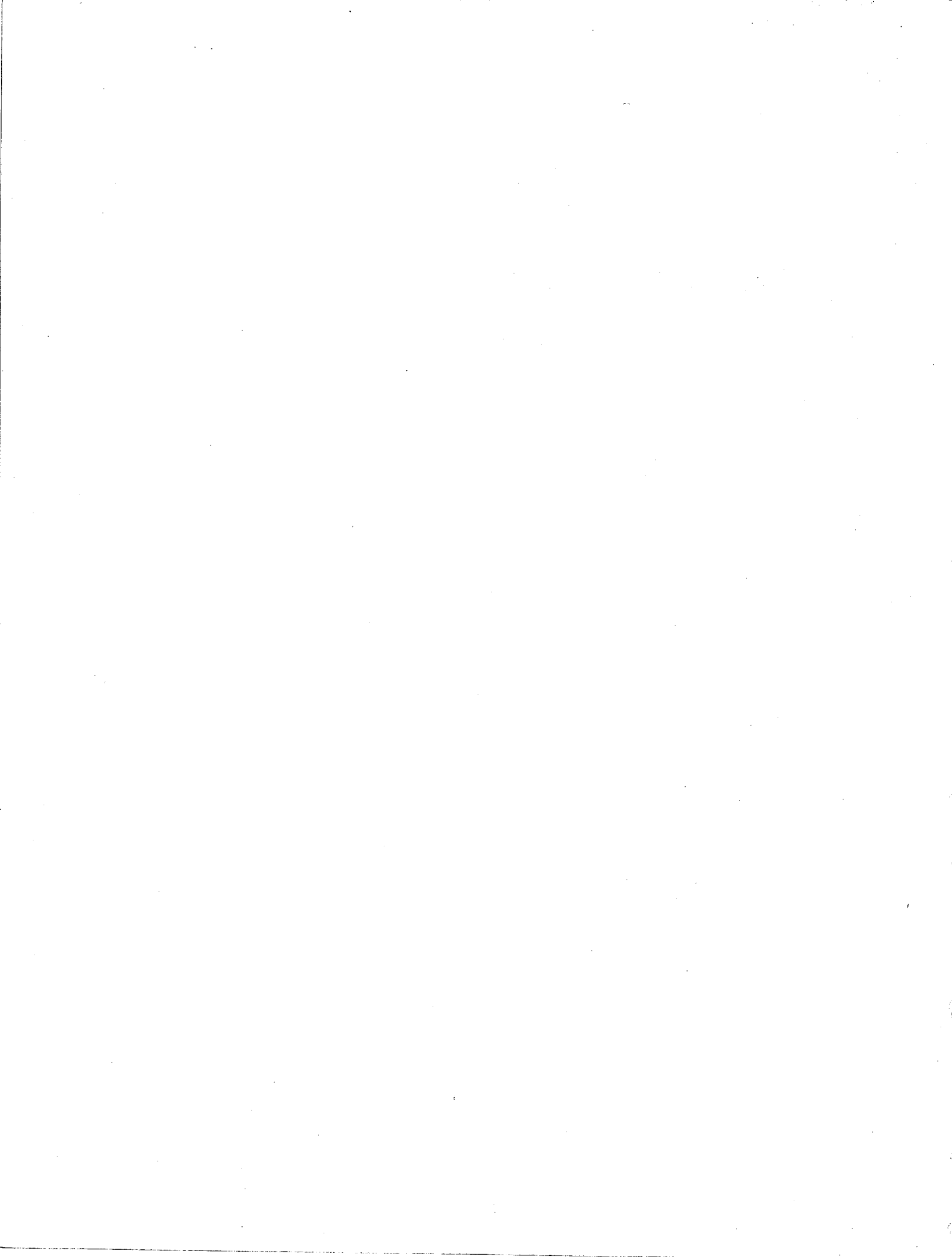
THE MISOSYS QUARTERLY



In this issue:

- LS-DOS 6.3 Column from LSI
- Extended Memory access for LS-DOS
- Upgrading Little Brother to LS-DOS 6.3
 - Converting Mail File Data to LB!
 - A "CAT" for LDOS; "KILL" for LS-DOS





THE MISOSYS QUARTERLY

Volume I, Issue iii

Winter 1987

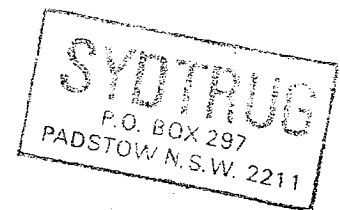
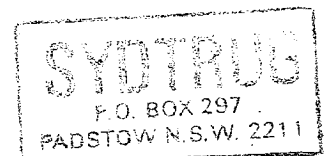


Table of Contents

| | |
|--|-----|
| The Blurb | 2 |
| Announcing New Products | |
| LDOS™ 5.3 Model 4 Hardware Interface Kit | 6 |
| RSHARD™ Hard Disk Driver Package | 7 |
| The MARK III Collection | 8 |
| The MARK IV Collection | 8 |
| MISOSYS, Inc., PRICE LIST | 9 |
| Letters to the Editor | 11 |
| Our Compuserve Forum | 31 |
| LDOS™ Information | 35 |
| LS-DOS™ Information | 52 |
| The LSI Column | 57 |
| LSI Patches to LS-DOS 6.3 | 65 |
| Applications for the User | |
| Making KILL6 and CAT5 by Jim Gyaffney | 67 |
| Upgrading Little Brother to LS-DOS 6.3 | 68 |
| Converting MailFile to Little Brother. | 69 |
| The Programmer's Corner | |
| Extended Memroy Management for 6.3 | 75 |
| Getting into MS-DOS Assembly | 84 |
| The Tower of Babel | 39 |
| MISOSYS Products' Tidbits | |
| EnhComp - BASIC compiler | 93 |
| Little Brother | 96 |
| PRO-WAM - Window and Application Manager | 100 |
| The Hardware Corner | 104 |
| The PATCH Corner | 112 |



The Blurb

This issue is correctly titled the "Winter 1987" THE MISOSYS QUARTERLY. Of course the first day of Spring was yesterday (as I write this column). That puts TMQ I.iii about two months behind schedule. Thus, I'll be the first to apologize over its lateness. We seem to have a lot of those apologies. Maybe it's because we try to do too much at one time. Yes, that's it.

A lot has been crammed into the past few months here at MISOSYS. We released LDOS 5.3 in January. We released The LDOS 5.3 Model 4 Hardware Interface Kit also in January. February brought the release of RSHARD, UNREL-T80, UNREL-CPM, and ED/ASM-86. We started shipping DED86 in March. We put together this issue of TMQ in March.

That leaves RATFOR-M4, RATFOR-86, and DSM-86 left to get released; these products have all been announced. I hope I am able to report in the next issue that they got released in April, no fooling. Also on the drawing board for implementation is a new version of PRO-WAM. That's the next project we work on. Later on in the year, we will be re-examining Little Brother.

This column is going to be a lot smaller than that appearing in previous issues. First, we're not going to bore you with gloom; we are going to let you know what's on the horizon here at MISOSYS. Second, we have established a separate "Letters to the Editor" section; they used to appear in The Blurb. We also ran out of room!

Speaking of room, we had to forego a few things which were supposed to be in this issue. For instance, Rich Deglin had forwarded us an "rm" utility written in C. Part of that was a new function for MC, the stat() function. His piece will be deferred. I also took one of the filters which was part of FILTER DISK 1 and rewrote it for Model 4 operation. That and an accompanying article discussing the writing of Model 4 filters is also deferred. We had also planned an article on customizing TED. Paul Wade sent us an application written using The BASIC Answer. That has also been placed in the input box for the next issue. Look for

these and more. We hope to put TMQ I.iv back on track to be out during the Spring.

TMQ subscriptions stand now somewhere between 650 and 700. We have now shipped over 1200 PRO-WAM packages, 150 Mister ED's (not too potent there), 200 MC's, and 750 LDOS 5.3 Upgrade kits. Here's an interesting observation. Last week I asked Brenda to give me a stroke count of 5.3 registrations broken down by computer type (i.e. not by packages shipped but by packages registered). Turned out to be 155 Model III's and 291 Model 4's. That's pretty close to 2:1 in favor of the Model 4. About 75 Interface Kits have been shipped. That figure has kept very close to one tenth of 5.3 shipments - consistently.

Now comes the bombshell. We have in the past always been so busy developing and implementing products that we never had the time to get a good handle on our product sales and marketing. We're still busy, but we have at least started using our computer horsepower to provide us some raw data on product sales. Here's how that goes. The invoicing program generates two output files. One is a record of the invoice; the other is a record for each item shipped per that invoice. One of the downstream programs mentioned in last issue's TMQ provides a monthly printout of all units; the quantity shipped and total dollar value is included. One report covers a rolling three month period with quarterly totals.

In the middle of this month, I had 5-1/2 months of data. These reports have been eyeopeners. It has been quite interesting to see the effect of each individual product to the "bottom line" over a period of almost a half year. The time came to clean house. We realized how costly it was to continue to make feeble attempts at selling products you, our customers, were saying they didn't want to buy. Thus, the big ax fell. We have now created two limited editions of products; a Mark III Collection for the Model I/III folks and a Mark IV Collection for the Model 4/4p/4d folks. Each collection has programs and files from twenty of our previous products. I hope you caught the word "previous". There will be no product

information on either collection other than the list of program files contained therein. If you don't know what they are, don't buy it.

Effective April 1, 1987, the products which have been bundled together into the respective collections will no longer be offered for individual sale. We also went further and discontinued some other products: ADE, CON80Z, DW2PS, GRASP, GRASPF1, Inventory Master, Binders, LDOS QRC, MailFile, Individual LDOS QUARTERLIES, SAID as a separate product, UltraTrek, VRHARD, The Gobbling Box, and individual copies of THE SOURCE. The products that are now being sold by MISOSYS are listed in our new price sheet which appears in the NEW PRODUCTS section. Note also that there are some new prices going into effect starting July 1. In some cases this reflects simply an increase in shipping charges. However, the bulk of the changes are price increases. You'll also note that as of July 1, we are discontinuing HartFORTH and PRO-HartFORTH; that's for other reasons.

We are out of our 86-2 catalog. A new 87-1 catalog is in the mill. The new catalog will be reflecting our slimmer product line. We also expect to separate out the MS-DOS product line into its own catalog. Since we need catalogs fast, this will probably be done before we start on the PRO-WAM revision. So if you are waiting for a catalog from us, please wait a little longer.

I received a few letters from folks complaining about the Compuserve chitchat which has appeared in previous TMQ's. I also received a few letters thanking me for that chitchat. It even prompted a few folks to sign up for Compuserve. You can't please everyone. I received one letter telling me to keep news about my family out of TMQ. Another said that such information makes it feel that there is a real live person at MISOSYS. Yes, we are real here; if you're talking to a female voice, say hi to Brenda (she likes to talk about the family and is interested in yours, too); if you talk to a gruffy male voice, you probably caught me in a bad day (like after I asked the last three callers to read the README/TEXT file which is

documented in the FIRST paragraph of the LDOS 5.3 documentation) - most of the time I'm pretty pleasant.

I will continue to put in to each issue of TMQ some of the things that I feel are necessary. When I proposed TMQ, it was targeted as a 64-page publication. I gave each of the first two issues an extra 36 pages. This issue may even top that! I have always been of the opinion that you can't please all of the people all of the time; but only some of the people some of the time. Who knows, we may even decide one day to accept ads for insertion in TMQ. After all, I understand that Lotus' magazine had an ad for VP Planner!

Here's a statement we need to make. There is only one company authorized for worldwide sale of the LDOS 5.3 Upgrade Kit. That company is MISOSYS, Inc. A legal copy of the LDOS 5.3 Upgrade Kit is packaged in the following manner. The master LDOS 5.3 disk is a blue disk with a MISOSYS label imprinted in blue. The documentation update is printed in blue ink on 60 lb offset blue stock. A legal copy includes a blue WARRANTY REGISTRATION CARD that's approximately 5"x8" and is addressed to MISOSYS, Inc. A yellow flyer advertising TMQ, the RSHARD package, and the LDOS 5.3 Model 4 Hardware Interface Kit accompanies each LDOS 5.3 Upgrade Kit. Each LDOS 5.3 master disk has a unique and distinct serial number encoded in the disk contents. Anyone other than MISOSYS who is knowingly giving away or selling a copy of LDOS 5.3 is in criminal violation of the law; they are also placing at risk the ability of MISOSYS to continue to support the TRS-80 marketplace.

Disk NOTES 7

We still seem to get some queries as to just what is DISK NOTES. Here's the answer. Each issue of THE MISOSYS QUARTERLY contains program listings, patch listings, and other references to files we have placed on DISK NOTES. Some people enjoy typing in long listings. Sometimes you may have need for only a short patch. If you want to obtain all of the patches and all of the listings, you may conveniently purchase a copy of DISK NOTES.

There is a cost involved. DISK NOTES is priced at \$10 PLUS S&H. The S&H charges are \$2 for US, \$3 Canada and Mexico, \$6 elsewhere. If you purchase the corresponding DISK NOTES with the coupon which accompanies this TMQ issue, you can save \$2.50; the cost is only \$7.50 + S&H. Here's what's on Disk NOTES 7:

BANKTEST/CCC - tests exmem
 EXMEM/ASM - source to exmem/cmd
 EXMEM/CMD - installs SVC108, @EXMEM
 FIXES7/TXT - THE PATCH CORNER fixes
 FLL/JCL - from TMQ I.ii
 HELLO/TXT - source to "Hello world"
 INIT54/JCL - installs 4-head 5-meg HD
 LSIFIX/TXT - fixes to LS-DOS 6.3
 MAILFILE/DEF - part of LB template
 MAILFILE/LB - part of LB template
 MAILFILE/PRO - part of LB template
 MAILFILE/PR1 - part of LB template
 MAILFILE/VDO - part of LB template
 MCBANK/ASM - exmem interface for MC
 MCBANK/REL - above assembled
 MCRELS/REL - revised RELs for MC
 MEMDUMP/ASM - source to memdump/cmd
 MEMDUMP/CMD - dumps memdisk to disk
 MF2LB/BAS - converts MailFile to LB
 MFINDEX/JOB - part of LB template
 STRIP/CCC - strips #asm blocks

QUARTERLY Coupon

Don't forget the coupon which accompanies this issue. It qualifies you for a \$2.50 savings on DISK NOTES 7. For those of you who have not already obtained your set of THE SOURCE TO TRSODS 6.2, you will probably never find a lower price - half off of our current \$99.95 price. Remember, you must return the coupon to qualify.

Family Update

Here's where the folks who are not interested in my family can stop reading and proceed to the NEW PRODUCTS section.

Our cat, Spitter, seems to have gotten a lot friendlier lately. Some of our friends knew us for over a year before they knew we had a cat. That's because Spitter is the kind of cat to run off and hide at the first unusual sound. Probably old age creeping in. After all, she's eleven now.

Over the winter, I had ordered a big climbing and swing set from Sears (a holiday present for the kids). Now I am not one to ordinarily try to assemble a swing set when there's snow on the ground, but I did want to make sure that it was not missing any parts. During a few weekends when the weather was comfortably in the high 30's, the swing set went up. As luck would have it, one little bag of parts needed to put up the sliding board (the one piece I really wanted to get up) was missing. A quick call to the company down in Louisiana got me the parts bag in about a week. By then the two big snows had hit the Washington DC area (Sterling is a suburb of DC) and dumped about two feet of snow on us. Needless to say, the slide waited.

Now that Spring has sprung, the folks around the neighborhood are making their presence known out doors. The swing set is getting plenty of use. Late afternoons and weekends find a few of the neighborhood children over here. We are the first house in a cul-de-sac consisting of nine houses. There are about seventeen children ranging in age from -5 months (there's two on the way) to 17. Stacey turns four in June. Right now there's only three younger than her (including here sister) and by the Fall there will be two more. Stefanie turns three in October. I think this will be a pretty active summer for outdoors activities. After the intensive development efforts of the past nine months, I think I will be making more of a presence outside than last year. In fact, we are considering taking a week off sometime in August to take the kids down to the shore (that's a Philly term; others say "beach"). We haven't yet decided exactly where, but the contenders are probably Ocean City, MD and Virginia Beach, VA.

Since September, Stacey has been going to a "Children's Early Education" class three days a week for 2-1/2 hours a day. That's to get here into an environment with other three year olds. It's not "formal" education; just coloring and painting and playing, and learning songs, etc. Both Brenda and I think that the environment is necessary and has been helpful to her growth. When I was growing up in Philly,

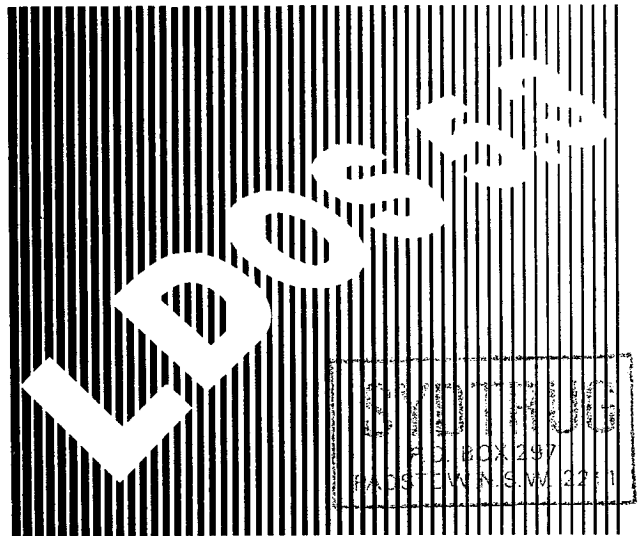
my family lived in a row house. That created a high concentration of houses (i.e. people density). It seemed like there were 50 zillion kids always around. Well, the CEE school does allow Brenda to spend "quality" time with Stefanie during those mornings when Stacey is at school.

Stefanie goes to "Fun for two year olds" which is two days a week for an hour each day. You may recollect from our first TMQ that Stacey also went to FFTYO last year. There's a waiting list for that class which is organized and run by the county. Stef is enrolled in the CEE school for next Fall (believe me, you have to start that enrollment business EARLY!). That means that both Stacey and Stefanie will be off to school three days a week in the morning.

I'll also be taking off some time to attend Stacey's graduation. I did go to her Christmas pageant but got there too late; the show which the kids put on was over. Boy did I feel bad about that; especially when I heard that Stacey told everybody, "My daddy didn't come." So if our phone doesn't get answered some morning coming up in early June, you know where I'll be.

Now for a little food update from last issue. Stefanie has been eating chicken and some fish. She still doesn't touch meat - except for a McDonald's cheeseburger (is she telling me something there? It is meat, isn't it?). Now my family is not too great on vegetables, except for me - I'll eat just about any vegetable. In fact, I'll eat just about any food that I can look at (that leaves out snails and mussels). So meal time here has been pretty good. When we have meat, we substitute apple sauce to serve to Stef. With her growth and general physical well being, I don't think she's being shortchanged. One other thing about mealtime here, that's one of the times that the TV goes OFF.

So, I hope that I haven't bored you. That's it for now. Keep those orders coming in; sooner or later Stefanie's bound to start eating meat!



The LDOS 5.3 upgrade kit is now available to take your Model III or 4 (in 3 mode) to the year 2000. LDOS 5.3 provides complete media compatibility with LS-DOS 6.3, the newest Model 4 DOS released by Logical Systems, Inc. With LDOS 5.3, you can add 12 years to the life of your software. Just look at these improvements over version 5.1.4!

DOS Enhancements:

- Date support through December 31, 1999; time stamping for files.
- Enhancements to LDOS now free up 14 additional file slots for data disks.
- On-line HELP facility for DOS and BASIC - 117 screens of help.

LIBRARY Enhancements:

- New FORMS, lets you change printer filter parameters.
- New SETCOM, lets you change RS-232 parameters.
- Improvements to LIST add paged displays, full-screen hex mode, and flexible tab expansion.
- MEMORY displays directory of terminate and stay resident modules.
- SYSTEM lets you direct the SYSGEN to any drive; adds a flexible drive swap subcommand; SMOOTH for faster disk throughput.
- DiRectory display enhanced with time stamps, file EOF, and more.
- We've also improved: AUTO, COPY, CREATE, DEBUG, DEVICE, DO, FREE, KILL, and ROUTE; and added CLS and TOF commands.

UTILITY Enhancements:

- We've added TED, a full screen text editor for ASCII files.
- LCOMM now gives you access to LDOS library commands while in terminal mode.
- PATCH supports D&F patch lines with REMOVE capabilities.
- DATECONV has been added to convert older disks to the new date convention.

BASIC Enhancements:

- Improvement to line editing with the addition of line COPY and MOVE.
- Very flexible INPUT@ added for screen fielded input.
- We've added a CMD"V" to dump a list of active variables with values - including arrays.

For \$24.95 (+S&H), the LDOS 5.3 upgrade kit includes a DOS disk and documentation covering the enhancements. Specify Model 3/4 or MAX-80.

P.S. - Don't return your old disk!



MISOSYS, Inc.

PO Box 239
Sterling, VA 22170-0239
703-450-4181 MC, VISA, CHOICE
Orders Only! 800-MISOSYS 1P-5P EST Monday-Friday

VA residents add sales tax. S&H: US \$2, Canada \$3, Foreign \$6.

Announcing New Products

LDOS 5.3 Model 4 Hardware Interface Kit

The software in this package enables you to make use of certain Model 4 hardware features while using it in the Model III mode running LDOS 5.3. It includes KI4/DVR to use the extra Model 4 keys (CTRL, CAPS, F1-F3), SET2RAM which switches to Model III RAM mode and adds @BANK and @EXMEM handlers (@EXMEM does getbyte, putbyte, getpage, putpage), a Model III mode version of BANKER (discussed in the last TMQ), and MemDISK (a driver to utilize one or two memory banks in a 128K Model 4 as a RAM disk, similar to the DOS 6 MemDISK/DCT utility).

Even though your Model 4 is capable of operating its video screen in an 80x24 mode, we have chosen not to support this operation in Model III mode due to the inevitable conflict with most commercial software packages in existence for the Model III operation as well as LDOS 5.3.

Normal Model III-type operation on your Model 4 uses a hardware mode where the machine duplicates a standard Model III - the first 14K of memory addressing is Read Only Memory (ROM). When you install the memory management handler provided with this package, it switches your machine to a mode where the first 14K is accessed as Read/Write memory. The major difference is that you now can alter the contents of this "ROM" image portion with pokes although it is not recommended that you do so.

The KI4/DVR keyboard driver completely replaces the KI/DVR keyboard driver supplied with your LDOS 5.3. It is used to enable access of all keys on your Model 4 keyboard. The three function keys, <F1>, <F2>, and <F3>, each generate a distinct value in both the unshifted and shifted depressions. The key values generated are identical to the values generated under DOS 6.x. Note that these values remain constant when the Extended Cursor Mode (ECM) is toggled ON even though the <ARROW> keys will duplicate some of the values. This ensures that programs which expect a particular function key value will continue to "see" that value when you (or a program) toggle the keyboard to ECM.

The key marked <CTRL> is now used to specify "control" instead of the two-key combination, <LSHIFT-DOWNARROW>. In fact, the latter combination will generate a key code value of 1AH. For the programmer, the code is the same value as that generated by <CTRL-Z> with the exception that it is not flagged as a control code. The flagging is only significant to those programs which make use of the Z80 flag state following a keyboard driver call.

The <CAPS> key is used to toggle the caps-lock mode ON and OFF. This replaces the <SHIFT-Ø> two-key combination as used by the KI/DVR. When caps-lock is ON, the twenty-six keys <a>-<z> generate capital letters <A>-<Z>.

The <SHIFT-Ø> two-key combination now toggles the value generated by the <UPARROW> key between 5BH and ØBH. The "ØBH" value is normal for a Model 4 operation while the "5BH" value is normal for Model III operation. The <CLEAR-SHIFT-Ø> three-key combination now generates a value of ØAØH (16Ød) which is the same as <CLEAR-SPACE>.

The SET2RAM/CMD memory management utility switches your Model 4 computer, which is operating in the Model III ROM mode, to a low-memory RAM mode. It populates that RAM with an image of what was in ROM. Furthermore, it inhibits the use of Model III cassette I/O while it re-uses the memory space previously used by the cassette routines. The space is now used for a memory management handler (two service calls, @BANK and @EXMEM, used for bank switching) which enables you to easily program access to the extra 64K of a 128K Model 4. The SET2RAM utility is usable on a Model 4, 4P, or 4D computer.

The MEMDISK/DCT RAM disk driver can establish either a 32K or a 64K disk drive emulated in memory. The RAM disk driver needs to use one of the eight logical drive slots available under LDOS. This driver uses the same installation protocol as that used by LS-DOS 6.3; thus, providing a common set of commands across both Model III and Model 4 modes.

The BANKER/CMD memory bank utility will allow you to reserve or release a single

memory bank or a range of consecutively numbered memory banks. BANKER will also display a map providing the status of all memory banks.

The LDOS 5.3 Hardware Interface Kit (order #M-12-110) is currently priced at \$29.95 + \$2 S&H (\$3 Canada; \$6 foreign).

RSHARD - Hard Disk Driver Package

RSHARD is a utility package that provides hard disk drivers which support certain hard drive assemblies manufactured by the Tandy Corporation for use with both LDOS 5.3 and LS-DOS 6.3. Specifically, RSHARD supports up to two drive units - each of the same physical step rate and interfaced via the Western Digital WD1000 or WD1010 hard disk controller. The physical size characteristics such as number of cylinders and heads may differ.

The driver supports a single drive of up to 1024 cylinders with up to eight (8) heads. Because drive configurations may exceed the maximum limits of a single logical drive afforded by the DOS, the driver supports drive partitioning by both head and cylinder.

Your hard disk drive may be shared between both LDOS and LS-DOS as both operating system's use a compatible format and directory structure. However, since other vendors driver and formatter packages may construct different allocation schemes, you MUST use the RSHARD5/DCT driver with LDOS 5.3 and the RSHARD6/DCT driver with LS-DOS 6.3 when you are sharing identical surfaces.

Before you can use the hard drive, it must be formatted and written with system information (i.e. the BOOT/SYS and DIR/SYS files - the DIR/SYS file contains the directory for the logical drive). Formatting is performed via the RSFORM formatter provided with the package.

The included HDCHECK utility is designed to check whether each sector of the hard drive is readable (and optionally writable). Where an error is detected, the granule containing that sector can be optionally locked out for further use.

HDCHECK can be run anytime as a non-destructive WRITE test is utilized.

The included ARCHIVE utility is designed to provide a controlled archival of one or more files from a large capacity storage device to a smaller floppy disk drive. You may use the BACKUP utility provided with your DOS to copy hard disk files onto floppy diskettes. However, when a single hard drive file exceeds the total size of a floppy diskette, the ARCHIVE utility can be used to segment the file and transfer each individual segment to a corresponding floppy diskette. You may want to use the SIZE parameter with a value of zero to force ARCHIVE to select ALL visible files.

All segments of all files that are archived include a header record generated by the ARCHIVE utility. This header record is used essentially by the RESTORE utility to enable it to regenerate an exact image of the original file from its segments. The header record is also used to identify each diskette included in an archive set. Bear in mind that the archived files must be RESTORED prior to their use.

The RESTORE utility is used to reconstruct files that have been archived via the ARCHIVE utility. RESTORE combines all of the segments of an archived file into a single file. The restoral process does not require the restored file to be existing on the drive that is restored. All attributes of the original file will be restored. RESTORE can be used to reconstruct a single original file from the archived file set or it can be used to reconstruct all archived files contained in the archived file set.

There is no requirement that you must restore an archived file to the drive that the source was on. Archived file sets may be restored to other drives providing that the size of the drive is sufficient to handle the file or files. This provides a method to transfer large files from one hard disk drive to another.

Order RSHARD (catalog #M-12-013) for either LDOS 5.3 or LS-DOS 6.3 (or both). The current price is \$29.95 + \$2 S&H (\$3 Canada; \$6 foreign).

The MARK III Collection

Here is a limited edition two-disk collection of software with documentation which bundles together twenty products previously sold individually by MISOSYS, Inc., and Logical Systems Inc., for Model I and Model III operating under LDOS 5.1.4. This collection, attractively packaged in a three-ring binder, brings you a software value in excess of five hundred dollars for less than one fifth the cost. Just \$99.95 + \$10 S&H brings you the following programs and files:

| | | |
|---------------|---------------|---------------|
| ALLOC/CMD, | BINCONV/CMD, | BINPLAY/CMD, |
| BINPRINT/CMD, | BSORT51/CMD, | BSORT53/JCL, |
| CALC/ASM, | CALC/FLT, | CMDFILE/CMD, |
| COMM1/ASM, | COMM1/FLT, | CONVCPM/CMD, |
| CRLF/FLT, | CTLG/FLT, | CVTEXT/CMD, |
| DD/CMD, | DED/CMD, | DESCRIBE/CMD, |
| DICTATE/ASM, | DICTATE/FLT, | DMP2005A/FIX, |
| DOAUTO/CMD, | DOCONFIG/CMD, | DOEDIT/FLT, |
| DOSAVE/FLT, | DOSPEED/ASM, | DOSPEED/FLT, |
| DVORAK/XLT, | EPBCDIC/XLT, | EPBINCAT/CMD, |
| EPBINCAT/FIX, | FEDII/CMD, | FM/CMD, |
| FTS5/CMD, | FTS5/HLP, | HELPGEN/CMD, |
| IFC/CMD, | IFCLIST/CMD, | IOMON/CMD, |
| KISTORE/FLT, | KSMPLUS/ASM, | KSMPLUS/FLT, |
| LCOUNT/ASM, | LCOUNT/FLT, | LINEFEED/ASM, |
| LINEFEED/FLT, | LISTBAS/ASM, | LISTBAS/FLT, |
| LOWER/ASM, | LOWER/FLT, | MARGIN/ASM, |
| MARGIN/FLT, | MAXLATE/ASM, | MAXLATE/FLT, |
| MEMDIR/CMD, | MEMDISK/DCT, | MONITOR/ASM, |
| MONITOR/FLT, | NAME/CMD, | NODAM/CMD, |
| PAGEPAWS/ASM, | PAGEPAWS/FLT, | PARMDIR/CMD, |
| PDS/CMD, | PRTOGGLE/CMD, | PTRACE/CMD, |
| RD40/CMD, | REMOVE/CMD, | RSBINCAT/CMD, |
| SLASH0/ASM, | SLASH0/FLT, | SLOSTEP/ASM, |
| SLOSTEP/DCT, | STRACE/CMD, | STRIP7/ASM, |
| STRIP7/FLT, | STRIPCNT/ASM, | STRIPCNT/FLT, |
| TITLE/ASM, | TITLE/FLT, | TRAP/ASM, |
| TRAP/FLT, | UNKILL/CMD, | UPPER/ASM, |
| UPPER/FLT, | VIDSAV/CMD, | WC/CMD, |
| XLATE/ASM, | XLATE/FLT, | XONXOFF/FLT, |
| ZCAT/CMD, | ZGRAPH/CMD, | ZSHELL/CMD |

The complete Volume II six-issue set of the LDOS QUARTERLIES will also be included. No descriptive information on individual items of this collection is currently available; such information may be located in earlier catalogs of MISOSYS, Inc., and Logical Systems Inc. Approximately 100 units of the Mark III collection are available; when they are gone, they're gone!

This collection is sold as-is; with no warranties implied. All programs will work with LDOS 5.1.4; no utilization of LDOS 5.3 enhancements are guaranteed nor will any of the programs be modified in any way by MISOSYS, Inc.

Order The Mark III collection [#M-99-003] for \$99.95 prepaid only (no COD's) plus \$10 S&H via UPS ground or other surface mail. Final sale; no returns accepted.

The MARK IV Collection

Here is a limited edition two-disk collection of software with documentation which bundles together twenty products previously sold individually by MISOSYS, Inc., and Logical Systems Inc., for the TRS-80 Model 4 operating under TRSDOS 6.2. This collection, attractively packaged in a three-ring binder, brings you a software value in excess of five hundred dollars for less than one fifth the cost. Just \$99.95 + \$10 S&H brings you the following programs and files:

| | | |
|---------------|---------------|---------------|
| ALLOC/CMD, | ALTDISK/CMD, | ALTLD/CMD, |
| ALTRES/CMD, | BE/LMF, | BINCONV/CMD, |
| BINPLAY/CMD, | BINPRINT/CMD, | BSORT/CMD, |
| CALC/CMD, | CRLF/FLT, | CTLG/FLT, |
| CVT324/CMD, | CVTEXT/CMD, | DD/CMD, |
| DED/CMD, | DESCRIBE/CMD, | DMP2006A/FIX, |
| DOCONFIG/CMD, | DOEDIT/FLT, | DOSAVE/FLT, |
| EPBINCAT/CMD, | FKEY/CMD, | FM/CMD, |
| FTS/CMD, | FTS/HLP, | HANDY/CMD, |
| HELPGEN/CMD, | IFC/CMD, | IFCLIST/CMD, |
| INSTALLB/CMD, | IOMON/CMD, | KISTORE/FLT, |
| LSCOMP/CMD, | LSFEDII/CMD, | LSQFB/CMD, |
| MAPPER/CMD, | MEMDIR/CMD, | MINIDOS/FLT, |
| MOD324/CMD, | NAME/CMD, | OD/CMD, |
| PARMDIR/CMD, | PDS/CMD, | PROCESS/CMD, |
| PROCURE/CMD, | PRTOGGLE/CMD, | PTRACE/CMD, |
| RD40/CMD, | RSBINCAT/CMD, | STRACE/CMD, |
| SWAP/CMD, | UNREMOVE/CMD, | WC/CMD, |
| XONXOFF/FLT, | ZCAT/CMD, | ZGRAPH/CMD, |
| ZSHELL/CMD | | |

The complete Volume II six-issue set of the LDOS QUARTERLIES will also be included. No descriptive information on individual items of this collection is currently available; such information may be located in earlier catalogs of MISOSYS, Inc., and Logical Systems Inc. Approximately 100 units of the Mark IV

collection are available; when they are gone, they're gone!

any of the programs be modified in any way by MISOSYS, Inc.

This collection is sold as-is; with no warranties implied. All programs will work with TRSDOS 6.2 ; no utilization of LS-DOS 6.3 enhancements are guaranteed nor will

Order The Mark IV collection [#M-99-004] for \$99.95 prepaid only (no COD's) plus \$10 S&H via UPS ground or other surface mail. Final sale; no returns accepted.

MISOSYS, Inc., PRICE LIST effective April 1, 1987 and July 1, 1987

** indicates a price change effective 7/1/87

| MS-DOS Products | Catalog # | 04/01/87 | 07/01/87 | |
|-------------------------|-----------|------------|------------|------------|
| DED86 | #M-86-020 | \$59.95 F | \$59.95 F | (see note) |
| DSM-86 | #M-86-204 | \$59.95 F | \$59.95 F | |
| ED/ASM-86 | #M-86-030 | \$124.95 F | \$124.95 F | (see note) |
| FED86 | #L-86-010 | \$29.95 | \$29.95 | |
| FM86 | #L-86-050 | \$29.95 | \$29.95 | |
| Little Brother-MS | #L-86-510 | \$74.95 F | \$74.95 F | |
| LBMU-MS | #L-86-515 | \$29.95 | \$29.95 | |
| RATFOR-86 | #M-86-073 | \$99.95 F | \$99.95 F | |

Note: DED86 is \$49.95 until 04/30/87

ED/ASM-86 is \$89.95 until 04/30/87

| CP/M products | Catalog # | 05/01/87 | 07/01/87 | |
|-----------------|-----------|----------|----------|----|
| UNREL-CPM | #M-32-054 | \$39.95 | \$49.95 | ** |
| LCOPY | #M-32-063 | \$39.95 | \$39.95 | |

| TRS-80 Operating Systems | Catalog # | 05/01/87 | 07/01/87 | |
|---------------------------|-----------|-----------|-----------|----|
| LDOS-I 5.1.4 system | #L-10-010 | \$49.95 F | \$59.95 F | ** |
| LDOS-I 5.1.4 disk | #L-10-110 | \$14.95 | \$24.95 | ** |
| LDOS-III 5.1.4 system .. | #L-10-030 | \$49.95 F | \$59.95 F | ** |
| LDOS-III 5.1.4 disk | #L-10-130 | \$14.95 | \$24.95 | ** |
| LDOS 5.3 Mod 4 Int. Kit . | #M-12-110 | \$29.95 | \$29.95 | |
| LDOS 5.3 Mod3 Upgrade Kit | #M-10-033 | \$24.95 | \$34.95 | ** |
| LDOS 5.3 Max Upgrade Kit | #M-10-833 | \$24.95 | \$34.95 | ** |
| LS-DOS 6.2 Model II/12 .. | #M-11-002 | \$49.95 B | \$49.95 B | |
| SOLE | #M-10-021 | \$14.95 | \$14.95 | |
| FIX Disk II | #L-70-400 | \$14.95 | \$14.95 | |
| RSHARD - R/S HD driver .. | #M-12-013 | \$29.95 | \$29.95 | |

| Language Facilities | Catalog # | 05/01/87 | 07/01/87 | |
|---------------------|-----------|------------|--------------|----|
| CON80Z | #M-30-033 | \$24.95 | \$29.95 | ** |
| DSMBLR | #M-30-053 | \$29.95 | \$34.95 | ** |
| EDAS | #M-20-082 | \$74.95 B | \$74.95 D | ** |
| EnhComp | #M-20-072 | \$124.95 B | \$124.95 D | ** |
| HartFORTH | #M-20-071 | \$74.95 A | discontinued | |
| MC | #M-20-064 | \$124.95 D | \$124.95 D | |
| MRAS | #M-20-083 | \$74.95 B | \$89.95 D | ** |
| TBA | #L-21-010 | \$24.95 B | \$39.95 D | ** |

| | | | |
|---------------------|-----------|------------|---------------|
| LS-TBA | #L-21-011 | \$24.95 B | \$39.95 D ** |
| PRO-CON80Z | #M-31-033 | \$24.95 | \$29.95 ** |
| PRO-CREATE | #M-21-082 | \$74.95 B | \$74.95 D ** |
| PRO-DUCE | #M-31-053 | \$29.95 | \$34.95 ** |
| PRO-EnhComp | #M-21-072 | \$124.95 B | \$124.95 D ** |
| PRO-HartFORTH | #M-21-071 | \$74.95 A | discontinued |
| PRO-MC | #M-21-064 | \$124.95 D | \$124.95 D |
| PRO-MRAS | #M-21-083 | \$74.95 B | \$89.95 D ** |
| RATFOR-M4 | #M-21-073 | \$99.95 F | \$99.95 F |
| UNREL-T80 | #M-30-054 | \$39.95 | \$49.95 ** |

| Applications | Catalog # | 05/01/87 | 07/01/87 |
|------------------|-----------|----------|----------|
| ===== | ===== | ===== | ===== |
| DSM51 | #L-35-204 | \$59.95 | \$59.95 |
| LED | #L-30-020 | \$29.95 | \$29.95 |
| QuizMaster | #L-51-500 | \$24.95 | \$24.95 |

| | | | |
|-------------------------|-----------|-----------|--------------|
| DSM4 | #L-35-205 | \$59.95 | \$59.95 |
| Little Brother-M4 | #L-50-510 | \$74.95 B | \$74.95 F ** |
| LBMU-M4 | #L-50-515 | \$29.95 | \$29.95 |
| LS-Host/Term | #L-35-281 | \$59.95 | \$59.95 |
| LS-LED | #L-30-021 | \$29.95 | \$34.95 ** |
| Mister ED | #M-51-028 | \$59.95 B | \$59.95 B |
| PRO-WAM | #M-51-025 | \$59.95 B | \$74.95 F ** |

| Utilities | Catalog # | 05/01/87 | 07/01/87 |
|---------------------------|-----------|-----------|-----------|
| ===== | ===== | ===== | ===== |
| diskDISK | #L-35-211 | \$39.95 | \$39.95 |
| FASTBACK | #L-30-055 | \$29.95 | \$29.95 |
| Mark III Collection | #M-99-003 | \$99.95 H | \$99.95 H |
| UTILITY-I | #L-32-070 | \$29.95 | \$29.95 |

| | | | |
|--------------------------|-----------|-----------|-----------|
| LS-diskDISK | #L-35-212 | \$39.95 | \$39.95 |
| LS-FASTBACK | #L-30-056 | \$29.95 | \$29.95 |
| LS-UTILITY | #L-32-150 | \$29.95 | \$29.95 |
| Mark IV Collection | #M-99-003 | \$99.95 H | \$99.95 H |

| Publications | Catalog # | 05/01/87 | 07/01/87 |
|--------------------------------|-----------|-----------|--------------|
| ===== | ===== | ===== | ===== |
| LDOS 5.1.4 Manual | #L-40-020 | \$30.00 D | \$30.00 D |
| LDOS QUARTERLY Volume II | #L-49-200 | \$9.95 C | discontinued |
| smallDOS 5.1.4 Manual ... | #L-40-040 | \$10.00 A | \$10.00 A |
| THE SOURCE 3-Volume Set . | #L-60-020 | \$99.95 F | \$74.95 F ** |
| NOTES FROM MISOSYS (set) | #M-49-000 | \$4.00 A | discontinued |

Freight codes: A = \$2.50; B = \$3.00; C = \$3.50; D = \$4.00
 E = \$4.50; F = \$5.00; G = \$7.50; H = \$10.00
 All unmarked are \$2.00 each
 Canada/Mexico add \$1 per order
 Foreign use US rates times 3 for air shipment

We accept MasterCard, VISA, and Choice credit cards
 Checks must be drawn on a US bank.
 COD's are cash, money order, or certified check to first-time customers

Letters to the Editor

Discontinued Software

(Fm: Theodore Masterton) When I decided to hang out with my Model 4p for a couple more years, I never anticipated that software would disappear from the market place. I guess it was clear to me that we would not see any new III/4 software but I have been horrified by the tendency of retailers to simply discontinue Z80 stuff. This is especially critical in the CP/M field. Absolutely amortized programs like Wordstar, Newword, and Dbase II simply gone.

Why don't retailers simply repackage the software with cheap doc's, or disk docs, or no docs at all and sell it out the backdoor. Maybe stuff would sell for \$40-\$100 bucks, like the new MSDOS stuff from the likes of Paperback Software. Instead, it seems the biggies have decided on a scorched earth policy; if it won't sell for \$500 then we will destroy the product.

Even Radio shack is DC'ing the upgrades to some packages they sold for 100-200 bucks a year and a half ago. Why? Is it not just a matter of a clerk walking over and copying a disk for me and charging ten bucks or so?

These are not rantings and ravings or rhetorical questions. I would be interested in learning the economics of software selling. Especially for those old, low sellers, who have paid off their development costs years ago!

(Fm: MISOSYS) We sell packages in that price range but few folks are willing to even pay that! Thus, when the documentation currently in stock gets used up, I doubt we will reprint it for most products. For certain packages, even in a very large company, the support cost and distribution cost is much more than the return on a sale if a product entertains low sales.

I remember a statement made by a Microsoft biggie about 5-6 years ago. Don't bother to even consider a product unless you expect to sell thousands. If MISOSYS ever adhered to that dictum, we would never

have released ANY software product except LDOS. Why? Because we have never sold thousands of anything! With 500,000 Model 4s out there and a few hundred thousand I's and III's, I sure would like to know what the heck is going on.

(Fm: Les Mikesell) It's not really a matter of choice. For the producers, the cost of advertising and distribution is higher than anyone would guess, and retailers can't afford to stock anything that doesn't turn over. Some of the new low-priced MSDOS software will probably make it as a mass market commodity if the suppliers do not have to provide support.

(Fm: Ken Hipple) Roy, After reading your last 2 messages I felt I should give you my viewpoint on the Mod 3/4 software subject. Two years ago, or even one, I would have bought EnhComp in a second but the time has come that I must seriously consider going MS-DOS even though I love my Mod III and think LDOS far exceeds MSDOS in almost all respects except sub-directories. The reason why is the availability of MSDOS software. Within the last year it has started to become a problem finding good packages for the 3/4 while they are proliferating for MSDOS and PC prices are dropping to within my range.

The 64K limit on 8 bit machines is another problem. If I had to make a guess I would say I'll have my 3 for another 2-3 years and then I'll move over to a PC. Looking at it this way then every time I go to buy a software package for the Mod 3 I end up thinking that I could put that money toward a PC instead. That means that unless I feel I just can't live without the package I don't get it (exactly what has happened so far with EnhComp). I figure that between now and the time I sell my 3, I'll spend between \$250 & \$1000 on it for software & hardware (repairs not included). There is a good chance I'll even end up getting EnhComp.

Unfortunately you & I must deal with the fact that the Mod 3/4 market is starting to shut down. In truth that is more of a problem for you but I can tell you this. If I still have the 3 when you stop supporting LDOS then I won't have it for much longer. I don't want to discourage

you but I'm not going to sit here & tell you that I'm going to be buying 1000s of dollars of software from you when I know that isn't true.

(Fm: MISOSYS) Ken, certainly I can understand that viewpoint. And appreciate it as well. That's why we have about 4 new MS-DOS products coming out in January-February time frame. Details on 2 of them were in Issue I.ii of TMQ. However, I still cannot justify people stealing my software products as I have eye witness testimony to that effect. Besides, PRO-WAM came out almost two years ago and should have sold in the tens of thousands.

Responses to TMQ I.ii Blurb

(Fm: Gary W. Shanafelt) I have been meaning to reply to your Blurb in the Fall QUARTERLY ever since I read it. You speculated there about the extent to which software piracy had hurt your sales. At the risk of adding to doom and gloom of the Z80 world, I would like to report the situation as I know it in this part of the world [Abilene TX]. The problem is not piracy, but the drying up of the whole TRS-80 market.

I can pretty much put all of the people I know with TRS-80's into one of three categories: (1) Those who really love their machines and continue to tinker with them and look for the latest programs. These are a small minority and I know none of them personally; they are people I have corresponded with about various problems. (2) Those who USED to love their machines and tinkered with them. Here is a larger number. Most of them, used NEWDOS as their operating system of choice; and they have either sold their TRS-80's or put them in storage and are now loving and tinkering with MSDOS machines of one sort or another. (3) Finally, the people who still use their TRS-80's. These had (and have) no interest beyond one or two basic functions and actually find TRSDOS 1.3 (or 2.3 on the Model I) sufficient for all their needs. They have no idea what non-Radio Shack software is and are uninterested in buying it. They also have no interest in piracy because they don't even know what is available for pirating.

To put it in other words: The people I know who were interested in advanced computer programs have gone over to MSDOS; those who still use TRS-80's see them as little more than glorified typewriters. The market for new, innovative software and utilities like those you produce, if the people I know is any indication, is simply not there.

Hence my earlier letter urging you to join the MSDOS world. That's where the future is. That's where I'll undoubtedly move in the next few years as desktop publishing comes down in price... and when I get there, I would like to be able to buy the same kinds of quality software from you that has made my experiences with the Model 4 TRS-80 so enjoyable.

Enough of that. If you have time to answer questions in future QUARTERLY Blurbs, here's one: DED86 seems to be an MSDOS equivalent of LS-FED. Features I've found especially useful on LS-FED are the ability to list load addresses of bytes as I move the cursor in machine language files, and the ability to disassemble instructions at the same time. Is there something about MSDOS that prevents similar features from being built into DED86?

(Fm: MISOSYS) Gary, I echo some of your comments; especially where the future is. That's why our April advertising includes a 2/3rds page ad covering eight MS-DOS products - six of which are now released. We have just kept too low-key on the subject before.

FED86 is closer to LS-FED than is DED86; albeit FED86 does NOT include disassembly; DED86 is more powerful and flexible but also does not include disassembly (it's a 67K .EXE file now!). I'm not so sure that the "disassemble instructions" would be that useful. No MSDOS file has absolute loading loations; a .COM file loads relative to the next available paragraph. An .EXE file has a header record listing a chain of file locations needing segment fixup operations. Thus, "load addresses" are not as useful as they would be on the Z80. We do have a symbolic disassembler built in to our ED/ASM-86 package.

(Fm: J.R.Jacques Baril) Your concern with piracy may be representative of a part of the reason for low sales. I suspect that the larger part of the reason is that most system users simply do not care for utilities. Our office uses quite a few computers, both Model 4's and some MS-DOS. None of the users give a hoot about the operating systems or utilities. When the company gets a utility (or application) it buys one and feels that it can be used on all it's machines.

I applaud both TRSDOS 6.3 and LDOS 5.3 but I really do not believe that most users will get beyond the extended date support. I believe that very few of us will make use of the added features. I guess I am not being encouraging but nevertheless I really appreciate your support and admire your durability. I sure hope you keep it up.

(Fm: Philip C. Jachem) If memory serves me sufficiently about the airy-fairy days of the Mod 4 introduction; LSI's move to its high-class building; the unveiling of LDOS/TRSDOS 6, and assumed application to the RS Model 2 & 12 and "other computers"; and the imminent emergence of a Z800 (whatever happened to it?), I now wonder whatever happened to the people who became "un-registered users" of LSI (and/or LDOS) which Mr. Schroeder estimated almost a third of a decade ago at "well over 40,000" to the current Mini-Blurb estimate of 12,000 registered customers of LSI.

(Fm: MISOSYS) Simple. When MISOSYS acquired the retail operation of Logical Systems back in March of 1986, we acquired the "registered customer" database as well. That base is still on the IBM PC which sits behind my desk (having ported the LB data to my 20 Meg drive). I can still bring up that database. It has 44,346 entries! If I were to send a flyer out to all of those entries, the postage alone would be \$5,543.25. This doesn't even begin to take into consideration the costs of printing and mailing overhead for 44 thousand items. We decided to trim down the LSI data base entries to only those records having an activity date of 1985 or 1986. LSI used to update the activity field whenever a purchase was made. We

just couldn't afford to spend the money to reach their entire list. That's how the 44,346 names got reduced to 12,000. Add our 4,000 and the mailing went to 16,000 with some duplicates; at a cost of about \$6,000! Our current CUSTOMER data base contains 17,263 names as I write this. We may also age it one day to eliminate what we feel are "inactive" records.

(Fm: Raymond S. Suchy) To help compensate for the small margins you're realizing (Vol. I, Issue ii, p.2), I was wondering whether some of the work could be "farmed-out" for others (like me) to do? I was thinking perhaps the typing/layout of QUARTERLY, scanning the correspondence, and/or other various and sundry "nuisances" could be performed by others thus allowing you more time/money to develop products. I don't want to see the QUARTERLY/any other MISOSYS undertaking fail! If you could use some free/no-strings-attached help, please let me know.

(Fm: MISOSYS) Well there's the kind of offers I like to hear. Actually, those types of things are best done in-house (literally) since that is the concentrated effort which goes on within the two to three week period before the final printing. We have a big box for QUARTERLY input which we dump things into whenever we come across something which we feel merits the QUARTERLY's attention. What we could use is ACTIVE input of articles. Some of the "letters to the editor" indicate that our TMQ issues are over the reader's head. Alas, since we are so heavy into this stuff, it is difficult to get down to basics. Thus, there's a splendid opportunity for lots of folks to work up an "easy" article. In fact see the next letter for some valuable input on this subject.

Another thing to reflect on is how do you think those 80-Micro authors got to be authors if 80 Micro wants "experience only"? They may have started writing articles for their computer club newsletter or the LDOS QUARTERLY. Well THE MISOSYS QUARTERLY is another way to get your feet wet (or your hands, as the case may be). TMQ is sent to the editorial staff at 80 Microcomputing. It is also

sent to the Model 4 Product staff at Radio Shack in Ft. Worth. So when YOUR article appears here, you are getting a little fame and notoriety; hopefully on the positive side.

(Fm: Robert A. Marotto) I am a recent subscriber to your QUARTERLY, and I read with interest many of your articles. Especially interesting to me was your disappointment with the lack of subscribers. Perhaps I can give you a viewpoint you are not aware of. But first some background.

I have owned a Model III since the beginning. At the time I purchased it I was a very busy industrial executive. Although I was interested in all aspects concerning computers, I never had the time to learn about them. I also have a small business which was my justification for buying the computer in the first place.

I never learned enough about TRSDOS to take advantage of all the power it had, but bought LDOS when it was available and will probably never get involved enough to take advantage of it's power with the Model III.

Quite frankly your QUARTERLY is over my head technically - I didn't understand what I was reading. I quit my subscription to 80 Micro because I didn't think they paid enough attention to the Model III or the basic user. I do subscribe to CODEWORKS and that has been a great help in learning BASIC.

I think the small time user, the learner, and people like them and me are hungry for something they can understand. I also think there is a big bunch of these folks.

I have learned a lot just changing my many disks from TRSDOS use to LDOS use and even more updating to LDOS 5.3.

(Fm: Roy Colding) A few years ago I had never touched a computer and the thought of "learning programming" was a little scary; however the excellence of the products from your organization really helped, i.e. LDOS 5.1.4, TRSDOS 6.2.1, and

PRO-CREATE. All were well written so that a novice like myself could decipher and make things "work".

THE MISOSYS QUARTERLY is really the best buy going for the TRS-80 community, it is up to your developed standards, which puts it ahead of the rest.

Also, I really enjoy the Compuserve LDOS forum, a vast amount of information passes across the screen.

(Fm: Michel R. Coutu) The reason I am putting these lines together, is the interesting bit of information you put in TMQ somewhere. Unfortunately, I can't seem to find it just now, but it has something to do with a tape backup driver or system that you are working on. Is this true, and if so, could you talk a little more about it in the next release of TMQ. I am definitely interested in one of those systems, as a 15 Meg is a two night affair to back up.

(Fm: MISOSYS) What you are referencing is my statement concerning a driver for the Cipher Data Products Series 525-CT FloppyTape™ Cartridge Tape Drive. This device uses a DC-600A data cartridge to store approximately 21.15 Megabytes using 256-byte sectors. The device ideally plugs into the external floppy port of a Model III or 4 since it requires two drive selects in combination with head select to address one of the six tracks of the tape accessed in serpentine fashion (i.e. track 1 is beginning to end, track 2 is end to beginning, etc.). Of course with 512-byte sectors, the device stores 25.07 Megabytes (26.63 with 1024-byte sectors). Unfortunately, we have just been too busy to spend the time to write the driver. Maybe someday...

(Fm: Walter Gabriel) Would it be possible to order the DISK NOTES in advance and receive them with THE MISOSYS QUARTERLY?

(Fm: MISOSYS) Walt, that's a good question. Unfortunately, for a couple of reasons, that would not be practical. First, we would need another flag in the

database just for that. I don't think I want to add another just for that considering the QUARTERLY subscribers are a subset of our customer database. We could not ship the disk with the QUARTERLY as the weight would then be different for bulk mailing. Furthermore, we have maintained a long standing policy of not maintaining customer credit card numbers on file. We also feel that one more thing to be done different in handling the mailing of the QUARTERLIES would just add to its cost. Sorry, but the next best thing is to give you a checkoff box on the coupon which accompanies the QUARTERLY.

Language Notes

(Fm: Lee C. Rice, Ph.D.) Very special thanks to you for including Harry Clayton's FORTRAN subroutine library in TMQ I.ii (although you did not print the FLL/JCL file which he documented). Despite all the broohah about Modula 2, Pascal, or even C as the newest fads, many of us who come to micros from a large time-sharing environment have never abandoned FORTRAN as our programming workhorse. No reflection here on PRO-MC - we all tend to stay with what we know. I am plugging away at trying to become a midway-proficient C programmer, and am grateful to have PRO-MC.

I purchased an MSDOS machine for my home earlier this year. I already have a Model 4 at home, and a Model 3 in my office; but I do have to do some software development under MSDOS, which is a much kludgier DOS than TRSDOS6 or LDOS5. My practice is still to do development work on the Model 4, and then tote it across to the MSDOS machine when done. I'm delighted that MISOSYS is providing standards and support for banked RAM also, since I am upgrading the Model 4 to 1 Meg currently.

Your response to JB about Microsoft FORTRAN's handling of double precision numbers in TMQ is on target. Many students seem to think that double precision declarations are the high road to accuracy, and usually they are not. Extreme care must be taken with FORTRAN especially, since passing a single-precision variable to a double precision

routine produces even more garbage. MS-FORTRAN for TRSDOS6 and for MSDOS is a fine implementation of FORTRAN. It is a number cruncher in its own right, and makes highly efficient use of the computer resources. Indeed, benchmarks have shown MS-FORTRAN can produce higher accuracy than some mainframe dialects (VAX FORTRAN, for instance). I'm sure that your PRO-MC, in the hands of a competent C programmer rivals it; but the point is that nothing else will.

Having praised MS-FORTRAN, and having digested Mr. Clayton's remarks on the bug in \$EXIT in FORLIB, I should also mention another irritating bug in MS-FORTRAN for the Mdoel 4. Despite what the manual declares, it appears to support files whose logical records are no greater than 128. Any attempt to read or write LRL.gt.128 produces a runtime error. I suspect that this is because the good people at Microsoft hastily cross-assembled their CPM version for TRSDOS without paying attention to the LRL indicators in the TRSDOS directory (which correspond to nothing in CPM or MSDOS).

In any event, I did phone Microsoft about the bug. They are apparently aware of it, but argue that, since they have licensed MS-FORTRAN to Tandy, it's Tandy's business to support it (which means here, "get it to run")! A call to Tandy in Fort Worth brought no more encouragement. Apparently their decisions regarding software support are dictated by the marketing division, not by good software practices; and we all know where the Model 4 stands in the Tandy marketing division.

If any of your readers anytime comes up with a fix to make MS-FORTRAN conform to TRSDOS standards for LRL, I hope that you will publish it. In fact, a logical record length greater than 128 is hardly ever needed for FORTRAN formatted I/O, but it should be available just for consistency with the TRSDOS conventions for file handling.

A couple of comments on EnhComp. I'm not a BASIC programmer, and don't use even the interpretive BASIC often, which is why EnhComp is one of the few pieces of MISOSYS software I have not ordered. On

the few occasions that I do use BASIC, it is for two reasons: it's superior (to FORTRAN) string and screen handling capabilities, and the ability to debug at the interpretive level before compilation (I do have MS-BASCOM for the Model 4, and MS-QuickBASIC for MSDOS). The only problem I could spot with EnhComp was its lack of full compatibility with interpretive BASIC. You lose the ability to debug and spruce up at the interpretive level, which is a major loss. Maybe it's offset by the more powerful assembly features, but I don't use an assembler that often anyway. If one is going to abandon the interpreter level, why not just abandon BASIC and go with FORTRAN: the syntax isn't all that different, and it's much more efficient too.

The upshot of this is that interpretive contexts are ideal for debugging and initial software development, whereas compiled or assembled contexts are better for efficiency and speed. I use a LISP interpreter on our VAX systems, and it is a pure delight in this respect - debugging and user interaction at all levels of development makes even interpretive BASIC look like a toy. Unfortunately there are no decent implementations of LISP for TRSDOS6 (hint, hint...) despite the fact that LISP is all the rage in AI programming generally.

I should also note that there is a tendency in the MSDOS environment toward C interpreters - two are currently available and two others have been announced. If PRO-MC is the success which it richly deserves to be, then perhaps MISOSYS might think about an interpretive subset of C for program development under TRSDOS. I may be talking through my hat here, since perhaps the memory requirements for such a C interpreter are beyond the reach of an 8-bit machine. My only experience here is with LISP, but I know that the demands which it can place on a system can be enormous.

Finally, you announced a RATFOR preprocessor in TMQ I.i, but there is no mention of it in TMQ I.ii. I hope that you go through with it. The only RATFOR translator I know for TRSDOS machines was written by the now-defunct company which

produced SOFTSCREEN and SOFTTEXT for the Model 3. I still use those under LDOS, and would certainly be happy to have RATFOR operating under TRSDOS6.

(Fm: MISOSYS) Sorry about omitting FLL/JCL. We may be able to dredge it up and put it on DISJ NOTES 7. Perhaps with our UNREL package now released, someone will take up your challenge and explore FORLIB to find the reason for the LRL <=128 restriction. If anyone does and submits it to us, we will publish it in a future QUARTERLY.

As far as EnhComp goes, you severely underestimate the compiling ease introduced by the supervisor mode of EnhComp. John Harrell stated, "EnhComp's superior user interface makes it an easy-to-use package." I feel that it's quick compilation and interactive compile supervisor mode makes program development almost as easy as interpretive BASIC.

I think that the reality of the Model 4 marketplace (or absence of same) precludes us from considering any further development of PRO-MC. We have spent some time fixing up some library bugs and will do some further cleaning before too long; however, there is just an insufficient market to warrant spending time on a C interpreter, a LISP interpreter, or a FORTRAN compiler. Such is life. After reading some of the other letters from our readers concerning the state of the TRS-80 marketplace (except the ones from fuming Model I owners concerning LDOS 5.3), who can blame us for dispensing with internal development of major items.

Last, but not least, RATFOR is coming. If all goes well, this issue will have a New Product Release. Hey, we are advertising it in 80 Micro! It's just that it got a little late. I'm sure that James Beard, the author of the RATFOR package, will be relieved when we start shipping it as he has spent an immeasurable amount of time on the software and documentation. We have been doing the final grooming of the installation and reference manuals associated with RATFOR. I hope it will be shipping by the time you read this.

LDOS 5.3 and the Model I

(Fm: MISOSYS, Inc.) The following heated comments are in response to our position on not developing, implementing, and supporting a Model I LDOS 5.3 product unless we receive 1000 prepaid orders as discussed in the last QUARTERLY and posted on our CompuServe forum.

(Fm: Joe Sewell) There are still quite a few Model I's out there, even though they haven't been manufactured for a long time. Maybe I'm just being old-fashioned, but I think support of an existing customer base should be just as important as supporting new customers. If the old customers tell their friends that such-and-such company stopped supporting them, would those friends become new customers? Highly unlikely. Yet that is what's being done with the Model I. I admit, I can see why new software is down, especially from companies that simply cannot afford the development time and finances. But support of existing software is a different story.

As for 80 Micro, the software isn't 100% portable between Model I's and III's. Programs that stick with standard BASIC stuff work OK, but most of the good programs don't stick with standard BASIC; if they use BASIC at all, they many times have POKES that simply don't work on a Model I. Differences between the ROMs also cause an unwanted hassle. Finally, incompatibilities between TRSDOS 1.3 and TRSDOS 2.3 hindered things to the point where I just gave up on 80 Micro (by the way, I'm not the only one who has these feelings); 80 Micro just doesn't have ENOUGH Model I useable information anymore to warrant subscribing.

(Fm: MISOSYS, Inc.) Sorry but our decision is based on staying in business to serve the most customers versus going under and serving NO ONE. Your dialog about 80 MICRO not having enough to satisfy your needs sounds like the old chicken and the egg. If their polls show that 80% of their readers use Model Is then they would satisfy them. If their polls show that 2% of their readers use Model Is, then you can guess what they'll write about. Your

opinions about pokes only supports my philosophy which says portability is preferable to poking around where you don't belong. If every program honored the DOS, you would probably have found that Tandy computers would be supreme and Apple would have been buried long ago. The tremendous fragmentation among the various Tandy computer users based on different DOS products resulted in NO formadable presence of a compatible set of users.

(Fm: Robert W. Vail) You have totally betrayed me and all the other Mod I owners out there. I may be one of a minority who still uses his antique but I am also one of your oldest customers dating back to one of your first EDAS's.

And since I have just purchased a Tandy 1000-SX, I could continue to offer a market to you for your MS-DOS products as well as the antique ones.

One reason I continue to use the Mod I is the fact that I have all the data for all property owners in our landowners association on AFM. It will be worthless to me without an accurate dating stamp.

The irony is that all I really want is a patch to allow me to enter a date after 12/31/87. I really don't need the other "frills" you are offering to the Mod III/4 owners.

Incidentally, if you decide not to honor the needs of customers such as me, I shall never order another product from you and you may then remove me from your mailing list.

I am sure that I will not be the only person to hold such views.

In closing, let me emphasize my shock at your unfair abandonment of your old and faithful customers. You are really one of the last people from whom I would have expected such action. Better LSI should have kept LDOS!

(Fm: T.J.Hodges) In the second issue of TMQ I read of all the wonderful things that will be in LDOS 5.3; then I read of

the requirement of a 1000 order minimum for a Model I version production run. Of course I am not expecting you to receive such a large initial order. The 80 Micro ads do not mention the Model I and you have only about 500 subscribers to TMQ; surely they are not all Model I owners!

In the event that the software is not available, will there be any guidance to converting the Model III version on a "do it yourself" basis? Perhaps an article in TMQ would be the way to handle it. Anyone choosing to tackle this type of project would be somewhat sophisticated and only facts, not procedures be presented.

(Fm: Terry W. Moore) I have had a Model I for over six years and I was a little upset when I read that there might not be a Model I version of 5.3. Yes, I understand that a certain number of orders are needed to justify it; however, the information in the QUARTERLY was not specific. Do you mean that I should send in my money now and hope that another 999 people will order it too? If they didn't, would I lose my money? How will you get 1000 orders for a Model I version when your ad doesn't even hint at the possibility? Remember, you only have 600 subscribers. Sorry, I'm sure you are more than aware of that.

For my part, the most needed item in 5.3 is a new date routine. Have you thought of selling one to the Model I people or printing one in the QUARTERLY? I personally would not ever need to calculate dates prior to 1987. Would this make it easier to write? Would selected portions of the Model 3 version work with Model I 5.1.4? If they would, I still might buy a copy.

What this all boils down to, is that I LOVE my model I! I don't want to see it die. Heck, I like the 3/4 machines too. But I cannot justify buying one at the current Tandy price (or from a discounter for that matter) when I could buy an Amiga for the same price.

In conclusion, give us some more information on 5.3 and the Model I. Also, if possible, could you print a list of all

known newsletters for the 1/3/4/; 80 Micro is fast becoming worthless for me.

(Fm: Robert B. Patton) Quite honestly, I am flabbergasted. I own a TRS80 Model I computer and it sounds like you have chosen to make the Model I an obsolete computer as of Dec. 31, 1987 because I will no longer be able to use the date! I think there will be several of us out here that will be in a bit of a bind unless you at LEAST offer a patch to allow the date prompt to be valid after the end of next year!

Secondly, one of the big plusses with LDOS is its compatability between the systems. While I do not as of yet own a Model 4, I do have definite plans of purchasing one. However, if I will not be able to share data files between my Model I and my Model 4, I may just as well go to a MS-DOS type of a machine. I had hoped on being able to upgrade my system without having to re-create everything I now use!

I only hope you get enough letters from other Model I users so that you will offer the upgrade. Don't forget, for every letter you get, there are probably several other users out here that are too lazy to write but that would probably buy the upgrade if it is offered.

Do you require a financial commitment or just a written commitment? Your newsletter implies that you want a financial commitment but then you say "no orders will be returned nor will notification be made." I do not relish the idea of spending \$26.95 and then never seeing the money again should you decide to not offer the upgrade! I am sure that you wouldn't do that but that is the way your wording makes it sound.

I have another question on your new date procedure. It isn't clear to me from the newsletter what will occur to files that have a date from between 1980 and 1987. Will the LDOS system now see and display all dates properly between 1980 and 1999 or will it just be correct between 1988 and 1999? Also, it sounds like OTHER systems will see ALL dates as being between 1980 and 1987 (even those that are

actually between 1988-1999 will appear as being between 1980 and 1987).

I hope to hear from you soon saying that you will be offering the 5.3 upgrade for the Model I. The improvements you went over in the newsletter sound really good, I sure do hope that I will be able to use them.

(Fm: Winfield Smith) Don't know how you'll get 1000 orders from a subscription list numbering 600, but you do have a larger mailing list and I hope will be able to include a brief line in your print ads, plus a little publicity on the BBS's and elsewhere. In any event, I sure hope you get your 1000 soon.

(Fm: William F. Fields, Jr.) I am unsure how to feel about your policy about Model I upgrades to LDOS 5.3. I don't understand why there is a need for \$24,950 of orders plus \$2,000 shipping charges to justify serving those of us who still use our "old faithful computer" that more or less started the personal computer revolution.

Part of my uncertainty about how to feel about your Model I update policy is that I'm not sure how reasonable it is to expect 1000 orders to materialize. If it's remote to impossible, then I feel quite misled by the advance word about this upgrade because it contained no indication of the anti-Model I bias that would have kept me from spending lots of bucks on additional hardware and software. And if I'd known a couple of years ago that LDOS support would become this tenuous, I probably would not have bought an interface that can use no other DOS than LDOS (the LX-80).

As you may detect, I feel my trust has been somewhat hurt by (what appears to me to be) your abrupt abandonment of Model I support for LDOS after you had indicated implicitly that it would continue to be supported.

On to other issues. I liked the idea of Ken Arntsen about including some public domain software on the DISK NOTES disk. I was sorry to read on and note that you

differ in your opinion on it but I guess unless you get a groundswell of interest in it, it won't happen.

Finally, I had a trivia question that you might wish to answer in the next TMQ: Whatever happened to LDOS 5.2? Did I miss it or was it never released?

p.s. It's interesting to see how a hobby can influence one's children. I grew up knowing only that daddy went off to "work" each day. To this day, I don't think I fully understand what he did each day while at "work". I work in the data processing department of a company whose name I'm sure you would instantly recognize; however, because my son sees me at what he calls the good ol' Model I, he has a better understanding of my activities at work. Out of that and his experience at school with personal computers came him asking for his own computer - another Model I. He enjoys writing BASIC programs on it and thinks he wants to grow up to be a programmer like his dad. I imagine your children as they grow up cannot help but benefit in very subtle ways as well as obvious ones from your experience in technology. For some reason, I thought of that as I wished you and yours a Merry Christmas and Happy New Year.

(Fm: MISOSYS) Well, a lot has been said. From the "flaming" letters and comments received, I detect a lot of misunderstood assumptions. Let me try to correct some of this confusion.

To begin with, MISOSYS never cashes checks nor puts through charges unless it is ready to ship the product being ordered. That's our policy! Our statement about "not returning orders" meant just that, we will not send you back what you sent us unless your order is accompanied by a SASE. If the product doesn't materialize, the box of orders goes through the shredder. If you understand what we are saying, you should realize that you are out only the cost of your stamp, envelope, piece of paper, and the cost of one check (not the amount written on the check).

Okay, why this tough policy of requiring 1000 orders? We believe there's an easy explanation. Over the past few years, we have experienced a drop in both Model I and Model III software sales to practically nothing. As far as we were concerned, there is no marketplace for Model I and Model III software. On the other hand, we recognize that the 500,000 Model 4 computers which have been sold are all capable of running Model III software. Many Model 4 owners have told us they use their Model 4 in Model III mode. That's why we gambled on a major effort to groom LDOS 5.3 to be as similar to LS-DOS 6.3 as we could make it. This would be very beneficial to that owner; and that's where we find our current 5.3 sales. If we had continued to entertain reasonable Model I/III software sales over the last few years, there would have been no question; a Model I 5.3 would have been done. But that didn't happen.

Persons with a business background recognize the concept of "opportunity cost". What this means is that the costs of doing a particular job must take into account the loss of revenue opportunities which you would expect if your time was spent doing some other job. Were MISOSYS to spend its time developing, implementing, and supporting a Model I LDOS 5.3 job, that time would have to come from not doing some MS-DOS development. You have already read the comments of respected individuals telling us we should have moved on to MS-DOS long ago. Look back at your old 80 Micros. Where are all of those companies who used to advertise back then? Well, they are either out of business because they could not find enough buyers or they moved on to other product lines (like MS-DOS, UNIX, or Apple). MISOSYS long ago recognized the absolute necessity of moving to MS-DOS for software development. That's why we bought a Tandy 2000 the first month it was released. It has been unfortunate for us that we were so determined to continue developing and supporting the TRS-80 that we just had NO time to develop for MS-DOS. But we should have left a long time ago. That means that there would have been no MRAS, no MC, no UNREL, no Mister ED, no PRO-WAM, no RSHARD, and no LDOS 5.3! We have finally drawn the line. We have had

to. It's either rapidly shift to MS-DOS or drop out - like everyone else has. We just don't have the time to waste any more on supporting a marketplace which is so small that it's not economically justified. MISOSYS does not have deep pockets. It has been the sweat of Roy Soltoff and the other free-lance authors who have provided MISOSYS with products since 1978. The buck stops here, folks. Like it or not, I'd rather stay in business than give up.

Now a lot of the confusion about dates stems from a lack of understanding as to just what is used in the DOS. Let's take a few moments out to examine date storage. The year is referenced to 1900; thus, 80 is 1980, 81 is 1981, etc. Since the storage field is a byte, it can contain a "number" from 0 to 255; thus, the date storage in memory can represent 1900 though 2155 (a date range quite useful for those persons currently living and those who will be born over the next 8 generations).

Forgetting about directory dates, the Model I DOS (every DOS) provides an @DATE routine (this is actually in ROMC on the Model III) which generates a date string in the form "MM/DD/YY". This DOS function is used by BASIC's "DATE\$" function. The coding of the @DATE routine would take the year stored and convert it to a 2-digit ASCII representation of its value. The actual code would correctly generate a string if the stored year is any value from 0 through 99 (i.e. 1900 through 1999). An error would be introduced if the stored year exceeded 99. A year value of 100 would get converted to a string of ":0"; the colon being the next ASCII character higher than "9". Even a casual programmer would recognize that the @DATE algorithm is primitive; but useful over the expected life of the product. It starts with a character value of "0" in the tens column and proceeds to increment it by one every time 10 can be subtracted from the year value before it underflows. That's how 100-109 would generate a string value of ":0" through ":9". Similarly, the next decade would generate ";0"-";9".

Now let's introduce directory dates. The original structure of file dates as used by LDOS was introduced by Randolph Cook in his VTOS 3.0. It may have even been in a never released TRSDOS 3.0 (Model I) which I actually have never seen, but just heard about. Nevertheless, Randy's scheme packed the month, day, and year into a 12-bit field. The scheme allowed three bits to store a year since 9 bits are required to store the month and day (check out the numbers: 1-12 take 4 bits; 1-31 take 5 bits). The remaining 4 bits of the 16-bit "word" were (and are) used for the MOD flag, date-not current flag, CREATE flag, and one spare which was subsequently assigned as the PDS flag). The directory date field is updated by the DOS whenever a file which has been written to is closed. The @CLOSE directory date updating routine gets the date data from the @DATE storage area. Since only three bits are available for the year, something has to restrict the year to the range 1980-1987. That restriction is imposed by both the DATE library command and the "Date?" prompt on BOOT. LDOS makes use of dates in two ways. (1) When you enter the current date, it updates the @DATE storage area and subsequently displays the date in string form such as, "Thu, Mar 12, 1987". (2) When you invoke a DIR command, the directory date is unpacked to a string form of "dd-~~mmm~~-yy". This directory date is also used for parameterized control in BACKUP, DIR, and PURGE.

In order to take care of both usages of dating, major changes had to be incorporated in the DOS. There is absolutely no way that both usages could be patched. On the other hand, if one wanted to limit the DOS' use of dates to just the first usage, that is indeed easily done by a patch! MISOSYS has NEVER stated that we would not do such a patch for the Model I. It is just unnecessary to do it right now. There's nine months left in 1987! That is exactly what we would do if the required number of Model I orders don't materialize. Such a series of patches would essentially eliminate directory dating; eliminate the use of the date parameter in PURGE, BACKUP, and DIR; eliminate the display of dates in the DIR command; eliminate the date string generation on date entry during BOOT and

DATE; and would extend the DATE input through 1999.

I think my discussion here should cap the flaming comments we are receiving over our decision to require a demonstrable audience of buyers before we further erode our ability to remain viable in the software marketplace. The TMQ is not the only place where we have noted this Model I decision. It was also addressed on our Compuserve forum. I certainly expect word of mouth and discussions in the few computer clubs that exist for the TRS-80. Sometime later this year MISOSYS will also be mailing a Mini-NOTES flyer to every one in our database (17,000+). Our Model I position will be so noted. We have also discussed the issue with Dave Essex who writes the Pulse Train column in 80 Micro. We have NO intentions of wasting ad space on the issue until, and if, we have a product to advertise. That's the best we

can do.
 SYD TRUE
 P.O. BOX 297
 BOSTON, MASS 02111
 Now let me respond to some of the other questions raised in the previous comments of our Model I audience. LDOS 5.3 will correctly display file dates between 1980 and 1999. Any other DOS which uses the same directory date storage convention as discussed above, will display LDOS 5.3 and LS-DOS 6.3 directory dates only between 1980 and 1987 for all dates ranging from 1980 through 1999 unless they also update their system. What about NEWDOS80? What about DOSPLUS? What about MultiDOS? When was the last time you discussed NEWDOS with Apparat?

No one missed LDOS 5.2. Since this new release of LDOS was to correlate with the 6.3 release of LS-DOS for the Model 4, we decided it would minimize confusion by skipping 5.2 and designating the LDOS release as 5.3. No, it had nothing to do with UNIX Version V.3.

We have nothing new to report concerning our position on the public domain software stored in the DL sections of our Compuserve forum. We don't have enough time to spend on profit-making ventures, let alone time to organize disks of PD software. I wonder how Montezuma Micro does it? I believe that some computer club

(or multi-club organization) ought to do that sort of thing. Most clubs have a pile of PD disks in their library. The local club here (NCTCUG) has a large library. Why don't you club organizations band together and create a national consortium of TRS-80 clubs - one purpose of which would be to consolidate all of the fragmented libraries of PD software into one organized collection. That's who should be doing it, not MISOSYS! Now if you clubs don't know of your sister organizations but do want to organize such a national consortium for a PD library, have your leader drop me a note providing your address. We will publish the TRS-80 club list in a future QUARTERLY. That's the best we can do.

Software Theft

(Fm: Martin L. Beauchamp) Let me curb my enthusiasm for a moment and delve into a more serious topic. My hobby <microcomputers> and my profession <cop> combine from time to time in unusual ways.

As technology advances, so do the crooks. The use of micros in crime is on the increase, and its not unusual to execute a search and seizure warrant, only to find that the criminals have been storing their records on disk. Most use a relatively simple encryption code, easily broken. Once in a while, some really innovative programming crops up.

My concern is this. On a recent raid, we recovered a TRS80 Model III with four floppy drives and a 5-meg hard disk. Standard stuff really, but in the course of dumping the hard disk, I found some of your software that had been pirated. Maryland has some pretty stiff penalties for unauthorized access AND for unauthorized possession of pirated software. We're talking criminal here, not just a civil case for copyright infringement.

How would you feel, if in the future, you had the opportunity to prosecute for this type crime? This case is dead, because the software theft was part of a plea bargain package. If the situation should arise again, however, would you agree to press

charges, or, at least give a deposition for the State? Interesting possibilities, no?

(Fm: MISOSYS) We would most certainly pursue such an issue.

GRPLIB, TRS-80 info, etc.

(Fm: Paul Bradshaw) In the latest MQ (Fall '86, page 45), Harry G Clayton, Jr. mentions that he is working on a MC to GRPLIB/REL interface. Since I will be purchasing PRO-MRAS/ PRO-MC within the next two or three months, I'd be VERY interested in such a product. Any chance that you would carry it as an actual product in your line rather than just printing it in MQ? Any chance of carrying a MAKE-type utility in the future? Just let me add my vote for a MISOSYS Pascal.

On page 84, you mention that "there is no easy way to redo a configuration without reBOOTing." This is precisely why I wrote UNSET.BIN, available in the DL's here [the LDOS forum on Compuserve]. It shows you what modules are in memory, and how they are connected to the system. I find I can redo my entire configuration in any way I please without having to reBOOT. UNSET is no panacea (it can't handle the hi\$ module of PRO-DD&T, and I have no idea how it would work with ZSHELL), but it is a great aid.

Any chance you'd advertise the subscription in 80 MICRO (as part of your normal full page ad) to try to increase circulation?

(Fm: MISOSYS) UNSET may be a useful and good tool; however, since there are many modules which interface in non-standard ways, it can't handle everything. My statement reflects the hopes of most folks that a product can do everything under the sun.

As of the first of the year, we have dropped the full page ad as too costly. The ad runs about \$3000 and the products being advertised don't even draw enough to pay the ad - let alone production,

support, overhead, amortization of development,... We could not economically advertise TMQ in 80 MICRO as too much space would have to be devoted to explain what it is. At rates of approximately \$1700 for a half page, that shuts the door. Our funds have to be conserved for productive, profitable products. TMQ would only be profitable at an order of magnitude increase in subscribers. At this point, I don't think that the entire TRS-80 8-bit marketplace is WILLING to drum up enough supporters to subscribe. I am usually disgusted when I hear folks tell me they don't subscribe to 80 MICRO because it has too little for them. For pete's sake, without an adequate subscriber base, a magazine goes under! Look what happened to BASIC COMPUTING! Look what happened to COMPUTER USER! These were TRS-80 specific magazines. If the TRS-80 users don't want to support the magazines that attempt to cater to their machines, then the users don't really deserve to be supported. I think that the subscription fee charged by 80 MICRO is peanuts compared to the hundreds of dollars we waste every year. Their subscription fee is certainly peanuts compared to the amount invested in our TRS-80 machines. If only 40,000 subscribers have Model 4s, what happened to the other 460,000?

If someone has a PASCAL for us to publish, we will consider it. The market is too small for MISOSYS to consider commissioning someone to write us a PASCAL compiler. If someone has a MAKE for us to publish, we will consider it. The market is too small for MISOSYS to consider commissioning someone to write us a MAKE. In other words, at this point in time, we have no NEW TRS-80 8-bit product under consideration. However, since there are thousands (maybe hundreds) of good hackers out there willing to spend their time in "labors of love", if they produce a useful "product", we will consider publishing it. We no longer have the resources to FUND the development of TRS-80 8-bit software. That still puts us in a good position to PUBLISH your work.

TRS-80 Information

(Fm: Gary Phillips) Unfortunately I haven't been a QUARTERLY subscriber because I can't justify the price of the QUARTERLY for mere curiosity (much the same reason that I don't already own THE SOURCE volumes). I try to direct my expenditures into areas from which I will derive as much utility as possible. However, the odds are good that I will succumb in a weak moment and decide to buy THE SOURCE before it becomes unavailable. I certainly never regretted THE PROGRAMMERS GUIDE TO TRSDOS 6.

(Fm: MISOSYS) You will regret not getting a QUARTERLY subscription. We don't overprint the quantity by much over the subscription base. Thus, if you delay too long, you will lose out on early issues. As it is, Issue I.i is gone.

(Fm: Paul Bradshaw) Can't justify the QUARTERLY? In only two issues, I've totally regained my investment! The specials offered to QUARTERLY subscribers (15-50% off selected MISOSYS products) are reason ENOUGH to subscribe (I've picked up LS-FED II and PRO-DUCE both for only \$14.98!). Not to mention the 100 pages jam packed with useful information and programs! We're talking 100 pages of info - not a single "ad" to be seen. TMQ is invaluable to each and every user of MISOSYS products! Tell me again you can't justify it? <grin>

(Fm: Gary Phillips) Note that my 80-Micro subscription is paid up for two years in advance. However, there is some validity to criticism applied to the magazine since it left Wayne Green's hands and capable management. The quality of technical expertise on EITHER TRSDOS or MS-DOS machines is rapidly declining. I have repeatedly seen answers to reader inquiries and reviews of products with statements that literally made me shudder. They have a policy of refusing reviews and technical data from authors who are "unknown" to them, so my letters have been wasted. 80-Micro is rapidly turning into just another "user" oriented MS-DOS

magazine, which will guarantee its demise because there are too many in that market already. I detest MS-DOS and IBM anything, but have no choice other than to keep up with it. 80-Micro is not a suitable choice there either. (I prefer PC Tech.)

Although I understand there have been disagreements between you and the editor/publishers of Northern Bytes, I would still urge you to keep at least a small ad there, where you will reach serious Z-80 users who can appreciate the quality of your product.

And now that you've made me feel sufficiently guilty, what's the current subscription to TMQ cost?

(Fm: MISOSYS) A TMQ subscription is \$25/year for 3rd class bulk mailing in the US, \$30/year for 1st class mailing. As far as 80 Micro goes, some folks will always have differences of opinion but one has to continue to apply pressure for what you want. I too was saddened when Wayne Green left. Although he took the brunt of a lot of criticism, he did what he thought was right and did not buckle under pressure from his major advertiser and focal point - Tandy. Why was I saddened when Wayne left? Why because advertising rates jumped up about 100% for me over the course of one year. CW Communications felt that 80's rates were way too low. Of course when Wayne had the pub, its content did seem "thicker". Size relates to the amount of available advertising. 80 maintains a relatively consistent ratio of editorial to advertising. When ads drop, editorial must drop also.

I really don't know what Jack Decker's problem is. My hunch is that its the same as the difference between cat lovers and dog lovers. In any event, I'd advertise in TRS-80 computer club newsletters before I put any ads in Northern Bytes. Besides, few of my products work under NEWDOS80 because of our extensive use of @PARAM

(Fm: Richard Buchman) How can you say that 80 Micro shouldn't support the Model I? If not for that machine there wouldnt be any other machines and perhaps the old CBM

Pets might have taken over. I own 3 model I's and though they might be a little old (1979), I can say that they are one of the best machines to learn hardware hacking on. Also, owning a machine that is in the Smithsonian in DC makes my tears come out of my eyes.

(Fm: MISOSYS) Any magazine has to respond to its reader composition. Latest figures show 80's readers at about 43% Model 4/4p/4d, 40% MS-DOS, and the remainder split between II/12/16/6000, I/III, and Coco. Such is life. A profit-making corporation must exist to be in business - the IRS demands that. Entities exist to serve the past and pet themes. These entities are called non-profit organizations. Support your local computer club!

Nostalgia

(Fm: Ray Pelzer) In a sense, its kinda fun for all of us "old fogeys" of the computer world (I guess I count, cuz I'm almost 29) who remember pretty blinking lights on front panels. Like that old plaque I had says: "... Das Rubbernecken Sightseeren keepen hans in das pockets. Is nicht fur gefingerpoken und mittengrabben. Ist causen poppencorken mit pittzensparken. Relazen und vatch das Blinkenlights."

Leap years a la 2000

(Fm: MISOSYS) The following discussions took place on our LDOS forum on Compuserve after I made a remark stating that the year 2000 will not be a leap year since it is divisible by 400.

(Fm: H. Brothers) The rule is "years divisble by 4 are leap years, except years divisible by 100 which are not, except years divisible by 400 which are." The year 2000 will be the first year that fits that final exception since the "universal" acceptance of the current calendar, which was completed in the early 18th Century. The Russian calendar, by the way, doesn't include the final test and, unless they change, the world will again be on differing calendars on Feb. 29, 2000. From

K & R, page 37: if (year % 4 == 0 && year % 100 != 0 || year % 400 == 0) it's a leap year

(Fm: Pete Granzeau) Years divisible by 400 ARE leap years in the Gregorian calendar.

(Fm: H. Brothers) Okay, if K&R wasn't a sufficient print reference, then how about this from "Webster's New World Dictionary of the American Language, College Edition" publication date unknown since that page fell out years ago, and the closest dictionary at hand right now: "a leap year is a year whose number is exactly divisible by four, or, in the case of century years, by 400." I also can quote the algorithms out of several other programming references if K&R wasn't enough.

(Fm: David Hardy) Following is the exact language of California Government Code, Section 6802: "The years 1900, 2100, 2200, 2300, or any other future hundredth year, of which the year 2000 is the first, except only every fourth hundredth year, are not leap years, but common years of 365 days. The years 2000, 2400, 2800, and every other fourth hundredth year after 2000 and every fourth year, except as provided in this section, which, by usage in this state is considered a leap year, is a leap year consisting of 365 days."

In California, 2000 is a leap year! I suspect the law is the same in every other state. When I get time I'll check the United States Code and let you know the federal law. No charge for this legal advice in view of your remarkably efficient order service. 5.3 and the QUARTERLIES arrived today!

(Fm: Adam Rubin) "Reference?" Did someone say "reference?" Here's what surely counts as the definitive reference on the subject. It's from the Papal Word of Gregory XIII, On the Correction of the Calendar (my translation): "...but of every four centuries three pass by without leap year and the fourth has a leap year, so the years 1700, 1800, and 1900 are not leap years, but the year 2000 is again the

usual leap year." This became the official calendar with Great Britain's "Calendar (New Style) Act" of 1750, which applied to the colonies as well.

(Fm: Dave Hardy) I have looked all over heck and gone for a federal statute defining leap year-without success-no regulation either. Unless it is in the Statutes-at-large somewhere and was never codified, I don't think it exists, and I am unwilling to undertake that research.

I did have a paralegal pull up on LEXIS the 38 federal cases since the mid 30's that contain a reference to "leap year" and then I looked at them on the screen. The most interesting item is that the District of Columbia had an ordinance in the exact language of the statute 21 Henry III, which was to prevent prejudice to pleaders because leap year had 366 days instead of 365-treating Feb 28 and Feb 29 as one day. There is an Indiana case quoted in Walker v. Hazen, 90 F2 502 (1937, D.C. Cir) which describes the statute as a relict of "more barbarous days"! (I'll bet there was also an English statute in the terms of the Papal Word but which Henry VIII probably repealed!)

I think though that we can advise Mr. Soltoff, that, subject to the qualification that free advice is worth every nickel that you pay for it, he is on sound legal ground in treating the year 2000 as a leap year. Let us trust that other programmers are as alert or 2000 may be the year of the jackpot in the Frederick C. Brown or Heinlein sense, I forget which.

(Fm: Adam Rubin) Well, it looks like we've come up with a pretty firm conclusion, anyway. One source I just thought of (while typing this message!) is newspapers from 1899, which would surely have an article explaining why 1900 isn't a leap year. (One book I looked at reprinted a page from a calendar that gave February 31 days, but that had been a printer's error.)

I don't think there was any English statute on the matter until the "Calendar New Style Act, 1750" (Act 24, George II, Chapter 23), as--well, let's say that religious differences delayed the acceptance of the Papal Word.

One other point about the year 2000 is often misunderstood. The year 2000 is the last year of the twentieth century, and the twenty-first century doesn't start until January 1, 2001. The first century was the years 1 through 100. The Astronomy Library here had a folder of newspaper clippings from 1899, explaining the same point.

Anyway, I agree that it would be fairly safe to assume that 2000 will be a leap year. The problem, software-wise, is likely to be worse with 2100, which (of course!) is not a leap year. I wonder how many programmers will slip up there?

I learned all these calendar details back around, let's see, 1977 or so. This was for a program to print out a calendar -- you know, the usual computer-printout calendar. The formula was essentially what you used. My program was in APL, though, so it can't be displayed here! The fun part of APL was that I fit the entire program onto one (very long) line! Later, I wrote a program for a Roman calendar, with days before the Kalends, Nones, Ides, years in A.U.C., etc. (That was one line, too, though it was a bit of a kludge.) Anyway, I think there are unofficial plans for 4000, 8000, etc. to be common years, but I'm not too worried about those, Yet!

(Fm: Adam Rubin) Well, the first hundred years were 1 through 100, the second were 101 through 200, ..., the twentieth were 1901 through 2000, and the twenty-first will be 2001 through 2100. Check the newspapers in about twelve years and a few months for details.

No apologies needed for adding to the thread! This is yet another installment of a subject started earlier this month. Remember the discussion about the year 2000, and whether it was a common year or a leap year? This came from a question about DOS date handling ending in 1999,

even though the new date field can handle the following few decades. Anyway, Roy asked for a reference for 2000 being a leap year. Dave Hardy quoted a section of the California State Code, and I gave the relevant section of a Papal Word of Gregory XIII from 1581. (That's Pope Gregory as in "Gregorian Calendar.")

For further details about the story of our calendar (Why did September 14, 1752 follow September 3?), see the Encyclopedia Britannica article on "Calendar." Hope this answers the question!

(Fm: MISOSYS) I give in. I give up. Forgive me... I accept that the year 2000 is a leap year. February in the year 2000 does indeed have 29 days! PRO-WAM CAL application says it does!). Where were you guys when I needed a good trademark attorney? Now to those TMQ readers irritated at all of the space devoted to the topic, let me commend to your attention that this will probably be the most pervasive coverage of the year 2000 for at least a decade. Probably by 2001, this TMQ issue will be cited as a reference. You don't want me to do an injustice to some future historian, now do you?

How to get ZAPPED

(Fm: Jeff Beadles) Is there any way, as a registered owner of several of MISOSYS's programs, that I can get the patches and zaps withOUT having to subscribe to TMQ? I kinda hate the idea that I have to pay for fixes for bugs in programs that I already shelled out the bucks for. With the work that I've been doing with the M4 lately though, I can't really support a subscription to TMQ. (I've been hard and heavy in Unix Sys V, on an NCR Tower that's here in my house now.) Just wondering, as the only other solution that I know of is to pay CIS to download the patch files.

(Fm: MISOSYS) Easy one Jeff, the patches from THE MISOSYS QUARTERLY have been uploaded onto this board. If you don't want to pay for the download, you can

always purchase the associated DISK NOTES (\$10 each + \$2 S&H). If you know someone who has the patches, then get a copy of the patches from them (not the whole DISK NOTES, please) [note: please feel free to distribute the FIXx file found on any DISK NOTES]. Finally, if you know of a particular patch(es), drop me a SASE and request a paper copy.

(Fm: Jeff Beadles) Ok, I was just wondering and all. I've downloaded the patches before, it's just the idea of the thing, after paying >\$200 for 2 programs (PRO-MRAS & PRO-MC) that I then have to pay to have the programs "fixed". I guess that I was just expecting to get a little post-card type of dealie in the mail with patches for the registered products that I own.

(Fm: MISOSYS) Our patches won't fit on a post card. Also, we don't keep electronic records as to what products every one in our database has so it would be kind of difficult to extract. I feel that the TMQ publication with the DL here and direct response when queried is the best we can do.

QUARTERLY EXPIRATION

(Fm: Ray Pelzer) Is there any sort of customer number I can use in the future to make things easier for you? I saw 87/08A on the mailing label of my last QUARTERLY, but that simply looked like an expiration date for the subscription.

(Fm: MISOSYS) We don't keep a customer number - we keep a customer NAME. Now really, you don't want to be known as a number, do you? Now unless your name is really long, that "87/08A" is both the expiration and mailing class for a TMQ subscription.

Corrections to TMQ I.ii

(Fm: Ken Kane) I have Disk Notes 6, for TMQ I.ii but have been unsuccessful using BANKER/CMD. I installed the [Alpha Tech] patches, complete with a fresh BOOT with

<CLR>. Banker alone reports my 15 banks, maps them OK. Reserve function is OK. However BANKER (FREE="x-y") does not change the map even though "x-y" is a previously reserved range of banks. After showing the map I see Error 2BH.

With the AT patches in place, both SPOOL and Superlog 4 can be installed into higher banks with the simple parrameter, e.g. (BANK=4). This is easier than the use of BANKER to block out lower sets, and would be appreciated in a future release of PRO-WAM.

This is not your problem, but when the AT patches are installed, ALLWRITE cannot use its Area 2 and 3 alternate editing function. I will call PROSOFT and see if they can remedy this. My main reason for getting the Alpha Tech board was to keep PRO-WAM in place while having full access to the ALLWRITE editing banks. It is possible to use "the Anitek RAM driver" without the AT patches and have full use of ALLWRITE, but not BANKER, etc.

(Fm: MISOSYS) There is a bug in BANKER. If you look at the listing on page 34 of TMQ I.ii, you will find the line 00052 which reads [LD (RESFRE1+1),A]; but it should read [LD (RESFRE2+1),A]. You can easily work up the patch. Also note that line 67.5 should be added to read [LD C,A]. We will consider your PRO-WAM parameter request in the next release of PRO-WAM. There really will be another one in 2-4 months.

Humor not in uniform

(Fm: Adam Rubin) [re the misreference of MD-DOS for MS-DOS ("D" being next to "S" on the keyboard)] Gee, that's not what I heard. I thought MD-DOS used the 101st element -- Mendeleevium-DOS, artificially produced and radioactive. I couldn't find out much about LD-DOS, though. Lead-DOS is another product entirely; it's marketed as Pb-DOS. Perhaps it's some sort of mnemonic for entering control characters: Left shift-Down arrow. Does anyone else have any better ideas about MD-DOS and LD-DOS?

(Fm: MISOSYS) I rather like "Pb-DOS". In fact, I can change my name. Similar to Jesse-Bob, I can become "Plumb-bob".

The following was forwarded to us by Lee C. Rice, Ph.D. It was written by Guy L. Steele, Jr. [MIT] (with apologies to Joyce Kilmer).

A LISP POEM

One thing the average language lacks
Is programmed use of push-down stacks.
But LISP provides this feature free:
A stack - you guessed it - is a tree!

An empty stack is simply NIL.
In order, then, the stack to fill,
A CONS will push things on the top;
To empty it, a CDR will
Behave exactly like a POP.

A simple CAR will get you back
The last thing you pushed on the stack;
An empty stack's detectable
By testing with the function NULL.

Thus, even should a LISPer lose
With PROGs and GOs, RETURNS and DOs,
He need his mind not overtax
To implement recursive hacks.

He'll utilize this clever ruse
Of using trees as moby stacks.
Some claim this method slow.
Because it uses CONS so much,
And thus requires the GC touch.

It has one big advantage, though:
You needn't fear for overflow.
Since LISP allows its trees to grow,
Stacks can to any limits go.

For LISPers everything's a breeze;
They neatly output all their trees
With format-free parentheses
And see their program logic best
By how their lovely parens nest.

While others are by GOs possessed,
And WHILE-DO, CASE, and all the rest,
The LISping hackers will prefer
With COND their programs to invest
And let their functions all recurse
When searching trees in maddened quest.

To men of sensibility
The lesson here is plain to see:
Arrays are used by clods like me,
But only LISP can make a tree.

RAM wars

(Fm: Robert B. Anthony) Looking forward to more comments on XLR8 Mod 4/4P board vs Alpha Tech .5Meg or 1Meg board.

(Fm: MISOSYS) I'd like to get a few articles as input from our readers. We just don't have the time to write everything here. Note the next letter for more information on the subject. I also believe that this issue's THE HARDWARE CORNER has some information on the boards.

(Fm: Lee C. Rice, Ph.D.) A running account of the ongoing saga of TRSDOS6.3/LDOS5.3 and the ram boards. The Alpha Technologies ramdriver software does not work with TRSDOS6.3 or LDOS5.3. Alpha Technologies told me that a new ramdriver was in the works, but that it had been produced, and was marketed by, Anitek.

So I phoned Anitek, and they said that their new ramdriver package had forms for TRSDOS6.3 and LDOS5.4. That package was ordered and arrived. As promised, the drivers work with TRSDOS6.3 and LDOS5.3 but neither of them will work on a system with a Radio Shack hard drive. So now I have a 1Meg Model 4 with a 15Meg hard drive which I can operate with 1Meg if I don't use the hard drive, or as a hard drive system if I don't use the memory.

I am also a licensed owner of LeScript, and I suspect that Anitek wrote the ramdriver as they wrote LeScript - by ignoring all the DOS calls and addressing hardware directly. I know the inherent risks of that method of programming, which is why I use Allwrite most often (it honors all DOS conventions, and interfaces nicely with ZSHELL, which LeScript completely ignores).

This letter is just a warning which you might want to pass on to other MISOSYS QUARTERLY readers. I'm no assembly

programmer, so I am probably stuck; and will probably go back to TRSDOS 6.2 and LDOS 5.1.4 so that I can have both memory and mass storage.

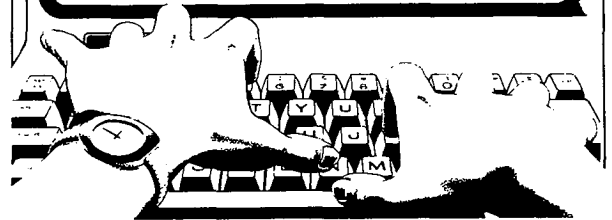
When I phoned Anitek to order the ramdriver, they explained that they had received so many orders for the memory board that they had run out of stock no fewer than three times. That leads me to believe that there are going to be a lot of surprised Model 4 owners out there.

I know that MISOSYS is busy with other projects, and I would never make any suggestion to slow down the appearance of RATFOR; but... Some day you should consider writing a ram driver for the TRSDOS6.3/LDOS5.3 family. It would blow the existing ram drivers straight out of the water, and you'd surely have my order. For what it's worth, you can address the extra memory under TRSDOS 6.3 from Microsoft FORTRAN by using the "CALL OUT" library subroutine - but that's a far cry from a ramdrive.

If I ever do find something that works, you'll be the first to know.

(**Fm: MISOSYS**) There is a public domain ramdrive driver on our CompuServe forum. Mention is made of it in THE HARDWARE CORNER. It's free, it works, and it uses the DOS @BANK service call made possible by the Alpha Tech patches which I published.

Your Model 4 computer
may not speak **XZ#M%S**
But with MISOSYS
language products, she will
speak **ASSEMBLER, BASIC,**
C, FORTH, and RATFOR



PRO-CREATE - The "standard" macro assembler used by professionals and novices alike. Nested macros, nested includes, nested conditionals. Full screen editor; cross reference. \$74.95 + \$3 S&H

PRO-DUCE - A 2-pass labeling Z80 disassembler from disk or memory with screening input for data areas. Generates /ASM files. \$29.95 + \$2 S&H

PRO-MRAS - Powerful relocating macro assembler development system REL module compatible with Microsoft! Includes full screen text editor, REL librarian, VM linker with overlay capability \$74.95 + \$3 S&H

UNREL-T80 - Converts MRAS or M-80 REL object files to /ASM. Use on your own REL modules, FORLIB, GRLIB, BASCOM, BASRUN, etc. \$39.95 + \$2 S&H

PRO-EnhComp - An enhanced BASIC compiler with a built-in assembler for Z80 in-line code mixed with BASIC. LOGO-like turtle graphics, strings to 32767 chars, multi-line functions, keyed/tagged SORT, REPEAT-UNTIL, structured IF-ENDIF, labeled statements, double precision functions. \$124.95 + \$3 S&H

LS-TBA - A structured BASIC translator. Labeled statements, Conditional translation, pseudo global and local variables, 14-char var names \$24.95 + \$3 S&H

PRO-MC - A full K&R C compiler with nearly 200 functions. Structs, unions, bitfields, enum, dp floats and functions. Wildcards, I/O redirection, args, overlay support. Requires PRO-MRAS or M-80 \$124.95 + \$4 S&H

PRO-HartFORTH - A full 1979 standard FORTH compiler using indirect threaded code. Runs under the DOS. Has floating point, access to DOS files, screen editor, block graphics \$74.95 + \$3 S&H

RATFOR-M4 - A professional implementation of RAtional FORtran. Provides stucture and greater portability to FORTRAN programs. Fully documented with tutorial user manual. Requires FORTRAN compiler \$99.95 + \$3 S&H

Note: Model I/III products may be available on request.



MISOSYS, Inc.

PO Box 239
Sterling, VA 22170-0239
703-450-4181 MC, VISA, CHOICE
Orders Only! 800-MISOSYS 1P-5P EST Monday-Friday

VA residents add 4½% sales tax. S&H: Canada add \$1;
Foreign use S&H times 3

Our Compuserve Forum

CALL WAITING

(Fm: Craig Tenbroeck) Being new at LDOS and at Compuserve, can anyone tell how way to disable CALL WAITING while I am on line? Bell of PA is my utility.

(Fm: LDOS Support) Call Forwarding is one viable option. Another is that some CO's offer the ability to disable Call Waiting for the next outgoing call by using #70 or some such. Check with your local representatives to see if this function is available in your area.

Extensive discussion on ComuServe has concluded that the "Skip call waiting" function: may or may not be available in your area, may or may not use #70 if available, and may or may not be linked to the arrival of "automatic long distance carrier selection" in your service area.

Try it and see what happens!

(Fm: Ken Hipple) SOME phone systems have the ability to disable call waiting. To do it you enter a code before the call and then while that call is in progress call waiting is disabled.

Unfortunately, we don't have that ability here yet and I can't remember the tone sequence you need. I'm sure someone else on here will though.

(Fm: Marc Nowell) In the Dallas/Ft. Worth area, if you dial the digits *70 on a touch-tone phone which has call waiting enabled, you get three short beeps and then another dial tone. Call waiting is then disabled for THE CURRENT CALL ONLY! Thus, you don't have to "reset" anything for Ma Bell. I use a terminal program that lets me set the dial prefix string for my Hayes-compatible modem, so I set it to

ATDT*70,

in which the "AT" is the ATTention command, the "DT" stands for Dial Tone, the "*70" dials the correct sequence, and

the final "," has the modem pause so the new dial tone kicks in before dialing. Works like a charm!

User ID prompt protocol

(Fm: LDOS Support) The "User ID:" message is always sent with even parity, and if you are set for eight bit no parity, it will look like trash. Go ahead and type in your number it will be taken OK and then you will get the password prompt and all will be well from then on.

Disabling the "MORE" prompt

(Fm: Theodore Masterton) I would like to be able to list the DL area data without my computer stopping for the "<CR> for more" message. I think it used to do this. Can anyone help?

(Fm: LDOS Support) The MORE prompt may be removed by doing a SET PAGED OFF, or you may disable it permanently by setting your defaults over on page CIS-9.

Forum HELP

(Fm: LDOS Support) To participate in discussions here, just log on and read the messages that have been left since you last logged on. The RN command will start this process. You can get help at any step of the way by entering HELP or H or ? in many cases.

At the prompt after each message, you can use the RE command to REply to it, leaving your comments.

If you have lower case in your machine, you will probably want to GO CIS-9 and pick the options to enable lower case operation.

To download, (you are using LCOMM under LDOS, right?) you can get more information by reading the Data Library bulletin from the Bulletins menu. This tells you how to get MNETA/CMD for use under LDOS. If you already have a terminal program that can

do either a CIS protocol or XMODEM (CIS "A" or "B" preferred), use it.

If you get stuck at any step of the way, just yell again for help! Looking forward to seeing you on the forum...

HEX Listings for MS-DOS

(Fm: Jim Beard) For all you folks who miss a Hex listing capability in MS-DOS, try HEXLIST.BAS in DL9. It is a short standard BASIC program which produces the same format as the TRSDOS 6 "LIST <fname> (H)" command. The format is leading with the record number: first byte address, then the bytes in hex, then ASCII listing, if appropriate.

MISOSYS ordering on CompuServe

(Fm: Rick Campbell) I would like to know how to go about ordering the PDS system from your company? If at all possible I would like to order it using compuserve or from you.

(Fm: MISOSYS) First, Use my PPN - 70140,310. You would have automatically addressed the message directly to me if you responded to my message with RE. Then you can order it right here. Your message MUST be saved with the SP option to make it PRIVATE. Give me name, address, etc. If COD, so state. If charge, give card number and expiration. Also give telephone number (daytime). Specify Model 4 or Model I/III. The I/III version is PaDS; the Model 4 version is PRO-PaDS. Either is \$29.95 + \$2 S&H. \$2.00 extra for COD.

Note: Do not send orders my EasyPlex - We have a long standing policy of not using that service. If you prefer, you can telephone orders direct to us at 800-MISOSYS during our ordering hours (1P-5P M-F Eastern Time).

Saving Money

(Fm: William Chao) I am new to CompuServe and so far, I find it really interesting with all these discussions. The only thing I am wondering about is if there is a list

of all the files that are available in the download library besides going thru DIR or BROWSE. It would be nice if there is a file that contains what's in the download library so people don't have to spend a lot of time reading thru it (I am on a college student budget).

(Fm: Bob Haynes) There is a file somewhere in the CIS sigs submitted by John Deakin [74015,1624] called "HELP5.DOC" which contains some superb tips and techniques for saving money while using CompuServe. I would strongly encourage you to send him an EMAIL message, he could tell you exactly where it's located. Sorry, I don't remember where I found it, but I do refer to my copy often.

To get your "catalog" of files, simply set up your software for capture to disk, and type the following from the main function menu: "DLx" [x is the library number you want]. "SCAN *.* /DES" [scanning is the same as browsing, but without all the pauses. /DES tells SCAN to include descriptions of files].

After exiting CIS, use your text editor to delete any extraneous data, then LIST filename (P) to get your printout.

To the best of my knowledge, there is no specific file in the DL's that catalog contributions.

(Fm: jeff brenton) S will list just the file name + size + date uploaded + number of downloads. S/DES will list the same, plus keywords and description. CAT is the old command that [S]can replaces, but CAT does still work.

(Fm: LDOS Support) You can obtain a list of the files yourself, as long as you capture it to disk or the printer so that you can use it later. To do this, you can use the "CAT /DES" command. If you want them in alphabetical order, use "CAT *.* /DES" instead. If your computer stops at the end of a screenful with a MORE! prompt, you can respond with "SET PAGE OFF" to get continuous output.

If you have a reasonably clean phone line, capturing this listing will be less expensive and faster than would be downloading a file containing the listing. Also, it will be much more up-to-date.

Once you've got a complete list, you can use the "CAT *.* /DES/AGE:7" command every week to get just those files added in the last seven days. Of course, each of these commands would be issued repeatedly in each DL of interest.

Our new MS-DOS section

(Fm: jeff brenton) With four IBM forums, section 5 of TRS80PRO, DECPC and any number of other MS-DOS forums on CompuServe, I don't think a separate MS-DOS section here is worth it. After all, we'd only use it to cuss about how poorly designed it is!

(Fm: MISOSYS) The April issue of 80 MICRO (which comes out in March) should have an ad from MISOSYS which shows about SEVEN-EIGHT MS-DOS products published by us. That includes LB86 (Little Brother), LBMU86 (the LB maintenance utility), FM86, FED86, DED86 (noted in the TMQ), ED/ASM-86 (noted in TMQ), and RATFOR86, and DSM86. We do expect to begin shipping DED86 and ED/ASM-86 by the end of the month [ED/ASM-86 and DED86 are both released products as of mid-March]. LB86, LBMU86, FED86, and FM86 have been available for about two years. So with all of the MS-DOS products from us (which you can run on your PC or R/S compatible), MISOSYS needs a section on our SIG to aid in supporting our MS-DOS customers. It's thrust is not to "cuss about [MS-DOS]" but to discuss the MS-DOS-compatible products published and supported by MISOSYS.

VIDTEX PLUS at 1200 baud

(Fm: Ken Vaughn) Does anyone have a patch handy to change the default baud rate from 300 to 1200? I'm running VIDEOTEX PLUS on a Model 4.

(Fm: Tom G.llaudet) Ken, the patch is: Patch Vidtex/cmd (X'3003'=77). You can also do the same thing from an AUTO LOGON File which is not quite so permanent...use: |S776C| as the first line. It's all outlined in the documentation. The 300 Baud patch is (X'3003'=55).

XMODEM for downloading/uploading

(Fm: Allen Foster) I have what appears to be operating copy of XMODEM.COM but when I try to download any of the programs all I get is a timeout and incomplete file deleted. What am I doing wrong?

(Fm: LDOS Support) Hi allen, you are downloading to XMODEM/CMD at your end, not XMODEM.COM, right? Also, to use XMODEM you must tell CompuServe that you are not a VIDTEX terminal. You do this by GO CIS-9 and setting your terminal type to "other".

An alternative is that you can logon with a ";TTY" appended to your number. For me, that would be 76703,437;tty instead of merely answering 76703,437 to the User ID: prompt.

Also, you must specify XMODEM at the File Download Protocol Menu. If you don't get such a choice when attempting to download, then your terminal type is set wrong.

Finally, you must set your communications parameters via

```
SETCOM (W=8,P=N,BREAK=0)
```

along with the desired baud rate. This should get you going.

(Fm: Allen Foster) I tried SETCOM (P=N,W=8,BREAK=0) and still get time out. I have A 4P and am config on sig as "OTHER". I have tried both XMODEM.COM and MODEM.COM. both just time out. Any other thoughts?

(Fm: LDOS Support) What are you trying to download, and what are the exact steps are you using? To download, let's say, a file

called MOOSE.COMD from DL 0, you would follow these steps if you are using the transient MODEM/CMD that is run from inside of COMM: (1) GO PCS49 (2) DL 0 (3) BRO MOOSE.COMD (4) At the (R D M) prompt after the description, pick the XMODEM option (#4 I believe) and hit enter. (5) In COMM, hit <clear><shift><0>, and at the command prompt enter the command "XMODEM R MOOSE/CMD" (6) The file transfer should start in about five to twenty seconds.

(Fm: LDOS Support) To properly upload with XMODEM, you must start the transaction at the CIS end first. At the DL prompt, type

UPL MOOSE.FIX/PROTO:XMODEM/TYPE:ASCII

You will receive a message indicating that the upload is starting. At your end, do whatever your terminal package needs to start the transfer. That should do it, other than you will be prompted for keywords and a description. Note that if the file is not text, but is a program or other full eight bit binary data, the command line should read

UPL MOOSE.COMD/PROTO:XMODEM/TYPE:BINAR

instead.

VIDTEX Downloading

(Fm: Carol D.) Perhaps, while we're on the subject of file names for your computer, somebody should warn new members and neophytes NOT to copy the name given on the file. Those names are listed as NAME.cmd, for example, and the problem is that SEEMINGLY harmless "." When I tried to run something later, MY COMPUTER WOULDN'T LET ME! Told me "file access denied"! A period designates (to my Model4) a PASSWORD! In the above example, password would be CMD and you cannot run without it! I know NOW it was a jerky goof, but oh the FRUSTRATION!.

Patches in hardcopy

(Fm: Mark P. Fishman) I have been following this forum since I got my 6.3 (actually earlier, but in detail

recently), and it moves FAST! I missed a couple of messages in a two-day lapse -- they were off the edge of the board before I got back. I hope LSI plans to mail the patches to 6.3.0 on paper to all registered users (including those who don't use CIS, poor souls). Better yet, maybe you can pay Roy to put them in TMQ and give more people another reason to subscribe!

To all who care, I am using an Aerocomp 30 Meg HD with Montezuma Micro's drivers for "TRSDOS 6.x" and LDOS "5.1.x" but with LS-DOS 6.3 and LDOS 5.3; one 5 Meg partition overlap between the two. No problem yet. (This is on an early 4 with 4 DS floppies internal -- TEAC 1/2-high, 2 40 trk and 2 80 trk.)

I learn more here than I can anywhere else. Keep it up, all, and thanks.

(Fm: MISOSYS) There is supposed to be an LS-DOS 6.3 section in the next issue of THE MISOSYS QUARTERLY [that's issue TMQ I.iii, and it's in here!]. Bill and I worked out the details.

JCL -> LCOMM and abort

[the following was a response to a problem report of LCOMM aborting to LDOS ready when the user tried to designate the file name for a file transfer. LCOMM was invoked with JCL. Note that the same problem could arise when invoking BASIC via JCL]

(Fm: LDOS Support) You are starting LCOMM up with a JCL, and in the JCL you do not have a //STOP execution macro. The JCL file should look like this:

```
SET *CL RS232x/dvr (parm, parm, ... )
LCOMM *CL
//STOP
```

This will fix this up, preventing the abort after <clear><6> <clear><9> or <clear><7> <clear><9>.

Dialog on Forum commands

(Fm: Ted Pinkert) I've heard UA discussed, but didn't quite catch what it was for. I'll try it tonight on "Your report". I assume a RI will get me the (R D T) prompt, otherwise I'll have to try OP;ST then RFxxxxx.

(Fm: LDOS Support) Yes, an RI will get you an (R D T) unless you are in non-stop mode (that's OP;SM N;T), which is easily turned off (via OP;SM A;T).

(Fm: Ted Pinkert) I'm normally on nonstop. OP;ST;T (later OP;NS;T) also work. Don't know why I thought I had to try RF - that set my HI message counter. Nice that that parameter is now under user control in OP.

(Fm: H. Brothers) You can also set HI directly from the function prompt. In fora where I don't read all sections of the board, I always end a session with "HI;L;G xxxx" which translates as "Set HI to the last message on the board and then go to "xxxx". [ed. note: due to a recent discussion concerning the frowned upon usage of "SIG" versus the CompuServe recommended usage of "Forum" to describe the various interest groups, Hardin has coined the usage of "fora" as a plural for "Forum"].

(Fm: LDOS Support) I've been confusin' folks, methinks. Make that (UA RE T) for the message board, and (R D M) for the DL areas.

(Fm: Adam Rubin) Am I using the Forum's <S>can <F>orward command incorrectly, or is there a problem with this? A few days ago, right after a RTN, here's what happened:

Function: sf

Forum messages: 74938 to 75445
Start at what message number
(N for new to you): 75435

Function:

I know there were messages in this range, as I'd uploaded 75438 to 75445 fifteen minutes ago, and they'd shown up when I did a RTN. What's going on here?

(Fm: Tom Gallaudet) Adam, You have to use "SF" after logging into a Forum, not after uploading messages. When you first enter a Forum and find 20 new messages, you can type "SF N" and scan the new headers then you can "RN" or "RTN" and see the same messages. If, however, you "RN" first then "SF N" won't work. I don't now why thats just the way it is.

BTW, If you send a X'03' (break) [or CTRL-C] instead of a <CR> after establishing the carrier connection when logging on to CIS, you won't get the "Host:" prompt. It saves a little time. [nor will you get re-prompted for "Host?" after you "Bye"]

(Fm: Adam Rubin) Thanks for your comments. I've finally figured out how the high message pointer works -- RI and RM don't change it, but RN, RTN, RF, and RR do. The problem here, though, is that I was trying to scan forward from a specific message ("SF75435"). This shouldn't be affected by which messages I've read, as long as there are messages in that range that I'm able to read. Either I'm misusing the SF command, or it doesn't work the way it did before.

I've been using ^C [CTRL-C] when logging on all along -- it wasn't until a few weeks ago that I learned anything else was possible! You must have seen the message where I asked about OCLC. A recent assignment involved the use of OCLC, which uses the CompuServe network (among others) for access. This required the "Host name:" prompt, as I didn't want CompuServe at that point! Anyway, I'm sure your message will help quite a few of the users here.

LDOS Information

New 5.3 KILL command

(Fm: MISOSYS) Yes, KILL fs1\$ fs2\$... is included. If "fsi\$" fails due to an error, then "fsi+l\$" through "fsn\$" are ignored. I believe that it should work that way. Sort of like JCL aborting when a program returns an error condition rather than blindly going ahead and continuing.

RESET

(Fm: Alan Brown) LDOS 5.1.4 seems to have a global RESET bug on a Model III: (0) Insert single sided 'QFB' image of LDOS 5.1.4 master in drive 0. (1) Press reset <orange button> (2) RESET (3) DEVICE shows normal settings (4) FORMAT :1 as double sided disk (5) BACKUP SYS0/SYS:0 :1 per Les Mikesell. phone call 8/30/83. (6) BACKUP :0 :1 (NEW,I,S) (7) Remove single sided QFB image of LDOS 5.1.4 from :0 (8) Insert new double sided image of LDOS in drive :0 (9) Press reset <orange button> (10) RESET (11) DEVICE shows 'SYS Error'. Why does the global RESET work from the master and not from the backup?

(Fm: LDOS Support) That's not a bug, but a normal result of how RESET works. (1) Don't use global RESET. (2) Never use global RESET. (3) It isn't a good idea to use global RESET. (4) I distinctly frown on using global RESET.

Here's why: Global RESET restores all the system parameters to their defaults. Or so it seems. Some things that global RESET does and doesn't do.

HIGH\$ goes back to top. This unprotects all high memory. If RS232T is up there, it gets overwritten and a character comes in over the RS232, blammo - one way ticket to nowhere.

All the DCBs and DCTs are restored to the system defaults. That means a single sided disk in drive zero. You just turned off your ability to access the second side of the disk, and interpret the directory of the disk correctly. The moral of this story? Don't use global RESET unless you

really, really understand what is going to happen. A much better solution is to boot with the <clear> and <enter> keys held down. Achieves the same end, but ensures that the system is truly "clean".

Printer Control & PR/FLT

(Fm: John H. Shepherd) My experiences in trying to send function control codes to my printer using LBasic or MINIDOS may be of interest. Typically I would LPRINT CHR\$(27)"B"chr\$(2) to switch my GEMINI printer to the ELITE print size. Sometimes this works, other times it just prints B.

After a lot of digging I found the trouble. If PR/FLT is active with MARGIN set non-zero it sends CHR\$(27) as the first char of the line & then send SPACES to fill out the margin. A simple solution is to send a space (or any greater char) before the CHR\$(27) as the first of a line.

Since PR/FLT waits for a char greater or equal to SPACE before padding the margin this puts the padding ahead of the ESC character. PR/FLT behaves the same way if you use <CLEAR><SHIFT><P> in MINIDOS.

SLOW BACKUP and Interleave

(Fm: Ted Pinkert) When using only TRSDOS6-FORMATTED DISKS, both the regular backup and the (X) backup clock in at 63 seconds, starting when track 0 is read and ending when track 41 is verified. But when using an LDOS-FORMATTED DATA DISK and a TRSDOS-formatted System disk, the (X) backup takes 101 seconds. I think that the LDOS and TRSDOS interleaves are different, n'est-ce pas?

(Fm: MISOSYS) Of course the interleaves are different. TRSDOS 6 is expecting to be running on a 4 MHz machine and thus has the ability to take advantage of a faster machine to adjust the interleave. That results in one less rotation to read a track. LDOS, expecting a 2MHz machine, uses a different interleave. The QFB which comes with LDOS 5.3 will take advantage of the "faster" interleave if SYSTEM (FAST)

is set. FORMAT does not, though. Thus, a QFB'd disk under LDOS 5.3 and a TRSDOS 6 disk would have the same interleave.

(Fm: Ted Pinkert) I was wondering if the interleave in 5.3 is set for optimum speed on a III running in SLOW mode. If so, will there eventually be a patch to optimize interleave for a 4 in III mode running FAST. I don't have a feel yet for my new DS drives, so I don't know if the slower track dumping is entirely due to the extra granules per track.

Thanks for taking time to reply, and hope your answer will be of interest to others, too. (With my multi-floppy system I will do anything possible to get backup time to a minimum.) (Just had a thought - maybe I should be using LS-DOS 6.3 to format all my disks - would that help?)

(Fm: MISOSYS) The interleave is a function of how the disk is formatted. FORMAT makes no change and hasn't been changed. QFB will take advantage of putting a "faster" interleave in when formatting if your in FAST mode. If you FORMAT disks using 6.2 or 6.3, they will be using the faster interleave even in Model III mode.

DOCONFIG/CMD with 5.3

(Fm: Ted Pinkert) Can I still trust my OS to DOCONFIG/CMD III.5.1 using 5.3 on a 4? This is one program I don't want to test by trial-and-error! Thanks.

(Fm: MISOSYS) The front end of the SYSGEN module in 5.3 had to be changed to accept the DRIVE parameter pass through from SYSTEM. Look for the corrective patch in this issues's PATCH corner (DOCON53/FIX).

SMOOTH - What's it for?

(Fm: Ted Pinkert) My understanding of the problem corrected by SMOOTH is that if it happens at all, there will suddenly be a marked slow-down in disk access, as opposed to a slight or barely noticable slow-down. Also, that an alternative

correction to the problem would be to set the drive speed to something other than exactly 300 rpm (say, 303 or 297 rpm). Am I correct in this? In other words, unless something drastic happens, I can go ahead and run with SMOOTH=OFF and not worry?

SYSGEN of SMOOTH

(Fm: Allen Foster) Is it possible to SYSGEN the SMOOTH option? Seems like all the other system parameters can be saved. However upon rebooting SMOOTH is always gone. Also, would there be a patch to make the directory command default to A=NO like in 5.1.4?

(Fm: MISOSYS) It is possible to make SMOOTH sysgenable. I thought that it was put into the CONFIG table. I just checked the code and it wasn't. I'll get a patch up for that soon. I posted a patch already for patching the DIR command. My next posting will be in [this] issue of TMQ.

Use of CMD"V ..." in BASIC

(Fm: Alan Kaplan) I was running the following simple basic program under the new basic:

```
10 for i=3 to 1000 step 2
20 for j=2 to sqr(i)
30 if i/j=int(i/j) then 60
40 next j
50 print i;
60 next i
```

I broke and did a cmd"v" to check the variables. When I tried to resume with "cont" I got a "next without for in 60" error. When I break to use other basic extensions, I can resume without error.

(Fm: MISOSYS) I don't believe that you can break then use CMD"N", CMD"X", or CMD"V" as these facilities clobber BASIC then reload it with a "BASIC *" invocation.

(Fm: Alan Kaplan) Okay, I suppose one could interpret the docs like that. I guess I had a preconceived notion of what

cmd"v" could do. But I see what you mean. It is still a valuable tool for BASIC programming.

Date-not-current flag

It may be useful to know that under LDOS 5.3, I dropped the use of the directory bit for indicating the date not current. Thus, you will never see the plus signs in the date field of a DIR under 5.3.

The TIME? prompt

(Fm: LDOS Support) I did find one possible change, I don't know if you consider it significant enough to change.

In LDOS 5.3, the TIME command and the TIME prompt at boot will take the time with either the colon or periods used as a separator. If the colon is used, the seconds are optional. If the period is used, however, all six digits of the time must be entered.

In LS-DOS 6.3, the seconds are optional regardless of the separator used.

(Fm: MISOSYS) I don't think that it is deliberate. But I doubt I will change it at this point.

Where'd the CAT go?

(Fm: Jim Gaffney) My 5.3 arrived yesterday and while I'm very pleased with the package, I was a bit disappointed that a "CAT" command had not replaced the DIR sans (A) parameter. Guess that a few "File not found" errors when I switch back and forth won't kill me

(Fm: MISOSYS) There is ABSOLUTELY NO MORE ROOM in SYS1 to add the entry. Believe me, I spent many hours packing SYS1 to put into it all of the things which I did. I probably spent the most time of any module on that one. I even pondered over (dread the thought) shortening the "LDOS Ready" prompt so I could add another library table entry.

Where's MODELA/III?

(Fm: Byron P. Peebles) Is it legal issue that prevented MODELA/III from making the logical (to me) trip to LDOS 5.3? I really could have used it, since my LS-DOS 6.3 wouldn't boot.

(Fm: MISOSYS) Why would I put MODELA/III file on a 5.3 disk? It comes on every 6.2 disk you get. You can easily copy it over. Besides, did you see how much room there was left on the LDOS 5.3 disk? Where was I supposed to squeeze in the 14K of MODELA/III? Also, MODELA/III is a copyrighted program of Tandy Corp. I can't just steal their work, can I?

(Fm: Gary Phillips) I haven't received my 5.3 yet, but I'm sure you have the appropriate code in the bootstrap to cause that ROM to do its thing. Did you see my note to Joe about the MODELA/III that was included with LS-DOS 6.3? Apparently it is somehow different from the one from 6.2 that I have been using. (ORCH-90 under LDOS 5.1.4 runs fine with the old image, but crashes the system with the new one. This is a consistent occurrence.)

(Fm: MISOSYS) No I do NOT have any code in the bootstrap of LDOS 5.3 to load in the MODELA/III file. That code is in the BOOT ROM of YOUR Model 4P. Your 4P loads in the MODELA/III file when it finds a non-T6 disk during booting.

(Fm: Gary Phillips) I know the BOOT ROM does the loading. But the way it identifies the need to perform that load is by looking for a CALL or JP instruction in the boot sector that will branch to an address lower than a certain value. So when I say I'm sure you have the appropriate code in the boot sector, I simply mean that you have such an instruction there, whether it is in fact executed or merely a "dummy instruction".

(Fm: bob snapp) What is looked for is (hex) CDxx00. It would be very difficult

to have a Mod III boot sector without an occurrence of that code.

MAX-80 ROM image

(Fm: John H. Shepherd) I greatly appreciate your very prompt response to my order for an LDOS 5.3 Upgrade kit for MAX-80 but have a problem installing the MSOFT/ROM. I used BACK-UP (5.1.4 version) to make a working copy of the LDOS-530 disk, killed copy23b/BAS and M80DVR/FIX leaving 10.5K free space (the original had 6.0K free). Using COPY (5.1.4) I put DUMPROM/CMD on my 5.1.4 boot disk & invoked it. This generated a file MSOFT/ROM (10.5K) on the 5.1.4 disk.

I then booted LDOS-530 and invoked DO = MERGEROM, got an error message "File not in directory". On checking I find no file "MERGEROM" nor anything like it on any disk. In fact there is no \$/JCL file on the LDOS-5.3. Can you help please?

(Fm: MISOSYS) One or two disks got out without the paper addendum noting that the MERGEROM/JCL file was omitted. Here it is:

```

CMDFILE
N
D
MSOFT/ROM
(three blank lines)
D
SYS0/SYS.SYSTEM
E
//EXIT

```

That was three null lines between MSOFT/ROM and D. The JCL just uses CMDFILE to append the ROM image to SYS0. Note that you will have to leave 10.5K free on the SYSTEM disk for the merged SYS0.

What won't work on x.3?

(Fm: Theodore Masterton) If my software won't run under 5.3 I will run under 5.1.4. That's ok. I just was hoping I could just put everything over on 5.3 and learn operationally if the program was incompatible without worrying about destroying the data on the disk. I am, I

think, a sophisticated user, not a programmer, and will not recognize whether or not something "special" is happening when my programs are accessing the directory.

For example, I am using Powersoft's Powerscript mods to Scripsit for about two thirds of my work. It creates a directory. If I move to 6.3 and 5.3 and find the internal directory functions won't work I will be very sad but not angry with you. I would prefer not finding out by destroying a days worth of work in the process. That is all.

(Fm: MISOSYS) I would venture to say that programs which just access the directory and use the information should "run", albeit not be able to use the extended dating. Also, programs which worked like the DOS in testing whether a program is password protected would also give erroneous results. Be cautious of programs which do powerful things and may need to intertwine with the DOS (like ZSHELL and maybe DOCONFIG). When in doubt, either ask a question about a specific program or test it out on data which can be destroyed. In general, leave your hard drive off when testing something - or write protect it.

DIR (A=N) patch

(Fm: Jim Gaffney) Would it be possible for us "po folk" to get a patch to make (A=NO) the default -- like the one that came with 5.1.4? I'm referring, of course, to the DIR command.

(Fm: MISOSYS) Gary Shanafelt advised me that an appropriate patch would be: "PATCH SYS6/SYS.SYSTEM (D08,92=00 00:F08,92=FF FF)". I have not checked this out at all, so you should double check it.

README/TXT corrections

(Fm: Jeff Beadles) I think that you better change the "README/TXT" file ASAP. Take a look at your directions on how to make a

DS BOOT disk. You have a FORMAT, BACKUP SYS0, and then "BACKUP :S :D". Shouldn't that be "BACKUP :S :D (S,I)"?

Also, this version refuses to allow me to set my HD as the SYSTEM drive. "SYSTEM (DRIVE=7,DRIVER="TRSHD5")"... "SYSTEM (SYSTEM=7)". When I do that, it appears fine. Then, when I try to execute a command, say "DIR 0", it gives me an error message about RS232T (which isn't even loaded!). Also, then I tried to format the HD. I went through the FORMAT command, (after re-booting with a floppy 0 SYSTEM disk), and loading in TRSHD5. The format program said "destination is a hard drive... are you sure?" I replied YES, and it said "formatting cyl 00" (something like that). Anyway, it then returned a "device not present" error. Funny, I can get a dir of it. Any ideas?

(Fm MISOSYS) The README/TXT file has already been updated to reflect what you said.

My suspicion about your problem is that your backup to drive 7 resulted in SYS6/SYS going to three extents. SYS6 is a little bigger now. That may be your problem. The "strange" behavior exhibited may be explained by unusable LIB files. Both SYS6 and SYS7 must occupy no more than 2 extents. Check that out and get back to me.

Double-sided boot disk

(Fm: Ted Pinkert) Could you amplify on how to make a double-sided BOOTable system disk under LDOS 5.3? I have read your README/TXT file, and tried to do what it said, but I cannot seem to create a bootable disk by using this technique. For the record, my README file says to issue the following commands:

```
BACKUP SYS0/SYS:s :d (SYS)
BACKUP :s :d
```

What I don't understand is how the first command can have any effect, since the second command would seem to replace ALL allocated tracks or granules on the destination disk, and therefore wipe out

whatever was accomplished by doing the first command. At any rate, it does NOT seem to work.

(Fm: MISOSYS) The second command line should read: BACKUP :s :d (S,I). See my previous response.

NEWCLOCK and x.3

(Fm: Richard Buzzell) I hate to bother you since I can see that you are VERY busy. BUT I have been trying to get my two new operating systems set up on my mod 4 to no avail. The problems I am having are: (1). My newclock will not work on either LDOS 5.3 or 6.3 on boot up. (2) cannot change the dir command NOT to have the A command on in LDOS 5.3 3. When the HD is the system drive a device command gives a system error under LS-DOS 6.3. If you could help me out I would be GREATLY thankful.

(Fm: MISOSYS) Since the system initialization part of both LDOS 5.3 as well as LS-DOS 6.3 has changed, whomever generated patches for the NEWCLOCK needs to take another look at it again. The patch to 5.3's DIR command to default the Aparm to OFF is buried in this section. If the HD as the SYSTEM drive gives a SYSTEM ERROR when you do a device command, check for a defective SYS6/SYS file. If DIR gives no error, make sure that SYS6/SYS is NOT taking 3 extents (a DIR command will tell you). If it is, then purge it and try to BACKUP SYS6 again. It's got to fit in TWO extents. If that's not the trouble, I can't think of anything else.

(Fm: Richard Buzzell) I recently [developed] a patch and have uploaded it here. Look for it soon. Uploaded as NEWCLK63/FIX.

SYSTEM (SYSTEM=d) during JCL

(Fm: MISOSYS) Invoking a SYSTEM (SYSTEM=d) command under 5.3 under JCL when the JCL file is on either the system drive or the drive being switched will now properly

continue the fetching of keyin text from the JCL file. You still can't SYSGEN while JCL is active. Too bad LSI couldn't add it to 6.3 due to timeframe.

LDOS 5.1.4

(Fm: William Chao) Will someone who knows the answer to the following question please send me a reply: I just purchased a LDOS manual from Radio Shack (disk missing) at a huge discount. It seems the manual covers LDOS 5.1.3 and that's it. If I buy 5.1.4 disk from MISOSYS, will I receive the additional info on changes from 5.1.3 to 5.1.4?

(Fm: Gary Phillips) The latest MISOSYS catalog indicates that you can get documentation updates from 5.1.3 to 5.1.4 for \$2. If you need a disk anyway, you might as well spring for LDOS 5.3 rather than 5.1.4, at \$24.95 plus \$2 for shipping. Add in the extra \$2 for the documentation update, and file the 5.1.4 update before inserting the 5.3 information that comes with the diskette, and you should be in good shape. By the way, LDOS 5.3 is worth every penny ten times over, though I'm sure it would never sell at \$249.

The future of LDOS

(Fm: Ted Pinkert) I think it would be great, now that Tandy has cut loose LSI, if LS-DOS and LDOS would evolve more and more toward total compatibility. I think it would require even more cooperation between LSI and MISOSYS (meaning more work, I guess). But it might be worth it in the long run. I don't think it would necessitate any merging from a business standpoint.

There sure seems to be a lot of confusion between LS-DOS and LDOS in the minds of the users who stop by the SIG to find out about the upgrades. I can even spot you insiders making "slips of the fingers" and getting them confused (rarely).

So how about it? What is the Five Year Plan with you folks? Or even the One Year Plan?

(Fm: MISOSYS) LDOS 5.3 was released on January 5, 1987. One year plan? Come on, since when would you expect another release in the mill before a year is up. I'll give you one clue for the "master" plan. The system date is now supported up through 12/31/99. Need I say more? It's going to take at least 15 months before I will even see the results of the buying public. Right now we've shipped about 400 [750 by mid-March] 5.3s. LSI has probably shipped between 4000 and 6000; but then they've been advertising for two more months and I put their market at about ten times the Model III mode market. We expect 5000 sales of the 5.3 upgrade kit. Come back and ask me my five year plan after we ship 5000 packages.

(Fm: Ted Pinkert) I have no doubt that's true, but IF there is ever a 6.4/5.4 or whatever, it would be a good opportunity to smooth out the last little variances.

(Fm: MISOSYS) No DOS will ever have everything in it that folks want. We strive to compromise by putting in enough to satisfy most folks in the time allotted to development. Were we to continue to put in features, the product would never get out the door. As it is, I am most satisfied with the job that was done here with 5.3; although there are always going to be some folks who will take me to task for one thing or another (e.g., "I love the new INPUT@ in BASIC but why did you have to make BASIC bigger?").

(Fm: LDOS Support) Other than the BASIC, which will always suffer from the differences between MicroSoft level four and level five implementations, and the differences caused by hardware (memory map, library structure, overlay area size), the commands could be somewhat improved as far as standardization. There are limitations, such as the library command table is full on LDOS, there would be no room to move SYSGEN out as a

separate library command outside of SYSTEM as it is on T6, for example.

(Fm: Ted Pinkert) Well, some things are best taken one day at a time. It's truly none of my beeswax. The history of this enterprise must make some story, though.

As to 12/31/99, I fully expect to continue hacking on this monster until long after that, should I live so long. By then, you'll probably be writing DOS's for 256-bitters. I'd better start stocking up on M4 motherboards (I already "stocked up" on power supplies, heh). To get to the point, the main thing that makes using the computer enjoyable to me is LDOS/LS-DOS. I can do no more than say THANKS for your role in creating it. Please try to remember that the next time I come up with some nitpicky bug.

SYSTEM (FAST) and the 4P MODELA/III

(Fm: Gary Phillips) Running LDOS 5.1.4 and 5.3 on a model 4P, I'm getting very confused about SYSTEM (FAST) and the software clock. I thought it used to be that when I booted 5.1.4 on the 4P, it recognized that I had a model 4 and set SYSTEM (FAST) automatically, which was fine only then the clock was never right. Since there was no timestamping, I turned off the Time prompting and let it go.

Now with timestamping added in 5.3 I understand you also did something to fix the clock so it would be right on a model 4 system running with SYSTEM (FAST)...at least I think that's what I understood. BUT, it appears that the new ROM image that came out with LS-DOS 6.3 ALSO does something to fix the clock problem.

As near as I can tell, if I boot LDOS 5.3 with the new ROM image, it defaults to SYSTEM (SLOW). If I switch to SYSTEM (FAST) the clock runs at half-speed! Obviously this is not a problem, as long as I remember to use the right ROM image. What all this is leading up to, really, is a question about disk formatting.

I have been using LDOS 5.1.4 on a model 3 to operate a BBS. Last week, (right while I was installing 5.3 ... hmmm!) the model 3 croaked. I am now running the system on a borrowed model 4P while attempting to decide whether I should fix the 3 or buy the 4P to run with. The BBS software is still running under 5.1.4 and I first started it up with the old ROM image, but of course the clock wouldn't keep time. Rather than give up the faster CPU, I switched ROM images to the new one. Now the clock is right, but disk I/O seems slower than it was on the model 3 (of course, with the model 3 lying here in 20 pieces, that is a subjective statement). Anyway, my question: is the skew factor in LDOS optimized for a 2 MHz clock? If so, will I get better performance by formatting the diskettes for the BBS under LS-DOS? Or is that inviting trouble? And is there any way to tell whether you have SYSTEM (FAST) set under LDOS 5.1.4?

(Fm: MISOSYS) First, let's set the record straight. LDOS 5.1.4 did no checking to see if it was running on a 4 so it could set FAST. It was hard patched into the SYSO init code and ALWAYS came up FAST. Of course the timer ran amuk since it used the ROM routine for time keeping.

When I rewrote the DOS for 5.3, I fixed the time operation by putting in a time slice routine. 5.3 comes up in SLOW since its designed for a Model III! It's very easy to change by setting SYSTEM (FAST,SYSGEN). Do it.

Next, let's clear up the terminology. The physical sequence of sectors on a track is termed "interleave". The "skew" factor relates to the sector offset as you go from track to track. Most folks don't even know about that.

For instance, if the first sector on track 0 is sector 0 and the skew is 3, then the first sector on track 1 is 3. The sector skew is adjusted so that the time taken up to step in one track is the amount of time it should take the disk to revolve from the end of the last logical sector of one track to the beginning of the first logical sector on the next track in. Clear?

The FORMAT interleave under LDOS 5.3's FORMAT utility is optimum for 2 MHz clocks only. FORMAT has no adjustment. The FORMAT under LS-DOS 6.x is designed for optimum at 4 MHz; there is no adjustment. QFB as supplied on LDOS 5.3 adjusts its interleave for 5-1/4" drives to be optimum for SLOW or FAST (2 or 4). Any disk formatted under 6.x and read under 5.3 in SLOW will be VERY SLOW. Likewise, a disk formatted under 5.3 will be SLOW to read under 6.x.

It was too late in the design cycle to add the additional code to 5.3's FORMAT. If you really need that kind of performance, use QFB to dupe a few spare disks and use them - don't use FORMAT (that assumes you will be using FAST).

On the MODELA/III issue, it has been noted that the release on 6.3 has other errors and the older one should be used. Certainly it's possible that Tandy adjusted the ROM image routine for timekeeping - but why would they? TRSDOS 1.3 doesn't use FAST! We have been told that that is what they did as well as alter the ROM keyboard driver.

TED/CMD

(Fm: Shane Dawalt) I found a small problem with TED1.1. TED doesn't know what to do when the BREAK key is pressed with the Insert Mode on. If this happens, the cursor remains as the Insert mode cursor, but anything entered will overstrike the text. Pressing the <CTRL><A> keys will not change the cursor, but will put it back into the Insert Mode. Pressing the <CTRL><A> keys again will change the cursor back to the standard overstrike mode cursor and everything works as advertised.

(Fm: MISOSYS) This problem is fixed up with TED5A/FIX. See THE PATCH CORNER.

Model III Cursor and CTL key in LDOS

(Fm: Ted Pinkert) As promised, here are a few small bug reports, mixed with some questions:

1. When a cursor character below X'20' is selected (SYSTEM (BLINK=xx)), the cursor display momentarily changes to a block cursor whenever <ENTER> is pressed.

2. On boot, the system seems to accept whatever is stored at DATE\$ even if it is invalid information (like when LS-DOS 6.3 was running just prior to booting LDOS 5.3). Whatever the reason, it has a tendency to display garbage for the date, rather than asking the operator for fresh input.

3. I can't get the insert-mode toggle to work in TED/CMD using CTRL-A (or SHIFT-DOWNARROW-A). It just stays in overwrite mode, and the cursor remains an X'5F'. The display does flicker just slightly when CTRL-A is pressed. All I have in memory are the standard modules: KI/DVR and PR/FLT and SVC.

4. When changing parameters using FORMS, all parameters must be re-entered each time on the command line. Any parameters which are not re-entered are reset back to their default values. This is contrary to LS-DOS/TRSDOS, and I don't like it as much (although you may have intended it as a "feature").

5. When FORMAT finishes and displays "Initializing System information", what is the meaning of the two plus signs, as in ".....++....."?

6. Thanks for internalizing the "MEMDIR" function into the OS! I used to have to put that little program on every one of my System disks! There are several other such features now in the OS (and BASIC) where they belong, and I really do appreciate the fine job you have done with this upgrade.

(Fm: MISOSYS) The BOOT detection of the saved date is not perfect. Switching between operating systems can certainly affect memory such that the 5.3 save area could look like a valid date.

Can't explain why CTL-A won't work for you in TED. Since everything is a constant, it puzzles me. One of my beta testers told me the same thing was happening to him. Never

could figure out what was his problem and it seemed to go away with another beta release. It never has happened to me. TED uses the flag register to detect the difference between a CTL-A and a BREAK - since they both return a value of 01H. You may try to write a little ASM program to get and display the results of a @KEY call with both CTL-A and BREAK response. Then get back to me with your results.

That's the way FORMS works. I note your opinion and will consider it an input if there ever is a 5.3.1.

The two '+' signs indicate the writing of the GAT and HIT sectors. The preceding periods indicate writing of BOOT sectors. The subsequent periods indicate writing of directory sectors. If there is an error while writing the system information, you now have a better idea what stage FORMAT is in. I should have mentioned it in the docs. Now I'll get 5000 questions about it; however, I didn't feel it significant to mention.

I tried to get LSI to put my MEMORY enhancement put into 6.3 but it was too late for them.

(Fm: Ted Pinkert) Here is some additional work on the cursor-below-X'20' problem. First, I should have said that the cursor momentarily displays the graphics block (X'BO') whenever ANY key is pressed, not just the Enter key. As noted, this only occurs when the cursor is set to a value less than X'20'. There are some other anomalies:

1. BLINK=0|1|255 displays nothing (I think I know why, and will be no fix).
2. BLINK=n followed by BLINK=OFF should set the cursor character, n, and then make it non-blinking. This works fine, except when the cursor character is X'02' - X'1F', in which case BLINK=OFF sets the cursor back to X'BO'.
3. Just an opinion, but I think X'84' would be much better than X'88' for BLINK, SMALL.

TED/CMD has the problem even under the cleanest possible conditions - booted my LDOS 5.3 master copy and installed KI/DVR which TED requires - that's it. Reminder: this is a Model 4 in Model III mode (I haven't bought a copy for the III which is gathering dust). Next, I hand-coded a short Z80 routine and tested key returns in Debug by displaying them on the CRT. All returns are as expected for the Model 4 character generator: Shift-DownArrow-A through Z produce the characters listed as l-26 in the manual. Ctrl-A just produces the letter "A". I noticed some screwyness with the Left/Right/shift and codes 27-30, but this is irrelevant here. I could not test Break, since this returns one to Debug via a background task.

(Fm: MISOSYS) Ted, Here's the reason why the blinking cursor will switch momentarily back to a block when you enter a key if the cursor character is in the range 00H-1FH. The ROM video driver has a routine at 49EH-4A7H which displays the cursor; however, if the value is below 20H, it displays a block (0B0H). It doesn't change the cursor character saved at address 4023H.

On the other hand, the interrupt task blinking routine in ROM at 354FH-3556H makes no test for the cursor character value. The interrupt routine would obviously take precedence as to the perceived cursor being displayed except when a character is displayed by a call to @DSP - which is the case whenever you enter a character.

Obviously Tandy wanted to restrict the cursor from being a control code value; however, they either neglected to check in the interrupt routine or forget to remove the check from the ROM routine.

(Fm: Ted Pinkert) Dumb explanation time. I was pressing RIGHTSHIFT-DOWNARROW-A which does indeed return keycode X'01'. Apparently, everyone else in the world uses LEFTSHIFT-DOWNARROW-A, and I am supposed to do so as well! Anyway, LEFTSHIFT works, and RIGHTSHIFT doesn't. So it was "my" dumb mistake.

However, the fact that it took an experienced computer user and downright hacker this long to figure this out seems to prove a point: DON'T write software which differentiates between the LEFT shift and the RIGHT shift! It is NOT FAIR to expect us "losers" to know the difference, and why SHOULD there be any difference? I don't know if this problem originates in the KI/DVR or in TED/CMD. But I do think it IS a problem and should be fixed. (Please don't tell me to turn to page xx in the manual where it says you must always use the left shift key.)

(Fm: MISOSYS) I am surprised that the RIGHT-SHIFT DOWNARROW with A generates a X'01' since only the LEFT-SHIFT DOWNARROW has been used for CTRL since day 1 of the Model III. Apparently, page 4-3 of the LDOS manual mentions <LEFT-SHIFT><DOWN-ARROW><*> for the screen print but omitted the "LEFT" in front of the word "SHIFT" on page 4-7 where it noted the abbreviations. I even recollect TRSDOS 1.3 as using only the LEFT shift key for CTRL functions. It has been so standardized, that I wouldn't have thought anyone expected RIGHTSHIFT-DOWNARROW for use as the CTRL key. Sorry.

(Fm: Ted Pinkert) When you said that, I first thought, "Am I going crazy - I've been using the RIGHTshift-downarrow since day one!" Then I thought, "Wait a minute, he said Model Three!" So I went downstairs, blew off the dust, and looked at my III. Lo and behold, the DOWNARROW key is right next to the LEFT shift key! And - on the Model 4 - the DOWNARROW is next to the RIGHT shift key.

I think that one's cerebellar cortex remembers the relationship between two keys, NOT whether the keys are on the right or the left. I'd bet that many other folks experienced this switching of SHIFT-DA when going from the III keyboard to the 4. I'd also bet that others were equally dismayed when trying to stretch LEFTshift-DA-*

I concede that I unconsciously switched from left to right shifts when I got the Model 4. I DON'T concede that RIGHTshift is NOT functional on the Model 4 in III

mode. Are you still using a Model III most of the time? How can you state "that I wouldn't have thought anyone expected RIGHT SHIFT DOWNARROW for use as the CTRL key." I use RIGHTshift downarrow ALL THE TIME in Modem80, LED, EDM, etc, etc, all of which use LDOS' KI/DVR! Not only do I personally use RIGHTshift-DA, but this combo is definitely more natural than LEFTshift-DA on the Model 4.

So why do some programs and functions, like TED and screen-print, use the LEFTshift only, while so many other programs allow RIGHT? Do all of the latter really use their own KI drivers (or their own ROM <hoho>)?

Please, let's get this resolved to everyone's satisfaction (or at least to MY satisfaction <<grin>>). How about fixing the %&\$#'"&%* LEFTshift-downarrow, too, while you're at it. Thanks in advance for your help and forbearance.

(Fm: MISOSYS) The difference is that the KI/DVR only sets the CTL flag when the LEFT-SHIFT DOWNARROW combo is depressed since that is what was supposed to be the CTL key. Any program which checks the flag register to differentiate a CTL key versus any other key combo which generates the same character code, will not interpret RIGHT SHIFT DOWN ARROW as CTL. For instance, the left arrow generates a character code of "8D" as does <LEFT-SHIFT-DOWNARROW-H>; however, only the latter sets the control flag. Sorry, but that's just how its always been. TED differentiates between CTL-A and BREAK which both generate a character l value.

(Fm: Gary Phillips) At least on a 4P, whether the RIGHT shift works with downarrow as a control key seems to depend on the ROM image version in use. With the ROM image driver, no distinction between right and left will occur with some versions of the image, while it does occur with other versions. I noticed this by chance, but since I seldom use RIGHT shift for CTRL in my (non-gate array) 4P, I didn't pursue it.

(Fm: MISOSYS) Funny about Tandy, isn't it. The use of LEFT SHIFT was the reason that they split the hardware between both SHIFT keys. It also makes it possible to detect the action of depressing both left and right keys simultaneously. If I didn't want PRO-WAM to be behaved, I would have used that as an activation operation since NO ONE under TRSDOS 6 uses it.

(Fm: Ted Pinkert) NOW things are coming clear! Thanks for the lucid explanation of both KI/DVR's habits and TED's (the program not the Pink). Now I suppose there are some compatibility reasons for you to keep the status quo with the left shift. Is there anything inherent in the system which would keep me from altering my personal copy of KI/DVR to accept both left and right shift keys as "control" keys? Would *KI come crumbling down if I start rooting around in there? It's a bitch to trace through that thing, but I've done it before. (Even have a yellowing disassembly of it. Trouble is, I never did understand more than about half of what was going on in there.)

(Fm: MISOSYS) Ted, I'll be honest with you about the SHIFT-LEFT ARROW combo. Since we never use a "new keyboard" machine under Model III mode, that control key combo never struck us as being stupid for a "new keyboard" machine. If you look at the KI/DVR code at about 53E3H, you will find a BIT 0,(HL) instruction. Change that to a BIT 1,(HL) instruction and you will change KI/DVR from checking the LEFT SHIFT key to the RIGHT SHIFT key. More of a patch would be necessary to have it accept both. An exact patch to change CTL to RIGHT-SHIFT-DOWNARROW is now in the README/TXT file for 5.3 (see the following).

Abbreviated README/TXT

Note: Don't forget that all files associated with disk BASIC use a filename of BASIC - not LBASIC. If you are upgrading from LDOS 5.1.4 or earlier, you may rename your old LBASIC files to BASIC. The password to use is ".BASIC". This is for LBASIC/CMD, LBASIC/OV1, LBASIC/OV2, and LBASIC/OV3.

Note: If you are using a Model 4 with clustered ARROWS which are adjacent to the <RIGHT-SHIFT> key, you can apply the following patch to have KI/DVR use <RIGHT-SHIFT><DOWN-ARROW> for use as the control key instead of <LEFT-SHIFT><DOWN-ARROW>:

PATCH KI/DVR.DRIVER (D01,EC=4E:F01,EC=46)

Note: BASIC will default to using a file extension of "/BAS" for program loading and saving. Please make sure your BASIC programs are named with this file extension or invoke BASIC with the command: BASIC (E=N)

Due to numerous requests concerning how to install 5.3 on a Radio Shack 4-head 5-Meg drive, we have added a JCL file onto this disk. The "INIT54/JCL" configures a 4-head 5-Meg drive using "TRSHD3/DCT". This JCL assumes the drive is to be partitioned as 4 logical drives (0, 1, 2, 3) and you have two floppies (4, 5). To install LDOS 5.3 using this JCL file, BOOT with a backup copy of 5.3 to which you have added a copy of your TRSHD3/DCT driver. Then issue the command: "DO = INIT54". Please read all of the documentation supplied with YOUR hard drive before calling us if you do NOT understand what to do. We are not responsible for your HARD DISK.

```
. JCL to reconfigure a 4-head
. 5-Meg R/S hard drive
SYSTEM (DRIVE=4,DRIVER="TRSHD3")
1
1
1
SYSTEM (DRIVE=5,DRIVER="TRSHD3")
1
1
2
SYSTEM (DRIVE=2,DRIVER="TRSHD3",DISABLE)
1
1
3
SYSTEM (DRIVE=3,DRIVER="TRSHD3",DISABLE)
1
1
4
BACKUP :0 :4 (S,I)
DATECONV :2
DATECONV :3
DATECONV :4
DATECONV :5
SYSTEM (SYSTEM=4)
```

SYSTEM (DRIVE=1,SWAP=5)

. From LDOS Ready,
 . type the following command line:
 .
 . SYSTEM (SYSGEN,DRIVE=4)
 .
 //EXIT

PRFLT/fix error and SET2RAM

(Fm: MISOSYS to Verne Wertman) After investigating your problem with SET2RAM installed and then attempting to use the PR/FLT filter, I have come to the conclusion that your problem is caused by two things. First, you will need to apply the PRFLTA/FIX which corrects an error in the XLATE parameter of PR/FLT. This bug caused two addresses in low memory to be corrupted - obviously not harmful unless low memory was RAM! We had already worked up that patch recently.

Now then, that still doesn't solve the entire problem which you were having. Seems that when a Model 4 is switched into that particular RAM mode, the printer status is no longer obtainable from the memory address of 37E8H. The ROM printer driver uses an IN from the printer port. The PR/FLT filter made use of the 37E8H address which was portable across both a Model I and a Model III. I have added PRFLT/fix to alter PR/FLT so it will also use the actual Z80 port.

If you experience any problem with other programs directly accessing the printer port, check whether they read port 0F8H or memory address 37E8H for the printer status. With SET2RAM, they must use the port.

(Fm: Verne E. Wertman) Thank you for your letter. My SET2RAM and PR/FLT works great now. Please be advised though that there is a typo error in your PRFLT/fix patch.

(Fm: MISOSYS) You're right. We caught that already. Seems that we had corrected the error during our testing of the patch but the FIX file did not get corrected. Unfortunately still was that the wrong patch file got applied to the MASTER disk

we use for duplicating. Thus, if your LDOS serial number is between TCOA0527 and TCOA0590 you need to apply the following fix:

PATCH PR/FLT.FILTER (D03,67=F8:F03,67=F0)

SYSGENning other configs

(Fm: Irwin B. Burton) Over the years I have purchased several MISOSYS products which I couldn't get along without! I have a question, and maybe you could answer it in the MISOSYS QUARTERLY as I am a subscriber to that.

Recently I was reading some material on DOS+ 3.5. It was talking about the fact that with DOS+ you can "sysgen" several configuration files under different names and then call up the one you want by its name. I wondered if this might work for the LDOS family.

So I tried it on LS-DOS 6.3 by taking several 6.3 configuration files I have and copying them out to another diskette under different names. It seemed to work fine, I could change the configuration just by calling a different file. I then tried the same thing on LDOS 5.3. It didn't work. It hung up and wouldn't return to LDOS Ready without a re-boot. I'm sure there is something in the way 5.3 handles configuration as opposed to 6.3 that causes this.

Is there an easy explanation for this? And is there any way to change 5.3 to allow for this kind of operation?

(Fm: MISOSYS) You were just plain lucky to get it to work on 6.3. Actually, it really didn't work. Here's why. The CONFIG file which is generated by the DOS contains an image of what's in high memory (from HIGH\$+1 to OFFFHH). It also contains an image of various blocks of system memory (e.g. the device table, the interrupt vectors, the SVC table, odds and ends). It also has an image of the drive control table. However, this table image doesn't load into the DCT region when the CONFIG file is loaded. That's because it's possible that the new configuration

overwrites a high-memory module being used by the disk driver or changes the contents of the drive-0 DCT slot. That must not be allowed to happen.

There is a small piece of code executed during the booting process which loads the CONFIG file then moves the loaded DCT image into the DCT memory location. Another job done by this boot stub is to call the @ICNFG vector. That vector is used to chain through any memory module which needs some initialization. For instance, the RS-232 driver needs to be primed. Most hard disk drivers get primed. ZSHELL is tied into @ICNFG so it engages its hooks into the system. I am sure there are many more examples to illustrate.

Now if you just invoke the config file name, as if it were a CMD file, various things could happen. First, the transfer address for a 6.x or 5.x CONFIG/SYS file is X'0000'. Thus transfer would be to address 0. The results are different on Model III versus Model 4 when you go to address 0. If you LOADED the file, the result would depend on how different the "loaded" config file was from the current configuration. Note that the DCT would not change in either 5.3 or 6.3 nor would the @ICNFG vector be "stroked". You see then, it really didn't "work" for 6.3 either!

The solution? Well you should have read some dialog here about our DOCONFIG product. This does exactly what you want to accomplish - in most cases. Some installed modules can only be disconnected by either their own (OFF) facility or be a re-boot. DOCONFIG does have some very good capabilities. Configurations can be changed WHILE YOU ARE RUNNING JCL. You can even save the state of the machine while you are running ANY program which allows you a DOS library call (i.e. BASIC) and return to that program at some future point (sort of like a snapshot). DOCONFIG is part of our MSP-01 (Model III) or PROGENY (Model 4) packages.

@PRT documentation error

Bill McQueen could not get the printer status when he used a method documented in the LDOS manual on page 6-66. The manual

states for @PRT, "This routine will output a byte to the printer. If a zero value is passed, then the printer status will be returned. This method is recommended over checking the port directly." The statement is only correct for a Model I with or without the printer filter installed. Under both Model I and Model III, the best way to obtain printer status is by an @CTL call as follows:

```
LD   DE,PRDCB$
XOR  A
CALL @CTL
```

After the return from @CTL, the "Z" flag has the status. "Z" means the printer is available; "NZ" means it is not.

DUAL - and other LINKs

(Fm: Patrick H. Larkin) Now here's a question I'll bet you haven't run into before. Is there a way in LDOS 5.x to LINK *DO with *PR from within BASIC to LPRINT everything being PRINTED in a manner similar to the TRSDOS 1.3 command "DUAL"? I have tried to LINK with the CMD"dos command" command, and was refused. I have a program in TRSDOS 1.3 BASIC that I have tried to convert to LDOS, and am unable to get thew *DO and *PR to both display at the same time unless I LINK at LDOS Ready, and I am trying to avoid displaying preliminary dialogue (input prompts and responses) necessary to get the printed output - which I would like to see on the screen at the same time. Any suggestions?

(Fm: MISOSYS) Sure. Here's the easy solution. At LDOS Ready, type the commands,

```
ROUTE *SO (NIL)
LINK *DO *SO
```

After that, you can ROUTE *SO to *PR to turn "dual" ON and RESET *SO to turn "dual" off. The following little BASIC program will demonstrate this.

```
10 PRINT"This is not dual'd"
20 CMD"route *so *pr"
30 PRINT"This is printed as well"
40 CMD"reset *so"
```

```
50 PRINT"This is not printed"
60 END
```

@DODIR and x.3

Anyone who has really scrutinized the documentation covering the 5.3 release and LSI's 6.3 release may have wondered about changes to @DODIR since there's no documentation covering any changes to this vector (nor were there any changes made).

One of DODIR's functions is documented as returning the first 16-bytes of each directory record followed by the ERN. Certainly that information included the date field. However, the x.3 directory record contains an extended field not part of those 18 bytes. Now if DODIR was changed to include the new information, any existing program which used DODIR would not work.

Therefore, LSI and MISOSYS agreed to not change the behavior of @DODIR - even though it doesn't return "useful" date information. At least it won't affect the operation of existing programs. Programs which use @DODIR to retrieve and use date information will have to be changed anyway. Thus, it is recommended that in those circumstances, the programs be changed to either read the DIR/SYS file or directly read the disk.

UN-DATECONV

(Fm: Kenyon F. Karl) Because I run the public domain library for the Tandy - Radio Shack User Group of the Boston Computer Society, I will have a requirement to convert disks from LDOS 5.3 back to the 5.1.x format. If DATECONV only works in one direction, I would appreciate whatever help you can provide for my requirements. I do have TMQ I.II and I do write Z80 assembly language.

(Fm: MISOSYS) My PPN is 70140,310; but don't send me EMAIL. Log onto PCS-49 and leave me a message saved as private.

Now for UN-DATECONV, use the information on the revised directory structure. Then do the reverse process. (1) Reset bit 3 of GAT+OCDH. This turns off the "extended date" flag. (2) Shove an encoded password into DIR+18/19. Suggest you use the disk master password (GAT+OCEH/OCFH), the file password (in DIR+16/17 of each record), the 2-byte value "EO 42", a password of "PASSWORD", or re-ATTRIB each file. It shouldn't take too much to generate such a program. We'll leave it as an exercise and print any that are submitted to the next QUARTERLY.

Where's the missing file?

(Fm: Tony Sowers) DIR to read your 5.3.0 disk says files = 79/128 implying 49 files in use. In fact, only 48 files are on the DIR (S,I) display. Is this an error or am I missing something?

(Fm: MISOSYS) We just can't hide anything, can we? Actually, the explanation is easy. Although we didn't have room in 5.3's SYS1 to put in an Extended Command Interpreter (ECI) interface to SYS13/SYS, we decided to reserve the SYS13 Hash Index Table slot on SYSTEM disks. This reservation is not done to DATA disks. The algorithm used in DIR and FREE to calculate that header counts up the valid HIT entries. It adds one for SYS13. The directory record slot for SYS13 has no entry; thus, a SYS13 does not show up in the listing.

If we ever come up with a method to add an ECI, it will be easier for us to "patch" it in by keeping that SYS13 slot "activated". I don't think you'll miss it.

FEDding the DIR/SYS file

Ever try to use FED 1.0 on the DIR/SYS file? You can, you know. Just give it the directory password of "SYSTEM". Of course if the disk you are FEDding was formatted by TRSDOS 6, the password is not "SYSTEM", it's "LSIDOS". An easy way to double check is look at the password field of DIR/SYS (using FED). The field will be "AE 01" for "SYSTEM" and "F6 37" for "LSIDOS".

Timekeeping with 50Hz

(Fm: K.W. Arntsen) I have not had time to try out all the new developments but I can report one bug. The clock keeps nowhere near real time under our 50Hz supply. Unlike LS-DOS 6.3, there is not even a utility which attempts to correct this problem. (I say attempts, because LS-DOS is equally inaccurate even after using the supplied utility.)

Now that all files are date and time-stamped it seems a shame that one cannot keep an accurate clock outside of the U.S. For your information the TRS-80 counts almost exactly 50 seconds for every 60 seconds elapsed and does not take long to produce an almost meaningless time.

(Fm: MISOSYS) The timing of the Model III and the Model 4 is derived from the real time clock interrupts. The interrupts are hardware oriented and based on the AC line frequency; thus, the Model III interrupts only 25 times per second at a line frequency of 50 Hz versus 30 times per second at a line frequency of 60 Hz. In the Model III, the seconds are ticked off by means of a software countdown timer located in the Model III ROM! Thus, in the Model III mode, one would think that Tandy would provide a proper ROM for machines destined to markets where 50Hz power is in use.

Since the timer is coded in the ROM, the only way around it is by means of a kludge routine. The timer value counts down from 30. The value is stored at address 4216H. What is needed is an interrupt routine that speeds up the counting. If you write a little interrupt task that checks that value, you can speed up the count and arrive at a more accurate clock. Just have the routine look for a value of 30. When it's 30, change it to 25 - the value needed for 50 Hz line frequency.

Another solution, if your running under 5.3 on a Model 4, is to obtain the Model 4 Hardware Interface Kit from us. Use the SET2RAM module to switch your Model 4 over to the RAM mode. Then change the value at address 355DH from 30 (the initial value of the countdown timer) to 25. Then you

would need no interrupt routine to kludge a correctiuon.

As this problem is obviously experienced by others, perhaps we will cover such an interrupt routine in the next THE MISOSYS QUARTERLY. Now if anyone has already written it, submit it to us to save us the time.

Where'd @ABORT go?

(Fm: Robert F. Winn) My question is about LDOS 5.3. I do a lot of assembly language programming for my applications and I noticed by accident that the vector @EXIT and @ABORT are identical in this release of LDOS and in earlier versions they were different. Is this change correct or has a bug slipped in. I would like to think that it is correct but I thought, ask to be on the safe side.

It has vector @EXIT = 3E 93 EF and vector @ABORT = 3E 93 EF, both of them are the same. I believe earlier versions had vector @ABORT = 3E A3 EF. What do you think? By the way, I am very pleased with 5.3 and the changes you made. You deserve a job well done and a very big thank you.

(Fm: MISOSYS) My, you are an astute observer. Prior to LDOS 5.3, @EXIT and @ABORT were two distinct entry points into SYS1/SYS; however, both entries within SYS1 jumped to the same code. This meant that although external to SYS1, both @EXIT and @ABORT were distinct; internal to SYS1, the handling routines were common. Everything in the DOS which interacted with those two service calls went through the two vectors. Thus, the separate entries into SYS1 were redundant. This redundancy was carried over from the early days; there was no necessity to eliminate it.

With the revisions necessary to SYS1 for the LDOS 5.3 release, the redundant entry of @EXIT and @ABORT were a luxury which could no longer be retained; I needed the four bytes! Everything using @EXIT and @ABORT will work just as it had in the past.

Miscellaneous problems with 5.3

(Fm: Flt Lt A W Gransden RAF) The new features are excellent - especially the directory date field. Unfortunately, I have identified a few problem areas which I list below for your attention.

1. There appears to be an omission in PR/FLT which prevents the XLATE option from being used with the command FILTER *PR PR (XLATE=X'1B07'). [Andrew went on to provide detailed information and included a patch. That's PRFLTA/FIX in THE PATCH CORNER].

2. As a result of the redesign of PR/FLT, LSCRPT/CMD will not function with the filter without the following patch: LSCRPT53/FIX - Patch to LSCRPT/CMD - 25 Jan 87. This ensures that SCRIPSIT/CMD patched with the LSCRPT patch 4 (published in LDOS Quarterly Vol 2, No 1, Pg 47) to operate correctly with PR/FLT Version 5.3.0. The patch alters the index addressing of the filter parameter block 'number of characters per line' and 'margin' values. These are saved on entry to LSCRPT and restored on exit. This patch was written by Andrew Gransden, currently residing in West Germany.

.LSCRPT53/FIX

D12,66=17:F12,66=19

D12,6C=17:F12,6C=19

D12,70=18:F12,70=1A

D12,76=18:F12,76=1A

D10,F4=17:F10,F4=19

D10,FA=18:F10,FA=1A

. Eop

3. DOCONFIG/CMD fails to work under 5.3 [this is discussed elsewhere in this issue].

4. FORMS, a potentially useful library command, has the annoying habit of resetting the unspecified option parameters to the default values. This requires the complete command to be retyped if a single parameter was incorrectly entered. Was this intended?

Additionally, FORMS fails to check for a RESET *PR (e.g. PR/FLT installed and then set non active) and displays a nonsensical parameter list. The PR DCB parameters are

reset to default and the previous PR/FLT parameters remain unchanged. The latter are displayed but have no meaning. This should be corrected.

5. The command LIST TEST/CMD:1 (HEX) causes the top line to scroll off the screen if the last page contains 12 displayable lines.

6. The command KILL *SI gives an incorrect display and aborts.

7. The DEBUG display corrupts if the alternative character set is active. This hinders analysis of configurations using this alternative set.

8. I am aware that the 5.3 update notes say high memory modules must abide by the protocol convention. However, MEMORY corrupts when a trace is lost unlike MEMDIR/CMD which reports the error and returns gracefully.

(Fm: MISOSYS) You are absolutely right about PR/FLT; that was a bug which was caused by the revision in addressing the parameter list within the filter. My PRFLTA/FIX as shown on the list of fixes is a correct fix.

Thanks for your LSCRPT53/FIX; I'll try to get it into the next THE MISOSYS QUARTERLY.

A patch to fix up DOCONFIG for LDOS 5.3 is included in the list which is enclosed. There are also fixes for ZSHELL as well as a few Model 4 products. We are aware of problems with other products; however, we can only attend to them in time. Be patient. THE MISOSYS QUARTERLY will convey whatever patches we have available.

Sorry you don't like what we did in FORMS. There is a limit as to just how much was able to be done with the DOS. For \$24.95, we believe we gave about \$100 worth of value. FORMS works as intended. A "RESET *PR"? I hate to say, "big deal". This is not something which needs "correction". You may like its behaviour changed; however, it doesn't mean that it's broken or unusable. I suggest that you either

live with it or don't use FORMS. After all, you can set your parameters just like you did before - when you install PR/FLT!

You always have use of the PAUSE key for LIST.

Your bug report concerning "KILL *SI" has been investigated. SYS6C/FIX corrects this.

DEBUG always behaved as you described when the "special" character set was active (not the "alternate" as you described). This is because DEBUG uses space compression codes for screen formatting. The SYS5 DEBUG module just has no room for any change to move away from space compression codes. It's just something you'll have to live with.

I find it ironic that you claim "the LDOS 5.3 MEMORY command corrupts when a trace is lost unlike MEMDIR which reports the error and returns gracefully". My reasoning is that the MEMORY command uses the identical code as MEMDIR. I took a look at sampling a lost trace. Maybe you didn't use identical circumstances when you were testing your claim. What I found was that the algorithm for recovery rejected a "possible module" if the module name had any character which wasn't an ALPHABETIC! Under LDOS 5.3, all of the system-provided modules use a DOLLAR SIGN as the first character. Of course, the trace recovery routine would reject such a module name and continue on. MEMDIR, obviously, has that flaw, too. I coded a fix (SYS6C/FIX) to have MEMORY accept as valid, strings containing values from 21H-7BH. It may then accept a false "module" on rare occasions; I don't think you will complain.

Defaulting FORMAT to SIDES=2

(Fm: Gordon Collins) The following is written with the admission that I do not understand why it works, but if you can make something of it, please publish it to help others. I have 80 track, double sided drives and if I used FORMAT :n (Q=N), it did not work as I expected (format an 80 track, double sided disk) under LDOS 5.1 or TRSDOS 6.2. As you will see from the

attached screen dump of FORMAT/CMD, I have zapped four bytes as marked. This works with both LDOS 5.1.4 and TRSDOS 6.2.0.

(Fm: MISOSYS) [Gordon's screen dump zapped the first appearance of "FD CB 04 AE" to "00 00 00 00"]. Gordon's zap will work providing the drive slot is already logged for two-sided operation (i.e. a DEVICE command shows SIDES=2). The zap eliminates the code which sets up the DCT for single-sided before checking parameter entries. Thus, the number of sides being formatted would be whatever was left in the DCT; not a totally reliable operation.

Here's a better way. If you change FORMAT's handling of sides so it initializes to sides=2 and changes it to sides=1 if the parameter entry was (SIDES=1), then FORMAT will always do 2-sided unless told otherwise. Here's the patch for both LDOS 5.3 and LS-DOS 6.3.

```
. FORMAT53/FIX
D09,D6=EE:F09,D6=AE
DOA,49=01:F0A,49=02
DOA,4F=AE:F0A,4F=EE
. Eop
. FORMAT63/FIX
DOA,8A=EE;FOA,8A=AE
DOB,06=01;FOB,06=02
DOB,0C=AE;FOB,0C=EE
. Eop
```

6.3 users, note that the statement separator is a semicolon; a colon is used for 5.3.

LS-DOS Information

TOP OF FORM FEEDS

(Fm: David R. Reynolds) On my Model 4 in Model 3 mode, I can input LPRINT CHR\$(12) and my print will advance the paper to the top of the next sheet. If I do this in Model 4 BASIC mode, the printer does not advance the paper to the top of the next sheet. Can you tell me why? What can I do to make this work?

(Fm: MISOSYS) You need to install the FORMS filter. The Model III ROM printer driver has a routine to expand a CHR\$(12) to a series of line feeds. The Model 4 printer driver does not alter any code so the driver will work with all sorts of graphic output. However, if YOUR printer does not accept a HARD form feed, you need to install the FORMS filter to provide a SOFT form feed. Its easy with the two commands:

```
SET *FF FORMS
FILTER *PR *FF
```

and you will be off and running. The FORMS command also will display the parameters of the installed filter.

(Fm: LDOS Support) Roy's answer is right on the button! You may want to make the command LPRINT CHR\$(12); instead of just LPRINT CHR\$(12). Without the semicolon, you can have a problem with some programming techniques and some printers that will result in the first page lining up fine, but the second page will be off by one line. Subsequent pages will match page two.

Stacking Commands

(Fm: Scott Russell) I was wondering if you know of anyway to stack 2 or more commands from a do file. The reason I ask is, I was trying to make a ram disk the system disk and then remove overlays that were SYSRES'ed and the SYS0/SYS file from an AUTO'ed DO file. However, when it issues, "SYSTEM (SYSTEM=2" the job aborts. I know it makes sense since the system has

switched logical drives but, It would be nice to execute another DO command automatically. I was also thinking, would it be possible to stack multi commands via @CHNIO from a filter? And could you call the filter from program?

(Fm: MISOSYS) If you're using TRSDOS 6.2, then you could benefit from our SWAP utility which has been part of PRO-GENY for about three-four years. Want to buy it? If you're using LDOS 5.1.4, then I included the SWAP facility within the SYSTEM command of 5.3.

6.2.x *PR timeout

(Fm: Clifford I. Knight) I have an Olympia RO daisy wheel connected to a M4P... problem is that when underlining an entire line the 256 byte buffer in the printer fills and the DOS printer driver times out ("Device NOT available") and either characters are dropped when dumping from ScripAid (my patched ScripSit) or a return to "Ready" is made (when sending a "routed" file to the printer). Has anyone else experienced this? I am not wild about re-inveting the wheel, and thought I'd ask before disassembling DOS.

(Fm: bob snapp) Try using the spooler with a k or so of memory buffer.

(Fm: LDOS Support) The suggestion of adding the spooler is a vaild one. In addition, I developed patches to extend the printer timeout some time ago, and they are here in DL6. Try a BRO /KEY:*PRINTER* on our Compuserve forum.

TRSDOS 6.2 and hard drive

(Fm: Bruce Travers) I have this weird problem when my hard disk is write protected and I try to open a file. The open does not work and the reason given is that the disk is write protected. The strange thing is this open is for R/O so I don't think it should make a difference. IE if my disk is write protected, scripsit will not load a file properly, it is all

gibberish. This also happens in other programs where all it is supposedly doing is reading the file. (Like the DTERM freeware package, it won't work if my disk is write protected!) COMMENTS?

(Fm: bob snapp) The open sets a flag in the directory entry. Just forget about w/p on the drive, and all will be just fine. Use SOFTWARE w/p if necessary.

(Fm: LDOS Support) The program in question may not be able/know how to open the file for read only access. For example, in SuperSCRIPSIT or SCRIPSIT, there is no way to specify that the file to be opened is not going to be changed.

When TRSDOS 6 opens a file for r/w access on a logical drive, and the drive is write protected, an open error is returned to specify that the program only got read-only access to that file. Whether or not a particular program knows how to handle that status info is another matter.

BASIC's eof() problem

(Fm: Bob Blackburn) I refer to the EOF problem noted in the following excerpt: "ERRORS AND COMMENTS ABOUT TRSDOS 6.x. Errors in RS Manuals by Adam Rubin [74126,2244], January 19, 1986. Some additions, formatting and editorial comment by Joe Kyle-DiPietropaolo, LDOS Support [76703,437], January 28, 1986. 2-90 EOF: Unfortunately, EOF is FALSE if positioned past the end-of-file. For direct access files, it is safer to compare LOC and LOF."

(Fm: bob snapp) EOF was not intended to be used with direct files, only with sequential files. For the direct files, it is normally for i = 1 to lof(fn):get n,i:next. for sequential files, the coding is usually something like: if eof(n) then close n line input #n,xx\$. Anyone using EOF with a direct file was corrupting its intended purpose. As far as I am concerned, it should be 'undefined', and / or return an error code....

(Fm: Shane Dawalt) I think you are talking about BASIC. BASIC has it's own little file handling routine which is not the DOSes problem. I have my doubts that RS will force Microsoft to change what you are talking about, although I do agree with what you are saying. An idea might be to define a function which returns logical value of the comparison. Something like:

```
DEF FNEOFCHK()=(LOC=LOF)
```

This would return a -1 if true or a 0 if false. Then, wherever you needed to check the EOF status in your program, you could use: IF FNEOFCHK() THEN GOTO "EOFHANDLER" ELSE "CONTINUE". It is assumed that you will replace "EOFHANDLER" and "CONTINUE" with the appropriate line numbers.

This is only an idea. It seems like it should work although I have never tried it out. I'd lay odds, however, that it works fine.

Model II LS-DOS 6.2

(Fm: Marc Nowell) Do you have a list of "misbehaved" Model 4 software that WON'T work under LS-DOS 6.2? I'd like to use Profile 4+, but I can imagine that it'll crash quick. Any others? Also, how would a reasonable person port their Model 4 software over to a Model II/12, if it doesn't support 5.25" external drives?

(Fm: MISOSYS) I don't have such a list since I hardly use much outside of MISOSYS products. Tops on the list would be SuperScripts. Next would be LeScript. Double Duty would be out. Our PRO-CURE would be out.

To move files over to the Model II could be accomplished in one of two ways. (1) Use a direct connect RS232 cable and xfer via a comm protocol. Make sure you SETCOM (BREAK=0)!!! (2) Find someone using a MAX-80 with both 5's and 8's. Aside from us, there are a few around. We will as a courtesy, move any files over to the LS-DOS system disk (or your provided 8" disks) when you order an LS-DOS (II) if you provide "MASTER" copies of the 5-1/4" source disks.

(Fm: Ward Griffiths) I received my copy of LS-DOS 6.2 for the Mod II last week, and finally am putting it through a decent test. Not having a compatible HD controller on my 16 (lease return, \$300 with 8meg board), I yesterday hooked the 5meg from my 4p to the 6000 at the office and copied most of the files onto 8inch floppies. Initial tests tell me that I've got some work ahead of me, and must get at least volume 3 of the Source code. Programs that do not work are 1) any REQUIRING the Mod 4 128k, such as DoubleDuty and memdisk, and 2) any which go around the system screen, printer and keyboard drivers. Those which seem to work (no extensive testing done yet) include the Allwrite editor, the Mod 4 version of OLD Scripsit, the Mod 4 Xenix Console emulator (no cursor), VisiCalc (only the base memory is used).

Programs which LOOK ok but freeze the system are Multiplan 1.7, Vidtex, Scripsit Pro. Programs which look like they will work with a few patches include the Allwrite formatter (needs patch for different parallel printer location-serial will PROBABLY work ok), and DeskMate, which locks due to inability to find [F3].

Of course, all of the programs on the TRSDOS 6 utilities disk which I've tested work fine: QFB6, UNKILL, and MOD324. I haven't tested TBA or BSORT. Of course, several programs will need some cosmetic patches because of the different screen character sets, but if I can put up with SCRIPSIT-16 on a DT-1, I can put up with a lot. Thank you for selling the product: I may start to get some real use out of my Mod 16 while waiting to budget a hard disk for Xenix.

I look forward to buying more MISOSYS products, as soon as I pay off the Tandy 2000 I just put on layaway. The source code to TRSDOS will come in handy when I feebly try to write my own command.com to use real operating system commands instead of MS-DOS.

(Fm: MISOSYS) You'll never get DD to work on the Model II under LS-DOS 6.2 even if you get the extra memory installed (the

ARCNET board) as DD goes to the hardware. When you look at that list of software products that violate the DOS standards, you begin to know how I feel when I can't get folks to listen to me when I preach standards and portability.

(Fm: LDOS Support) Ward: see my file in DL 8, EXTRA.64K for information. This file contains instructions on how to modify an extra memory board to get 128K for LS-DOS 6.2 on the Model 2/12. This will allow memDISK to work, though probably not DoubleDuty.

Multiplan 1.6 worked, 1.7 might after you add the extra memory. Patches for M4 Vidtex are here DL6 as VTPLUS.TXT

The [F3] keystroke can be generated by the keyboard driver. TBA and BSORT will work fine.

I have a Tandy 2000, so perhaps we can help you on that front also.

(Fm: Ward Griffiths) I had a feeling that a program that used the [CAPS] key as a shift with the function keys probably wouldn't be too cooperative. Sure would be nice to be able to page through that 512k attached to the 68k processor. (Maybe if I rewrite BASIC and VisiCalc to use the 68k as a math co-processor?)

On rereading the page of documentation, I see that you are right about [F3]. Wish I presently had time to play with it during the week. I'll have to dig through my crates of 5.25 disks for an MP 1.6. I haven't looked at the DL's yet. I suspect that the info on modifying the 64k (VisiCalc) board may not apply to my Mod 16 256k RAM boards, so I'll probably have to modify MemDisk instead. Either that or spend long hours poring over the Mod 16 schematics, which are about 3 times as big as the Model II schematics they are an appendix to.

A lot of 5.25 disks is one of the minor side affects of spending 5+ years as a Radio Shack Instructor & CSR. I do wish I'd catalogued the suckers.

(Fm: bob snapp) I am intimately familiar with the 68k memory boards and their access from the Z80 processor. If you have SPECIFIC questions, perchance I can save you some time.

(Fm: LDOS Support) The mods in question would apply only to a specific board, and not to a 68000 memory board. You may wish to take bob snapp's suggestion and perhaps modify @BANK to support the 68000 RAM.

(Fm: bob snapp) His first MAJOR concern should be that the 68k RAM is accessible by the the Z80 in 16k pages, not 32k pages. That will make life much more difficult, perhaps impractical for what he wants to do.

Hold on. If ALL you want to do is a memdisk, that's of course no problem. If you want bank swapping via svc's JUST LIKE ON THE MOD 4, that's simply physically impossible!

(Fm: LDOS Support) That also means that the driver is going to have to go in the low I/O driver zone, but that is usually the case. I don't think that it will be that difficult to implement.

(Fm: bob snapp) That's what I expected you to say. The way that the 68k memory is accessed is kinda strange. 16k at a time is switched in, from x'8000' to x'BFFF', but a full 32k of REGULAR Z80 memory is then unavailable. The latter we found out to our dismay in interfacing a 9 track mag tape unit to that t6k, and we were hoping to port an 8k buffer at x'C000'. When the 68k memory was being accessed, there was simply NOTHING in that area. We didn't want to keep banking in and out, so we gave up and moved the tape buffer from /z80ct1 to /xenix.

(Fm: LDOS Support) Ah, in that case, @BANK can't be used to support it. The memdisk or other driver will have to manipulate the banking directly.

File already open

(Fm: Harmon Ruble) I am using TRSDOS 6.2.1. Every time she goes to get a file to upload it comes back with "file already open". Need help.

(Fm: MISOSYS) If you invoke a DIR command targeting the drive you are trying to write to and you see a question mark following a file's name, then it indicates that you have the network flag set and a file was previously "open" but not "closed". Such files need to be RESET.

Now then, If you're network flag is set then it is possible that XT4 is trying to OPEN the file twice! You can confirm the setting of the network flag from DOS Ready by invoking the command, MEMORY (A="N"). If the displayed value is such that it is an ODD number in decimal, then the network flag is set. You would probably want to reset it by invoking the command, MEMORY (A="N",B=d) where d = the previous value displayed reduced by 1.

If this is not the case, then perhaps XT4 is setting that bit in memory due to either a programmed event or a memory corruption (unplanned). In any event, you now have some more info to investigate.

(Fm: Harmon Ruble) I use MultiDos to transfer from NewDos 80 to TrsDos 6, that is when I have this problem. Think you have me on the right track.

(Fm: MISOSYS) MultiDOS is probably setting the "file already open" bit in the directory. Our NOTES FROM MISOSYS, Issue II had a program called CONV DOS which was able to read files off of a NEWDOS80 double density data disk. It was revised again with the later version put on DISK NOTES 3. That's available for \$10 + \$2 S&H. Of course, I believe that the version was for the Model III operation but it could be easily altered. The other plan of attack with files < 85K in length would be to put them, on a single density disk with ND80 then copy those files to T6.

Bug in TED/CMD

(Fm: Bryan Headley) Saw 6.3 at the office. Your TED is the same as before; you know it has a bug with screen refresh? No? Go into TED, and type more than 80 characters (in words, or continuous, it doesn't matter), and press <enter> on the second line. Voila; the first line disappears. If you scroll up, it still is invisible. Only if you do a page scroll or scroll-to-top does the line you typed re-appear. Guess what? Same bug in TED/APP.

(Fm: MISOSYS) Bryan, I can only fix problems that people report. No one ever reported a problem with TED/APP and we didn't experience it here; thus, it's kind of difficult to fix problems you are unaware of. A fix for TED/CMD is in the LSI section.

(Fm: Dick Newman) I wrote this message on the "6.3 included TED" and I confirm Scott Russell's message about strange things. Every thing seems to clear up with a Shift Up Arrow, but after editing, things look funny for a while.

(Fm: MISOSYS) As TED is a MISOSYS product, I'll be the one to fix that. Too bad someone who noticed the bug in TED/APP never reported it to me as it would have been fixed before it got released as a /CMD file in 6.3 and 5.3.

Program revisions for LS-DOS 6.3

(Fm: Gene Szedenits, Jr.) Are the revisions to various utilities shaping up as patches for the QUARTERLY or as new versions to be purchased?

(Fm: MISOSYS) Revisions to our other products which require changes for either working with the x.3 releases or changes to take advantage of the extended date are shaping up slow due to the priority of other projects. This issue has patches for DOCONFIG (PRO-GENY) and PRO-PADS on the LS-DOS 6.3 side and ZSHELL, DOCONFIG (MSP-01), SAID 1.1, and BSORT51 on the LDOS 5.3

side. Programs which may need changes to utilize the extended date (such as PRO-WAM, FM, IFC, DESCRIBE, ADE, DISKDISK, LS-HELP, UTILITY 1, PARMDIR (MSP-01, PRO-GENY), ALTDISK, UNREMOVE (PRO-ESP), and ZCAT have not been looked at yet. Any product which needs revision for extended date access would have to be redesigned. This may or may not be done depending on the past acceptance of the product in the marketplace. Judging from past performance of most of the products mentioned above, few will be revised. Those that do will probably have a \$15 upgrade fee because the product price will be raised about \$10. I don't expect to begin looking at them for at least another two months. Certainly future issues will have more on this topic. The next project is a total revision of PRO-WAM with new features and more applications. Information on that will be in the PRODUCT TIDBITS section.

Ending JCL files

(Fm: Byron P. Peebles) When I use TED or EDAS to edit JCL, when the job finishes about a third of my screen contains what looks like an aborted Star Trek game - random characters (and I'm sure a lot of non-displayed stuff). I think there must be something wrong with the way the file is listed in the directory, such as EOF or LRECL. Am I close?

I have never experienced what you stated about EDAS or TED and editing of JCL. Your remark, "when the job finishes...". Are you talking about the JCL "job"? If so, what JCL command are you ending the JCL, "//EXIT" or "//STOP"? Or are you just dropping off? If you use EDAS and do not save the file with a "W!!filespec" command (or something like that - check the manual), you will write a CTL-Z at the end. TED writes a NULL. The NULL should be OK for JCL, not the CTL-Z.

Questions, questions...

(Fm: Bruce Travers) I have just received and installed LS-DOS 6.3 on my hard disk system. Overall I am very pleased with the enhanced DOS. A few questions and concerns

come to mind which I hope you can help me out on:

1. I like to 'customize' my DOS and was wondering if such customization would invoke the security protection scheme. For example, I use my own DOS prompt and definately intend to do something about that ugly boot up logo (everyone is entitled to their opinion right ?) such as put a graphic LS-DOS in tiny letters. What would be more helpful is if I can find out exactly what NOT to touch. I already know the serial number and such is taboo. It sure would ease my mind to know that that is all that is taboo. (Since making minor modifications I have already found one /CMD file that had 2 blank sectors in it causing a load file format error, and the system locked up once. I hope this isn't because I changed the DOS prompt to DOS ready with a bell character so I could get a beep).

2. I also own DiskDISK and was wondering if LSI planned to release a patch to the format program so that it uses 6.3 format. Right now I get around the problem by formatting then dateconv'ing.

3. Already I have seen patches posted in the message section of this SIG to correct a few minor bugs in the DOS. Would it be possible for you, or Roy to upload sets of patches rather than simply saying here they are and listing them? This would make it easier to install the patch as I read my messages online and hate scribbling patches down.

NEWCLOCK on LS-DOS 6.3

(Fm: Richard Buzzell) I do not know hi or low port. How could I find out? I have tried to no avail to get either the new LDOS or LS-DOS to work to my satisfaction. Any help you can give would be greatly appreciated. I cannot get the newclock to work.

(Fm: Allen Foster) Run the BASIC program porthelp which came with newclock. If the 12/24 and am/pm bit is port 117 you have lo port otherwise you have hi port. For LDOS 5.3 don't use the JCL which they give

you. Instead just run HIGHTIME/CMD and then sysgen. Set time & date off with System first. That should do it! I haven't had time to check on 6.3 but will get back!

(Fm: Pete Betz) I get it now! You're booting up with DATE and TIME=NO and letting CLOCK4 straighten things out. That works for the TIME, but CLOCK4 doesn't update the DATE, so if you do it that way there's no date in the system when you boot up from POWER OFF. I still think we need to revise X62/FIX to get it working on 6.3.

(Fm: Allen Foster) Don't use any patches! Just run the file called CLOCK4/CMD and then SYSGEN. CLOCK4 is a driver that loads into HI memory.

(Fm: Allen Foster) The date gets updated on my system along with the time just fine. Maybe you have a different version of NEWCLOCK.

(Fm: Richard Buzzell) Well I recently (today) uploaded a patch. Uploaded as NEWCLK63/FIX.

(Fm: Pete Betz) Thanks a bunch for NEWCLK/BAS. Everything works great now.

* - * - * - * - * - * - * - * - *

LSI's Column

The LSI LS-DOS Column

Hello from LSI. This will be where you will be getting info from LSI now and in the future. MISOSYS has been kind enough to offer to publish an LSI column in each issue of THE MISOSYS QUARTERLY. So, here we are.

There are a few things that should be understood from the beginning though by all readers. First LSI and MISOSYS are separate companies and we have different policies and procedures. Secondly, things that are expressed in this column are the

statements of LSI alone and MUST NOT BE ATTRIBUTED TO MISOSYS in any way. Third, any comment, good or bad, regarding the material presented here must be addressed to LSI directly. MISOSYS will assume no responsibility for the content of this column.

LSI will try to use this space to provide useful information for users of our LS-DOS 6.3.x operating system. Technical subjects will be addressed from time to time and all patches to the system will be in this column. As LS-DOS was just recently released this column will deal with the few corrections that were required to early releases of the system and general customer information. We will also address the most common questions that are being asked of our customer service department.

We won't forget the BASIC programmers either. Here's one for those who use BASIC, at ready try !<ENTER>. The "!" is the single character abbreviation for SYSTEM. The "!"<ENTER>" will take you back to DOS level with a lot less keystrokes. This is an undocumented feature.

Some folks have commented that our sign-on logo (or lack of it) is not to their liking for one reason or another. Let me explain. First, by decreasing the amount of data stored on the disk to make a "logo" we were able to save a GRANULE (1.5k bytes) on the MASTER disk, as SYS0 is now much smaller. Secondly we used only normal ASCII characters instead of graphic characters so that the sign-on could be universal for LS-DOS on different machines (if that should happen) without being dependent on a certain graphics set. Many have asked if they can change this, the answer is: of course, change it to what ever you like. You will find it in the last sector of SYS0. The bytes in question will be loaded directly to video memory during boot up. The number of characters per line and the number of lines should be equal to or less then the count that is on the MASTER. Be careful of the load block data (a 01 aa aa), there is a load block header like this for each line. Even these can be changed, of course, if you know what you are doing.

In a product as complicated and as large as an operating system there will always be problems found that are in need of correction. In an ongoing effort to support LS-DOS 6.3.x LSI will from time to time issue a revision statement such as this. You may, in most cases, make the revisions yourself or you may send your MASTER disk in for updating. The patches that are given here apply to customers with an ID # below 16500. However all LS-DOS users should read the following about update procedures and LSI.

LSI has categorized revisions to fall into two categories. These consist of CRITICAL REVISIONS and NON-CRITICAL REVISIONS. A CRITICAL revision is a revision that corrects a problem that could cause damage to, or loss of data. ALL other revisions are considered NON-CRITICAL. In the event that a CRITICAL revision is issued all registered LS-DOS 6.3 users will be notified by mail. When we create NON-CRITICAL revisions there will be no individual notification. ALL revisions will be available on the LDOS forum on COMPUSERVE and will be published in our column in THE MISOSYS QUARTERLY, which you can subscribe to by contacting MISOSYS at (703) 450-4181. Should you run into a problem with 6.3 that you consider a "bug", our Customer Service Department wants very much to hear from you. Most likely we will already have corrected that problem and will fire off a copy of the correcting patch by First Class mail. If you are the first to make us aware of the problem we will give it our most sincere attention. We will also be glad to provide complete revision statements (patches, etc.) to anyone requesting them, make sure you include your customer service ID#, with your request, it lets us know what revisions apply to your copy of the system.

To install the patches included here you must first create each PATCH file exactly as shown as an ASCII file. Then save it with the patch name specified in the patch. Each physical line shown must be terminated with a <ENTER>. Each individual patch has been separated by a line of = signs. The /JCL file that follows will properly install all these patches. After all files have been created do a:

DO MAKE63I (D=d,S=d) <ENTER>

where the two "d"s in the example are replaced by the Source (where the /FIX files are) and the Destination (where a COPY of your Master 6.3 disk is).

Also, you should note the special parameter that is given to the PATCH command for the installation of the patch to BASIC/OV2. The (O=N) parameter must be used with this patch or it will not install. This is a very special type of patch that is being applied to a special type of file.

The effort needed to create these files and get the patches correctly installed is tedious, time consuming and subject to human error. For these reasons LSI has instituted the following update policy for LS-DOS:

LSI will update your MASTER disk for a service, handling and shipping fee of JUST \$5.00 per disk update. With each update we will include an outline of what corrections have been made. To receive this service you must return your original BEIGE LSI MASTER disk along with a check or money order for \$5.00 to:

Logical System, Inc.
Update Department
PO Box 55235
Grand Junction, CO. 81505

LSI will also provide a REPLACEMENT MASTER disk to a registered user, for \$10.00. It will be shipped to your REGISTERED address only. You must include your Customer Service Number (CS#) when requesting a replacement disk and include a check or money order for \$10.00 with your request. (NO CODs, CREDIT CARDS OR P.O.s). Outside of U.S. or Canada, add \$3.00 for update or replacement.

It should be noted that this is the same disk update policy that LSI has had for over five years. Because we have a simple and inexpensive way for our users to update their system in an accurate and controlled manner, we will NOT help in any way with the creation, installation or

implementation of upgrade patches for the system. What we are saying is that if you cannot deal with, or are uncomfortable with patch installation totally on your own, then SEND IN YOUR DISK FOR UPDATING, as stated above.

We would like to point out that we do provide LS-DOS 6.3 as an "AS IS" product. No warranty of ANY kind is stated or implied. LSI has always tried to provide a reasonable level of support for our products and we hope to continue to do so, but we are not obligated to any service, update, error correction or ongoing support policy. Our policies may change from time to time and we feel that changing our policies on "AS IS" products is our right. LS-DOS 6.3 is provided to you at a very nominal fee. We simply do not make enough profit to provide unlimited, free services at this level of pricing.

Please consider the position of the authors of CPM and MSDOS, neither provide any where near the level of phone and mail customer service that LSI provides and their products cost much more. When CPM and MSDOS are updated or revised (even to correct their own errors) in most cases, you must simply BUY the new version, as of now the "latest and greatest" version of MSDOS is somewhat over \$100.00. At LSI we have paid a total of OVER \$1800.00 over the last five years to keep current with MSDOS/PCDOS revisions. In the same time frame a TRSDOS 6.x and LS-DOS 6.3 user could not have paid more than a total of \$95.00 to keep his DOS current, quite a difference we think. Try calling one of the other DOS authors and getting your questions answered, at least we are willing to TALK directly to our users.

We are sure you can cite some situations where other software companies have provided excellent service and stood behind their products, with nominal or no fees, but remember LSI has a NINE YEAR track record of such service. Also, how good has your support been on other \$29 or \$39 products?? So, please do not complain to us about our support policies, if anything our support policies and services will become LESS liberal, and MORE expensive in the future, due to simple

economics. LSI also reserves the right to totally terminate support for any product sold "AS IS", without notice.

LSI RETURN POLICY

Should you find the LSI support policy, or the "AS IS" condition under which LS-DOS is provided, unacceptable to you, OR FOR ANY OTHER REASON, you may return your LS-DOS 6.3, within 10 days of receipt, for a full refund. If you purchased on a credit card, please include the card number and expiration date so we can properly credit your account. Note: You must return the product completely intact and in resalable condition to receive a refund. NOTE: This policy became effective on 02/10/87.

Simply stated: WE WANT YOU TO BE A SATISFIED CUSTOMER.

A NOTE ABOUT BREF

BREF (the BASIC variable and line cross reference utility) has also been repaired so that it will properly deal with programs that contain a line number 0 or a reference to ON ERROR GOTO 0. However we were unable to make these corrections to BREF as a patch because the patch would have been huge and totally unmanageable, so we had to reassemble. If you use this utility in your programming work and wish to have the corrected version you will have to send your disk in for updating as stated above.

ABOUT LSI CUSTOMER SERVICE

Our customer service department will be happy to answer any question about the use and function of LS-DOS 6.3.x. by either written request or by phone. When making a written request be as specific as possible and include as much information as possible: screen prints, copies of disks, printouts, exact keyboard sequences, error messages, etc., are all helpful. Remember, vague questions will get vague responses. When calling customer service it is best if you can be at your computer so we can have you check things, however this is not a requirement. Always include your CS#

(NOT your serial #) on your correspondence with LSI or it will have to be ignored, sorry, there are no exceptions. Also, PLEASE provide us with a phone number where we can reach you, a DAY and EVENING number helps. We return all answers to the registered address for that CS# and each one IS checked.

IMPORTANT: WHEN PHONING OUR CUSTOMER SERVICE DEPARTMENT YOU MUST USE THE PHONE NUMBER FOR CUSTOMER SERVICE (SEE PAGE 16 OF YOUR 6.3 DOCUMENTATION). CALLING OUR MAIN NUMBER WON'T GET YOU TO THE RIGHT PLACE. THE CUSTOMER SERVICE DEPARTMENT IS IN A SEPARATE PART OF OUR BUILDING AND HAS A SEPARATE PHONE SYSTEM.

Please do NOT ask questions about products other than LS-DOS 6.3 as we probably do not know the answers. If you have a program that does not work on 6.3, it is most likely not the fault of LSI. The entire program interface, as documented for 6.2.x is identical to that of 6.3.x. So if a program fails to run identically on 6.2 and 6.3, it is probably the fault of the author of that program, NOT LSI. To get the program to run correctly on 6.3 you will have to contact the author/supplier of that program. If that can not be done you may have to be content to run the product on 6.2.x. The fact that it may run just fine on 6.2 and does not run on 6.3 is of great concern to us, but does not mean 6.3 is at fault in any way. Any program that does not run correctly on 6.3, but did on 6.2, has probably violated documented interface standards and the program is at fault.

There are THOUSANDS of programs being run on the Model-4 and it would be impossible to support all of them, especially when we are providing LS-DOS at such a small cost to the user. In most cases we do not even have the program you are asking about, much less the source code for that program. Our policy will be very firm; **Users can NOT rely SOLELY on LSI for assistance with products other than LS-DOS 6.3.x itself.** This does not mean that if we have information that can be of help to you that we won't tell you, we will. But in most cases we just won't have any knowledge that would be of help.

There are places to get useful information about applications running on your Model-4. Users of the Model-4 system should subscribe to "80-MICRO" magazine. You should be a member of COMPUSERVE and check in to the LDOS forum bulletin board (PCS-49) on a regular basis. This is an excellent place to ask questions, any questions, and get answers from other users. Another great source of accurate and timely information is THE MISOSYS QUARTERLY.

We will answer technical questions about LS-DOS, BUT IN A GENERAL SENSE ONLY. This means we will try to answer questions if the answer is to provide general information about our system and its function. We will not answer specific PROGRAMMING QUESTIONS. If you wish to program over LS-DOS 6.3 you should have the technical reference manual from Tandy (26-2119) and the "Programmers Guide to TRSDOS 6.x" from Diskcount Data and the "SOURCE CODE TO 6.2" from MISOSYS; without these you will be lost. Great amounts of accurate technical information and a special LS-DOS section dedicated to your 6.3 system are also available in THE MISOSYS QUARTERLY, from MISOSYS.

Normally we will not provide patches to custom configure your system. Patches to change system defaults and to modify system functions may become available from sources other than LSI. Please do not expect LSI to get involved with these in any way. However, LSI may from time to time issue an optional patch to alter a default or modify a function. If we do, we will publish it and support it. LSI will decide if, when, how, and what to offer an optional patch for.

ANSWERS TO THE MOST COMMON USER PROBLEMS

QUESTION: I just received my 6.3 disk and I get a parity error (or other read error) when I try to make a copy of the master, or my master disk won't even boot? **ANSWER:** The drives that we produced your MASTER disk on are kept in near perfect alignment and checked daily. This type of problem is most likely to occur if your Drive :0 is slightly out of alignment. Try this:

(1) BOOT UP with a 6.2 disk as you normally would. (2) Place your LSI MASTER 6.3 in drive :1 (3) Do the command: DISKCOPY.UTILITY :1 :0 <ENTER> (4) At the insert disks prompt, place a blank disk in Drive :0 (5) Press <ENTER>

If this procedure goes through without an ERROR message then you have succeeded in duplicating your MASTER disk to your drive :0. The disk in drive :0 will now be a fully functional and usable system in your boot drive. BUT, your drive :0 should be realigned.

If this procedure does not complete without ERROR then your MASTER disk probably is defective or was damaged in shipping. DON'T PANIC - just let us know and we will send you a fresh one in a RIGID, ANTI-STATIC package, and do it quickly.

QUESTION: How do I enable more floppy disk drives with my LS-DOS 6.3? **ANSWER:** The easiest way is to refer to the SYSTEM command in your manual for information on ENABLING and DISABLING drives. The SYSGEN command will then save these desired states and initialize them when you boot up. The commands you would use to enable 2 additional drives as drives #2 and #3 are:

```
SYSTEM (DRIVE=2,ENABLE) <ENTER>
SYSTEM (DRIVE=3,ENABLE) <ENTER>
SYSGEN <ENTER>
```

This is all there is to it; simple and straight from the 6.2.0 manual.

Another way would be to "PATCH" the operating system so that the drives are enabled from the start. Although this is NOT the recommended method, it is safe and can be used without problems. To make these patches you would execute the following commands:

```
PATCH BOOT/SYS.LSIDOS:0
(D02,84=C3:F02,84=C9) <ENTER> and PATCH
BOOT/SYS.LSIDOS:0 (D02,8E=C3:F02,8E=C9)
<ENTER>
```

Note: The first patch line ENABLES drive #2 the second patch line ENABLES drive #3. You must re-BOOT after installing for these patches to take effect.

One other method of turning on extra drives or other "environment" arrangements is to put them in a JCL file called INIT/JCL and then set an "AUTO DO INIT" on your boot disk. This method is most frequently used at LSI.

QUESTION: I want to get a copy of the current 6.2 user and/or technical reference manual, what do I do? **ANSWER:** Simple, get in touch with Radio Shack, the current 6.2 user manual is available as part #26-0316, the technical reference manual is part #26-2119. If you do not have these or have old versions LSI would recommend that you obtain them while you can.

QUESTION: DATECONV gives a message "Can't convert dates on non-6.3 SYSTEM disks"? **ANSWER:** Very simple to correct. At one time the disks' directory was marked as being a "SYSTEM" type disk. This can occur in several ways, none of which are damaging to your data. Why it happened is really not important at this point so let's cure it. LSI suspected that this may become a problem for some of our users so we provided an "undocumented parameter" to override the checking in for this DATECONV. Use the command:

```
DATECONV :d (CS) <ENTER>
```

The (CS) is the parameter, it simply stands for Customer Service. This overrides the protection for NON-6.3 system disks, so be careful. Check any disk you use this on to make sure that it has only the /SYS files BOOT and DIR, if any other /SYS files are on the disk DO NOT USE THE CS PARAMETER!!! These /SYS files must be PURGED before you proceed. The (CS) parameter is to BE USED ON DATA DISKS ONLY.

QUESTION: How do I make a double sided system disk that will be laid out correctly? **ANSWER:** The main reason to have

a system disk laid out "CORRECTLY" is so the system modules will be clustered around the directory to reduce access time to these frequently used /SYS files. Here is a very simple JCL which will do just that.

```
. DOUBLE/JCL - Procedure to create a
. double sided SYSTEM disk
. SOURCE (S) and DESTINATION (D)
. must be set on JCL COMMAND LINE
.
FORMAT :#D# (SIDES=2,A,N="LSDOS63",Q=N)
MEMORY (A="A",B=17)
BACKUP /SYS:#S# :#D# (S)
MEMORY (A="A",B=11)
BACKUP :#S# :#D# (I)
MEMORY (A="A",B=1)
.
. A correctly structured, double sided
. SYSTEM disk has been created.
. END
```

The above commands can of course be entered as individual commands at DOS ready with the same results. For convenience and accuracy we at LSI tend to use this JCL. The important part of this procedure is the use of the MEMORY command to set the "A" flag. This is the Allocation flag for the operating system and the number in this flag is the track on the disk where the file placement system will start to search for space on the disk.

This is a very valuable thing to know about and use. This will allow you to structure your own disks in any manner that you wish, and to restructure and repack disks for efficiency. The normal setting of the "A" flag is for track #1.

YES, those of you with hard drive should use a similar method when applying the system to your drive. Start the /SYS backup 2 tracks below the directory of your hard drive, by setting the "A" flag accordingly. Do the rest of the backup starting about 5 tracks below the directory. If you did not do something like this when you first set up your hard drive, you should reformat and do so now. This optimizing of the system on your hard drive will greatly increase the overall speed of your system.

QUESTION: On 6.2 I had a patch to have FORMAT default to double sided, double density and 40 tracks. What is the patch for this on 6.3? **ANSWER:** If the patch that you were using for 6.2 was: PATCH FORMAT/CMD.UTILITY (D09,5B=02:F09,5B=00) <ENTER> then on 6.3 it would be: PATCH FORMAT/CMD.UTILITY (D09,65=02:F09,65=00) <ENTER>

FORMAT will now default to DD/DS-40 track format. If you want a different format you will now have to specify same in the command line for format (see FORMAT in your 6.2 manual). **NOTE: THIS IS WHY LSI DOES NOT RECOMMEND THIS PATCH!!** Users forget that they have made this patch and then call. us saying "Your system is NO GOOD it won't even let me format a single sided disk".

QUESTION: I followed all the instructions for updating my hard drive system but I still can't seem to get it to work properly. What do I do? **ANSWER:** The instructions for updating your hard drive system will work in almost all cases, but there are some systems that will require another method. If your system is one of the odd ones you can try a RE-SETUP.

Note: Before doing any upgrade to a hard drive, it is expected that you will have made a complete backup of all the important data and files off of that hard drive before you begin. This is only common sense.

First you **MUST** start with a **CLEAN** exact copy of your LS-DOS 6.3 DISK made with DISKCOPY. Next using PURGE or REMOVE get rid of some files you won't be needing on this disk, keeping in mind that this is to become your **BOOT DISK**. We suggest you REMOVE DOS/HLP, this should give you ample room on the disk.

Now, using COPY, put your hard disk DRIVER program onto this disk, this is the program used to set up your hard drive that has the extension /DCT. Then copy the hard disk formatting program that came with your system onto the disk.

Now you must refer to the documentation that came with your hard disk software and perform a manual SETUP of the hard drive. You must set up the drive in such a manner that it will be partitioned in the identical manner as it was before you began your upgrading. If a SYSTEM (SYSTEM=d) command is requested as part of the setup **DO NOT DO IT YET.**

If all has gone well you should now have your hard drive accessible now. Test this by doing a DIR on each of the logical drive numbers now designated to be on your hard drive. If everything seems correct you should now put 6.3.0 onto the hard drive surface that contained your 6.2.0 do this with:

SYDTRU
P.O. BOX 297
PADSTOW N.S.W. 2215

BACKUP :0 :d (S,I) <ENTER>
If you have followed the instruction drive 0 should still contain a fresh 6.3.0 that is missing the DOS/HLP file. Replace the "d" in the backup command with the present logical drive number for the hard drive partition that will become drive 0.

After the BACKUP has completed you should do a:

SYSTEM (SYSTEM=d) <ENTER>

Replace the "d" above with the same drive number used in the backup above. This will swap drive 0 with drive "d". Test things a bit by doing a DIR:0. If all is well you should get the directory of your hard drive SYSTEM platter. If this checks out OK then is time to save things, so do a:

SYSGEN (DRIVE=d) <ENTER>

Again replace the "d" with the same drive number as before. This will cause this SYSGEN to go to the 6.3.0 floppy that is in the PHYSICAL drive position :0. You should now be able to RESET the system and it should end up running correctly.

If you will be using the HELP command you should copy DOS/HLP onto hard drive :0 from a fresh copy of 6.3.0. This will complete the setup of the system. You now can reboot and then add any addition setup you may require in your SYSGEN or create an AUTO function. Just make sure that any

SYSGEN or AUTO you do is specified to the BOOT floppy drive #.

If the above procedure fails in any way you will have to contact the company that provided you with your DRIVER AND FORMATTER for the hard drive. You may have to RE-FORMAT the whole drive or something else. DO NOT call LSI unless you are using the OFFICIAL Radio Shack driver and formatter, these are the ones we wrote and understand. There are many other drivers and formatters out there for drives other than Radio Shacks or for special purposes. We can't help with those, sorry.

QUESTION: How can I make a bunch of blank, formatted disks, real quick? **ANSWER:** Format a blank disk in the normal fashion using the FORMAT command in LS-DOS 6.3. Now simply use DISKCOPY to make copies of this blank disk. DISKCOPY is very smart and only duplicates tracks that contain real data, blank tracks are just formatted and verified. Try it, you will be amazed at the speed.

QUESTION: How long is LSI going to support LS-DOS 6.3.0 on the Model-4? **ANSWER:** Honestly, at this time we don't know. Three things will be given consideration in determining when we will stop. First, as long as there are enough questions and interest from users to justify keeping a customer service department running we will try to do so. Second, we will have to feel confident that there are NO serious problems with the Operating System that would require our attention. Lastly we MUST consider economics: LS-DOS 6.3.0 is not an expensive product, LSI HAS TO SHIP OVER 500 COPIES PER MONTH JUST TO BREAK EVEN. When the cash flow stops the support may have to stop, there simply may not be an alternative. But we will do the best we can.

Full support of LS-DOS will certainly continue through all of 1987 and most likely through most of 1988, possibly much longer. But ongoing support MAY not be free. LSI may have to begin to charge for support. We reserve the right to charge for our services, but customer support

will always be available for the first 30 days of ownership free of charge.

We are considering offering a Support and Update Newsletter (The S.U.N.) which would contain tips on usage, technical information, patches to the system and answers to common questions and problems for users of LS-DOS. In the future, a publication like this may be the only way that LSI will provide update information to our users. We would have to charge about \$20 to \$50 per year for this type of service publication. Please, let us know if you would be interested in such a publication.

QUESTION: The anti-piracy protection scheme on LS-DOS 6.3 scares me. I'm afraid something might go wrong with it and I will lose valuable data? **ANSWER:** Not to worry. You are right to assume that the protection system is VERY sophisticated, it cannot even be found using any conventional software technique. LSI spent much time and effort in the design and testing of this protection system and we can assure you that:

"No matter what happens, no legitimate users will ever lose data or have any other problem with this protection system" - "pirates or users who violate our license agreement will have trouble with the system"

We will know if, and when, a violation occurs based on the problems that are reported by the user. But, under no circumstances will the PROTECTION SYSTEM ever damage DATA.

OUR PARTING COMMENT

ALL USERS: Please do yourself a BIG favor and carefully read the following sections of your TRSDOS Version 6 Reference Manual: BACKUP / FORMAT / SYSTEM / SYSGEN.

A very large portion of the questions we get from users could be answered if the user had a clear understanding of ALL the functions in these four areas of the TRSDOS and LS-DOS systems.

LSI patches to LS-DOS 6.3

```

=====
.*** MAKE63I/JCL ***
.-JCL- Procedure to create a LS-DOS 6.3.0 Level - "I" from a Level - "H"
. THIS JCL AND THESE PATCHES ARE FOR DISKS WITH CUSTOMER SERVICE NUMBERS
. LESS THEN 16500 ONLY !!
. NOTE: Destination to be patched and Source of patch file must be given on
. JCL command line, ie - DO MAKE63I/JCL (D=d,S=d) <ENTER>
. Replace the "d" in example with correct drive numbers
PATCH SYS5/SYS.LSIDOS:#D# using SYS51/FIX:#S#
PATCH SYS9/SYS.LSIDOS:#D# using SYS91/FIX:#S#
PATCH MODELA/III:#D# using MODELA1/FIX:#S#
PATCH TED/CMD.UTILITY:#D# using TED1/FIX:#S#
PATCH DATECONV/CMD.UTILITY:#D# using DCONV1/FIX:#S#
PATCH BASIC/OV2.BASIC:#D# using BASIC1/FIX:#S# (O=N)
PATCH BOOT/SYS.LSIDOS:#D# (D02,1E=49:F02,1E=48)
PATCH TED/CMD.UTILITY:#D# using TED2/FIX:#S#
PATCH DISKCOPY/CMD.UTILITY:#D# using DCOPY2/FIX:#S#
PATCH BOOT/SYS.LSIDOS:#D# using BOOT2/FIX:#S#
. This next patch may or may not install. If it does not install it is
. because the copy of DISKCOPY you have is already corrected.
PATCH DISKCOPY/CMD.UTILITY:#D# using DCOPY1/FIX:#S#
. *****
. **** Your LS-DOS is now a version 6.3.0 / Level - I ****
. *****
.END

```

Here are the current patches that are applied by the above JCL.

```

=====
.** DCOPY1/FIX - 01/07/87 - For level-H - DISKCOPY/CMD.UTILITY - PATCH #001
. This will cause DISKCOPY to work correctly on double sided drives.
. This PATCH to be installed on DISKCOPY/CMD.UTILITY - File dated 12/01/86
D04,83=28 16 FD CB 03 A6 CD 53 2B;F04,83=CA 7F 2A CD 53 2B C2 0C 26
=====
.** SYS51/FIX - 01/13/87 - For Level-H - SYS5/SYS.LSIDOS - - - PATCH #002/A
.Patch to correct the "P" command in extended DEBUG
. NOTE: THIS IS A TWO PART PATCH (SYS51/FIX and SYS91/FIX)
.This patch must be applied to SYS5/SYS.LSIDOS - File dated 12/01/86
D00,09=0D;F00,09=17
D00,12=36 2C 4C 53 49 01 08 BD 23 F5 CD 3D 06 F1 C9
F00,12=32 2C 33 2C 34 2C 36 20 62 79 20 4C 53 49 2A
=====
.** SYS91/FIX - 01/13/87 - For Level-H - SYS9/SYS.LSIDOS - - - PATCH #002/B
.This patch must be applied to SYS9/SYS.LSIDOS - File dated 12/01/86
. NOTE: Patch #002/A must be installed on SYS5/SYS also.
D01,F6=BD 23;F01,F6=7F 20;D01,F9=BD 23;F01,F9=7F 20;D02,0A=BD 23;F02,0A=7F 20
D02,16=BD 23;F02,16=7F 20;D02,26=BD 23;F02,26=7F 20;D02,36=BD 23;F02,36=7F 20
D02,3B=BD 23;F02,3B=7F 20;D02,3E=BD 23;F02,3E=7F 20;D02,52=BD 23;F02,52=7F 20
D02,5E=BD 23;F02,5E=7F 20;D02,63=BD 23;F02,63=7F 20;D02,70=BD 23;F02,70=7F 20
D02,77=BD 23;F02,77=7F 20;D02,86=BD 23;F02,86=7F 20;D02,91=BD 23;F02,91=7F 20
D02,A4=BD 23;F02,A4=7F 20;D02,A7=BD 23;F02,A7=7F 20;D02,AA=BD 23;F02,AA=7F 20
=====
.** MODELA1/FIX - 01/20/87 - For level-H - MODELA/III - - - - PATCH #003
. This patch to be applied to MODELA/III - File dated 12/01/86
. Corrects the messages in the RAM image in Model-3 mode, on Mod-4Ps.
D01,33=03 49 49 49 2F;F01,33=2F 44 69 73 6B

```

D01,38=34 2F 34 50;F01,38=20 42 41 53
 D02,16=2F 44 69 73 6B 20;F02,16=49 43 0D 28 63 29
 D02,1C=42 41 53 49 43 0D;F02,1C=20 31 39 38 36 00
 D38,B9=37;F38,B9=36

=====
 .** TED1/FIX - 01/15/87 - For Level-H - TED/CMD.UTILITY - - PATCH #004
 . This patch corrects buffer handling and screen updating in TED.
 . This patch to be applied to TED/CMD.UTILITY - File dated 12/01/86

D02,3F=C3 6C 2E;F02,3F=C3 C1 28
 D08,76=E5 7C 87 87 84 87 4D 6F 26 00 29 29 29 06 2F 09 1A 77 E1 C3 C1 28
 F08,76=00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

=====
 .** DCONV1/FIX - 01/20/87 - For Level-H - DATECONV/CMD.UTILITY - - PATCH #005

. This patch will cause DATECONV to update the GAT at 'GAT+0CBh' to a '63h'
 . This patch to be applied to DATECONV/CMD.UTILITY - File dated 12/01/86.
 D01,37=CD 00 2A;F01,37=CD 26 27
 X'2A00'=3E 63 32 CB 29 C3 26 27

=====
 .** BASIC1/FIX - 01/25/87 - For level-H - BASIC/OV2.BASIC - - - - PATCH #006

. This PATCH to be installed on BASIC/OV2.BASIC - File dated 12/01/87
 . This patch corrects the problem in FIND and SEARCH where ALL occurrences
 . of a variable may not have been found.

D03,6D=C3 59 23 00;F03,6D=FE 84 28 F7
 D05,59=FE 84 CA 68 21 FE FF C2 71 21 23 C3 60 21
 F05,59=00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

=====
 .** BOOT1/FIX - 01/20/87 - For Level-H - BOOT/SYS.LSIDOS - - - - PATCH #007

. This patch will cause the display LEVEL of 6.3.0 to change from "H" to "I"
 . This patch to be applied to BOOT/SYS.LSIDOS - File dated 12/01/86.

D02,1E=49;F02,1E=48

=====
 .** TED2/FIX - 02/01/87 - For Level-H - TED/CMD.UTILITY - - PATCH #008

. This patch corrects CURSOR handling and MEMORY handling in TED.
 . This patch to be applied to TED/CMD.UTILITY - File dated 12/01/86

D03,90=E3;F03,90=EF
 D07,12=D4;F07,12=CD

=====
 .** DCOPY1/FIX - 02/08/87 - For level-H - DISKCOPY/CMD.UTILITY - PATCH #009

. This will correct buffer locations and usage in DISKCOPY
 . NOTE: This affected 80 track double sided drives only.

. This patch to be applied to DISKCOPY/CMD.UTILITY - File dated 12/01/86
 D02,F3=00 7F;F02,F3=C4 6C;D03,74=00 2F;F03,74=5A 2E;D03,BD=00 2F;F03,BD=5A 2E
 D03,C0=60 2F;F03,C0=BA 2E;D04,01=00 30;F04,01=5A 2F;D04,35=00 30;F04,35=5A 2F
 D05,3D=00 54;F05,3D=5A 53;D05,4F=03 54;F05,4F=5D 53;D05,58=CD;F05,58=27
 D05,A1=00 54;F05,A1=5A 53;D05,F2=00 54;F05,F2=5A 53;D06,93=00 54;F06,93=5A,53

=====
 .** BOOT2/FIX - 02/20/87 - For level J - BOOT/SYS.LSIDOS - - - - PATCH #010

. This patch will change the amount of "STACK SPLIT" that is used during
 . the enable/disable ram routine to more balance stack availability.

. Software products such as PRONTO/PROWAM will work after this patch.
 . This patch to be applied to BOOT/SYS.LSIDOS - File dated before 02/10/87
 D06,2C=40;F06,2C=60

Applications for the User

Making KILL6/CMD and CAT5/CMD

by Jim Gaffney

Compuserve (70515,1036)
(GENIE: JIMGAFF)

LS-DOS 6.3 and LDOS 5.3 are two fantastic operating systems, but they each lacked a feature which I find to be essential -- a KILL command for LS-DOS and a CAT command for LDOS. I switch back and forth between operating systems a lot and having a syntax work on one machine and not on the other drives me to distraction! Besides, if "REMOVE" is so great, why isn't the command from inside the BASIC interpreter 'REMOVE "filename"'? If you try other than 'SYSTEM "REMOVE filename"', you'll get a syntax error while 'KILL"filename"' works just fine. Now I realize that space constraints precluded including these features in the "vanilla" DOS versions, but I decided there must be a way to provide the features for myself.

I started with the problem of no "KILL." Apparently, incorporation of the "ID" command by LSI in version 6.3 took the jump table slot which was formerly occupied by "KILL" in version 6.2. There could therefore exist either a KILL command or a "REMOVE" command in SYS1/SYS, but not both. I zapped the "REMOVE" with a "KILL " (as have a lot of users), but that produced a problem in that some programs (notably Stephen Milliken's SHELL) look for a "REMOVE" command to disable files. I could foresee a string of patches to work around the patch to SYS1/SYS -- not desirable at all!

About that time, I received a letter from Jerry Wagers (who is currently stationed in Korea) which reminded me of a "KILL/CMD" file that had been available on the LDOS SIG access in the days of TRSDOS 6.1 -- before 6.2 provided a KILL command possibility. It takes up a gran of space on your disk, but that isn't much and what the heck, I'm blessed with DSDD floppies and a HD anyhow!

I looked through my library of PD files only to discover that I had apparently KILLED KILL/CMD when I moved to TRSDOS 6.2. What the heck, I'd never need it again, right?? I next looked in the DL's of the LDOS SIG, but it had disappeared

from there, too. So I left a message to Joe Kyle DiPietropaulo (did I spell it right, -jjkd-?) to ask if it had been archived away somewhere. A lot of members responded "why don't you just change REMOVE to KILL?", but that wasn't what I wanted. Finally, Gary Phillips suggested that I could use CMDFILE to extract the REMOVE code from SYS6/SYS and just rename it to KILL/CMD. Bless you, Gary!

I knew that SYS6 was an ISAM file, but I had never done any file extraction/library creation before. Armed with the PRO-CESS docs, I set out to my task. My biggest problem was figuring out which ISAM overlay I wanted. "The Source" came to the rescue; overlay 18 is the REMOVE code.

I next fired up PRO-CESS and executed the following interchange:

```
<L>oad a file
"specification?"
SYS6/SYS.LSIDOS
".. enter #>"
18
"enter new address"
<ENTER>
"specification?"
KILL/CMD:0
"written"
<E>xit to DOS
```

And that did it -- I had an operating KILL/CMD on my :0 drive.

Now, when I type 'KILL filename' DOS first looks at the library. Not finding KILL there, it executes the KILL/CMD file to disable the desired file. I did add one more bit of cosmetic surgery -- I changed the text in the application file from "Removing:" to "Killing:" so that I was sure which file was doing its job. I did it using FED, but the following patch will do the same thing if you feel uneasy using FED in the ASCII modify mode.

```
.KILL6/FIX -- patches KILL/CMD to display
."Killing:" rather than "Removing:"
.SYNTAX PATCH KILL/CMD USING KILL6/FIX
D01,87=4B 69 6C 6C 69 6E 67 3A 20 01 00
F01,87=52 65 6D 6F 76 69 6E 67 3A 20 01
```

If you want to do it absolutely right, you can remove the extra NOP generated by the above patch by packing the file with PRO-CESS. You could pack it even more by removing the header load module, but that would be a copyright violation and would serve no useful purpose since the file is already well under 1 gran in size and it will occupy a whole gran on disk in any event.

Armed with this success, I set about creating a CAT/CMD for LDOS 5.3; I wanted LDOS to behave similar to LS-DOS. This time, I didn't have the source code for LDOS to help me find the right ISAM overlay, but since both operating systems had the same genesis, I reasoned that they might use the same ISAM numbers for similar functions. The premise proved correct, ISAM #21 is the DIR function for LDOS 5. Before I extracted the code from SYS6/SYS, I patched it so that the parameter default was (A=NO), i.e., so that the display is filenames and flags only like the LS-DOS 6.3 CAT command. This was accomplished by applying the following patch:

```
. CAT5/FIX -- patches LDOS 5.3 SYS6/SYS
.   to make (A=NO) the default for DIR.
.   Syntax is PATCH SYS6/SYS.SYSTEM CAT5
D08,92=00 00:F08,92=FF FF
```

Since LDOS and LS-DOS formats are interchangeable, I used PRO-CESS to extract the desired ISAM overlay (#21). The method was similar to what I had used for extracting the REMOVE overlay, except that this time I specified:

```
<L>oad the file SYS6/SYS.SYSTEM
ISAM #21, and
<W>rite CAT/CMD with the drive
containing the LDOS disk as the
destination.
```

My final step was to go back and "unpatch" the LDOS SYS6/SYS so that "DIR" would again produce a listing with (A=YES) which is similar to that produced by LS-DOS6 with the DIR command. The syntax there was:

```
PATCH SYS6/SYS.SYSTEM CAT5 (REMOVE)
```

Finally, I realize that I could have

written short programs to accomplish the same functions as my new programs by using the @CMDI SVC and that they would have been very short pieces of code. In order to make them flexible, however, I would have had to add code to parse the parameter line and act upon those parameters which would have appreciably increased the size of the code. With a lot less work and with the aid of PRO-CESS, I have two applications files which suit my needs very nicely, making LS-DOS and LDOS behave in a very similar manner.

Upgrading Little Brother to LS-DOS 6.3

The TRS-80 version of the Little Brother Data Manager is supplied on three disks: a RUNTIME disk used while running LB; a CREATION disk used while defining a new LB data base; and a STARTUP disk used to initialize and startup LB when running on a 2-drive 128K Model 4. The STARTUP disk comes supplied with a minimum TRSDOS 6.2 operating system. The 2-drive startup process automatically installs a MEMDISK and bulk loads it with a SYSTEM drive image which had previously been stored by LSI in a file named BACK/IMG.

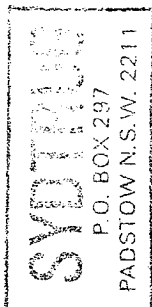
LB users stepping up to LS-DOS 6.3 would like to utilize 6.3 with LB. The hard drive user had an easy conversion process; all that was needed was to follow the standard instructions issued by LSI for upgrading a hard disk to LS-DOS 6.3. Unfortunately, there has not been any procedure to upgrade the 2-floppy Model 4 LB system to 6.3; at least not until now.

The hard part of a procedure is the upgrading of the process which automatically installs the MEMDISK and loads it from the SYSTEM image file. Not only does your STARTUP disk need to be upgraded to 6.3, but the BACK/IMG file needs to be upgraded as well with a 6.3 MEMDISK image. That requires a special program. Since LSI never considered the necessity of the user upgrading the DOS release at some future point in time, the utility they used to generate the BACK/IMG file was not included.

The following is a hex listing of a program called MEMDUMP. Type it into a

file using any text editor of your choice (TED as supplied with LS-DOS is good for this). Then convert it to a binary file using BINHEX (a few versions of BINHEX were printed in the last issue of TMQ). An alternative is to obtain the /CMD version of the program which is on DISK NOTES 7. Note that in order to save from switching over to single columns, we have printed each line on two rows; thus, contiguously connect two lines when you input. A full line will contain 60 characters.

```
01B900262191263E0AEF3E01EFFE30
280BFE312807FE3228033E16EF32B5
2611AC26060021CC263E3AEFC27926
0100013E42EF3E4BEF3E44EFD90101
00210080D901000121CC2611AC26E5
D9D1C5F33E66EF010001EDB03E66EF
FBC1D93E4BEFC279260B78B1280F79
FE8020DAD9010200210080D918D03E
3CEF3E16EF2182263E0CEF3E15EF44
69736B20492F4F206572726F720D44
7269766520746F20686F6C6420696D
616765203C302D323E20034241434B
2F494D473A310D02020026
```



Once you have obtained the MEMDUMP program, follow this step by step procedure.

(1) Make a backup copy of your current STARTUP disk. This will be your new startup disk when the procedure is complete.

(2) BOOT your new STARTUP disk. When it prompts you to place a RUNTIME disk in drive 0 (the bottom or left drive), do so. Note that this will still be under 6.2.

(3) When the LB main menu appears, escape back to TRSDOS Ready by depressing <BREAK> twice. Now BOOT with a copy of the LS-DOS 6.3 disk.

(4) Invoke the command:

```
SYSTEM (DRIVE=2,DRIVER="MEMDISK")
```

Respond to the selection request with "D" to select a 2-bank memdisk. Request "D" for double density. When you get the "Do you wish to format it <Y,N>?" prompt, respond "N" for NO! You don't want to format it.

(5) Now issue the command:

```
BACKUP :0 :2 (SYS,OLD)
```

This will update the previous contents of the MEMDISK (which still contained the 6.3 SYSTEM image) with the new system files from your LS-DOS 6.3.

(6) When that backup is complete, issue the command,

```
DATECONV :2 (CS)
```

(7) Now place the new STARTUP disk in drive 1 (the top or right drive) and issue the MEMDUMP command, first making sure that the MEMDUMP program mentioned above is on your SYSTEM disk. When asked what drive should the image file be written to, answer with "1". The BACK/IMG file on the new startup disk will now be updated to a 6.3 SYSTEM image.

(8) The last step is to upgrade the STARTUP disk to 6.3 by the standard procedures noted in LSI's documentation. Essentially, all you need do at this point is the commands,

```
BACKUP :0 :1 (OLD,SYS)
DATECONV :1
```

I would recommend that you now boot up the new STARTUP disk holding down the <ENTER> key (to inhibit the AUTO) and go through the procedure noted in the Little Brother Installation Manual (i.e. invoke LBINSTALL). That should do it and you will be using LB under LS-DOS 6.3.

A final note. Keep these instructions handy in case there ever is another release of the DOS which requires major upgrading.

Converting MailFile to Little Brother

Here's a little exercise we did for a couple of reasons. First, it may show a few folks just how easy it is to define a Little Brother Data Base. Second, it provides a way to port your old MailFile data files over to Little Brother for either Model 4 use or MS-DOS use (with little or no work required).

Now certainly, it would make more sense to embellish the data fields to suit your own needs; something you couldn't do with MailFile. On the other hand, since I can't second guess your needs, you'll have to be content with an emulation of MailFile using Little Brother; or define your own record layout and then adapt the conversion program discussed later. What I have done is defined a data structure identical to MailFile; defined an add/edit/update screen which looks like MailFile, and defined two output reports: one-across mailing labels and an 80-column directory; the resulting set of LB data files can be considered a MailFile template. I also wrote a BASIC program to take an existing MailFile data base and convert it for bulk loading into the Little Brother emulation; the bulk load is a "job" file.

The five files which make up the MailFile template are on DISK NOTES 7. They are MAILFILE/DEF /LB, /PRO, /PR1, and /VD0. In addition, I have typed in a LB job file to automatically generate two index files. The first index sorts the data by name; the second sorts by ZIP code. Both sorts are just what you would experience by the constant sort of MailFile. The job file is named MFINDEX/JOB.

It's kind of tough to demonstrate the ease of defining a data base without at least telling you the steps which I went through. Thus, the following scenario outlines what I did. I'll include the menu screens as I work through the scenario so those readers who do not have Little Brother will be able to follow.

When you invoke LB, you will see the main menu. Here's what it will look like:

Little Brother - LSI Database Version 1.0.0
Copyright (C) 1985 by Logical Systems, Inc.

1) Select Data Base Name

- | | |
|-----------------------------|----------------------------|
| 2) Add Records | 8) Define Screen Formats |
| 3) Update or Delete Records | 9) Define Print Formats |
| 4) Print Records | 10) Define File Format |
| 5) Sort or Select Records | 11) Set Screen/Add Index |
| 6) Run Automatically | 12) Change Password |
| 7) Expand Data File | 13) View Field Definitions |

Name: MAILFILE Index:None Screen:1 Allocated:100 Used:76

Enter Selection Number ..

This main menu screen actually shows the results of the entire job. That's why the bottom status line shows 76 records in use out of 100 allocated. I already loaded a MailFile database into this demo copy.

Now the first step we need to do is menu item 10 - define the data file record layout. This then prompts for the data base name (where I specified "MAILFILE") and the data base password which will be needed later to view definitions and make screen changes. I also entered "MAILFILE"

as this password; that, then will be the name and password on the DISK NOTES 7 template.

The <D>efine command is used to initiate LB prompts for field information. The LB prompted entries are easy to specify; you just enter a field name, that field's type, that field's length, and whether it is protected or not from normal editing. Thus, specifying the nine MailFile fields gave me the following chart:

| | | | | |
|---|----------------|---------|------------|------------|
| 1 | Name: Name #1 | Type: L | Length: 27 | Protect: N |
| 2 | Name: Name #2 | Type: L | Length: 27 | Protect: N |
| 3 | Name: Address | Type: L | Length: 27 | Protect: N |
| 4 | Name: City | Type: L | Length: 15 | Protect: N |
| 5 | Name: State | Type: U | Length: 3 | Protect: N |
| 6 | Name: Zip Code | Type: L | Length: 11 | Protect: N |
| 7 | Name: Phone | Type: N | Length: 14 | Protect: N |
| 8 | Name: Ext | Type: N | Length: 4 | Protect: N |
| 9 | Name: Code | Type: L | Length: 6 | Protect: N |

<RET> continues .

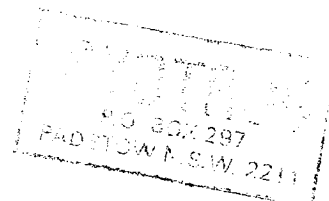
Note that the L, U, and N types stand for literal, upper case literal, and numeric. For those not in the know, other types supported by Little Brother are "A" alphabetic, "B" upper case alphabetic, "R" right justified, "D" dollar, "F" floating point, and "C" calculated (using add, subtract, divide, multiply).

It's easy to layout a screen; all you do is move the cursor about the video screen using the ARROW keys and type in your information. When you want to designate a data field, a simple CTRL-F will prompt for the "field" number. LB also allows you to switch on/off inverse video highlighting, delete field designations previously designated, insert/delete any characters which you typed, and insert/delete entire lines. I easily constructed the following screen which can be used for adding records, viewing records, and editing records.

Now that the data base has been defined, the next reasonable course of action is to define the screen format to be used for adding data, searching for data, and editing data. We exit the "define file format" module and select number 8, Define screen formats.

MailFile Emulation using Little Brother

- 1> Name #1 :
- 2> Name #2 :
- 3> Address :
- 4> City :
- 5> State : ...
- 6> Zip Code :
- 7> Phone :
- 8> Ext :
- 9> Code :



Define Screen 1

Help Quit Define Edit Print Save

Now that the data base file has been defined and the input screen layed out, the next step is to design any needed output reports. Quitting the "define screen" option, I'm back at the LB main menu. I select option 9, "Define printer format".

report is about as easy as laying out the input screen. The cursor is moved about the video screen and any characters which you want typed on the output are just typed into the screen. Fields from a data record which are to appear on the printout are designated by a sequence such as, "^4^", which designates field 4.

Generating the structure of an output

Since an output report can (and usually does) exceed the 80 columns of your video screen, LB provides a windowing technique to provide for wider reports. In fact, three separate windows are displayable to give you up to 208 virtual columns for your report.

The twenty columns available for designating rows of information can also be divided into header, record data, and footer areas. I wanted a header which

included the name of the data base, the current date and time the report was generated, and a page number. I also wanted a footer to print a separating line followed by a blank line. Thus, I divided the six lines of information which I typed into the video screen into two lines for a header, two lines for printing record data, and two lines for a footer. Here's the results of my 80-column "MailFile directory" report.

```
<MailFile 80-Column Directory> <File:~f~> <~d~ ~t~> Page: ~p~
=====
~r~  ~1~                ~2~                ~7~                ~8~
      ~3~                ~4~                ~5~ ~6~                ~9~
=====
.....
_____ Print Screen - File # 1 _____
Header : 1-2   Text : 3-4   Footer : 5-6   1 Rec: N Row :1   Column:1
<V>iew, <E>dit, <F3> Save, <ESC> Quit - Enter Command: .
```

The series of dots represents blank screen lines which I have omitted to avoid wasting QUARTERLY paper. Incidentally, these "screen images" were captured by routing the *PR device to a disk file before entering Little Brother and then invoking a DOS "screen print" command whenever I wanted to capture a screen. I subsequently loaded the "screen print" disk file into ALLWRITE for further editing (i.e. adding the scenario text

which you are now reading). This technique of screen printing to a disk file is quite useful (now if we could add that feature to MS-DOS!).

I also wanted to create the "output report" for printing mailing labels. Duplicating the procedure used for the 80-column report, I worked up the following screen:

```
~2~
~1~
~3~
~4~, ~5~ ~6~
.....
_____ Print Screen - File # 2 _____
Header : None   Text : 1-6   Footer : None   1 Rec: N Row :1   Column:1
<V>iew, <E>dit, <F3> Save, <ESC> Quit - Enter Command: .
```

The Little Brother "define printer format" option includes a function which outputs the entire printout definition to your printer; complete with a sample record. I took the liberty of requesting this for

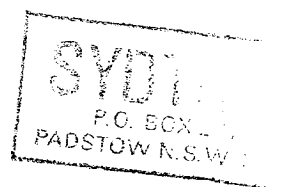
the mailing label. This report shows the individual printer parameters (which I haven't touched on), the record field descriptions, and the sample report.

Printer Parms, Field Formats and Sample Screen Format for File - MAILFILE
Print Format File # 2

Physical Lines Per Page .. 6
Printed Lines Per Page ... 6
Physical Line Length 60
Columns Per Record 60
Left Margin Setting 0
Indent on Wrap-around 0
Records Printed Across ... 1
Repeat Record Count 1

| # | Name | FLEN | PLEN | S | Format | Calc |
|---|----------|------|------|---|--------|------|
| 1 | Name #1 | 27 | 27 | N | | |
| 2 | Name #2 | 27 | 27 | N | | |
| 3 | Address | 27 | 27 | Y | | |
| 4 | City | 15 | 15 | N | | |
| 5 | State | 3 | 3 | N | | |
| 6 | Zip Code | 11 | 11 | N | | |
| 7 | Phone | 14 | 14 | N | | |
| 8 | Ext | 4 | 4 | N | | |
| 9 | Code | 6 | 6 | N | | |

%%%%%%%%
%%%%%%%%
%%%%%%%%
%%%%%%%%, %% %%%%%%%%%



The "S" column noted above stands for "strip". This designates whether the corresponding field will have trailing spaces stripped from the printing. It would commonly be done on a first name field so that there would not be a big gap between the printing of the first and last names. The mailing label which I have defined here would be similar to that generated by MailFile. This has been an emulation of MailFile, hasn't it?

There's one last thing which I did for this data base. Back at the main menu, I selected option 11 to "set the screen". What this does is establish a particular screen automatically whenever you go into the add or update modes. Setting a default eliminates the generation of the screen selection prompt. Since I have defined only one screen, I let Little Brother automatically designate it.

Oh yes, one more thing. I promised you a BASIC program which constructs a LB job

file for adding data from any MailFile data base. I call it MF2LB/BAS. This program (which follows) reads the MailFile control file (filename/CTL) from your MailFile base and reads records in alphabetical order. In doing it this way, it ensures that only active records will be extracted from your MailFile data base. The program also includes the strange code which unpacks the telephone number field used in MailFile and keeps it a standard "N" field for LB; thus, the telephone number field can contain only digits and minus signs.

The MF2LB/BAS program is written using Model 4 BASIC. It is on DISK NOTES 7, in case you don't want to type it in. The file was saved in ASCII so it could be read by "alien" BASIC's. This dialect of BASIC should also be easily ported to MS-DOS - in case you want to convert all of your old MailFile data to the MS-DOS version of Little Brother.

```

5 REM MF2LB/BAS - Program to convert MailFile data files to Little Brother
6 REM Copyright 1987 MISOSYS, Inc., All rights reserved
7 REM Last updated: March 9, 1987
10 CLEAR 3000:DEFINT A-Z:DEFSTR S,D:CLS:GOTO 100
20 IF CVD(S6)<10000 OR S6="" THEN P$=STRING$(8,32):RETURN
22 P$=STR$(CVD(S6)):P$=RIGHT$(P$,LEN(P$)-1):P$=LEFT$(P$,LEN(P$)-4)+"-"+RIGHT$(P$,4 )
24 IF LEN(P$)>9 THEN P$=LEFT$(P$,LEN(P$)-8)+"-"+RIGHT$(P$,8)
26 RETURN
50 LN=LEN(P$)
52 IF LEFT$(P$,1) = " " THEN P$ = "" :RETURN
54 FOR L1=LN TO 1 STEP -1
56 IF (MID$(P$,L1,1)=" ") THEN GOTO 58 ELSE P$=LEFT$(P$,L1):L1=1
58 NEXT L1
60 RETURN
100 INPUT "Enter the name of your mailfile ";N$
110 OPEN"r",1,N$+"/DAT",128:FIELD 1, 27 AS S0, 27 AS S1, 27 AS S2,15 AS S3,3 AS S4, 11 AS
S5,8 AS S6,4 AS S7,6 AS S8
120 OPEN"I",2,N$+"/CNT":INPUT#2,T:REM Get # of items
130 PRINT@(1,0),"Total number of items is ";T
140 OPEN"o",3,N$+"/JOB:2"
150 PRINT#3,"2";CHR$(13);"A";
160 FOR L=1 TO T
170 INPUT#2,TA,TZ:GET 1,TA
180 PRINT@(2,0),"Record ";L;": ";TA;
201 P$=S:GOSUB 50:PRINT#3,P$;CHR$(13);
202 P$=S1:GOSUB 50:PRINT#3,P$;CHR$(13);
203 P$=S2:GOSUB 50:PRINT#3,P$;CHR$(13);
204 P$=S3:GOSUB 50:PRINT#3,P$;CHR$(13);
205 P$=S4:GOSUB 50:PRINT#3,P$;CHR$(13);
206 P$=S5:GOSUB 50:PRINT#3,P$;CHR$(13);
207 GOSUB 20:GOSUB 50:PRINT#3,P$;CHR$(13);
208 P$=S7:GOSUB 50:PRINT#3,P$;CHR$(13);
209 P$=S8:GOSUB 50:PRINT#3,P$;CHR$(13);
220 PRINT#3,CHR$(27);
225 NEXT L:
226 PRINT#3,CHR$(26);
230 CLOSE:END
9999 SAVE"mf21b/bas:2"

```

The Programmer's Corner

@EXMEM - Extended MEMORY Management

Here's a real treat for the 128K Model 4/4p/4d folks. How many times have you wished for an easy way to gain the use of your computer's extended memory from languages such as BASIC and C? How many of you have succeeded? Well with this TMQ issue's Programmer's Corner, you will be able to.

Let's first take a look at why accessing the extended memory from BASIC and C is so difficult. I mean, after all, the DOS does provide a Supervisor Call (SVC) named "@BANK" for this purpose, doesn't it? You bet it does; but because of some operational interference in BASIC and C, @BANK cannot be used. Here's why.

In a standard 128K Model 4, the "extra" 64K is imaged as two different banks of 32K each. By means of some hardware port control coded in the DOS @BANK handler, the 32K Z80 address space from X'8000' through X'FFFF' can be configured to access either of those two banks or the normal upper half bank of the original 64K address space. Thus, there are three banks which can be switched into this address space. When one gets switched in, what was there gets switched out. Ah, there's the rub!

The BASIC interpreter (as well as compiled BASIC programs), compiled programs written in C, as well as other high level languages maintain a program stack pointer in high memory; that's it, the address space somewhere up in that region of X'8000' through X'FFFF'. It would certainly be catastrophic to have the memory normally being used by the stack to be switched out; sort of like having the rug pulled out from under you. The DOS must protect against this kind of foolishness; thus, any time @BANK is called to switch banks when the Z80 stack pointer (SP) is in that swap region, it just returns an error code and does not perform the requested bank swap operation. Kind of makes it tough to use the bank switching from BASIC and C.

Here's where this @EXMEM handler comes to your rescue. To begin with, it resides in low memory. It also switches to a low-

memory stack pointer of its own. Third, it provides for easy input/output of data from/to the memory banks available to you. The extended memory handler allows for easier access of extended memory storage than through the normal @BANK handler. It does this by giving you both character (one byte) and page (256 bytes) I/O by bank and relative offset. As the @EXMEM handler provides its own control over both disabling of interrupts and stack management, the programmer can ignore those maintenance items. This makes it easy to use extended memory from languages which use a high memory stack such as BASIC and C.

The Supervisor Call number (SVC) of the added @EXMEM handler is 108D. The following information notes the protocol of the four extended memory management functions.

Get a character from a bank

Registers affected: AF, B
 B => 1
 C => Requested bank
 HL => Relative offset in the bank, 0000H-7FFFH
 A <= The character returned
 Z <= Set if successful

Put a character into a bank

Registers affected: AF, B
 B => 2
 C => Requested bank
 E => The character to put
 HL => Relative offset in the bank, 0000H-7FFFH
 Z <= Set if successful

Get a page (256 bytes) from a bank to your buffer

Registers affected: AF, B, HL, DE
 B => 3
 C => Requested bank
 HL => Relative offset in the bank, 0000H-7FFFH
 DE => Address of your 256-byte buffer
 Z <= Set if successful

Put a page (256 bytes) from your buffer to a bank

Registers affected: AF, B, HL, DE
 B => 4
 C => Requested bank
 HL => Relative offset in the bank,
 0000H-7F00H
 DE => Address of your 256-byte buffer
 A <= The character returned
 Z <= Set if successful

Note that the relative offset can also be in the range 8000H-OFFFFH. The @EXMEM handler always sets the high-order bit of register H to ensure that the bank offset value is addressed into the bank swap image area. Function 4, putpage, checks to make sure that your bank copy won't wrap past X'FFFF'; although getpage will permit you to read past X'FFFF'. There is no error checking on the address of your buffer in either function 3 or function 4. If your buffer is below X'2400', it is possible to corrupt the system; however, since you may be wanting to swap out low memory, @EXMEM does not prohibit it.

@EXMEM always does double buffering through a low memory intermediate buffer (at address X'2300'-X'23FF') in case your buffer overlaps the bank swap image area. Although this may be a tad slower in operation when it is not necessary (i.e. when your buffer is totally below X'8000'), it enables the extended memory handler to be smaller; thus providing maximum space for any other low memory usage. After all, @EXMEM takes about 145 bytes.

If you were to look at the memory-resident code of the MEMDISK/DCT RAM disk driver (which is contained in Volume III of THE SOURCE TO TRSDOS 6.2), you would begin to realize that the MEMDISK driver has to do a lot of the bank switching and finagling that @EXMEM has to do. It is interesting to note that by using @EXMEM, a MEMDISK could be written which would (1) be a lot smaller since it could directly use the page I/O functions of @EXMEM; and (2) could actually reside in high-memory; thus freeing up valuable low-memory space. We don't have the space (nor the time) in this issue to devote to such topics;

however, we may introduce a revised MEMDISK in TMQ I.iv which will use @EXMEM!

Before going into the EXMEM/CMD program which installs the new SVC, let's examine a few potential uses. Using the SVC interface facility provided in LS-DOS 6.3's BASIC, it would be extremely easy to set aside 2048 bytes of array space (an integer array of length 1024, for instance). This could be used as a screen buffer; swapping to the screen with the @VDCTL service call. With @EXMEM installed, you could take a bank of memory and use it to store 16 different screens. A very short SUBROUTINE could swap eight pages of banked memory with your array storage. What could you do with that?

Need more data storage. It shouldn't take a half-decent programmer too long to come up with a short routine to swap a data array with banked memory. All sorts of interesting possibilities can open up if you use your imagination and ingenuity.

For the C programmer (using our PRO-MC, of course), we have provided a function interface in this issue along with a quick demonstration program for accessing banked memory. The file named "MCBANK/ASM" contains the following five functions:

```
int chkx(bank);
char getx(bank,offset);
char putx(c, bank, offset);
char * getpgx(buffer, bank, offset);
char * putpgx(buffer, bank, offset);
char c; int bank; unsigned offset; char
*buffer;
```

I think description of the arguments is pretty obvious; thus, I won't belabor the point here. The BANKTEST/CCC program which follows is a good illustration of the usage. A few points are in order. Chkx() returns TRUE if a bank is available or FALSE if it's already in use. Getx() returns with the character requested or EOF if the DOS gives an error. Putx() will return with the character put else EOF; this is similar to the behavior of both getc() and putc(). Both getpgx() and putpgx() return with a pointer to the buffer. This greatly simplifies the use of these two functions as you can determine by examining the BANKTEST/CCC program.

One last point before proceeding with the programs. We would hope that some of our readers who are skilled at making use of this added memory access capability would provide some welcomed input to THE MISOSYS QUARTERLY. Please let us know how you are

using this facility. By providing us with a small article (words please, not just a program listing), you will enormously help those readers struggling with advanced concepts.

```

/* banktest/ccc - simple banked memory test
 * Copyright 1987 MISOSYS, Inc., All rights reserved
 * Version 1.0 - 03/10/87
 */
#include stdio.h
extern char* memset(), getpgx();
main()
{  int c, bank, rc; unsigned loop; char *ram, pagein[256], pageout[256];
   for (bank=1; bank<=31; bank++)
/*
 * Test putx for all available banks
 */
   if (chkx(bank))
   {  printf("\nBank %d is available\n",bank);
      ram=0;
      for (loop=0; loop<=32767; ++loop)
      {  c = loop & 15;
         if ((putx(c,bank,ram++)) !=c) break;
         if (!(loop&0x00ff))
            printf("\xldBank %d, Page %d",bank,loop>>8);
      }
/*
 * Test getx for all available banks
 */
      ram=0; puts("\ngetx() tests\n");
      for (loop=0; loop<=32767; loop+=17)
      {  if ((c=getx(bank,ram++))==EOF) break;
         printf("\xldloop %d: %lx",loop,c);
      }
/*
 * Test page I/O for all available banks
 */
      ram=0; puts("\nPage I/O tests\n");
      for (loop=0; loop<=32767; loop+=256)
      {  if (putpgx(memset(pageout,loop>>8,256),bank,loop)==EOF) break;
         printf("\xldloop %d",loop);
         if (memcmp(getpgx(pagein,bank,loop),pageout,256))
            printf("Page error: %d\n",loop);
      }
   }
   else
      printf("\nBank %d is NOT available\n",bank);
}

```



```

00001 ;MCBANK/ASM - MC interface to SVC-108 (@EXMEM)
00002 ;*==*
00003 ;      Copyright 1987 MISOSYS, Inc., All rights reserved
00004 ;      Version 1.0 - March 10, 1987
00005 ;*==*
0066  00006 @BANK EQU    102
006C  00007 @EXMEM EQU    108
      00008 PUBLIC  CHKX,GETX,PUTX,GETPGX,PUTPGX
      00009 CSEG
0000' F1  00010 CHKX: POP    AF          ;Return code
0001' C1  00011      POP    BC          ;Get bank number
0002' C5  00012      PUSH   BC
0003' F5  00013      PUSH   AF
0004' 0602 00014      LD     B,2          ;Bank availability
0006' 3E66 00015      LD     A,@BANK
0008' EF    00016      RST    40
0009' 210100 00017      LD     HL,1          ;Init for success
000C' C8    00018      RET     Z
000D' 2B    00019      DEC    HL          ;Set to NULL
000E' C9    00020      RET
000F' F1    00021 GETX: POP    AF          ;Return code
0010' C1    00022      POP    BC          ;Get bank number
0011' E1    00023      POP    HL          ;Get address
0012' E5    00024      PUSH   HL
0013' C5    00025      PUSH   BC
0014' F5    00026      PUSH   AF
0015' 0601 00027      LD     B,1          ;Set for getchar
0017' CD1D00' 00028      CALL  DO_EXM
001A' C0    00029      RET    NZ          ;Back if error
001B' 6F    00030      LD     L,A          ;Put char into HL
001C' C9    00031      RET
001D' 3E6C 00032 DO_EXM: LD     A,@EXMEM
001F' EF    00033 DO_SVC: RST    40
0020' 210000 00034 ERRCK: LD     HL,0          ;Init for success
0023' C8    00035      RET     Z
0024' 2B    00036      DEC    HL          ;Set to EOF (-1)
0025' C9    00037      RET
0026' F1    00038 PUTX: POP    AF          ;Return code
0027' D1    00039      POP    DE          ;Get character
0028' C1    00040      POP    BC          ;Get bank number
0029' E1    00041      POP    HL          ;Get address
002A' E5    00042      PUSH   HL
002B' C5    00043      PUSH   BC
002C' D5    00044      PUSH   DE
002D' F5    00045      PUSH   AF
002E' D5    00046      PUSH   DE          ;Save char again
002F' 0602 00047      LD     B,2          ;Set for putchar
0031' CD1D00' 00048      CALL  DO_EXM
0034' D1    00049      POP    DE
0035' C0    00050      RET    NZ          ;Back if error
0036' 6B    00051      LD     L,E          ;Return with the char put
0037' C9    00052      RET
0038' F1    00053 GETPGX: POP    AF          ;Return code
0039' D1    00054      POP    DE          ;Get page buffer
003A' C1    00055      POP    BC          ;Get bank number
003B' E1    00056      POP    HL          ;Get address
003C' E5    00057      PUSH   HL

```

```

003D' C5      00058      PUSH      BC
003E' D5      00059      PUSH      DE
003F' F5      00060      PUSH      AF
0040' 0603    00061      LD        B,3          ;Set for getpage
0042' D5      00062 PAGEIO: PUSH      DE          ;Save page buffer
0043' CD1D00' 00063      CALL     DO_EXM
0046' D1      00064      POP       DE          ;Get page buffer
0047' C0      00065      RET       NZ          ;Back on error
0048' EB      00066      EX        DE,HL       ;Buffer pointer to HL
0049' C9      00067      RET
004A' F1      00068 PUTPGX: POP       AF          ;Return code
004B' D1      00069      POP       DE          ;Get page buffer
004C' C1      00070      POP       BC          ;Get bank number
004D' E1      00071      POP       HL          ;Get address
004E' E5      00072      PUSH     HL
004F' C5      00073      PUSH     BC
0050' D5      00074      PUSH     DE
0051' F5      00075      PUSH     AF
0052' 0604    00076      LD        B,4          ;Set for getpage
0054' C34200' 00077      JP        PAGEIO
0000      00078      END

```

```

00001 ;EXMEM/ASM - Adds @EXMEM SVC to LS-DOS 6.3 - 03/10/87
00002 ;*==*
00003 ;      Copyright 1987 MISOSYS, Inc., All rights reserved
00004 ;*==*
000A      00005 @DSPLY EQU      10
000C      00006 @LOGOT EQU     12
0015      00007 @ABORT EQU     21
0052      00008 @GTDCB EQU     82
0065      00009 @FLAGSS$ EQU    101
2300      00010 BBUFF$ EQU    2300H          ;Low memory swap buffer
          00011 ;*==*
2600      00012      ORG      2600H
2600 219626 00013 BEGIN LD      HL,HELLO$          ;Welcome
2603 3E0A   00014      LD      A,@DSPLY
2605 EF     00015      RST     40
2606 114B49 00016      LD      DE,'IK'
2609 3E52   00017      LD      A,@GTDCB          ;Locate low memory ptr
260B EF     00018      RST     40
260C 2051   00019      JR      NZ,NOTLOW        ;Error, can't find *KI
260E 2B     00020      DEC     HL
260F 56     00021      LD      D,(HL)          ;P/u pointer to
2610 2B     00022      DEC     HL              ; start of free
2611 5E     00023      LD      E,(HL)          ; low core
2612 ED534026 00024      LD      (LCPTR+1),DE    ;Save loc for later
2616 E5     00025      PUSH    HL              ;Save low core ptr
2617 219400 00026      LD      HL,MODEND-MODBGN
261A 19     00027      ADD     HL,DE            ;Start + driver length
261B E5     00028      PUSH    HL
261C 2B     00029      DEC     HL              ;Point to last byte
261D 227126 00030      LD      (SVEND+1),HL
2620 010013 00031      LD      BC,1300H        ;Max addr + 1
2623 AF     00032      XOR     A
2624 ED42   00033      SBC     HL,BC
2626 D1     00034      POP     DE              ;Rcvr new lc

```

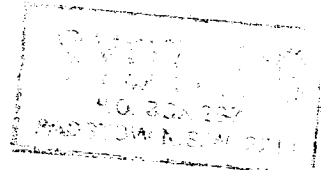
```

2627 E1      00035      POP      HL          ;Rcvr low core ptr
2628 3039    00036      JR        NC,NOROOM   ;No memory if => 1300H
                00037      ;*==*
                00038      ;
                00039      ;*==*
                Relocate absolute addresses
262A 73      00040      LD        (HL),E      ;Stuff low core ptr
262B 2C      00041      INC      L            ; with new low
262C 72      00042      LD        (HL),D
262D CD6C26  00043      CALL     RELO         ;Relocate vectors
                00044      ;*==*
                00045      ;
                00046      ;*==*
                Fixup @EXMEM's calls to @BANK
2630 3E65    00047      LD        A,@FLAGS$   ;Get flags pointer into IY
2632 EF      00048      RST      40
2633 FD661A  00049      LD        H,(IY+26)   ;Point to SVCTAB
2636 2ECC     00050      LD        L,102*2     ;Point to @BANK entry
2638 7E      00051      LD        A,(HL)      ;Get @BANK pointer
2639 23      00052      INC      HL
263A 66      00053      LD        H,(HL)
263B 6F      00054      LD        L,A          ;Update @BANK calling
263C 22C427  00055      LD        (SVC102),HL ; for speedier @BANK
263F 110000  00056 LCPTR      LD        DE,$-$      ;Low core pointer
                00057      ;*==*
                00058      ;
                00059      ;*==*
                Move module to memory
2642 215A27  00060 MOVMOD      LD        HL,MODBGN
2645 019400  00061      LD        BC,MODEND-MODBGN
2648 EDB0     00062      LDIR
264A FD661A  00063      LD        H,(IY+26)   ;Point to SVCTAB
264D 2ED8     00064      LD        L,108*2     ;Point to SVC_108 slot
264F 118C27  00065      LD        DE,EXMEM@
2650          00066 RX01      EQU      $-2
2652 73      00067      LD        (HL),E      ;Update SVCTAB
2653 23      00068      INC      HL
2654 72      00069      LD        (HL),D
2655 21FD26  00070      LD        HL,INSTLD$
2658 3E0A     00071      LD        A,@DSPLY
265A EF      00072      RST      40
265B 210000  00073      LD        HL,0         ;Indicate successful
265E C9      00074      RET          ; to DOS
265F 211F27  00075 NOTLOW     LD        HL,NOTLOW$
2662 DD      00076      DB        ODDH
2663 214027  00077 NOROOM     LD        HL,NOROOM$
2666 3E0C     00078      LD        A,@LOGOT
2668 EF      00079      RST      40
2669 3E15     00080      LD        A,@ABORT
266B EF      00081      RST      40
                00082      ;*==*
                00083      ;
                00084      ;*==*
                Relocate internal references in driver
266C DD21EE27 00085 RELO      LD        IX,RELTAB   ;Point to relocation tbl
2670 210000  00086 SVEND     LD        HL,$-$      ;Find distance to move
2673 225C27  00087      LD        (MODBGN+2),HL ;Set last byte used
2676 11ED27  00088      LD        DE,MODEND-1
2679 B7      00089      OR        A          ;Clear carry flag
267A ED52     00090      SBC      HL,DE
267C 44      00091      LD        B,H        ;Move to BC

```

```

267D 4D      00092      LD      C,L
267E DD6E00   00093 RLOOP    LD      L,(IX)      ;Get address to change
2681 DD6601   00094      LD      H,(IX+1)
2684 7D      00095      LD      A,L
2685 B4      00096      OR      H
2686 C8      00097      RET     Z
2687 5E      00098      LD      E,(HL)     ;P/U address
2688 23      00099      INC    HL
2689 56      00100      LD      D,(HL)
268A EB      00101      EX     DE,HL      ;Offset it
268B 09      00102      ADD    HL,BC
268C EB      00103      EX     DE,HL
268D 72      00104      LD      (HL),D    ;And put back
268E 2B      00105      DEC    HL
268F 73      00106      LD      (HL),E
2690 DD23     00107      INC    IX
2692 DD23     00108      INC    IX
2694 18E8     00109      JR     RLOOP      ;Loop till done
          00110 ;
2696 0A      00111 HELLO$ DB 10,'EXMEM 1.0 - Extended memory handler for LS-DOS 6.3'
          45 58 4D 45 4D 20 31 2E 30 20 2D 20 45 78 74 65
          6E 64 65 64 20 6D 65 6D 6F 72 79 20 68 61 6E 64
          6C 65 72 20 66 6F 72 20 4C 53 2D 44 4F 53 20 36 2E 33
26C9 0A      00112      DB     10,'Copyright 1987 MISOSYS, Inc., All rights
reserved',10,13
          43 6F 70 79 72 69 67 68 74 20 31 39 38 37 20 4D
          49 53 4F 53 59 53 2C 20 49 6E 63 2E 2C 20 41 6C
          6C 20 72 69 67 68 74 73 20 72 65 73 65 72 76 65 64 0A 0D
26FD 53      00113 INSTLD$ DB 'SVC_108 (@EXMEM) is now installed',13
          56 43 5F 31 30 38 20 28 40 45 58 4D 45 4D 29 20
          69 73 20 6E 6F 77 20 69 6E 73 74 61 6C 6C 65 64 0D
271F 43      00114 NOTLOW$ DB 'Can't locate low memory pointer!',13
          61 6E 27 74 20 6C 6F 63 61 74 65 20 6C 6F 77 20
          6D 65 6D 6F 72 79 20 70 6F 69 6E 74 65 72 21 0D
2740 4E      00115 NOROOM$ DB 'No memory space available',13
          6F 20 6D 65 6D 6F 72 79
          20 73 70 61 63 65 20 61
          76 61 69 6C 61 62 6C 65
          0D
          00116 ;*==*
          00117 ;      Extended memory access
          00118 ;
          00119 ; B =>func [1=getchar, 2=putchar, 3=getpage, 4=putpage]
          00120 ; C =>bank
          00121 ; HL=>bank offset
          00122 ; DE=>UBUFF$
          00123 ; E => char
          00124 ; Z <= OK
          00125 ;*==*
275A 1830     00126 MODBGN JR     EXMEM@
275C 0000     00127      DW     $-$
275E 03      00128      DB     3,'$XM'
          24 58 4D
          00129 ;*==*
          00130 ;      Get char from bank (C) at (HL)
          00131 ;*==*
2762 CDC627   00132 GETCHAR CALL  MOVSTAK      ;Switch to our stack
    
```



```

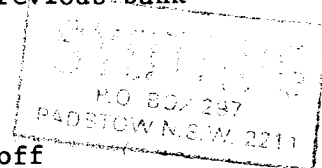
2763          00133 RX02   EQU    $-2
2765 CDC227   00134 RXB1   CALL   @BANK           ;Get requested bank
2768 C0       00135       RET    NZ             ;Back on error
2769 7E       00136       LD     A,(HL)
276A F5       00137 GETEXIT PUSH   AF
276B CCC227   00138 RXB2   CALL   Z,@BANK       ;Restore previous bank
276E F1       00139       POP    AF
276F C9       00140       RET
          00141 ;*==*
          00142 ;           Put char from (E) to bank (C) at (HL)
          00143 ;*==*
2770 CDC627   00144 PUTCHAR CALL   MOVSTAK       ;Switch to our stack
2771          00145 RX03   EQU    $-2
2773 CDC227   00146 RXB3   CALL   @BANK
2776 C0       00147       RET    NZ
2777 73       00148       LD     (HL),E       ;Char to memory
2778 18F0     00149       JR     GETEXIT       ;Reinstall previous bank
          00150 ;*==*
          00151 ;           Write page from bank (C) at (HL) to (DE)
          00152 ;*==*
277A CDC627   00153 GETPAGE CALL   MOVSTAK       ;Switch to our stack
277B          00154 RX04   EQU    $-2
277D CDC227   00155 RXB4   CALL   @BANK       ;Install the bank
2780 C0       00156       RET    NZ             ;Back on error
2781 CDB427   00157       CALL  MOVMEM
2782          00158 RX10   EQU    $-2
2784 F5       00159       PUSH  AF
2785 010001   00160       LD     BC,256
2788 EDB0     00161       LDIR              ;Move page into the bank
278A F1       00162       POP    AF
278B C9       00163       RET
          00164 ;*==*
          00165 ;           @EXMEM SVC entry point
          00166 ;*==*
278C CBFC     00167 EXMEM@  SET    7,H           ;Ensure upper mem address
278E 05       00168       DEC    B             ;Check on function 1
278F 28D1     00169       JR     Z,GETCHAR
2791 05       00170       DEC    B             ;Check on function 2
2792 28DC     00171       JR     Z,PUTCHAR
2794 05       00172       DEC    B             ;Check on function 3
2795 28E3     00173       JR     Z,GETPAGE
2797 05       00174       DEC    B             ;Check on function 4
2798 3E2B     00175 PERR   LD     A,43         ;SVC Parameter error
279A C0       00176       RET    NZ
          00177 ;*==*
          00178 ;           write page from (DE) to bank (C) at (HL)
          00179 ;*==*
279B 7C       00180       LD     A,H           ;Check for page 255
279C 3C       00181       INC    A
279D 2003     00182       JR     NZ,PUTPAGE
279F B5       00183       OR     L             ;Can't put into > OFF00H
27A0 20F6     00184       JR     NZ,PERR
27A2 CDC627   00185 PUTPAGE CALL   MOVSTAK       ;Switch to our stack
27A3          00186 RX05   EQU    $-2
27A5 EB       00187       EX     DE,HL       ;UBUFF to HL, offset to DE
27A6 CDB427   00188       CALL  MOVMEM       ;Install the bank, move
27A7          00189 RX11   EQU    $-2

```

```

27A9 C0      00190      RET      NZ      ;Back on error
27AA C5      00191      PUSH     BC      ;Save previous bank #
27AB 010001  00192      LD       BC,256  ;Page length
27AE EDB0     00193      LDIR    ;Move page into the bank
27B0 C1      00194      POP      BC      ;Recover previous bank #
27B1 C3C227  00195 RXB5     JP       @BANK   ;Reinstall previous bank
                00196 ;*==*
                00197 ; Routine common to page I/O
                00198 ;*==*
27B4 C5      00199 MOVMEM  PUSH     BC      ;Save previous bank #
27B5 D5      00200      PUSH     DE      ;Save UBUFF or offset
27B6 110023  00201      LD       DE,UBUFF$ ;Our buffer to DE
27B9 D5      00202      PUSH     DE
27BA 010001  00203      LD       BC,256  ;Move a page to our space
27BD EDB0     00204      LDIR
27BF E1      00205      POP      HL      ;Our buffer to HL
27C0 D1      00206      POP      DE      ;UBUFF or offset to DE
27C1 C1      00207      POP      BC      ;Recover previous bank #
27C2 79      00208 @BANK   LD       A,C     ;Simulate SVC handling
27C3 C30000  00209      JP       $-$     ;Reinstall previous bank
27C4                00210 SVC102 EQU     $-2
                00211 ;*==*
                00212 ; Routine to switch to our stack
                00213 ;*==*
27C6 F3      00214 MOVSTAK DI      ;Interrupts off
27C7 22D427  00215      LD       (HLSAV),HL ;Save reg_HL
27C8                00216 RX06 EQU     $-2
27CA E1      00217      POP      HL      ;Get RET address off of stack
27CB ED73EA27 00218      LD       (SPSAV),SP
27CD                00219 RX07 EQU     $-2
27CF 31E727  00220      LD       SP,STACK ;Set to new stack
27D0                00221 RX08 EQU     $-2
27D2 E5      00222      PUSH     HL      ;Set RET to MOVSTAK caller
27D3 210000  00223      LD       HL,$-$   ;Restore reg_HL
27D4                00224 HLSAV EQU     $-2
27D6 C9      00225      RET
27D7 00      00226      DC       16,0    ;Stack area
                00 00 00 00 00 00 00 00 00 00 00 00 00 00
27E7 E927     00227 STACK  DW      RESTORE ;Pointer to restore code
27E7                00228 RX09 EQU     $-2
27E9 310000  00229 RESTORE LD     SP,$-$ ;Restore old stack
27EA                00230 SPSAV EQU     $-2
27EC FB      00231      EI          ;Interrupts on
27ED C9      00232      RET        ;Return to @EXMEM caller
27EE                00233 MODEND EQU     $
                00234 ;
27EE 5026     00235 RELTAB  DW RX01,RX02,RX03,RX04,RX05,RX06,RX07,RX08,RX09,RX10
                6327 7127 7B27 A327 C827 CD27 D027 E727 8227
2802 A727     00236      DW       RX11,RXB1+1,RXB2+1,RXB3+1,RXB4+1,RXB5+1,0
                6627 6C27 7427 7E27 B227 0000
2600                00237      END       BEGIN

```



Getting into MS-DOS Assembly

This section will initiate a new column for our TMQ readers. Now that MISOSYS is devoting most of its efforts to marketing our MS-DOS product line, we need to devote space in future QUARTERLIES to MS-DOS topics. Since we sell a powerful but easy to use MS-DOS assembler, what better column to add to THE PROGRAMMER'S CORNER than a column on MS-DOS assembly language.

Some of the material presented here will be direct from MISOSYS; some will come from Phil Oliver, the author of our ED/ASM-86 package. We hope to encourage those of you who have been tinkering with 8086 assembly for awhile to offer up some of the hints and kinks you have learned. Don't be afraid to get your feet wet!

This will be a very short introduction into assembly language for the MS-DOS class of machines. I think many of our readers have acquired PC compatibles; many more will be doing so in the near future. The Z80 programming types have worked in a very comfortable environment - usually an

```

1: 0100-BA0B01          BEGIN: MOV     DX,HELLO      ;Point to message
2: 0103-B409            MOV     AH,9        ;Display string SVC
3: 0105-CD21            INT     33
4: 0107-B400            MOV     AH,0        ;Terminate pgm SVC
5: 0109-CD21            INT     33
6: 010B-48656C6C6F20776F
   726C640A0D24        HELLO: DB     'Hello world',10,13,'$'
7: 0119-                END         BEGIN

```

0 Error(s) Detected

Doesn't look like the convoluted stuff you see in the PC magazines, does it? ED/ASM-86 can directly assemble this into either a ".COM" file or a ".EXE" file. We'll be bringing you more easy stuff on 8086 assembly language in each subsequent issue of TMQ. This is just to whet your appetite.

integrated editor and assembler. If you have in any way taken a look at small PC assembly programs, I am sure you have been turned off by the complexity of the 8086 assembly language. Well, maybe it has not been the language so much, but rather the hassle with an assembler environment such as MASM.

Here then, to start you off in the right direction, let me show you a very simple 8086 ".COM" program written using our own assembler - ED/ASM-86 (still at its introductory price of \$89.95 + \$5 S&H until April 30th 1987 - \$124.95 thereafter). This assembler doesn't force you into some complicated environment of separate editor, separate assembler, separate linker, and separate debugger. ED/ASM-86 is an integrated editor (line and screen), assembler, linker, debugger, and disassembler; all in a single 60K+.EXE file. ED/ASM-86 can automatically handle so much of the quirky 8086 assembly protocol that you would hardly know you left the nice safe world of the Z80. Here then is "Hello world".

The Tower of Babel

Questions on Assemblers

(Fm: Theodore Masterton) I am busy hunkering down for the long haul with my 4p. Since owning an assembler seems to be a good survival tactic, I am about to go after one. I have no intention of learning assembly language but would like to be able to type in those 80-Micro listings I have been saving since 82.

For CP/M I have ASM.COM, a non-macro non-reloc type assembler, and will soon have M80, a Microsoft standard I hear. This will allow me to assemble some of that PD CP/M stuff out there.

Question 1: If a Z80 is a Z80, shouldn't I be able to assemble TRS-80 stuff with the CP/M assemblers? I know that the DOS's are much different, but won't calls to DOS routines simply assemble for TRSDOS or LDOS, be unusable on CP/M, but OK for Tandy? (YOU SEE I AM REALLY NAIVE ON THIS TOPIC!)

Question 2: If not, which Assembler should I own? It needs to be cheap and simple. Maybe PD? Or how about that little one that came with the Junk Box deal on Assembly Language Tutor? Will that one work for me?

Question 3, the really dumb one: When I read an Assembly Language listing I notice that everything is nicely laid out in columns. Yet I have never heard that those column placements were critical. Are they? I mean back in Fortran Days....

(Fm: Les Mikesell) (1) CP/M & TRSDOS assemblers would assemble the same code, but there is a difference in the disk storage for executable programs and the assembler/loader must know how to store in the correct format. The /REL files created by M80 can, in fact, be loaded by L80 on either system.

(2) EDAS is probably the easiest assembler to use unless you need to link to microsoft compiler code (then you need M80 or MRAS).

(3) Most (not all) assemblers treat "white space" (spaces or tabs) as field separators and are not too picky about column placement.

(Fm: LDOS Support) (1) Yes. The problem is going to be getting the executable over to LDOS/TRSDOS 6 so that you can run it.

(2) You should own Pro-MRAS. Somewhat inexpensive (\$75) but not simple. Then again, it isn't any worse than M80, so you are paying for that already, so to speak. The Tutor assembler is pretty much useless outside of the Tutor.

(3) No, the assembler does neaten some things up for you. You do need to get labels flush left, and everything else indented with some white space, namely spaces or tabs. None of this Col 1, 6, 7

(Fm: Gary Phillips) The CP/M assembler (at least, the free ASM.COM one) cannot generate load modules in TRSDOS load format. CP/M object files are normally generated to load at address 0100H and simply contain all the binary bytes of memory from there to the end of the program in one continuous stream. ASM.COM produces a standard Intel HEX file (actually ASCII hex, not binary, with some added loading information), and LOAD.COM converts that into the binary stream file that CP/M uses. Actually, you CAN take the HEX file output by ASM.COM, and run it through your own conversion program to produce a suitable TRSDOS load file, but I don't know of commercial software to do that for you. And in any case, you would then have the added problem of getting the file from your CP/M disk onto a TRSDOS disk (different disk formats).

In short, I recommend you get a TRS-80 assembler if you want to use TRS-80 assembly language programs in source form. If you were serious about assembly language programming, then there is only one to consider, and that is the Pro-Create assembler from MISOSYS. Most of the source files you see in magazines now are written for this assembler. However, you can get by with one of the less expensive

ones IF you are willing to make an occasional change to suit the peculiarities of your own chosen software. Choose one that accepts both numbered (a la EDTASM) as well as unnumbered (a la EDAS or PRO-CREATE) source files, and preferably one that also handles conditional assembly and macros, because many magazine programs use those features and if you don't know much assembly language you will have difficulty translating into the simpler format. The simple non-relocating, non-macro assemblers are OK for just learning, or for doing simple routines to be called from BASIC, but are pretty clumsy if you are working with significantly complex programs.

Assembly language is "free-form" in that columns are not critical, although there are traditional columns to use in formatting your source. Labels must begin in column 1. Instruction mnemonics or macro calls in column 2 or after.

Mixed case + PRO-MRAS

(Fm: Scott Russell) I just got PRO-SAID with PRO-MRAS and I'm VERY new to it. I don't specify any MODE. Does it default? Like to TEXT? Anyway, my problem was that PRO-MRAS flagged my label "_Zero" as being a "Bad Label" which I didn't think it would be.

(Fm: jeff brenton) SAID defaults to a TEXT-like mode; TEXT allows you to specify a default file extension, such as /TXT or /LTR, while the default mode is NO extension unless YOU put it there.

(Fm: John Garner) SAID is semi-smart. In ASM mode, it converts lower case to upper EXCEPT in comments and quoted strings.

MRAS switches -CI and -GC

(Fm: Bob Haynes) I'm confused by the -CI and -GC switches in MRAS. Am writing a small PRONTO application, so MLINK/MLIB utilities are not invoked. My source code

(normally w/o OPTION statements) is OK - invoking:

```
MRAS filename/ASM -we -ne -gc -ci
```

produces a good /CIM file with no errors.

The -gc docs SAY -gc is automatically invoked by the -ci switch, but this doesn't seem to work; invoking

```
MRAS filename/ASM -we -ne -ci
```

produces "Relocation reference error"s, with a bad /REL file. Inserting the switches as OPTIONs doesn't help either. Invoked "MRAS filename/ASM {no switches}" on the following:

```
TITLE <filename>
OPTION WE
OPTION NE
OPTION GC
OPTION CI
; rest of code here...
```

and got "branch out of range" and "multiple definition" errors, and a bad /REL file. Deleted "OPTION CI" and re-tried, and got "Relocation reference error"s with a bad /REL file (the OPTION GC was completely ignored).

The only way I can get a good direct assembly is by using both -gc and -ci on the command line, and considering the usual reliability of your software, I'm inclined to believe I must be doing something wrong. What gives?

(Fm: MISOSYS) This was covered on page 77 of TMQ I.ii. The -GC switch must be entered in from the command line as it is too late for a useful detection via the OPTION statement. The default of CSEG is set before the source file is read. Once the file is accessed, that segment default can only be altered by a xSEG statement. It's not a bug in the code, just an oversight in the docs. It should have been noted.

(Fm: Bob Haynes) Thanks for the prompt response. Will adjust my docs to reflect this minor restriction. I'm still not

clear, though, on why a -CI from the command line does not invoke the -GC option. That is specified in both the -CI and -GC descriptions. Or am I overlooking something else?

(Fm: MISOSYS) Bob, I'ts just not in the code. You'll just have to use both -CI and -GC on the command line until (and if) we do another version of MRAS.

C Language

Retrieving date/time with MC

(Fm: H. Brothers) I'm writing a program in MC to work with LS-DOS 6.3. The program needs to retrieve the date & time of files on disk. Since MC doesn't have a function to directly get the date & time of a file (does it?), my plan is to use a ML module that opens the file with @OPEN, gets the drive and DEC from the FCB, closes the file, then gets the directory entry with @DIRRD. Then the program would lift the date/time from the directory entry and send it back to the MC program. There's no problem with doing this, but I keep having the feeling that I'm missing something that would make the task much easier -- am I?

(Fm: MISOSYS) You are, of course the function may need updating for 6.3. Take a look at fstat(). This function obtains the status of an open file. The data is returned in a structure and is a standard UNIX System V function. Of course, not all of the data documented under UNIX is available since our DOS doesn't support all of it. The st_mtime element currently contains the mod date. That could be upgraded to include the time piece as well (off hand, I believe that the element is a long packed like the other time fields).

(Fm: H. Brothers) I had looked at fstat() briefly and then figured that since it didn't know about the 6.3 date stamp, it wouldn't be useful. Thanks for turning my attention to it again. Obviously, the st_uid field will hold the new time stamp from dir +18,19 and it should be a simple

matter to extract the actual values from there.

I have a question, though. The 6.3 docs that I have say that bits 0-2 of dir+2 should no longer be considered the primary year field. That makes sense, but what happens to those bits? And since bits 3-7 still hold the day, what happens to the st_mtime field of the stat structure that fstat() returns?

(Fm: MISOSYS) Hardin, Under the x.3 release, the existing date field will be simultaneously maintained with the year masked to a three bit value. Thus, after 1987, the existing year field would look like 80, 81, 82,... instead of 88, 89, 90,... The reason for that is to allow earlier releases or other DOS's to continue to read the disk and get something somewhat meaningful. Obviously, when I redo the fstat() function, I need it to be smart concerning the disk vintage.

Wildcard fix for MC & M80

(Fm: MISOSYS) I got around to checking out a reported problem of difficulty with the WILDCARD option of MC when using it with M80 (i.e. the MC80/JCL file). I traced the problem to a missing instruction in the M80/H file. Please add the line,

```
LD HL,@_WILDCARD
```

to precede the line 'JP \$WILDCARD###' near the end of the file. Make sure that the '#asm' and '#endasm' statements remain in lower case. That will cure your wildcard problem.

Puzzled on PRO-MC assembler needs

(Fm: Gary Phillips) Hardin Bros. gave your C compiler a really great review in December 80-Micro. I take it they really mean to say you need EITHER Pro-Create OR M80/L80, rather than BOTH. Does the MC package include a linker of its own like some other C compilers, or will the linker from Pro-Create do the job? I am (I think I've mentioned before) a die-hard Pascal

user, but there just isn't a good Pascal for LS-DOS and I much prefer LS-DOS to CP/M where I have Turbo Pascal. Consequently, I may have to switch to C. I have Alcor C, but it's not really suitable for serious work because it's much too slow and bulky. Now if we could just convince you to give us a standard Pascal on the same professional level as Pro-Create and Pro-MC!! (Of course, I mean SELL, not GIVE.)

(Fm: MISOSY) First let's clear up some confusion. The old review pertained to our earlier C compiler - LC. MC is a complete implementation of K&R with some extensions and an extensive library with most functions compatible to UNIX System V. MC requires the use of either MRAS or M80. LC came bundled with EDAS. LC did not use REL libraries but MC does. There is a review of MC in the February 1987 issue of 80 MICRO.

If someone out there wanted to implement a good PASCAL for us to publish, we would be certainly interested in it. The only PASCAL for TRSDOS-compatible DOSs besides Tiny PASCAL and ALCOR's (Tandy's) was PASCAL-80 written by Phelps Gates and last published by New Classic. We'll take another.

Compiling separate files

(Fm: Les Mikesell) The usual practice when compiling c files that are to be linked together is to put all of the #defines, typedefs and struct definitions in a "/H" header file which is #included by each of the /C files.

What's ctype.h all about?

(Fm: Joe Smith) Can someone tell me what the CTYPE/H file contains in the C package? I'm using Alcor and am trying to compile some program in it. I can't readily see anything that would be in it, so thought I'd ask for help

(Fm: jeff brenton) delete the #include <ctype/h> line - it is not needed.

ctype.h is a header that #defines macros for the issomething() functions, like isascii(), isupper(), etc. Most C compilers include library routines for these functions, and will use them if CTYPE.H is not included.

The macro versions in CTYPE will (usually) be faster than calling functions, but will take up more space in the program. They can also cause a lot of insidious bugs if they are not implemented properly; the classic macro problem is:

```
toupper(*s++);
```

which will produce proper results with a function-type toupper(), but may cause BIG problems if it is implemented as a macro!

On assigning character pointers

(Fm: Bryan Headley) A question about MC. Here is some partial code:

```
char *argv[], *s;
char inbuf[81];
gets(inbuf);
argv[1] = inbuf;
    ^ MC doesn't like this...
```

BUT, if I change things around to:

```
gets(inbuf);
s = inbuf;
argv[1] = s; /* MC likes this */
```

Have I been away from C for so long I'm getting goofy, or is MC doing something nasty to me?

Just for your info, I'm re-writing the nice "cc" that appeared in TMQ #1. It only handles up to 8 files. Here I am, trying to compile MicroEMACS (with it's 20 some-odd modules) and am stuck. What I intend to do is to implement the "-f" option, that reads options in from stdin as well as the command line. This means getting the command line, and associating argv[]s to the individual words in the command line. Fun and games, almost works right now, except for some bug-a-boos with FREE

(something to do with my using a higher element of argv[] than what was allocated by _main... no biggie)

(Fm: Bryan Headley) Been playing with cc again. Got around the FREE problem by passing the address of argv to my routine. My keyboard parsing routine, which reads more options for cc from stdin, determines the size of argv, malloc()s an equivalent space, plus one element, and sets about parsing command options (I realloc() whenever I have a new element of argv, basically). Here's the trip: "p", my local copy of argv, contains all of the command parameters. (I know, because I can print these to screen) I return the pointer to p back to argv, as in "argv = keyparse(&argc, argv);. Argv has the same address as "p" after that statement, but the array of pointers argv doesn't point to anything! How wild! Question: are the effects of malloc() and realloc() local only to the calling function? That is, when I return to my caller, does the carriage return back to a pumpkin? I don't recall this behaviour in UNIX, but that doesn't mean much...

(Fm: MISOSYS) malloc() and realloc() are not local to the calling function. On the other hand, my recollection is that you cannot fool around with argv and argc as they are not allocated via alloc() and thus are not part of the linked list of heap storage. Thus answer is just a rapid one without too much hard thinking.

(Fm: Bryan Headley) Roy - You are correct. Since leaving the message, I noticed the error of my ways. Has to do with argc being base-1, and argv[] being base-0! Fine, I set the pointers up, then I malloc() OVER that pointer! Didn't figure it out until I set up the MED wammie, and looked at the buffers after every malloc and realloc. MED, you know, is extremely tough to use in this fashion. I think I'll write a MED-like beast, that only does "display hex address x"____", and takes up maybe an 1/8 of the screen. With MED, I have to move the page, and then do an "@" to move to offset to what I want to see...

Plus, it takes the whole screen, and my memory isn't THAT sharp!

Anyway, got around argv[] allocation by allocating a like buffer, and then copying the contents of argv[] into it. Stands to realloc() fine that way.

So, I have my variation of 'cc' working. What I want to do is now bring the whole beast into System V standards. So, this means working with .i (/tok), .s (/asm), and .o (/obj) files, supporting the -l flag (specifically I want to do -lm to do an #option mathlib, but have the ability to ask for the userlibs as well) Have an idea on this, too:

```
#ifdef M
#option mathlib
#endif
#include SOURCE
```

with the idea of doing a 'mcp -dM -dSOURCE=thomas/ccc -o=thomas' Any ideas/comments? Ought to make a good article for TMQ.

(Fm: MQISOSYS) Keep plugging on the -cc stuff. I could use an article or two.

More on the "FILE" declaration

(Fm: Bob Zinn) MC: following program fragment gives "illegal function call" msg.

```
typedef char *FILE;
FILE *fp, *fopen();
main()
{
    fp = fopen();
}
```

The construction works using #define just fine. With the "updated" mc (per MISOSYS QUARTERLY I.ii) fixes and page 73 col 1 change to stdio/h, it no longer works.

Your DISK NOTES 6 should have included a copy of my stripper program along with cc, ccl, and comments.

(Fm: Rich Deglin) Here are the patches (hopefully the final ones) for the typedef problems. They are VERY large, since I had to completely rewrite the getdeclarator() function in MCDESUBS/CCC. The solution to the problem was to delay application of a typedef until after parsing the declaration as if there were no typedef used. For example, the declarations

```
typedef char *FILE;
FILE *fopen();
```

are parsed in the following manner. The typedef for FILE is parsed normally, as "pointer to". The fopen declaration is first parsed as "function returning pointer to", then (and only then) is the additional level of indirection from the typedef applied, yielding "function returning pointer to pointer to" as expected. These declarations are thus equivalent to,

```
char **fopen();
```

Another example; the declarations

```
typedef long EC[2];
EC ec_array[10];
```

are parsed in the following manner. The typedef for EC is parsed normally, as "array of 2". The ec_array declaration is first parsed as "array of 10", then (and only then) is the additional level of array from the typedef applied, yielding "array of 10 arrays of 2" as expected. These declarations are thus equivalent to

```
long ec_array[10][2];
```

Obviously this required a major overhaul of the affected code.

The previous typedef patches in TMQ I.ii MUST be applied before these, since I overlay the same areas of code!

I also discovered that the upper limit of declaration complexity really is 16 levels. I've tested this. The compiler version prior to this fix accepted such.

(Fm: MISOSYS) The patches are designated GETDEC51/FIX and GETDEC52/FIX for MC; GETDEC60/FIX, GETDEC61/FIX, and GETDEC62/FIX for PRO-MC. Note that these patches are so long that they are not going to be printed in THE PATCH CORNER (I really cannot expect anyone to type them in). They will be on DISK NOTES 7 and will be uploaded to our Compuserve forum. Anyone not having access to either, can return their master disks (both of them) in a protective cardboard mailer and we will update them at no charge. It would help if you could include a return address label.

Getchar() and the ROM keyboard driver

(Fm: Bob Zinn) Subject: getc()/getchr(). I am not sure if I have reported this before but just in case: getchr() does not work properly unless set *ki ki is done.

```
clone >testfile
where clone is
#include stdio/h
main() /* copy input to output */
{
    int c;
    while (( c=getchar()) != EOF )
        putchar(c);
}
```

will constantly write to disk, 00 bytes, when using the ROM KI driver. Substituting getc(stdin) for getchar() fixes the problem! Sure gave me fits when I first ran across it, as the problem was "intermittent" depending upon which configuration I was using. (model I, 48k, LDOS 5.1.4)

Initialization of automatic variables

(Fm: Les Mikesell) A couple of nits about MC that may or may not have been mentioned before: "entry" seems to be a reserved word - any use of it as a variable name invokes an "illegal declaration or typename" error.

Initialization of automatic variables with global variables or the variables passed to the function does not work. I don't have a K&R handy to check if a constant is

supposed to be required, but the unix SysV compiler accepts other variables.

(Fm: H. Brothers) K&R says, "For automatic and register variables, the initializer is not restricted to being a constant: it may in fact be any valid expression involving previously defined values, even function calls." (p. 83)

(Fm: MISOSYS) Les, Let me look into your statement concerning initializers. I thought that Rich was going to remove the identifier, "entry" from the list of reserved words. That was his intention. On the other hand, Page 2-6 of the MC manual states that "entry" is one of the reserved words!

Problems with MCOPT

(Fm: Bob Zinn) MCOPT seems to have a problem (its such a nice program too!) It does not report errors that cause it to blow up. Including #asm - #endasm generated code seems to be a good way to blow its mind. It seems to expect certain sequences of operations, and if my code matches a beginning pattern, but it does not find the end.... blammmmmmm. It also does not check for or detect disk errors (they show up later!).

I have rewritten the crc routine in lu in assembler, and the up to 1 min of time that it took is down to about 5-10 sec. I suggest the crc in IFC be coded in assembler. The following works for the lu type of crc. (I also learned about several other types of crc--it is very easy to make small program errors that give completely different answers.

(Fm: MISOSYS) I don't recommend using MCOPT on any C module where you have inserted a #asm-#endasm block. MCOPT does expect certain sequences of code and I am sure it is possible for it to choke. It's really not a terribly "smart" tool; but was added in the hopes that it would be able to help out in most cases.

The CRC algorithm in IFC is coded in ASM - all of IFC is.

Speeding up MCP Preprocessor

Rich Deglin has investigated the slowness of the MCP preprocessor and has determined that 40% of the time is being spent in just satisfying the "LINE" function. For those of you who do not wish to utilize that capability, you can defeat it. Here's the patch which is applicable to both MC and PRO-MC versions,

```
PATCH MCP (D19,C3=C9:F19,C3=21)
```

A fix for #undef in MCP

(Fm: Rich Deglin) I recall that you mentioned at one time that a previously undefined macro (undefined with #undef or -U) couldn't be redefined with #define. I didn't look into the problem until it bit me; in the process of working with a public domain C program written by someone else, which used the defines "ON" and "OFF" with values different from those defined in MC's STDIO.H, I ran into the bug. The code stream looked like:

```
#include <stdio.h>
. . .
#ifdef ON
#undef ON
#endif
#ifdef OFF
#undef OFF
#endif
. . .
#define ON 7
#define OFF 8
. . .
case ON:
. . .
case OFF:
. . .
```

MCP took the #undef's, but the subsequent #define's failed to get entered into the macro table correctly. This caused the tokens "case ON" and "case OFF" to be passed to the compiler, which complained with "Illegal constant expression" or some such error. Thus, I was forced to fix the bug (how horrifying). In the process, I

found another bug in MCP. Correcting the first problem led to MCP giving an unnecessary warning when the undefined macro was redefined later, but only if the macro had been undefined using #undef, not -U, so I fixed this problem, too.

The fixes are designated MCPUNDEF5/FIX and MCPUNDEF6/FIX.

Update on MC libraries

A few minor problems have surfaced when the options REDIRECT OFF and FIXBUFS ON were specified together. The bug was traced to a coding error in the fopen() routine. Fopen() also contained an error associated with the O_KBECHO option; which occasionally was not functional, depending on circumstances. These bugs have been fixed.

We have also found a bug in gets() if the input is coming from a file and the input stream contains a "line" longer than 80 characters. This has been fixed.

Rich also located a wrong return code coming back from fclose() when an attempt was made to close a permanent system device (i.e. on *KI or *DO with REDIRECT OFF and FIXBUFS ON). This has been fixed.

The associated library modules, fopen5/6, fclose5/6, getc5/6, and gets are bundled together into one library named MCRELS/REL and put onto DISK NOTES 6. You can use MLIB to update your libraries from this set of modules. Similar to what was previously mentioned, anyone not having access to either, can return their master disks (both of them) in a protective cardboard mailer and we will update them at no charge. It would help if you could include a return address label.

Pending MC bugs

It has been reported and verified that some of the single-precision functions available in fplib via the non-standard "+f" compiler option [specifically fsqr() and fexp()] produce erroneous results. We have not yet investigated this fully, but we believe that it stems from the change

in register utilization between the older LC library and the MC library (BCDE vs DEBC). Until we get a fix for this, the workaround is to use standard double precision functions.

Jeffrey McLean reports that, "Scanf("%d,%dd",... and fscanf(stdin,"%d,%d",... do not handle non white space, non conversion characters correctly. The intent of these two statements is to read in two integers, separated by a comma. Instead of matching the comma in the input stream with the comma in the control string, these functions skip the assignment operation completely and go on with the rest of the program without the ENTER key being pressed."

The problem is due to a misinterpretation of the specs when _scan() was coded [that routine is common to both scanf() and fscanf()]. As coded, an input field is considered terminated only by white space; an "invalid" character terminates the input, ungets the "invalid" character fetched, and exits. The _scan() routine needs to be changed to terminate the input, unget the "invalid" character, but then process the field and continue - not exit. This should be fixed up soon; maybe even by the time DISK NOTES 7 is prepared (it requires revising the _scan() library module in LIBC/REL. That cannot be done by a patch.

Product Tidbits

BSORT51 and LDOS 5.3

John Welsh reports that BSORT51 will not work with LDOS 5.3's BASIC. The problem stems from the revision to the CMD"doscommand interface in 5.3's BASIC to enable you to issue a SETCOM or FORMS command and find modules in high memory. Here's a simple patch to BSORT which will fix this up:

```
PATCH BSORT (D00,7C=6E:F00,7C=68)
```

Maybe after applying the patch you should rename the program to "BSORT53" to avoid any confusion.

EDAS and PRO-CREATE

(Fm: MISOSYS) Here's some info in response to a bug report from Carl Bryson. I investigated his problem - that of getting a close fault error when using PRO-CREATE to assemble to disk with lower case accepted and entering a switch. The problem is not so much due to "lower case accepted", but to the fact that the command line was not converted to upper case. The switch, being left in lower case, was not matched as valid; thus, the "bad parameters" error. The "close fault" came about because the FCB at this point contained a pointer to the filespec string. It just so happened that the low-order byte of the pointer coincidentally had bit-7 SET. The error handler, which attempts to close any file left open, assumed that the FCB contained the data on an open file. The data was invalid; thus the DOS flagged the close fault. We may try to work up a patch to deal with this in the future. For now, be aware of this effect.

EnhComp

(Fm: Bob Zinn) I) Read with interest the dialogue about EnhComp's ON ERROR. I agree with BH that an ON ERROR handler that does not allow RESUME at any next statement is practically useless. Sure, you can detect an error, tell what kind it is and restart your program at some particular point where that kind of error is handled. But you can not easily write a general "handle

any error" routine that will allow "correct and continue" fool proofing.

Perhaps if EnhComp treated user numbered lines as if they had a numeric label were GO TO "numeric Label", then program supplied "hidden" line numbers, would ALWAYS be returned by ERL - eliminating the confusion between physical vs label numbers. You could include a function that would give the line number of a label, and another that gives the label (if it exists) of a line number.

(Fm: Mike Harrow) Your comments in the Fall TMQ about EnhComp on deciphering info displayed during compilation answered part of a question I had. Your closing remark in this section is exactly the second part of the question. Ah! -- What do the Data Table lengths indicate? If you don't answer me now - How about in the next MQ.

(Fm: MISOSYS) They indicate how much space is taken up by each table. Since we didn't go into the total details about each table, the info isn't too useful except during a debugging session. Then I can ask you for the info and it may give me a clue.

(Fm: MISOSYS) The reason that the "X" file mode cannot support the use of LOF(n) is because such a function relies on the knowledge of an exact end-of-file byte position. The DOS does not maintain the EOF byte when full sector I/O is used - that is the responsibility of the program. When record I/O is used, the DOS maintains the EOF byte. Since "X" mode is designed to handle record access of logical records having any length less than 32767, it must do its own record blocking and deblocking while it does full sector I/O via the DOS. The routines as written, make no use of any offset in a sector for maintenance of an EOF byte. Thus, there is no facility for detecting the EOF and thus an LOF(). The solution is to maintain the exact number of records being utilized in either the first record of the file (maintained for control purposes only) or in a

separate control file. The choice is yours.

The problems with too many editors is the reason that CED uses the standard BASIC editing commands as a foundation. You can always use any ASCII editor you choose to use for EnhComp; however, the S mode is only usable with CED since both the editor and the compiler need an interface to operate under S. That can't be done for any generic editor unless S was completely redesigned - which we don't plan to do.

I believe that the statement name "EQV" stands for "equivalence" and the statement "IMP" stands for "implication". They are terms arising in the field of logic, sometimes addressed as philosophy.

Here's a correction to last issue's BC62/FIX. It turns out that there was a mistype for the BC62/FIX when the file was made up. The line which started out "D29,E5=09 25" should actually read "D29,E5=09 2F" - the second patched byte should be a "2F" and not a "25". Since my original test was off a version manually patched by FED, I didn't "catch" the error. THE PATCH CORNER in this issue has the correct fix.

Here's some scoop on the video cursor while running a program compiled by EnhComp. EnhComp was developed by Phil Oliver for the Model III. I ported it to the Model 4. A big task when I had to first learn how the compiler worked so I could port it. Both the INPUT and LINEINPUT statement handlers in the SUPPORT/DAT library use the machine provided @KEYIN handler. On the Model III, this is a ROM routine. On the Model 4, this is a DOS service call. The Model III and 4 routines behave similarly except for their handling of the cursor. The Model III routine ALWAYS turns the cursor ON as the first thing it does and ALWAYS turns the cursor OFF as the last thing it does. It does this regardless of the state of the cursor upon entry. Thus, if your program had the cursor OFF, it would be turned ON after a @KEYIN call. The Model 4 routine was purposely coded to not touch the cursor. It could have kept the state, turned it ON, then restored it to the entry state upon exit - but that wasn't

done.

Now the EnhComp INPUT handler had code to turn the cursor ON before calling @KEYIN and code to turn the cursor OFF after return from @KEYIN. The LINEINPUT handler did not. Of course, on the Model III, both routines worked properly because the INPUT support code was redundant - the ROM turned the cursor ON then OFF - and LINEINPUT did not need the code. I missed this when I ported the compiler over to the Model 4. Thus, when you use INPUT, the handler engages the cursor but LINEINPUT does not. You can circumvent this problem by surrounding your LINEINPUTs with a PRINT CHR\$(14); and a PRINT CHR\$(15);.

PRINT: with EnhComp

(Fm: Hamilton Gaillard) The following typical statements are commonly used in interpretive BASIC but are catastrophic under PRO-EnhComp, producing unintelligible error statements in numerous and syntactically correct statements involving strings (but not in purely numerics).

```
10 PRINT"MESSAGE":PRINT:PRINT
20 PRINT"MESSAGE"
```

Line 10 must read

```
10 PRINT"MESSAGE":PRINT" ":PRINT" "
```

You may wish to include an alert in the README file. It took me an hour or two to recognize what was wrong. The clue actually came from experience as my printer will not accept LPRINT without the quoted blank.

So far, I've tried only one old program involving a lot of double-precision arithmetic. It loads and executes very rapidly, especially loading as it eliminates load BASIC, load MYPROG, and RUN.

Documentation is comprehensive and well organized, but I wouldn't recommend it for a rank beginner in BASIC.

Disabling RMARGIN in EnhComp

(Fm: MISOSYS) D. T. Froedge had a need to disable the RMARGIN function in EnhComp. Seems that he was using EnhComp to compile programs being used to dump an output of graphics control codes to a printer. A single "line" takes up to about 700+ characters of data; however, RMARGIN accepts a range of only 2-255. It would have been good if a value of zero would defeat the automatic carriage return; but that's not how it was designed.

Now if you can live with RMARGIN permanently disabled, the following patch will defeat the RMARGIN checking (i.e. no automatic carriage return will be issued regardless of the number of characters per line output).

```
PATCH SUPPORT/DAT (DOF,79=C9:F0F,79=D8)
PATCH SUPPORT/DAT (DOF,83=C9:F0F,83=D8)
```

The first patch line is for the Model III EnhComp version; the second patch line is for the Model 4 PRO-EnhComp version.

EnhComp's BASIC line #'s in Z80MODE

(Fm: MISOSYS) The sample program on page 6-1 of the EnhComp manual illustrates that you can reference a BASIC variable using the syntax, "&(varname)". You can also reference the address of a BASIC line number (an actual line-numbered line) by the syntax, "&(#line_number)".

FastBack

(Fm: Jack Lottey) For hardisk backup with FastBack, is it possible to put more than one file on a floppy? It seems like a lot of unused disk if I am copying a file with just a few records. Most of my backups are for Profile III+ HD and the screen, label, etc. files are quite short.

(Fm: LDOS Support) FASTBACK should only be used for files too big to fit on a single floppy. For this purpose, FM is ideal as you can have FM move all files smaller than a given size to floppies (faster than BACKUP can), and then use FM to write a

jcl to make FASTBACK move all the files bigger than a floppy.

FM - File Manager

(Fm: Bob Zinn) Subject: FM 5.1b: suggested enhancement: a "later date" mode (L) ? "fm :1 :2(1,move)" would move all files on 1 with a more recent date than their counterparts on 2. i.e. a file by file date comparison is done. A "new/later date" mode (NL) or (newer) "fm :1 :2(newer,move)" would move all new files and all later dated files.

IFC

(Fm: Bob Zinn) IFC - I made an error on copy - specified same input and output drive. It assumed one drive copy, and would not let me <break> out.

IFC - has loooong pause after read during copy. What the heck can it be doing for 1 to 5 and more seconds?

IFC 3.4c does not ask for password on protected files and has no documented provisions for viewing system files. Both of these are sometimes useful OPTIONS.

Error messages consistently unrelated to cause. Perhaps all error handling is messed up? Previous reported problems seem to indicate this.

(Fm: MISOSYS) IFC does a full CRC error check of dest vs source. That's what is taking long. IFC works like PURGE; it has the power to delete password protected files. That's by design. IFC has no documented way for viewing SYSTEM files since it does not support access to SYS files. I believe IFC erroneously assumes that @CKDRV returns a valid error number. This may be fixed via a patch; we'll put it on the hook.

SYSGEN KSMPplus?

(Fm: Dave Sanders) I would like to know if it is possible or advisable to sysgen the state of the keyboard configuration when

using the KSMPLUS3/FLT. Seems to me that it is easier to do this on bootup rather than using the SET and FILTER commands. However I read in the manual that it is not advisable to use the SET or ROUTE commands when a device is attached to a file. The terminology of the word file is what escaped me. Is in fact my ASCII listing of key definitions for the ksmplus3/flt a file? I would appreciate any help you can give me!

(Fm: jeff brenton) Yes, KSMPLUSx/FLT can be sysgened, both in 5.x and 6.x versions. If it couldn't, I would not have grown so attached to it over the years! Now I'd like something like KSMPLUS for MS-DOS, but I'm not willing to dedicate the 70-200K of RAM that MS-DOS programmers assume you HAVE to use for such nonsense.

Little Brother and LS-DOS 6.3

(Fm: Kevin Kierans) I have a question regarding 6.3 and LB. Is a new floppy STARTUP disk coming soon? This disk turns the extra 64k into the system disk (and its Faaassssttttt....). Course it's 6.2 not 6.3. Does anyone know just how that disk works and how it could be patched or fixed or whatever so that it turns the extra 64k into 6.3? I know how to do it with the slow copying over of the /sys files and then giving the system (system=drive) command, but it's really slow and leaves the physical drives numbered backwards. Anyone have any hints or clues? How was the BACK/IMG file created. How is it loaded into the extra 64 without pre-formatting it?

(Fm: MISOSYS) Kevin, This issue of TMQ has a special section devoted to Little Brother. One of the items discussed is the procedure to upgrade the Model 4 version of LB to LS-DOS 6.3. It uses a "memdump" utility provided here to create a new BACK/IMG file.

Little Brother and MS-DOS

(Fm: MISOSYS) I investigated a problem report from one of our new Tandy 1000

users of Little Brother (attempting to invoke LB and getting back the "A:" prompt). The solution turned out to be quite simple. On the other hand, since we had never received such a report before, we were unable to quickly give the solution on the phone when he first called.

The second page of the Little Brother Installation manual (MS-DOS version) shows the steps necessary to install LB on a 2-drive floppy system. Unfortunately, it neglected to advise that the DOS file, COMMAND.COM, must be present on the disk in either the drive you BOOT from or otherwise available based on a PATH command (which should be discussed in your DOS manual). The COMMAND.COM file is the DOS file which performs "shell" operations. It is needed to be able to access LB overlay files. Without it, LB cannot operate.

Simply put COMMAND.COM on the LB disk if you are putting the LB disk into the "A" drive.

MC Information

Because of the extent of information being printed concerning our MC C-language compiler, you will find it discussed in the Tower of Babel section.

MRAS

(Fm: Bob Zinn) MRS510 is missing from the TMQ I.ii QUARTERLY and disk, but seems to be needed.

(Fm: MISOSYS) Don't know how that happened. We have it in this issue and on DISK NOTES 7.

Mixed case + PRO-MRAS

(Fm: Scott Russell) I have a question about PRO-MRAS. Is there any way of allowing mixed case labels. I have a lot of EQU + MACRO files that I used with ALDS but I used mixed case. It made it easy for me to see what were equates as opposed to labels and stuff like that.

(Fm: MISOSYS) There is no way presently to use mixed case unless, of course, you use a text editor which doesn't case convert and you are careful that all OP codes, register names, and labels are upper case (what's left?). SAID can easily fix that up for you. Just invoke SAID filename (ASM). Save out your file and it will be upper cased for PRO-MRAS.

PaDS

(Fm: Bob Zinn) PDS(COPY) will not work on 37.5 k file although PDS(APPEND) worked fine with it. Copy says - file too large - out of memory. I can not see why the whole member needs to fit in memory in order to copy it.

(Fm: MISOSYS) PDS(COPY) was written with the entire concept of PaDS in mind - small library members. Thus, it was not written to make multiple passes through a small memory space. Just a design decision since PaDS was never intended to handle large files. I don't recommend putting 37.5 K files in a PaDS anyway. When you start using large members, you will generally wind up with too many directory extents which then slows the access down and defeats one of the concepts of the PaDS.

(Fm: Gary Phillips) Here's another minor bug report, Roy. Not urgent, but in case you didn't already know: the PDS(BUILD) command gets an error/abort apparently because it tries to set the user password of the created file under LS-DOS 6.3. It creates the file OK, though, and you can use it. You just have to do an ATTRIB to set the password protection if you want it. I might not have even noticed it, but I was constructing a macro library for *SEARCH today and kept having to dismantle the thing to change the code and then reassemble it, so I put the PDS commands in a JCL file. Sure enough, the JCL aborted after the BUILD. I didn't even notice the error message when I entered the command manually! This is probably a small patch, and if you don't get around to it, I may have time to hunt it down myself.

(Fm: MISOSYS) See PDS663/FIX in THE PATCH CORNER.

(Fm: Shane Dawalt) Paul Bradshaw and I have been talking about some of PRO-PaDS uses. One I thought would be nifty would be to put all PRO-MC header files into a PDS file. This, however, brought up the question of, "will it work?" I know MLINK can use PDS files for the libraries, but since headers are included into the source during preprocessing, can the preprocessor read PDS files? Does it matter? Perhaps LS-DOS sorts through the PDS and presents only the data requested by the application.

(Fm: MISOSYS) No, it won't work. The DOS has no way of sorting through a PaDS file. Such a capability was to be designed into PaDS version 2 which was never developed because the PaDS product did not sell in quantities sufficient to justify continued development. PaDS is an excellent cost-effective product for what it does; however, the using community told me they didn't want it. So we don't do any more development on it to add features. Also, MLINK doesn't use PODS for its libraries; IRL's are not the same thing!

PRO-ESP's CVT324 and listing

(Fm: Rayner L. Roueche) Recently I purchased the PRO-ESP package. I have been pleased with the performance of these programs with one exception which I hope you can help me rectify. When trying to use the program CVT324 to convert a Model III BASIC program to run on the Model 4, I find that the information scrolls off the screen before I can read the specifics that require further work for conversion. Pressing <SHIFT AT> does not stop the scroll as normally happens in BASIC. Under these circumstances I find it impossible to tell what the program is instructing me to do. Is there a patch that you can give me to install to rectify this situation?

(Fm: MISOSYS) Why not just LINK *DO *PR before invoking CVT324. That way, you'll

get a printer listing along with the screen listing. RESET *DO afterwards.

Another way is to use the built-in command line I/O redirection of the LC-compiled CVT324. The screen output can be temporarily redirected to the printer (or even to a disk file with a command such as,

```
CVT324 infile outfile >*PR
```

It's the ">*PR" which redirects the normal screen output to the printer. Substitute a disk file name for "*PR" and the screen output goes to a disk file, instead.

SAID Text Editor

(Fm: Alan Kaplan) The following happens when I use PRO-SAID 1.1 on my model 4d. Let us say that I want to delete all occurrences of the letter "l" or the character OAH, etc. in a file. If I start at the beginning of the file and the character to be deleted occurs twice in succession, eg. "ll", the function will only delete the first occurrence of the character but will be successful in all other places where the target character is surrounded by different characters. If I start again I will then be able to delete the remainder of the characters that originally occurred as a pair. Is this a bug or am I doing something wrong? For example, I put %0a<enter> in the search field and <enter> in the replace field, followed by all <clear><shift><4>.

(Fm: MISOSYS) That's how it's programmed to behave. When you do a search and replace, the cursor advances to the first character after the replace is done. A search always starts with the first character after the cursor. It was done this way to prevent blowing up on "replace A with AA", etc.

THE SOURCE to TRSDOS 6.2

(Fm: Gary Phillips) I'll admit you make me curious enough to want to buy a copy of THE SOURCE. How badly out of date will it be after the 6.3 upgrade?

(Fm: MISOSYS) My guess is that at a minimum, 90% of THE SOURCE remains totally unchanged. The remaining 10% has pieces with moved addresses due to the added code involving the date extension and time stamping. LSI's internal requirement was to avoid any change to the low memory region. Most utilities, drivers, and filters did not change a byte. It's also a special for this TMQ.

TYPEIN & Profile 4+

(Fm: Bob Sullivan) Using LED, I created a file for TYPEIN to process which would go into Profile 4 Plus (version 1.0.3) and build an index. It works until the X command is issued to exit Profile. Then the Job Done message appears, but there are still commands in the TYPEIN file. I solved the problem by making the TYPEIN procedure part of a JCL file. I'm just curious as to the reasons why.

(Fm: Les Mikesell) Problems with TYPEIN are usually caused by programs reading characters and ignoring them (perhaps to clear the type-ahead buffer at certain times). Others have mentioned this problem with Profile and have overcome it by adding extra characters at certain places in the file.

(Fm: John Garner) (Putting in my 2 cents worth)... Sadly, TRSDOS TYPEIN is not as powerful as LDOS TYPEIN. (I happen to know, as we have both at the office.) In the LDOS version, you can do something like this:

```
. This is a /JCL file
TYPEIN (LINES=2)
  program
  data for program
. If you see this, it's working
//EXIT
```

On LDOS, TYPEIN will suck up "program" and "data for program", then the comment will be displayed, then "program" will be executed and hopefully eat its data. On TRSDOS, this JCL will just sit until you

TYPEIN 2 lines from the keyboard (or get disgusted and press BREAK) and you will never see the comment. Maybe there's a good reason for this behavior, but I can't for the life of me figure out what it could be.

(Fm: Les Mikesell) I thought that TYPEIN (LINES=n) was supposed to work in the TRSDOS 6 version under JCL also. How about: "TYPEIN inputfile" when included in a JCL?

(Fm: LDOS Support) For the patches to fix the errors in P3+, Tandy should be able to supply them. Check with your local RSCC, and ask to look in their Customer Service Bulletin book under Profile Three Plus.

As far as JCL, Profile 3 Plus and TYPEIN goes, here's the scam: In the JCL file, put the line

TYPEIN MOOSE/DAT

In MOOSE/DAT, you actually insert the EFCn invocation command to enter Profile, followed by the keystrokes necessary to do whatever it is that you want done. Make sure that there are enough commands in there to complete totally and exit to the OS, at which time the next line of the JCL will get executed.

XFTS

(Fm: Bob Connors) I have received the copy of X-FTS that I ordered, and am very pleased with the performance. I will be mailing the registration card tomorrow. I do have a few questions though about what is happening, and I hope you can answer them.

First, let me explain to you how I am using X-FTS. I use it on the Lobo MAX-80 under LDOS 5.1.4M (natch) in conjunction with a Towne Crier V2.0 BBS system. This system, instead of linking *DO and *CL, for some reason just changes the *DO driver address to point to a m/l assist module in high memory. The previous driver address is also saved in that module so that characters are passed to it once the

assist module has processed it. I also have DO80M/DVR installed (purchased from RICLIN many moons ago), which is part of my configuration file and which is the driver that all the screen data is passed to. OK, with that in mind, the following are my questions:

(1) X-FTS is called using the LBASIC CMD"expr" feature. This works well except that upon returning to the BASIC program, the cursor has been turned on and no matter what I do, I cannot turn it back off. I have tried "PRINT CHR\$(15);" and even gone so far as to make "SYSTEM (BLINK=32)" and "SYSTEM (BLINK=OFF)" part of my LDOS configuration. It works fine until X-FTS is used again, and then I get the cursor back on even when scanning the keyboard using INKEY\$. The only reason I mention this is because when the system restarts, the cursor (either the large block I normally use, or the blank space as explained) prints at the end of every "PRINT@" line I use and it messes up the screen. The peculiar thing is that if I exit the program at that point, and re-run it, the problem goes away. Can you help with this? Am I doing something wrong? Or is there something about cursor control here I have not tried yet?

(2) I have a copy of Ward Christensen's docs on XMODEM and it says that normally, a minute time out is recommended for the initial NAK to be received by the sending system for file transmission. However, on the MAX-80, this time out is quite fast and seems to be only 30 seconds or less. Is this correct? I assumed that LDOS on the MAX had the timing routines adjusted in order to compensate for the faster clock speed. Or does XFTS use its own timing routines that don't account for it?

(3) Sometimes I have noticed that re-tries of blocks received do not seem to wait at all. They just zip from 1 to 9 and then the program aborts. Other times, this does not happen. I wish I could explain to you why, but I don't know (this is more of a comment than a question).

(4) Does the "R" parameter only apply to files being sent or does it also apply to files reception? And does it affect the

timeout on the wait for the initial NAK/SOH?

(5) Is there anyway to increase the time out for the initial SOH on file reception? Some of my users complain that there is not enough time to start sending before my system aborts the reception.

(6) When a caller to my BBS asks for a download, I use the command, "XFTS S FILESPEC (R=20)". This causes (as you know) the default to "QUIET" since I am in host mode. For everyone, the LOGO of XFTS does not display, and the file transfer occurs without a problem. But in one case, the LOGO does display and the program (XFTS) times out waiting for the initial NAK. This always occurs for the same person and only that person. Can you think of any reason why this should happen? This person can upload alright, but has never been able to download, except when I take the BBS off-line and do a direct transfer. Should the logo display in the quiet mode? If not, why would it for this person (could it be something caused by his terminal program)?

I am quite pleased overall with XFTS, and it has caused my BBS's membership to increase almost 500% since I started using it. Thanks for all your help! As usual, I have written a much longer letter than I had planned, but I really do need to ask. Sorry. I am impressed with the ease of use of X-FTS and the only thing I would recommend changing in it is the addition of CRC-16 protocol. Thanks a million for keeping all use LDOSers happy with so many excellent products!

(Fm: MISOSYS) XFTS turns the cursor on before it exits back to the "invoker". That's why it comes back on. My hunch is that since the Towne Crier BBS system took over the *DO driver, it is probably trapping the cursor off you are attempting to issue via the PRINT statement.

Here's how you can patch the FTS5/CMD file to keep it from sending the X'OE' through @DSP. At X'56F3', you will find the code:

```
LD    C,0EH
LD    A,C
```

CALL 33H

Thus, you have a couple of options. If you want any other character sent to @DSP on termination of FTS, patch it in at 56F4H. If you just want to inhibit the character display, change the value at X'56F6' from X'CD' to X'21'. This changes the "CALL" instruction to a "LD" instruction.

FTS has to continuously monitor for a received SOH character while it is in its timing loop; thus, it cannot use the OS's @PAUSE routine which is adjusted on the MAX-80 to match the timing characteristic of the Model III (which the MAX emulates). That's why the "1-minute" timeout appears like "30-seconds" for you. You can probably increase the value of the FTS countdown timer. At X'5375', the value 11776D (2E00H) is loaded into register BC. By doubling the 2EH to 5CH, you will get back to your 60 seconds; a higher value will add in more delay. You can easily use FED to make these changes.

The "retry" parameter applies to both the "wait for ACK or NAK" and the "wait for SOH". The SOH wait time will be three times the length of the ACK/NAK wait time (the countdown timeout routine mentioned above is called three times by the SOH checker).

I can't explain your other problems. Maybe someone else has experienced the behavior and has more concrete evidence.

PRO-WAM and PSORT

(Fm: Tom Gallaudet) I have a question about using PSORT which comes along with PRONTO. I built my own BASIC data base program which contained 90 Records (Names and Addresses) with a LRL of 192. I recently figured out how to use PSORT to sort the file. I put a 192 byte record in the beginning and had Relative byte 02=5A (number of actual data records), 10=C0 (LRL=192), 12=01 (first real data record) and the sort info. Including the first info record, the file was 91 records long and ended at the 64th byte of the 68th sector. Psort sorted the file perfectly but when I added one more record, bringing the end of the file to the end of the 68th

sector, I would get a "data file read error, record 91". I also incremented the number of records from H5A to H5B. In frustration, I added another record, updated the first record and everything worked fine so I kept adding records until the "EOF" = end of a sector and I'd get the data file read error. Why?

(Fm: MISOSYS) PSORT absolutely REQUIRES that the file have a logical record length such that the file's EOF byte ALWAYS ends at 255. PSORT is written in C and always uses files with LRL of 256 and EOF offset byte of 255 (i.e. full sector I/O). If your file does not conform to this, then it won't work with the current version of PSORT. Perhaps in the future, I'll redo PSORT with MC and use its fstat() function to get the actual file length. Then I can read the last sector with an exact byte count. Since the last sector of the file is always read with 256 bytes, that's when the read error occurs.

PRO-WAM/PRO-NTO - What is it?

(Fm: Harry Hopkins) OKAY, I admit to stupidity! I think I know what PRO-WAM or PRO-NTO(?) is and can do, but can someone make me a clear, 1000 words or less explanation. All this discussion is going on and I feel a bit lost. Perhaps I can increase the count to 1151 for you. I have 3 128K Model 4s (1 regular and 2 Ps). Maybe I can make it 1153 if the price is right and it's something I can use.

(Fm: LDOS Support) First off, you must have one of the extra 32K banks available. Given that requirement, what you get is a keyboard pop-up manager for mini-applications. For example, while running some program, you can "pop-up" your address book or phone book. When through, you can pick up exactly where you left off in your "foreground" program.

Mini-applications included are: phone book/dialer, card file, address book, calendar, appointment/tickler file, calculator (four function or programmer's), text import/export and a bunch of other stuff.

There is an add-on pack available that gets you five editors: file, disk, memory, screen and text, plus some extra tools.

Lastly, there are a few add-ons available here in DL 2 (our Compuserve forum) for merely the cost of connect time.

MS-DOS users will recognize this as a kind of "Sidekick-like" utility.

(Fm: Tom Gallaudet) Harry, Here comes an unsolicited testimony. PRO-WAM (PRO-NTO) is one of those programs that sounds neat in the mag ads and for \$59.95 Why not? Well, I recieved it, installed it and played a little and then decided that it was a really neat program but I wasn't sure what I wanted to do with it so it sat for a while. After a couple of months and several ad "rereads", I started to play with it again and now I could not live without it!!!! After pressing two "key combinations" whatever program (if any) you were running, stops and a little window appears in (or close to) the middle of your screen. If you hit the break key, the window goes away and your program continues where you left off never missing a beat.

I have an auto dial modem and use the DIALER all the time, I use the ADDRESS for anyone that I send any info to and code each entry for the type of info that I send so I can quickly print a list of who got what. I have the BRINGUP appointment tickler automatically run when I boot the system so that I'm forced to see what I am supposed to do each day.

It also has a CARD program which is a little text file which I use for storing notes on people like topics I want to discuss with my partner when we call. No matter what program I'm running, I can immediately call up his "CARD" and have the list. Besides what comes with the main program, there are several additional public domain programs that work with it.

PRO-WAM uses some high memory and all of the first 32k partition leaving the second 32K bank for a memdisk to store the most used PRO-WAM data files or as the memory

for the TEXT Editor. It is a very well written program and it will interface with every program I have tried with only one problem that I could complain about and I haven't complained because it's not that big a deal. The problem is: If you are using RS's Videotex Plus and you want to use the TEXT Editor, it's reset button time. Videotex just freezes solid. But you can use all the other programs with no ill effects. The bottom line is: it takes a while to figure out how to use it but once you get started the possibilities seem endless. It's very well written and well worth the price. (And it's worth more than 1000 words!) And I also got the MR. ED ad-on package which has TEXT, FILE, DISK, MEMORY, and VIDEO EDITORS. Hope this helps to answer your question.

Backing up Mister ED

(Fm: Pete Granzau) I received Pro-WAM and Mr. Ed today, and have been busy installing it (I have this HAI Megaboard, see, and the RAMdisk is a perfect place for it to go), and have come up with a funny-strange occurrence. Three of the files on the Mr. Ed disk don't appear in a "by-file" BACKUP, trying to put the files into the RAMdrive. I can COPY them there, but the normal method of loading the RAMdrive is with BACKUP from a DiskDISK (or is that DISKdisk?) "drive", which is, of course, a by-file BACKUP. The files are TED/APP, OOPS/APP and MED/APP. Have any idea why?

I moved the three files using COPY with the (CLONE=NO) parameter, and the copy works just fine in BACKUP. Why would the three files have a date sometime in 1980? That doesn't seem logical.

(Fm: MISOSYS) Pete, Here's what I found. We duplicate disks on a MAX-80 using an in-house version of QFB (actually a DUPE which makes 4 disks at once). The master is actually a diskDISK on a 10Meg drive. The MAX's battery which powers the hardware clock has long gone. We normally BOOT that machine with a JCL automatically going into the DUPE JCL. When we are finished duping, we go back to DOS Ready. If we forget to set the DATE, then the

actual system date is erroneous. If we then proceed to apply a patch on that system, the file's date in the directory gets reflected as "not current". All I can tell you at this point without violating proprietary agreements is that the bit in the directory under LDOS 5.1 which indicates that a file's date is not current is used for a different purpose under TRSDOS 6.x. That's why the files would not BACKUP by class. A COPY with (C=N) would correct that error.

(Fm: Les Mikesell) Sounds like the no-backup bit strikes again...

(Fm: Bryan Headley) Roy - For shame! All of these articles on Maximul on how to fix your battery, and you haven't fixed your MAX.

(Fm: MISOSYS) Who has time to fix MAXs'. With LDOS 5.3 to get out the door, the LDOS Model 4 Interface Kit, the RS Hard disk driver package, UNREL to document, ED/ASM-86 to test, DED86 to document, QUARTERLYs to write, customers to support, software to upgrade for x.3 compatibility, etc., do you honestly think I have the time to get the battery replaced. Why, I just throw the computer away and get a new one <grin>.

Removing PRO-WAM from memory

(Fm: Pete Granzau) Roy, for reasons of my own, I like to run my Model 4 with two filters on the *KI device. It has become evident that something was killing them (as if a RESET *KI had taken place) on occasion, and I have finally discovered the cause: Removing PRONTO does it. "Why," you may well ask, "would ANYONE want to ever remove PRONTO?" Well, when I fill a buffer here with messages, it turns out the file I save is about 1K too long for Lazy Writer; I remove PRONTO, and voila! enough space. Also, I have a BASIC program I occasionally wish to run which, it turns out, is too big to run with PRONTO installed. Is the process of installation and removal too complicated for this to be a feasible fix? I have been very assiduous

in ensuring that PRONTO is installed after anything else (the only thing that makes it possible to remove it at all!).

(Fm: MISOSYS) I see nothing wrong with removing PRO-WAM (PRO-NTO). That's why we put in the (OFF) parameter. However, since PRO-WAM is a filter on the *KI device, it needs to RESET *KI. This, of course, resets any other *KI filter. On the other hand, under DOS 6.x, you can easily re-filter the *KI device since the modules are intact in memory. Yes, we could have told you what was happening when the (OFF) was done. Sorry about that one.

(Fm: Paul Bradshaw) If you only have the early version of UNSET, I highly suggest you get the updated version 2.0. It fixes many bugs, and provides MUCH more information on high memory modules. I've never been bothered by <ctrl><p> conflicting with CIS operation - I make due with a <ctrl><o> and that seems to suit me just fine. In fact, I patched DOEDIT to allow me to activate it using ctrl<e> so that it would fit more with PRO-WAM! (kind of a reverse situation).

I'll get to work on an UNFILTER program (should it prove feasible). In Roy's "Guide", he outlines a method for removing a single device from a chain without affecting the other devices, but this is not implemented at the DOS command level. Will see what comes up.

(Fm: MISOSYS) Paul, It was my intention that the unfiltering of a device module would have gotten implemented in a future '.' release of 6.x. That's why the technique was discussed in TPGTLTV6. If I had been involved directly with the DOS in the post 6.0.0 era, I was going to use,

```
RESET devspec (module="modname")
```

or something similar. Unfortunately, I was no longer involved with 6.x after 6.0.0 (at least not in any official capacity). Thus, designs don't always get implemented. Karl chose not to implement the technique when he coded PRO-NTO. Maybe it will get added into the code when I

bring out a new version of PRO-WAM this year; or maybe not.

Reusing the screen with PRO-WAM

(Fm: MISOSYS) It's easy to open a window and still reuse the current screen - just sneaky. I save save the region from 3000H-33FFh into 2400H-27FFH. This then gives me 2K of space following the end of VED (I limit it to 1K of code). I then save the CRT before WCREATE with a @VDCTL. VED uses @VDCTL for updating; thus I @VDCTL the saved image after the WCREATE. That's why it "blinks". EXPORT cannot be done under programmed control. I know its a good idea, but not in.

PRO-WAM's BRINGUP/APP past 1987

(Fm: Lynn Sherman) Roy, Is there a patch available or on the way to allow the BRINGUP/APP to utilize dates beyond 1987? I'd sure hate to lose that application at the end of the year. In fact I might need to start adding 1988 entries by mid 1987.

(Fm: MISOSYS) We expect to be releasing an updated version of PRO-WAM sometime this year. Extension of BRINGUP plus a few new apps are in the planning stage. Specific word on this upgrade will be via a MiniNOTES mailing. For now, we intend to re-groom PRO-WAM, add support for reverse video, work to reduce its memory requirements, add about 3-4 new applications, and rewrite the manual (also port over to 5-1/2 by 8-12 size in a binder). We expect the new PRO-WAM to be priced at \$74.95 + S&H. An upgrade will get you a whole new package at a cost of between \$20-\$25 + S&H.

PRO-WAM and "bigram"

(Fm: Peter Amschel) So, then if I equipped myself with the 1 meg Ram buffer put out by Anitek, how big of a PRO-WAM address file could I have; and what other neat things could I do?

(Fm: MISOSYS) A PRO-WAM address file can

be as long as it can be provided it will fit on the disk as a file. ADDRESS/APP records are 128 bytes; thus for every 8 records, you use up 1K of disk space. Irrespective of a 1 MEG board, you can always do all sorts of neat things running PRO-WAM. Unless you've tried import and export, you haven't "lived".

(Fm: Peter Amschel) Can I import and export from PRO-WAM to LeScript? I know that PRO-NTO will not export and import with LeScript but I am hoping you will say that PRO-WAM will do it.

(Fm: MISOSYS) PRO-WAM is PRO-NTO with a name change. The lack of being able to EXPORT from PRO-WAM to Lescript has nothing to do with PRO-WAM. The problem is that Lescript is not a "behaved" program - it uses its own keyboard driver. Since EXPORT feeds the data back through the @KEY calls, the program you want to EXPORT back to MUST use the DOS's keyboard driver. Now you know why I harp so much on standards. Unless a program honors the DOS, it has to stand alone.

(Fm: LDOS Support) The problem is not on Roy's side, it's on Anitek's side. If you use the command line option that says to honor the system drivers you may get further, but I believe that turns off some features in LeScript.

(Fm: jeff brenton) PRO-WAM IS PRO-NTO - just renamed to satisfy a bank who believes Pronto is their trademark. And it won't export to LeScript (it WILL import FROM LeScript) because LeScript doesn't use the keyboard driver.

PRO-WAM and AUTODIALER

(Fm: Guy M.) When I bought Pronto! (now Pro-Wam), I patched DIALER/APP with DIALER4P/FIX as it is recommended in README/TXT in order to make autodialing work with my Model 4P internal modem and edited the "@" DIALER MACRO. Everything worked fine. Recently, I upgraded my system with the DC-2212 modem. I then tried to patch DIALER/APP with

DIALER12/FIX and edited the "@" DIALER MACRO according to the FIX file comments. Now, when I ask PRONTO! to dial a phone number for me, nothing happens. I only get the "DIAL COMPLETE. <H> to hang up." message. Anybody can help? Many thanks.

(Fm: MISOSYS) The DIALER12/FIX dated 1/28/86 should work.

The Hardware Corner

Z-TIME1 clock board

(Fm: Jerry Wagers) Kenmore Computer Technologies Z-TIME1 real-time clock is a super enhancement for the TRS-80 in a RCPM or other BBS environment; however, there is an apparent caveat with it. It does not appear to be compatible with non-gate array Model 4/4P machines. I have an original converted III and a gate array 4P and can only get it to function in the 4P. Has anyone else experienced this same problem? If so, were you able to get the clock to work in the non-gate array machine and how?

It apparently causes some problems with the timing in non gate array machines, such that the computer will not even boot with it installed. In fact, the bootstrap loads, but then all drives are selected and will continue spinning but not reading forever if you don't press the panic button. In the gate array machine, it functions superbly without any problems whatever, it WILL fit into the 4P along with the internal modem and a HI-Res board.

If anyone is interested, I have written a clock driver for this fine board which runs on TRSDOS. All the software shipped with the clock is for CP/M only. If enough folks want it, I'll upload the software to the database.

(Fm: Ken Hipple) I uploaded an ACH file into DLO. It is a set of patches & programs for using the ZTIME-1 clock board on the Model III. The main thing is the patches. They enable LDOS to read the time & date from the ZTIME-1 during boot. I

just got the board in 3 or 4 days ago. So far it seems real nice. BTW, since I have this I won't need that info you have. I was a little worried about my ability to modify it to work on the Model III so I'm glad I found this board. Thanks for looking it up though.

Question - I used ARCH3 on these programs & uploaded the ACH file it produced. Its the 1st time I've used ARCH3 & was wondering what the difference was between the ACH file & the 'crunched' ACC version? The ACC file was about a 10th smaller but I don't remember seeing any of these in the DLs so I played it safe and used the ACH version.

New Tandy 'pooters

(Fm: Paul Bradshaw) Just a silly question: What on earth do those new suffixes for the new TANDY computers stand for? EX, SX, HL? I can never keep the two Tandy 1000's straight (I always say EX when I mean SX, etc). How can I remember these things and who came up with them? (and WHY?) Just curious (as usual)

(Fm: LDOS Support) I have no idea what they really stand for, but I remember them as: Tandy 1000 Educational eXpanded; Tandy 1000 Speedy eXpanded; Tandy 3000 Hi-speed but Little

TEAC FD-55BV-06 drives

(Fm: MISOSYS) Thanks to all for the suggestions on my TEAC drives for use with a pure IBM-PC. The DS, RY, and FG jumpers are the correct ones. The TEAC tech support helped me out with another problem. Seems that one of the 6-32 screw hols on the TEAC drive MUST NOT take a screw longer than 1/4". That's because IMMEDIATELY on the inside of the threaded case directly in line of the screw is the PC board with the MOTOR ON circuit trace right there. Put in a longer screw and you break the trace; result, the motors don't turn on! A quick board repair brought both drives back in operation. The tech guy asked me if the drive had ever been mounted. Sounds like this problem is well

known about by them. Caution to all of you about this screw length restriction.

Model IV - Gate vs Non-gate

(Fm: Robert Beaubien) Can anyone tell me the difference between a "Gate array" and a "Non Gate array" model IV and how do you tell the difference. I have a Green Screen Model IV and to upgrade it to 128k all I had to do was move a jumper and install the chips. Do I have the gate array machine and if so what are the advantages or disadvantages of it.

(Fm: Ray Pelzer) The gate-array unit is the jumper-only, and has the RS-232 port out the back, while the non-gate unit needs the PAL chip and has the RS-232 next to the hard disk card edge.

Help with RS-232 specs

(Fm: David L. Cartwright) This information is located in the Model 4 Technical Reference Manual under RS-232C info. starting on page 169.

The Model 4 RS-232C/Modem interface conforms to EIA RS-232C standards and as such connecting the Model 4 to most modems should be a piece of cake. The RS-232C DB-25 connector pinout listing is as follows:

| | | |
|---------|---|---------------------------|
| PIN #1 | - | PGND (Protective Ground) |
| PIN #2 | - | TD (Transmit Data) |
| PIN #3 | - | RD (Receive Data) |
| PIN #4 | - | RTS (Request to Send) |
| PIN #5 | - | CTS (Clear to Send) |
| PIN #6 | - | DSR (Data Set Ready) |
| PIN #7 | - | SGND (Signal Ground) |
| PIN #8 | - | CD (Carrier Detect) |
| PIN #20 | - | DTR (Data Terminal Ready) |
| PIN #22 | - | RI (Ring Indicate) |

The ports used for the RS-232C device are E8H thru EBH. The Model 4 Reference Manual presents the information in much more detail.

2-Sided Drives

(Fm: Pat Amlung) I just got a hold of a

double sided drive for my Model III. Unfortunately, I can't get LDOS to format and verify the diskette. It works well with single sided operation, but not double sided operation. Does anyone have any clues? The drive I'm talking about is a Tandon 100-2.

(Fm: jeff brenton) Either replace the Radio Shack cable you are using (with one that has all 34 pins intact on all connectors), or flip it over at ALL connections. The pin that selects SIDES is not on a R/S cable, so you are actually writing BOTH sides' information on the FRONT of the disk. Only the last written can be seen by the controller, so the verify stage aborts.

See also my file DSDSK.HLP in DL6. it is under my other UserID of 73105,532.

Model I CPU Problems

(Fm: Gene Szedenits, Jr.) A friend of mine is having trouble with his Model I. Drive zero will whir to life at random times for no reason and his screen fills with repeated "@!@A"s. Any ideas?

(Fm: MISOSYS) Look for a broken pin on one of the ROMs. I recollect mention of some test in the Model I Tech Manual which would reproduce that screen image.

(Fm: LDOS Support) Oh, yeah, that's the @9's display. If there is no ROM or RAM working, but the CPU is running you'll get continuous alternating @'s and 9's on the display, with a strobe flicker.

Sounds like the standard Model I periodic maintenance is necessary. Take the sucker apart and clean both sides of every edge connector on the CPU and EI, then replace the CPU to EI cable. The cable goes intermittent from the vibration of typing, the pregnant and shielded flat cables are famous/the worst for this.

(Fm: LDOS Support) I'll go through my whole troubleshooting list for posterity's

sake.

Try this: hit the reset button while holding down the break key. If you get a memory size? prompt, at least we know that the CPU is running.

Hit enter. If you get a Level II BASIC signon message and a READY prompt, then the memory is at least marginally functional.

Type "? MEM" and hit enter. If the answer is below 16000, then communications with the EI is hosed up, or the EI itself has a problem. Replace the EI cable, check it's power supply and go again.

If the answer is above 16000, and one of the "Right" answers (which are 15570 or 15572 plus either 16384 or 32768), run the following program:

```
1 max% = mem - 100
2 if mem > 100 then print mem;:gosub 2
3 if mem < max% then print mem;:return
4 goto 2
```

This is a quickie memory test that fills up all memory with BASIC's stack. Wiggle the CPU to EI cable while this is running. Let it go all the way down, then all the way up and turn around to go back down again before you stop wiggling occasionally and go on to the next step. If the machine crashes, clean the connectors on the PC boards and replace the cable.

Now we go over to the EI. First, pull the RS232 board. Check the components for overheaters. The BRG and the DC-DC converter tend to short out and hose up the EI. If it seems ok, set it aside for later.

Check the EI for operation again. A bad RS232 can cause the whole EI to stop working. If it don't work we start with: disk isolation!

Ok, now to check out the disk functions. Pop the cover on the disk drive so that you can watch what's going on. Move the head off of track zero, and hit the reset button.

If the light on the drive doesn't light up, there is a configuration problem or the EI is pretty badly farkled.

If the drive lights up but doesn't restore to track zero, suspect the FDC and supporting circuitry.

If the drive restores to track zero but the screen does not clear, then the boot sector is not getting read into memory. Check the diskette and make sure that it is ok (the problem is how? finding another Model 1 person can be difficult. Perhaps somebody else here can send you a known good Model 1 TRSDOS 2.3 diskette.), also check the drive.

If the screen clears but then you get "disk error", the boot sector is going in OK, but it can't read the rest of the diskette. Again, could be bad diskette or drive.

If you are booting LDOS, and the system looks like it is booting fine until just before the LDOS Ready prompt comes up, you are missing index pulses coming back from the drive. Could be drive, cabling or an EI/FDC problem.

Let's treat checking out the drive itself internally as another topic. If you are ready to go ahead with that now, lemme know. The easiest route is going to be to find somebody with a known working Model 1, 3 or 4 that we can plug your drive into.

Model 4 RAM upgrade - Case I

(Fm: john g. gelesh) I just installed 8 4164 RAM chips in my 64K Model 4(26-1069A, green screen, redesigned keyboard). How do I access the extra memory? Everytime I try to use Memdisk I get a "cannot install Memdisk, requested bank in use." error message. What did I do wrong?

(Fm: David Huelsmann) If you have a non-gate array CPU, you have to remove the 4 position shunt from socket U72 and install the PAL chip. If your Z80 chip is in the upper left corner of the CPU board, you have a gate array CPU and no PAL chip is

necessary but there are a few other goodies that have to be done in a gate array board: You need to move the wire jumper from U5 pin 16 that goes to C39 and disconnect from C39 and solder to U33 pin 16. On even newer versions of the 4 (like the 4D) there is just a jumper to move (JMP13) from its present position to its alternate position.

Model 4 RAM upgrade - Case II

(Fm: Ted Pinkert) Unfortunately, the upper bank has something wrong with the memory refresh in the 2nd and 4th quarters of the 64K. (Those segments go bad 30 secs. after bootup, UNLESS one keeps accessing the memory, as does a memory test program!) So, it's now in the shop (not Radio Shack) getting fixed. I have a paranoid theory about 64K Model 4's.

(Fm: LDOS Support) I hope that they don't try to stick you for a replacement CPU board. The most likely problem is the wrong kind of RAM chips. There are two different kind of 64K chips: 256 row and 128 row refresh. Model 4 memory chips must be of the second variety, whereas the IBM PC and equivalents can use either type. Motorola RAM chips typically are used for the second category.

A set of the proper chips can be obtained from JAMECO for eight bucks plus shipping and handling. They do have a \$20 minimum order, and advertise in Byte, Computer Shopper, etc. They also have a kit complete with PAL (needed on older 4's) for \$15 + S&H.

(Fm: bob snapp) Please explain the difference between 128 row refresh and 256 row refresh chips: (a) how to tell from the schematic if the circuit is SPECIFICALLY designed for one or the other, or is 'generic', and (b) how to tell if a particular brand of chip is one or the other. I'm really more interested in the answer for 41256 chips, but I suppose that the principal is the same.

BOOT a Mode 4 from a HARD DISK

(Fm: Thomas L. Perry) To: bob snapp; I am interested in your desk top mod. If you find out the cost for the boot rom will you let us know?

(Fm: bob snapp) I was just gettin' ready to give the final report on this upgrade.

(a) Order part number MXP-0175, about \$7 at your local repair center. One of the books they might look in for the part number INCORRECTLY lists part number MXP-0195. Don't let them order that. Don't know what that one is, might even be a Mod II boot rom.

(b) Bend up pin 21 of aforementioned part so it doesn't go into socket, and run a wire from pin 21 to pin 24, attaching it so that pin 24 WILL still fit into the socket.

(c) Remove all three roms from your desktop '4' and put this one in the socket where the first (lowest numbered) of those three came out, leaving the other two socket empty.

(d) Use whatever software you have available that allows a 4p to boot directly from a hard drive. I use 'wdhdbtin', included in the POWERSOFT driver package. (BTW, the 'old' version of that won't work with 6.3 -- you need an upgrade).

(e) After this modification, this desktop 4 can NOT RUN IN MOD III MODE AT ALL, so if that's important to you (it isn't for me), don't bother doing this.

(Fm: Adam Rubin) Congrats on getting it all completed! The ambitious types could add an extra socket and a switch, if MIII mode really matters to them. I think your project would be a useful addition to one of the DLs here; have you thought about contributing it?

FDC 1793 Information

(Fm: LDOS Support) You can get information regarding the 1793 disk controller directly from Western Digital (714) 863-0102. Note that a complete set of the

"major" chips on this board (1793, 1691, 2143) is available from B.G Micro for about twelve bucks. They advertise in Byte and Computer Shopper. Note also that the side select lead is not generated by the 1793 in Tandy's configuration, it's generated externally by U6, a 74LS174. This is the only part common to both the internal and external side select outputs.

(Fm: Adam Rubin) WD lets their regional offices handle this kind of thing, but I don't know where you are. You could try (no results guaranteed!) their main office:

Western Digital Corporation
2445 McCabe Way
Irvine, CA 92714

After a few attempts, I managed to get their "Storage Management Products Handbook," which had all the nitty-gritty about all (as of date of publication) their floppy disk controller chips.

HD64180 information

(Fm: Duane Saylor) Did you know that the HD64180 also has 2 Asynchronous Serial Communication channels with baud rate generators up to 38,400 baud. The DMA is configured to work with these Serial channels. There is also 1 clocked serial port that can operate up to 200 k bits/sec (with clock of 4 Mhz). Two 16 bit counter timers are also on the chip. Thus you can see why there are 64 pins on this baby!

SYSATFIX and Alpha Tech board

(Fm: David Huelsmann) Roy, probably more of a problem with something I did but that I would pass it on for info: I have the Alpha SuperMem 1 meq board. I had installed your patches to the system for the memory bank expansions but I usually use the Security Systems software for installation and use of the RamDrive. I recently received PRO-MRAS and had installed SAID using SAIDINS. I always reserve the alternate 64K bank for use by application programs. When SAID came up,

it showed that I had 7 banks available which shouldn't have been the case - should only have been 2 banks. Somewhere, I managed to trash BOOT/SYS and SYS6/SYS while experimenting with SAID. After fixing up everything again, I went back to a system with no bank patches installed. Now SAID comes up with 2 banks available and everything is fine.

Question: Are the patches not as well behaved as they should be? or could there be some interference of the Patches and the Security Systems driver? Again, not a critical issue, more for info than anything else.

(Fm: MISOSYS) The Security Systems driver does not use the @BANK handler; thus, don't use the ATFIX if you are using that driver. The guy who wrote it died of a heart attack (Ben Mitchel) before he could revise the driver to use the "standard".

XLR8er "Ramdisk"

(Fm: Bill Schaper) The RAMDISK is the utility (a'la MEMDISK) that allows the use of the upper 8 (10) banks of memory with the XLR8er board. I have one installed in my Model 4. It is unfortunate that I cannot access the board with LDOS 5.3. I sure do like the changes in the new DOS.

(Fm: Jim Gaffney) As a matter of fact, drivers have been developed for LDOS which are comperable to the Model 4 versions. In fact, LDOS seems to run faster under the XLR8er than TRSDOS 6.2 does. I got the files from "the boys in Oklahoma", but

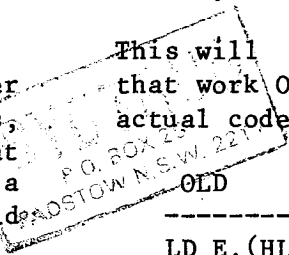
H.I. Tech probably has them, too.

More on the AT patch

(Fm: Doug Tittle) I find that your initialization code for the Alpha Tech banks doesn't work properly on my machine. Banks which are not present yield RANDOM results when read, and so are "detected" as valid by your code. I have a Gate Array 4P with 512K installed. As others may also have this problem, I submit the following patch to your code for SYS0 at 1E5AH:

DOC,E8=ED 41 5E 70 7E B8 73 20 09

This will probably NOT work on machines that work OK with your testing method! The actual code change is:



OLD
LD E,(HL)
INC (HL)
LD A,(HL)
CP E
LD (HL),E
JR Z,xxxx

NEW
LD E,(HL)
LD (HL),B
LD A,(HL)
CP B
LD(HL),E
JR NZ,xxxx

Please feel free to distribute this to others in need. I also have a patch for the 6.2 version which does both types of testing if anyone wants it. There isn't room in 6.3 for both tests!

(Fm: MISOSYS) Your machine should NOT be yielding random results on banks which are not installed. They should ALWAYS be yielding an X'FF'. However, I'll note your observations.

Still more on the AT patch

(Fm: Duane Saylor) I believe I have found an error in your patch for the Alpha Tech Megaboard which is apparently not

| | | | | |
|-------------|-------|------|-------------|---------------------------------|
| 0831 23 | 00410 | INC | HL | |
| 0832 C5 | 00420 | PUSH | BC | |
| 0833 0E43 | 00430 | LD | C,43H | ;ALPHA memory mgt port |
| 0835 ED40 | 00440 | IN | B,(C) | ;Get present bank |
| 0837 23 | 00450 | INC | HL | |
| 0838 70 | 00460 | LD | (HL),B | ;Save bank number |
| 0839 47 | 00470 | LD | B,A | ;A and B both now 0 |
| 083A 3A7800 | 00480 | LD | A,(OPREG\$) | ;P/U prt mask |
| 083D 3002 | 00490 | JR | NC,I2 | ;Bypass if NC (no stack switch) |
| 083F E67F | 00500 | AND | 7FH | ;Strip bit 7 to use as flag |
| 0841 77 | 00510 | LD | (HL),A | ;Save current state fm OPREG\$ |

I contend that line 450 and 460 should be interchanged. The code as it stands stores the current state of OPREG\$ AND 7FH over top of the present BANK. The code at 0858 to 0865 retrieves OPREG\$, DECrements HL and then retrieves the saved BANK number. Thank you any clarification you can provide on this matter.

(Fm: MISOSYS) I would agree with you, Duane. The bit of code which is reversed would get executed only if the stack pointer was above X'F3FC'. That actually should occur at times. I haven't analyzed the effect of the code reversal; however, since it hasn't caused a problem, I won't waste the time to investigate. On the other hand, your suggestion should be implemented.

RamDrive driver & LS-DOS 6.3

(Fm: Pete Granzau) Users of Security & Software's RAMDRV44/DCT have indicated it doesn't work in LS-DOS 6.3. I don't have LS-DOS 6.3 yet, so I can't test this out, but in the RAMDRV44/DCT I have, there is a test for a hex 62, and a jump if not equal. The following patch MIGHT cure part of the problem (of course, I make no representation as to its correctness, as I can't test): PATCH RAMDRV44/DCT.UTILITY (D00,83=DA:F00,83=C2) This should change the JP NZ,xxxx to JP C,xxxx, nu? You're still going to have problems with OTHER things wrong, but at least it won't reject

catastrophic since I have been using your patch and the RAMDRIVE software I obtained on this SIG for about 11 months. The code in question from the modified ENADIS_DO_RAM follows:

LS-DOS 6.3 as the wrong version of the DOS.

More on RamDrive driver & LS-DOS 6.3

(Fm: Ken Kane) I am using 6.3, a 15 meg hard disk and am on the brink of installing 1/2 meg of Alpha Tech RAM from AIA computers.

The software end is getting a bit confusing; the instructions from Alpha Technology don't mention any patches needed to TRSDOS; but I'll operate with Roy's patches in place. (Blindly, I might add, this is all over my head). So I'm startled to hear from you that yet another patch is needed. Anyway, if I need it, I'd sure appreciate having it! Would it be to FLOPPY/DCT or TRSHD6/DCT? Please be specific I can follow directions to install a patch but can't bridge any knowledge gaps here; this is NOT my field!

Do I need the contents of FDR6AT? What will I find when I get there?

(Fm: David Huelsmann) The AlphaTech board should not interfere with your hard disk at all (TRSHD6/DCT). The RAMDRIV/LQR file contains a device driver (actually several depending on your application) one of which is called RAMDRIV3/DCT. You will need Roy's patches from the last issue of TMQ. This will enable the ability for the RAMDRIV3/DCT to access up to 30 banks of additional ram. The one byte patch to

RAMDRIV3/DCT is to allow the use of the new file timestamping in 6.3. It becomes a lot like installing the MEMDISK/DCT that is described in your manual for accessing a 128K model 4 memory. RAMDRIV3/DCT was developed from the original MEMDISK/DCT that is supplied on your disk. Alpha Tech doesn't mention anything cause they assume you are going to buy their software to use the system. If you did and if the software they supplied was for LS-DOS 6.3, then you can just follow those instructions. If you didn't buy any software to access the additional memory, then, we are talking about using the public domain software that is RAMDRIV3/DCT, along with FDR6AT and MOVE/CMD. Clear as mud, right? Download the RAMDRIV/LQR, unsqueeze it, and delibrary it and we will go from there. Don't panic, we will get you going.

RamDrive driver & 6.3 saga continues

(Fm: Duane Saylor) I have an Alpha Technology board installed in my personal Model 4 and one in the Model 4 at work. Both were installed by RAI. When my board was installed, RAMDRV44/DCT was the only choice I had and so I used it until RAMDRIV3/DCT by Michael Jacobs was uploaded to Compuserve Information Service in March 1986. I immediately switched to RAMDRIV3/DCT for the following reasons:

1. It can be loaded by a mirror image backup from one of my 80 track drives. (I have 2 - 5.25 inch and 1 - 3.5 inch 80 track doubled/sided drives.) I create an 855k ramdisk and load the bottom 720k from a floppy. This leaves over 120k left for scratch.
2. It uses only operating system calls and no direct hardware dependent code. I have been around the Z-80 world since November 1977 and appreciate software that uses the operating system only for access to hardware because it makes for a much lower probability of conflict with other software. I took it upon myself to fix conflicts with COM/DVR by both ORCTERM and XT4 that were caused by the author not using available system resources. With the AT patches installed I reserve 3 banks for other uses. One of these normally goes to

PRONTO, 1 to SPOOL and 1 for a SAID text buffer.

3. My system boots from a cold start, loading the ramdisk and making it the system drive in exactly 3 minutes. The mirror image floppy is then removed and replaced with another for data files or other utility programs.

4. Warm boot if the system has already been loaded and power has not been lost takes 30 seconds. During boot a JCL file is executed and the operator must select a 1 for COLD or 2 for WARM boot.

Model III Network ROMC

(Fm: MISOSYS) Concerning the ROMC in a Model III/4; this can be ordered from National Parts in Ft Worth (817-870-5662). The part number of a standard Model III ROMC chip is MXP-0167. A Network III ROMC is part number AXX-7078. The cost for the standard chip is only \$6 plus shipping. That will turn your Network Model III into a normal one.

Note that although LDOS 5.1.4 will not BOOT on a Model III equipped with a Network III ROMC, LDOS 5.3 will. This is one of the changes we introduced into the LDOS 5.3 product. The catch is that I could not circumvent the Network ROM's dead key handler - the three bytes it uses are LDOS's @ICNFG vector - I couldn't change that.

The other good point is that the three ROMC memory regions which I had determined to be consistent across ROM releases still holds with that "network" ROM. The regions are used for cassette I/O and boot handling. In my hardware interface kit which includes an @BANK handler and an @EXMEM handler (char & page I/O from/to banked memory), I reuse these regions after switching to the Model III RAM mode.

The Patch Corner

General Information

The following information should be read before you type into a file, any of the patches noted in THE MISOSYS QUARTERLY.

It is unfortunate that our printer prints the letter "O" and the number "0" almost identically. Unless we utilize a filter to "slash" the number zero, the two are difficult to distinguish. However, when it comes to patches, all is not lost. In an LDOS 5 or TRSDOS 6 direct patch, the letter "oh" is not used in the patch code (it may appear in comments which are lines beginning with a dot). The direct patch format of TRSDOS 6 which we use in our patches is:

```
Drr,bb=xx xx xx xx xx xx ...
Frr,bb=xx xx xx xx xx xx ...
```

The patch is usually a pair of lines. The first line begins with the capital letter, "dee". This is immediately followed by the "rr" field (which stands for record). The "rr" field is always two hexadecimal digits. Actually, it can be a 4-hexadecimal digit number if the file to be patched has more than 256 sectors. Hex digits use nothing but the numbers zero through nine and the first six letters of the alphabet: A,B,C,D,E,F, or a,b,c,d,e,f. The record number is immediately followed by the "bb" field (which stands for byte). The byte field is also two hexadecimal digits - just like the record field. This is immediately followed by an equal sign, "=". The equal sign is immediately followed by the first patch byte (the "xx" shown above). The patch byte is again two hexadecimal digits. Where more than one patch byte is included on a line, it is separated from its predecessor by a single SPACE. The line is terminated with an ENTER.

TRSDOS 6 and LDOS 5.3 patch formats use a "find" line record. This is used to verify

=====

The following fixes are associated with LDOS 5.3

- . BACKUP/FIX - 01/16/87 - Patch to LDOS 5.3 BACKUP/CMD
- . Patch corrects use of DATE parameter.
- . Apply via, PATCH BACKUP.UTILITY using BACKUPA
- D06,DC=00:F06,DC=47
- . Eop

that the file being patched is actually the file you want patched. All of the bytes noted in the "F" line or lines must be matched in the file before any of the "D" patches will be utilized. The second line of the pair begins with the letter "F" which stands for FIND. The next six positions are identical to the preceding "D" line. Following the equal sign on the FIND line are pairs of hexadecimal digits which should align themselves with the preceding line.

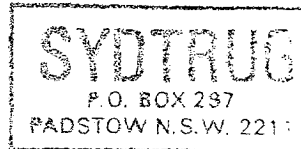
So far, the letter "oh" is not used. The only place outside of a comment line where you could find the letter "oh" used is if instead of showing the patch bytes as a series of hexadecimal pairs, it was depicted as a string. A string could be used if one was patching a string of displayable ASCII characters. For instance, the patch:

```
D03,14="This is a new string"
F03,14="extra space for what"
```

would replace the string, "extra space for what", with the string, "This is a new string". Strings are shown within double quotes. That's the only place where a letter "G" through "Z" could be used.

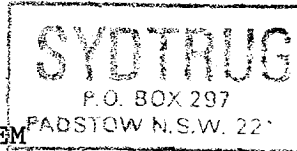
Also, even though TRSDOS supports the colon notation to put more than one patch line on the command line (e.g. "PATCH TEST (D01,27=56:F01,27=65)"), it does not support the colon separator when used in a FIX file (it does support a semicolon which is used under LDOS to signify a trailing comment); LDOS 5.3 supports a colon separator both in a command line patch and a fix-file patch. In order to conserve space in THE MISOSYS QUARTERLY, we may logically print more than one FIX line on a printed line; HOWEVER, ALWAYS USE A HARD <ENTER> FOR THE COLON WHEN TYPING IN A FIX FILE for TRSDOS 6.

- . BACKUPB/FIX - 02/11/87 - Patch to BACKUP/CMD
- . Corrects error in calculating destination disk size
- . Apply via, PATCH BACKUP.UTILITY using BACKUPB
DOB,31=7E 06 3D:F0B,31=4E 06 0D
- . Eop
- . BASOV4A/FIX - 01/14/87 - Patch to BASIC/OV4
- . This FIX corrects CMD"V" for scalars
- . Apply via, PATCH BASIC/OV4.BASIC using BASOV4A
D00,94=28:F00,94=20
- . Eop
- . FORMATMX/FIX - Patch to FORMAT for MAX-80 only - 03/09/87
- . Some MAX-80's have problems formatting 5" disks due to controller tolerance problems with the new format gap data. The old gap data can be used by applying this patch; PATCH FORMAT.UTILITY FORMATMX
- . The QFB program can be altered as well using QFBMX/FIX.
D05,8D=14:F05,8D=20:D05,AD=11:F05,AD=17
- . Eop
- . PRFLTA/FIX - 02/05/87 - Patch to PR/FLT
- . Fix corrects XLATE parameter:. Apply via, PATCH PR/FLT.FILTER using PRFLTA
D00,A0=F2 54:F00,A0=01 00:D00,A4=F3 54:F00,A4=02 00
- . Eop
- . PRFLTB/FIX - 02/14/87 - Patch to PR/FLT
- . Changes printer status check to IN A,(OF8H)
- . Apply via, PATCH PR/FLT.FILTER using PRFLTB
D03,66=DB F8 00:F03,66=3A E8 37
- . Eop
- . QFBMX/FIX - Patch to QFB for MAX-80 only - 03/09/87
- . See text in FORMATMX/FIX. Apply via; PATCH QFB.UTILITY QFBMX
DOB,12=14:F0B,12=20:DOB,32=11:F0B,32=17: DOC,32=14:F0C,32=20:DOC,52=11:F0C,52=17
- . Eop
- . SYS0A/FIX - 03/19/87 - Patch to SYS0/SYS of LDOS 5.3.0
- . Apply via, PATCH SYS0/SYS.SYSTEM SYS0A
D02,87=53:F02,87=51
- . Eop
- . SYS0AM/FIX - 03/19/87 - Patch to SYS0/SYS of LDOS 5.3.0Max
- . Apply via, PATCH SYS0/SYS.SYSTEM SYS0AM
D1A,21=53:F1A,21=51
- . Eop
- . SYS2A/FIX - 01/14/87 - Patch to SYS2/SYS of LDOS 5.3
- . Apply via, PATCH SYS2/SYS.SYSTEM using SYS2A
X'4FBB'=C3 B2 51:X'51B2'=CB 66 CA A0 4F 3E 05 85 C3 BE 4F
- . Eop
- . SYS6A/FIX - Patch to LIST command - 01/14/87
- . Corrects BREAK escape & 'C' continue
- . Apply via, PATCH SYS6/SYS.SYSTEM using SYS6A
D29,67=20:F29,67=A6:D29,AD=C9 54:F29,AD=1B 00
- . Eop
- . SYS6B/FIX - 01/18/87 - Patch to DIR library command
- . Corrects display of free space
- . Apply via, PATCH SYS6/SYS.SYSTEM using SYS6B
D05,DC=04 C5 7E 37 1F 38 01 13 FE FF 20 F7 2C 10 F3 21
F05,DC=C5 7E 37 1F 38 01 13 FE FF 20 F7 2C 10 F3 21 0D
D05,F0=0D 5B CD 94 57 32 13 5B F1:F05,F0=5B CD 94 57 32 13 5B F1 3C
- . Eop
- . SYS6C/FIX - 02/05/87 - Patch to MEMORY & KILL
- . Allows finding modules with '\$' as part of name
- . Corrects exit looping on device kill



Late Breaking Patch for LDOS 5.3 Max80
 PATCH SYS0/SYS.SYSTEM (X'4214'=00)
 [initializes scroll protect]

- . Apply via, PATCH SYS6/SYS.SYSTEM using SYS6C
D02,C0=21:F02,C0=30:DOF,6A=D5 DD E1:FOF,6A=CD F1 49
- . Eop
- . SYS7A/FIX - 01/16/87 - Patch to LDOS 5.3 PURGE/CMD
- . Patch corrects use of DATE parameter.
- . Apply via, PATCH SYS7/SYS.SYSTEM using SYS7A
D26,38=00:F26,38=47
- . Eop
- . SYS7AM/FIX - 01/16/87 - Patch to LDOS 5.3 MAX-80 PURGE/CMD
- . Patch corrects use of DATE parameter.
- . Apply via, PATCH SYS7/SYS.SYSTEM using SYS7AM
D26,32=00:F26,32=47
- . Eop
- . SYS7B/FIX - 01/18/87 - Patch to FREE library command
- . Corrects display of free space
- . Apply via, PATCH SYS7/SYS.SYSTEM using SYS7B
DOB,7F=04 C5 7E 37 1F 38 01 13 FE FF 20 F7 2C 10 F3 21 38 55 CD
FOB,7F=C5 7E 37 1F 38 01 13 FE FF 20 F7 2C 10 F3 21 38 55 CD 09
DOB,96=09 54 32 3E 55 F1:FOB,96=54 32 3E 55 F1 3C
- . Eop
- . SYS7C/FIX - 01/18/87 - Patch to SYSGEN library command
- . Adds SMOOTH parameter to configuration file
- . Apply via, PATCH SYS7/SYS.SYSTEM using SYS7C
DOA,75=01 00 F7 46:FOA,75=00 00 00 00
- . Eop
- . SYS7D/FIX - 01/18/87 - Patch to LDOS 5.3 SYSTEM
- . Corrects SYSTEM (BLINK,LARGE) parameter
- . Apply via, PATCH SYS7/SYS.SYSTEM using SYS7D
DOF,01=C3 16 53 00 00 00:FOF,01=32 1C 40 C3 1C 53:DOF,0C=F7 01 02 00 53 52
:FOF,0C=10 01 02 00 53 53:D19,19=F7 52:F19,19=10 53
- . Eop
- . SYS7DM/FIX - 01/18/87 - Patch to LDOS 5.3 MAX-80 SYSTEM
- . Corrects SYSTEM (BLINK,LARGE) parameter
- . Apply via, PATCH SYS7/SYS.SYSTEM using SYS7DM
DOE,FB=C3 10 53 00 00 00:FOE,FB=32 1C 40 C3 16 53
DOF,06=F1 52:FOF,06=0A 53:D19,13=F1 52:F19,13=0A 53
- . Eop
- . SYS7E/FIX - 02/03/87 - Fix to ATTRIB (SYS7/SYS)
- . Permits use of (PROT=NO) according to documentation
- . Apply via, PATCH SYS7/SYS.SYSTEM using SYS7E
D1B,FF=00 00:F1B,FF=28 11:D1F,6B=4E 4F:F1F,6B=41 4C
- . Eop
- . SYS7EM/FIX - 02/03/87 - Fix to ATTRIB (SYS7/SYS) MAX-80
- . Permits use of (PROT=NO) according to documentation
- . Apply via, PATCH SYS7/SYS.SYSTEM using SYS7E
D1B,F9=00 00:F1B,F9=28 11:D1F,65=4E 4F:F1F,65=41 4C
- . Eop
- . TED5A/FIX - 02/01/87 - Fix for LDOS 5.3 TED/CMD
- . Corrects <BREAK> in insert; block copy with no room!
- . Apply via, PATCH TED.UTILITY using TED5A
D03,BA=E0:F03,BA=EC:D07,2E=D4:F07,2E=CD
- . Eop



=====
The following fix is associated with PRO-EnhComp

- . BC62/FIX - Patch to PRO-EnhComp BC/CMD - 11/05/86 - Corrected
- . Patch corrects BREAK handler with -NX option specified
- . Apply via: PATCH BC BC62

D29,E5=09 2F:F29,E5=06 45
 D05,21=CD 06 45 0E BF C3 06 45:F05,21=00 00 00 00 00 00 00
 . Eop
 =====

The following fixes correct MISOSYS products for LDOS 5.3 or LS-DOS 6.3

. BSORT53/FIX - 02/24/87 - Patch to BSORT51 for LDOS 5.3
 . Apply via, PATCH BSORT (D00,7C=6E:F00,7C=68)
 . DOCON53/FIX - 02/19/87 - Patch to DOCONFIG for LDOS 5.3
 . Apply via, PATCH DOCONFIG using DOCON53
 D00,79=2A:F00,79=16:D00,84=43 59:F00,84=15 52:X'5943'=3E 59 32 2E 52 C3 29 52

. Eop
 . DOCON63/FIX - 02/18/87 - Patch to PRO-GENY's DOCONFIG for LS-DOS 6.3
 . Apply via, PATCH DOCONFIG (D03,91=D8:F03,91=C0)
 . PDS663/FIX - 02/14/87 - Patch to PRO-PaDS for LS-DOS 6.3's ATTRIB
 . Fixes PDS(BUILD) to revamp the internal ATTRIB command string
 . Apply for LS-DOS 6.3 only via, PATCH PDS.PDS using PDS663
 D1B,8A=70 3D 52 45 29 0D;F1B,8A=75 3D 2C 70 3D 52

. Eop
 . SAID53A/FIX - 01/16/87 - Patch to SAID V1.1 for LDOS 5.3
 . This patch lets SAID find the DOS KI/DVR driver in memory
 . Apply via, PATCH SAID using SAID53A
 D2A,35=49:F2A,35=33:D2A,C2=00 00 00 00 11 0D 00
 :F2A,C2=23 7E 23 66 6F 11 E7 FF:D2B,38=24 4B 49 34:F2B,38=4B 49 31 0D

. Eop
 . SAID53B/FIX - 01/16/87 - Patch to SAID V1.1 for LDOS 5.3
 . This patch lets SAID find the DOS KI4/DVR driver in memory after
 . installing SAID53A/FIX first. Apply via, PATCH SAID using SAID53B
 D2A,B1=04:F2A,B1=03:D2A,C8=0E:F2A,C8=0D

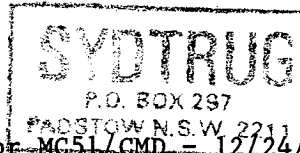
. Eop
 . ZSHELL53/FIX - 02/18/87 - Patch to ZSHELL 2.2 for LDOS 5.3
 . Apply via, PATCH ZSHELL using ZSHELL53
 D01,13=50:F01,13=4A:D09,86=50:F09,86=4A

. Eop
 =====

The following fixes are for MC and PRO-MC

. GETDEC51/FIX - rewritten getdeclarator() for MC51/CMD - 12/24/86
 . Appears only on DISK NOTES 7 (FIXES7/TXT)
 . GETDEC52/FIX - rewritten getdeclarator() for MC52/CMD - 12/24/86
 . Appears only on DISK NOTES 7 (FIXES7/TXT)
 . GETDEC60/FIX - rewritten getdeclarator() for MC60/CMD - 12/24/86
 . Appears only on DISK NOTES 7 (FIXES7/TXT)
 . GETDEC61/FIX - rewritten getdeclarator() for MC61/CMD - 12/24/86
 . Appears only on DISK NOTES 7 (FIXES7/TXT)
 . GETDEC62/FIX - rewritten getdeclarator() for MC62/CMD - 12/24/86
 . Appears only on DISK NOTES 7 (FIXES7/TXT)
 . MCPUN5/FIX - correct #define operation on previously #undef'ed macro

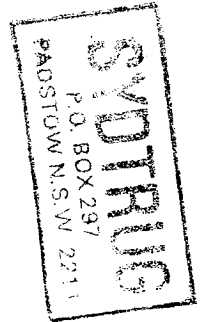
D1F,0B=21 E3 81 34 CD:F1F,0B=2A C1 6D E5 CD
 D1F,10=F9 81 22 C7 6D EB 35 7A B3 CA FE 71 21 06 00 19
 F1F,10=B0 81 F1 22 C7 6D 7C B5 CA FE 71 2A C7 6D 11 06
 D1F,20=CD 8F C2 01 01 01 02 77 71 00 ED 4A 28 41 ED 4A
 F1F,20=00 19 CD 8F C2 01 02 77 71 11 FE FF CD 82 C1 7C
 D1F,30=28 3D 3A 23 52 B7 28 37 00 00:F1F,30=B5 28 03 2A 23 52 7C B5 28 35
 D2F,D0=F1 E1 E5 F5 3E 00 B7 C0 EB 21 06 00 19 CD 8F C2
 F2F,D0=F1 E1 E5 F5 11 06 00 19 CD 8F C2 11 FF FF CD 7C
 D2F,E0=01 01 00 ED 4A C8 ED 4A C8 EB C9 E5 2A C1 6D E5
 F2F,E0=C1 7C B5 20 11 F1 E1 E5 F5 11 06 00 19 CD 8F C2
 D2F,F0=CD B0 81 F1 D1 C9 00 00 00 00 00 00 00 00 00



F2F,F0=11 FE FF CD 7C C1 7C B5 28 05 21 00 00 18 04 F1
D30,00=00 00 00 00:F30,00=E1 E5 F5 C9

. Eop
. MCPUN6/FIX - correct #define operation on previously #undef'ed macro
D1F,24=21 00 56 34 CD 16 56 22 E4 41 EB 35
F1F,24=2A DE 41 E5 CD CD 55 F1 22 E4 41 7C
D1F,30=7A B3 CA 1B 46 21 06 00 19 CD 5C 96 01 01 01 02
F1F,30=B5 CA 1B 46 2A E4 41 11 06 00 19 CD 5C 96 01 02
D1F,40=94 45 00 ED 4A 28 41 ED 4A 28 3D 3A 23 26 B7 28
F1F,40=94 45 11 FE FF CD 4F 95 7C B5 28 03 2A 23 26 7C
D1F,50=37 00 00;F1F,50=B5 28 35
D2F,E9=F1 E1 E5 F5 3E 00 B7;F2F,E9=F1 E1 E5 F5 11 06 00
D2F,F0=C0 EB 21 06 00 19 CD 5C 96 01 01 00 ED 4A C8 ED
F2F,F0=19 CD 5C 96 11 FF FF CD 49 95 7C B5 20 11 F1 E1
D30,00=4A C8 EB C9 E5 2A DE 41 E5 CD CD 55 F1 D1 C9 00
F30,00=E5 F5 11 06 00 19 CD 5C 96 11 FE FF CD 49 95 7C
D30,10=00 00 00 00 00 00 00 00 00 00 00 00 00
F30,10=B5 28 05 21 00 00 18 04 F1 E1 E5 F5 C9

Late Breaking Patch for SET2RAM
PATCH SET2RAM (D00,50=00 38:F00,50=FF 37)
[corrects BASIC's TIME\$ function]



. Eop

=====
The following fixes are for the Hardware Interface Kit

. LIKBA/FIX - 02/05/87 - Patch to BANKER/CMD
. Fixes (FREE) parameter:. Apply via, PATCH BANKER using LIKBA
D00,6E=66:F00,6E=65

. Eop
. LIKKA/FIX - 02/05/87 - Patch to KI4/DVR
. Changes "Screen print" to <CTRL><colon>:. Apply via, PATCH KI4/DVR using LIKKA
D03,F2=3A:F03,F2=2A

. Eop
. LIKMA/FIX - 02/05/87 - Patch to MEMDISK/DCT
. Fix corrects bank test on or <C>:. Apply via, PATCH MEMDISK/DCT using LIKMA
D03,2A=05:F03,2A=06

. Eop

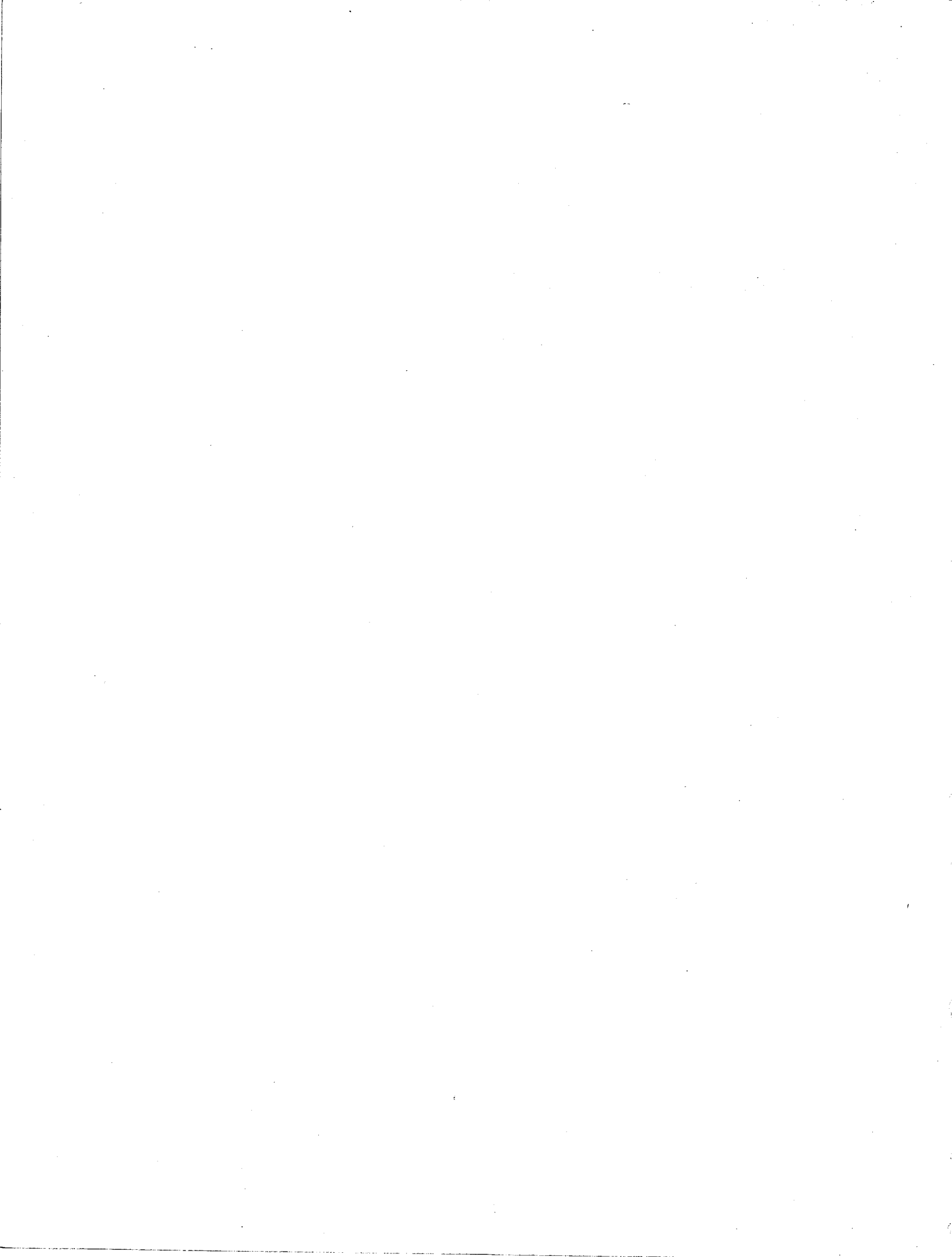
=====
The following fix omitted from TMQ I.ii is for MRAS

. MRS610/FIX - Patch to MRAS - 10/20/86
. Patch corrects a number of problems with COMMONs
. Apply via: PATCH MRAS MRS510
D02,46=75 8F:F02,46=01 8F:D1E,9C=CD 5D 8F:F1E,9C=22 2B 58:D1E,A7=66 8F:F1E,A7=11 5C
D1E,EE=53 8F:F1E,EE=DC 78:D20,67=CD 1C 8F:F20,67=E5 E6 03:D20,97=48 8F:F20,97=11 5C
X'8F1C'=E3 E5 47 E6 03 FE 03 C0 F5 C5 78 E6 1C OF OF 47
X'8F2C'=3A D4 67 B8 28 13 78 32 D4 67 87 21 D5 67 85 6F
X'8F3C'=8C 95 67 CD C6 6F CD 43 79 C1 F1 C9 E1 C1 C5 E5
X'8F4C'=78 CD 1F 8F C3 11 5C CD DC 78 3A D4 67 32 67 8F
X'8F5C'=C9 22 2B 58 3E FF 32 67 8F C9 3E FF 3C 28 07 3D
X'8F6C'=07 07 47 CD 24 8F C3 11 5C

. Eop

=====
The following fix is for GOBLER

. GOBLERJ1/FIX - Patch to correct joystick operation
. due to incomplete info received from Alpha Products.
. Apply via: PATCH GOBLERJ/CMD GOBLERJ/FIX
. At 6C2FH:D0D,1E=2F 4F 00 E6 1F:F0D,1E=4F E6 OF FE OF
. At 7F3BH:D1B,DE=CD 69 7F:F1B,DE=32 1D 80
. At 7F67H:D1C,0A=18 0A 32 1D 80 F6 10 C9:F1C,0A=00 00 00 00 00 00 00 00
. Eop



MS-DOS

LB86™ – A versatile, menu driven, flat file data manager that's easy to use. Supports 65K records, 1K chars/rec, 64 fields, 254 chars/field. Definable database records, input screens, and printout formats let you set up your data management with absolutely no programming required! To keep your data organized, LB86 lets you select and sort on up to 8 fields; keeps 5 index files per data base. Extensive on-line help. Use for mailing lists, customer lists, registration data, etc. \$74.95 + \$5 S&H

LBMU86™ – Utility used to build new LB-86 data files using the data from an existing one. Add or delete fields, convert field types \$29.95 + \$2 S&H

FED86™ – A sector-oriented file and disk editor for PC-DOS 2.x or equivalent. Display and edit any byte of any file or disk. ASCII or hex search. Print sectors; list files in hexadecimal \$29.95 + \$2 S&H

FM86™ – A useful aid to moving files or group of files from one drive to another. Mark files by name/time, wildcarding, old, new, by size, or individual selection. Purge marked files. Touch date/time \$29.95 + \$2 S&H

DED86™ – A powerful sector-oriented disk and file editor as well as a page-oriented memory editor. Position by cylinder/head/sector, logical sector, or cluster on floppies or hard drives. Jump around subdirectories. Edit any sector in hexadecimal or ASCII. Designate sectors to "keep" in a list for dumping to a disk file. Examine/alter memory and dump a RAM "keep" list to a disk file. Great for a text recovery after a program crash. Requires DOS 2.x or higher . \$59.95 + \$5 S&H

ED/ASM-86™ – An integrated editor, assembler, linker and debugger for 8086, 80186, 80286, and 8087 instruction assembly and disassembly. Has a full-featured line EDITOR with intra-line editing, block move and copy, partial save and load, tokenized source for saving memory and disk space (a full-screen editor is supported on CGA-equipped systems). Structured coding with the macro ASSEMBLER and direct .COM or .EXE write capabilities with direct assembly in memory makes your job easier. The DEBUGGER includes symbolic disassembly of arbitrary code to disk and use of complex expressions with previously defined symbols and/or register values. The LINKER is an assembler instruction that takes an ED/ASM-86 link file (.LNK) and treats it as if it were an original source file. ED/ASM-86 allows any logical combination of in-memory source codes, INCLUDES and LINKs, and can produce from these a single .COM, .EXE, .LNK, or in-memory program \$89.95 + \$5 S&H

RATFOR-86™ – A professional implementation of RAtional FORtran. Provides structure and greater portability to FORTRAN programs. Fully documented with tutorial user manual and sample programs. Requires a FORTRAN compiler \$99.95 + \$5 S&H

DSM-86™ – A high-speed disk virtual sorting utility that creates an index file. Sorts on compressed integer, single, and double precision numbers, as well as ASCII strings. Supports up to 65K records or RECLen up to 1K. Specify up to 24 select fields \$59.95 + \$5 S&H



MISOSYS, Inc.
 PO Box 239
 Sterling, VA 22170-0239
 703-450-4181 MC, VISA, CHOICE
 Orders Only! 800-MISOSYS 1P-5P EST M-F

VA residents add 4½%
 sales tax
 S&H: Canada add \$1;
 Foreign use S&H times 3