

THE MISOSYS QUARTERLY

In this issue:

Extend DATE\$ of Model I LDOS 5.1.4

Input SUBroutine for QuickBASIC

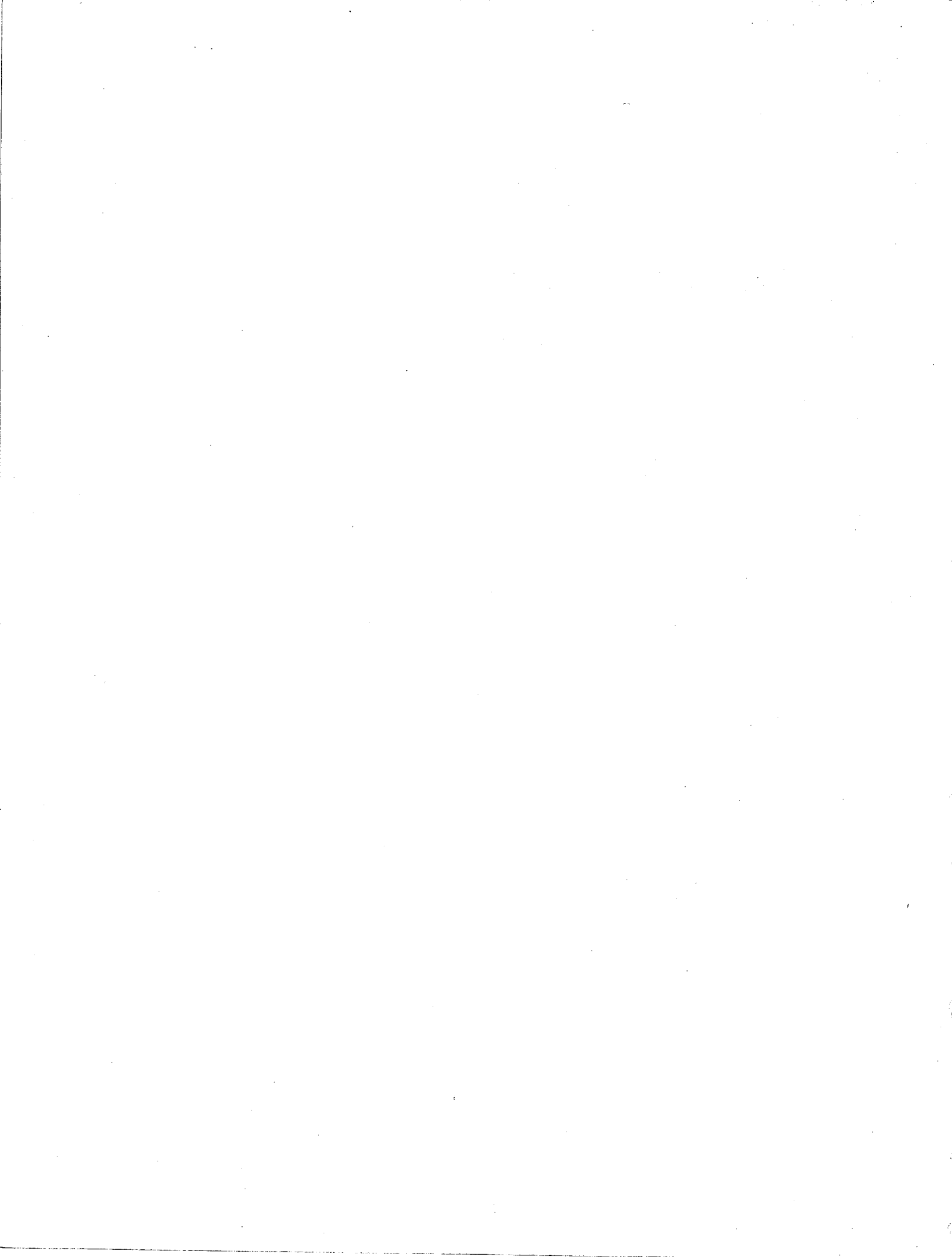
BASIC Interface to @EXMEM

HIRES Graphics for MC

Focus on speed



SYDNEY
P.O. BOX 296
PADSTOW N.S.W. 2211



THE MISOSYS QUARTERLY

Volume II, Issue ii

Fall 1987

Table of Contents

The Blurb	2
Letters to the Editor	5
Announcing new products	17
The Gobbling Box	17
Lair of the Dragon	17
HQ	18
MC Library update	18
GreatBac-86	19
XLR8	19
LDOS™/Model I/III Information	20
LS-DOS™/Model 4 Information	25
The LSI Column	31
MS-DOS™ Information	34
Applications for the User	45
Focus on SPEED	45
Hires Graphics for MC, by H. G. Clayton	61
Access EXMEM from BASIC, by Edwin Ketola	63
All Input Subroutine for QB, by W. D. Schroeder	67
MISOSYS Products' Tidbits	74
Assemblers: EDAS, MRAS	74
C Language: MC	76
Doubles to ints: an exact science?	81
diskDISK	83
LB Data Manager	91
SAID and TABs	92
PRO-WAM Window & Application Manager	93
The Patch Corner	100

Copyright © 1987 by MISOSYS, Inc., All rights reserved
PO Box 239, Sterling, VA 22170-0239
703-450-4181



The Blurb

Yes, it's really QUARTERLY!

We did everything we could; we pulled in all of the stops. And we wound up getting this issue to press precisely two months since TMQ II.i went to press. That means we are just about back on schedule. The last issue (II.i) was mailed on 9/28/87. I'm writing this on Sunday November 15th. The rest of II.ii is complete. This issue should be at G W Press by Tuesday. I should get it back and into the mail by November 24th or 25th (or sooner). Although I still didn't beat the requirement of getting the fall 87 issue out before the cord wood was needed. We got over a foot of snow here on Veterans' Day! This town is unbelievable! The Woodrow Wilson bridge, that channels interstate 95 traffic from South of Washington to North of Washington was CLOSED for over 12 hours! Folks were spending the night in their cars and trucks! Boy, I am sure glad I don't have to commute far to go to work.

Which brings me to the subject of mail delays. Here is what the Domestic Mail Manual (DMM) says about Third Class Mail in section 630, "Third-class mail may receive deferred service. The Postal Service does not guarantee the delivery of third-class mail within a specified time." I have heard from some TMQ readers claiming FOUR weeks to receive their TMQ. I have investigated Second-class mailing for TMQ. Here's what the DMM manual says in section 430, "The Postal Service does not guarantee the delivery of second-class mail within a specified time. Publications authorized second-class entry will be given expeditious distribution, dispatch, transit handling, and delivery insofar as is practicable". That isn't encouraging over third-class. Also, at TMQ's low subscription base, the overhead of mailing preparation would be exor-

bitant. So folks, if you can stand the additional \$5, please go First-class.

The impact of 80 Micro dropping coverage

There's not too much more I can add to the letters and messages printed in Letters to the Editor. Yes, the passing of 80 Micro may be as traumatic as passing from young adulthood to middle age. But all things must pass. Since we are but a flyspeck in the grand scheme of things, we all must learn to live with it. MISOSYS is still here. We expect to be shifting our advertising dollars originally intended for 80 Micro into some direct marketing; our mailings should keep you in touch.

On the other hand, do not be confused by our continued presence in the TRS-80 market. Our primary energies are devoted to increasing our presence in the MS-DOS marketplace. I expect that 1988 will be a much bigger year for us in the sale of MS-DOS products. We have also begun to take on the resale of other products of excellence. Where we find our customers are looking to us with questions, we want to be there to provide the answers - either with our own product, or one we feel merits our name.

LDOS Manual TABSETS

Some folks have asked about tab sets for their LDOS manuals. We have lots of them in boxes over at the warehouse. Up until now, we had no useful way to distribute them. That's because they don't fit into a standard 9 x 12 envelope. Well I now have a solution. They fit into a 9-1/2 x 12-1/2 envelope. I just bought 1000 of these envelopes and I want to give the tab sets away for free. Here's how to get your set. Send us a mailing label with your address on it along with United States postage for 3 ounces (or equivalent in international reply coupons) and a set is yours.

LDOS 5.1.4 Quick Reference Cards

We have thousands of them. They're not too useful any more since 5.3 was released, but if you want a FREE 5.1.4 QRC, send me a #10 envelope (or one at least 4 x 9) and 3 ounces of United States postage (or equivalent in international reply coupons). Be sure to put your address on the envelope for your free QRC.

Out of print TMQ's now available

We are providing back issues of THE MISOSYS QUARTERLY via copier reprint. The price is \$10 plus \$2.50 S&H in the U.S. and CANADA. For foreign zone D, the S&H rate is \$3.75; zone E is \$5.00. That price will be in effect also for regular back issues. Back issues will be available ad infinitum through one means or another. That includes issue I.i, I.ii, and I.iii as well, which are currently out of print. Here's a synopsis of past issues:

I.1 A cc for MC; Add SETEOF to EnhComp; Change baud rates with SETBAUD; unlock protected BASIC programs; WinCalc: a PRO-WAM application.

I.ii BANKER - RAM bank control; BINHEX revisited;; Expose on LDOS 5.3; Fractals in FORTRAN; Model 4P BOOT ROM exposed; Split REL libraries with SPLITLIB.

I.iii 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.

I.iv UNDATE reverses DATECONV; 80x86 assembly language; Converting LDOS filters to LS-DOS; Previewing output from SCRIPSIT; Learn MORSE with CODE/BAS.

II.1 David Hall on the 64180 CPU; Gary Phillips on XLR8 and 4P; Doug Tittle on sorting PRO-WAM data; WORD with DW II

Contents of DISK NOTES 2.2

Each issue of THE MISOSYS QUARTERLY contains program listings, patch listings, and other references to files we have placed on DISK NOTES. Note that to avoid confusion, we have revised the numbering scheme of DISK NOTES to correlate with the corresponding issue of TMQ. Thus, the "2.2" implies TMQ Vol II, Issue ii.

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.

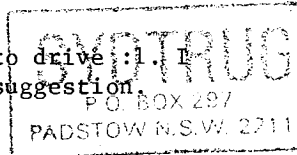
There is a cost involved. DISK NOTES is priced at \$10 PLUS S&H. The S&H charges are \$2 for US, Canada, and Mexico, \$3 elsewhere. If you purchase the corresponding DISK NOTES with the coupon which accompanies this TMQ issue, you can save \$2.50; the cost then being only \$7.50 + S&H. Here's what's on Disk NOTES 2.2:

BUFDATA - BASIC <-> EXMEM by Ketola
 FIX22* - fixes discussed in TMQ II.ii
 HIRES* - MC Hires by Clayton
 INPUT - INPUT.BAS by Schroeder
 RUBINS - Model 4/III RUBINS test
 RUBINS86 - MS-DOS RUBINS test
 SPEED* - Speed programs in C
 UNARC? - Utility splits "*" files

Note that we are including our UNARC utility which is provided with MC. This splits apart the files suffixed with an asterisk by a command such as:

UNARC6 <FIX22/TXT

The separated files are written to drive :1. I thank Claude E. Hunter for this suggestion.

Family Update, by Roy Soltoff

I think Brenda did a great job last issue on apprising our interested readers of the latest happenings here at the Soltoffs. But Brenda wasn't quite up to the task for this issue. Seems like a fifth Soltoff is in the mill. We're expecting the new arrival mid-June 1988. So the last few months have been quite interesting. Brenda's "morning sickness" has been like the 24-hour flu - for days on time throughout the past couple of months. I apologize for some of the delays in getting your orders processed (with the right product); I have been getting involved in that part of the job; Brenda's had other things on her mind.

Now one weekend, the flu hit; Stacey couldn't keep much down. She started on a Friday night. Since Stefanie seemed okay, I took her along while I went shopping for Brenda's birthday present. Bad mistake! About 45 minutes away from the house, she put breakfast into the back seat of the Jeep. Here I was stuck in traffic on the Beltway waiting to exit onto 95S and I had to pull off the road to clean everything up. Lucky for that roll of paper towels we keep in the car. The passers by must have puzzled over my actions as I had to remove Stef's clothes and wrap my coat around her. I had to drive back home with all of the

windows open and my head halfway out. I know many of you have experienced this sometime in your life. Seems like I was doing nothing but washing bed linen and clothes the next few days. Ah the joys of parenthood...

But there are some pretty great times too, if you just stop along the way to notice them. I believe that even though our individual life is minuscule compared to the span of time, our existence is meaningful since it's part of the continuum of life. Mankind, as a lot, improves over time because we are able to pass our wisdom and knowledge on to those who follow us. As one taking an active part in transferring some of what my parents taught me over to my children, I feel a great personal pride over some of the extremely small things my kids are doing; things I taught them. As a for instance, some time ago I taught Stacey to hold on to her shirt sleeve with her hand as she pushed her arm into her coat. In doing that, her sleeve would not ride up her arm. I know, it's such a little thing, but when I see her doing it, I can see my parents teaching that to me; it brings me a little closer to them.

Now Stefanie had her Birthday back in October; she turned 3. We had a whole house full of people - from little to big. I believe it was Ann Landers who said that children should be permitted to have as many friends to their party as their age; that would have predicted a party of four. But then there's our friends, and neighbor friends, and siblings, and... I think for Stacey's fifth birthday, we'll limit it to kids!

I'm going to keep this short; too many other things in this issue which need space. Let me close by wishing all of our friends, and other readers, a safe and joyous holiday season. You may not necessarily get all of the things you want, but I hope you get all of the things you need. This is a time to think of peace throughout the world. Last night I caught "Billy Joel From Leningrad, U.S.S.R." on HBO. Isn't it interesting that aside from the person on stage who translated Joel's between-song dialogue into Russian, I was unable to differentiate that audience from one here in the States. Perhaps one day we'll all be earth people...



Release 2

Window controller and Applications Manager

PRO-WAM supplied applications can turn your 128K Model 4, 4P TRS-80 into a sophisticated business or personal machine rivaling the best of them. That's because *PRO-WAM* comes with many useful and powerful menu-driven time savers and work organizers. *PRO-WAM* includes eleven applications, a complete HELP facility, a data file sort program, a 99-page user manual, and is easily installed. While you operate other programs, you can request its services with a single keystroke. *PRO-WAM* saves you typing with its EXPORT and IMPORT functions which allow you to move data across windows between programs. Requires 128K TRSDOS 6.2

PRO-WAM APPLICATION MODULES

- ADDRESS: Mailing Labels and Rolodex™Cards
- BRINGUP: Tickler File and Appointments
- CALENDAR: Any Month From 1582 to 4902
- CALCULATOR: Four Function Floating Point
- RPN CALC: Seven Function in Bin, Oct, Dec, Hex
- CARD: 480 Character 3x5 Cards for Notes and Data
- CHARSET: Display All Video Characters
- DIALER: Telephone Number List and Auto Dialer
- DOSAVE: Save Entire Screen to Disk
- TERM: A Really Small Terminal Program
- TYPER: Line-Buffered Typing to Your Printer

PRO-WAM..... \$74.95 + \$5 S&H

plus 7 new applications in release 2!

Here's Mr. ED

More powerful applications for your *PRO-WAM*. This pac includes an editor for your every need. You get:

- DED: Edit disk sectors on any drive
- FED: Edit file records in any disk file
- MED: Edit any page of memory in any bank
- TED: A pop-up full screen text editor
- VED: Edit the video screen; gives cut and paste
- CARDFORM: Populate CARD with a form
- DOLOAD: Brings DOSAVED files back to the video
- DEGENBU: Shrinks BRINGUP/DAT file

Mr. ED \$59.95 + \$3 S&H

What? You don't have PRO-WAM yet?

800-647-6797



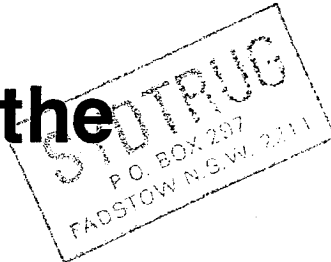
MISOSYS, Inc.

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

Letters

to the



Editor

The end of 80 Microcomputing?

Fm MISOSYS, Inc: Sorry for the length of this dialog; however the significance of the event precludes me from cutting anyone off.

Fm John Harrell To MISOSYS, Inc: Well, Roy, it does not look as if your letter did much good in the October issue. The new attack is right in the middle of the page and is called "Gender Agenda" -- an outright attack on Tandy's apparent sexism.

Things like "if you are a feminist and own a Tandy computer, you might want to evaluate whether you're supporting the right company" don't sit very well with me and certainly don't belong in a magazine that purports to be a Tandy representative.

Fm MISOSYS, Inc: I sometimes wonder what Maloney's background is when I see his ravings. I also note the minute correction they placed on page 111. They certainly could have made the correction more clear!

Fm --jjkd-- To John Harrell: I found their comments to be particularly misleading, given that there was no (or I noticed no) comparative statistics on other corporations. Maybe the folks at 80 micro were recently given a copy of "How to Lie with Statistics".

Fm Stephen Sutherland To All: Let me be the first to announce the end of my subscription. Since the announcement in the new Nov. issue that as of January, 1988, 80 Micro will go MS-DOS! No more Model I/III/4! They claim MS-DOS readership is nearing 60%. They claim Model 4

readership is below 40%. They claim Model I/III readership is below 20%. It still sounds as if half of their (former) readership was non MS-DOS. Let us all speak with our pocket books. I also canceled two gift subscriptions at the same time. Maybe the loss of 1/2 of their readership will tell them something.

As bad as some of you think they were, there was nothing else. They sure do make some dumb decisions!

Fm --jjkd-- To Stephen Sutherland: Ah, but there is something else... Ta DA! The MISOSYS Quarterly. Specifically for LDOS and TRSDOS 6 Users, and the de facto publication for all Z80-based TRS-80 Users. Just \$25 per year for U.S. Bulk Mail subscribers. Call (703) 450-4181 to order.

Gee, it's too bad, it's probably too late for Roy to get an ad into the December issue of 80 micro pumping subscriptions to TMQ (The MISOSYS Quarterly).

My guess - you won't have 80 micro to kick around anymore, say, past Jan of '89 at the very latest. They might be able to coast that long on existing subscriptions and newstand sales.

Fm Shane Dawalt To --jjkd--: Oh ... what timing for the plug of TMQ!!! My compliments on seizing the moment. BTW, anyone getting TMQ, good idea to pay the extra \$5 for First Class mail. I was on Bulk rate and the wait of 2 to 3 weeks is quite tortuous. Especially when everyone on CIS was oging and ahing and I couldn't see what about.

Fm Bradford A Morse To All: Re: 80-Micro and us Non-MESS-DOS users -----> Bye Bye! I won't be renewing!

Fm Kevin R. Parris To MISOSYS, Inc: Well, Roy-- at least after the New Year you will not need to worry about the incompetents publishing any more incorrect descriptions of your products!! By they way, does it make me "famous" or something to be quoted more than once in each of two consecutive issues of 'TMQ'? Maybe the death of 80Micro is just what you need to really liven up the QUARTERLY; not that there is anything wrong with it (GAD, please do not think I believe that! Had I not been fortunate enough to subscribe in time to get the very first issue, I would be begging for a reprint), but with more subscribers maybe you could afford to put even more wonderful stuff in each issue!

Fm MISOSYS, Inc: Actually, the number of subscribers doesn't necessarily equate linearly to content. Most magazines actually make their money via advertising. There are also tons of magazines that are 100% funded by advertising and are offered free to controlled circulations. In those cases, the editorial vs advertising ratio is probably 10-90 to 25-75 and the editorial content's impartiality is sometimes suspect. TMQ is 100% editorial. and the issue is totally funded by subscribers. A larger subscription base would give me a better feeling because I am concerned that all of the time I spend on it only goes to help 700+ subscribers. My time may be better spent elsewhere. Now as thin as 80Micro is now, I still question the viability or need of a publication catering to the Tandy MS-DOS market. What's different about that? Or is it that CW doesn't know how to exit gracefully?

Fm H. Brothers To --jjkd--: File under "Almost forgotten predictions about 80 Micro": In the summer of 1984, Bill offered to bet me \$100 that 80 Micro wouldn't survive to the end of that year. I've never been able to explain why I said no.

Fm Gary Phillips To all: I never get my copy until about a week after everyone else here has been talking about it. However, this bodes very ill for those of us who are happily using model 4 or model 3 hardware. There will be no forum left for the remaining advertisers who are supporting our market: Aerocomp, Montezuma, H.I.Tech, MicroLabs, etc. With no way to reach even a limited customer base, I predict most of these companies will quickly drop out of the market. Perhaps Roy can consider accepting advertisements into TMQ, but with his limited subscription base, that won't be enough. Everyone who still uses a Radio Shack Z80 machine should be subscribing to Misosys Quarterly NOW. (On pain of losing what support you have left!)

We had better think twice about how to get adequate communication with those companies whose products we still need to have available, as well.

Fm MISOSYS, Inc: One of the primary reasons for 80 Micro dropping support of 8-bit Tandy computers was not so much the lack of readers, but the lack of advertisers. Compare recent 80's with those of yesteryear. You'll note that the hundreds of advertisers supporting I/III/4 computers have long left the scene either because the I/III/4 owners weren't sufficiently buying or the labors of the compa-

nies were better utilized elsewhere. Magazines need advertisers as well as readers. But advertisers need buyers! When we were spending \$1800 a month for a half page ad for LDOS 5.3, you go calculate how many I had to sell just to pay for that ad each month. Then go figure if its worth it.

Fm --jjkd-- To Shane Dawalt: I tried to work in a plug for The Source as long as I was at it, but couldn't come up with a sufficiently elegant bridge. BTW, I've had my First Class (in every way) copy of the latest TMQ issue for several days now <twist>, <twist>, <twist>...

Fm --jjkd-- To Stephen Sutherland: Thanks, Steve, Roy does indeed deserve praise for a fine production. My only contribution is part of what filters in from CIS as the source, Roy does all the real work...

Fm --jjkd-- To H. Brothers: Ya, that does sound a bit early for the demise. My prediction is based on the belief that the folks at 80micro aren't prepared to make the investments needed to compete with the likes of PC Magazine et al.

Fm H. Brothers To MISOSYS, Inc: Obviously, I must be circumspect in my public statements on this topic, but there are a couple of things which I think many people here are missing. First, the differences between Tandy's MS-DOS machines and the other clones are much less important than the perceptions of their users. If the users feel that a TIK is "different" or if they feel left out by the other MS-DOS publications, they'll want a magazine aimed at them. Second, most of the editorial content of recent issues was aimed at MS-DOS users so the change won't be huge. And third, CW obviously did a lot of careful marketing studies before deciding on the change.

Fm MISOSYS, Inc: Hardin, When you get down deep enough, I think you will realize that Tandy's customers today are no different than any others. The MS-DOS marketplace is so generic that there just is no need to devote space specifically to Tandy computers. CW is just digging another hole to crawl out of. Rather than fragmenting their resources, CW ought to be spending their time and resources in one MS-DOS magazine, not 80 Micro and PCR. I thought that the move to PCR was a great idea. It allowed CW to get out of the "Tandy" image and break loose. If you look at both current issues of 80M and PCR, together they still have a long way to go to build into the

quality of a PCM or PCW publication. Keeping two just makes no sense. And I also think that Tandy is not foolish enough to think that they are able to sell their MS-DOS computers as being "different" from the rest. Price competitive? Sure! Available service? Sure! Available from your local store even out in the boonies? Sure! From a company who has in business for years and will still be around when you need them? Sure! Able to sell you everything? Sure! But a "special" computer different from the rest? No! And the folks know that. They buy Tandy computers because of their assessment of the "sure" points I raised. Tandy knows that. I know that. Their customers know that.

Besides, a company which prohibits magazines from their stores (Tandy) does not deserve a publication devoted to them!

Fm Gary Phillips To Roy Soltoff: Great issue! You will receive my renewal (upgraded to 1st class, please) in next week's mail. What do the renewal percentages look like? I was depressed by the negative comments you printed.

Serious question: I like your offer of trade-in credit for Alcor C on Pro-MC. What do I have to send, the original disks and manual? (The latter weighs a ton!) I'll also include my order for a reprint of TMQ I.1 and probably something else to use the \$5 coupon. Your offer to take other equivalent products in trade for Misosys (read good quality) software is real generous...hope it gets you some sales of value and you aren't just taking a loss on it. Rhetorical question: what other product would be suitable as a trade-in for PRO-WAM??

Fm MISOSYS, Inc: Renewals are going great, and new subscribers are picking up on it. If we continue to move like this, I'll see more than 1000 subscribers in early 1988.

I really don't think there ever was another product like PRO-WAM ever developed or marketed. As far as the trade-in policy, you need to trade in the whole package. You don't need to submit any binders, if any. Don't forget that you can send the package separately via third class mail or via UPS, which may cost less than 1st class.

Fm --jjkd-- To H. Brothers: I think that the use of "CW" and "careful" in the same sentence classifies as an oxymoron. That is, unless the sentence is something like: "Gosh, you had better be careful when reading a CW publica-

tion like 80micro, except for those columns by Hardin Brothers, the rest is often of questionable accuracy." (dare I add a <grin> or two?)

Fm Guy Mercier To --jjkd--: After these rumors about 80-Micro dropping support of TRS80 models I/III/4/4P computers, and all the great things I read here about The Misosys Quarterly, I just couldn't wait any longer and, today, I subscribed to TMQ. Now, I hope this %\$##! postal strike here in Canada won't hold my first issue too long. Thanks for your (always) wise advice. Looking forward to read all this good stuff in the next TMQ issue.

Fm MARK MUELLER To Guy Mercier: They ain't rumors anymore. The Nov. 87 issue announced that the little *%&\$'## 's in Peterborough just flipped the TRS80's off as of January. Nice. Real nice. Ten minutes after I read it I called Misosys and signed up. Now, I only hope that Roy can hold hold out. <hope, hope>

Fm Daniel L. Srebnick To MISOSYS, Inc: Well, 80 MICRO shall be no more. I am looking forward to receiving my fall copy of TMQ. Thus, the reason for my inquiry. A couple of folks I know have already received theirs. There was a goof up once before at the start of my sub, so I was wondering if you could either advise me that all is OK (maybe they have not all gone out) or check and see if there is a problem. I shall be more dependent on TMQ than ever for new info. Perhaps you should contact some of 80's advertisers re: TMQ advertising. As for me, I wonder if CW will give me a refund on the remainder of my 80 Mike sub.

Fm MISOSYS, Inc: You are on the list of mailees. I have enough to do without trying to turn TMQ into a magazine. After all, it is still only a support vehicle for my product line. But we are considering the acceptance of outside advertising.

Fm JEFFREY KLINE To MISOSYS, Inc: I just received the latest issue of TMQ and as usual, am very pleased with it. I seem to have just picked up on something of 80 Micro will no longer be publishing anything for the Model 3 & 4 computers after the end of December. Gone all MSDOS (UGH!). If so, I had an article that I was writing for it but will now submit it to you for TMQ. (I thought of you first anyway!). I too had seen many an article and review saddled with blatant errors throughout. Personally, I think it is this kind of editing that has frustrated many a Model 3 & 4 user and

helped convince them over to the MSDOS (MessDOS).

I have a Turbo MSDOS clone and bench tested several modes between the two. I like to compare practical operational features over Sieve, and such. Notice, I still like my model 4's. Regarding the XL8ER board, I think that I will wait for a while and better put my money towards MC! I looked at the libraries with ALCOR's C and you cannot break them apart. They retain the source code for them. So much for that! I will be taking advantage of the "Trade In" of yours with this one! Does MC have libraries for the graphics board in it? I might be picking up on this one from TMQ's articles. I have a Micro-Labs HR Board and hope that if it is supported, I can access the board just as with a RS board. I am big-time into graphics right now. Speaking of trade-in's, besides my ALCOR C, does the trade-in program have to be an equivalently price program or can I use my old EDTASM or something like that? I hate to give up some of the other stuff as it is still pretty good stuff! Believe me, I have spent a "wad" on these computers and must now (unfortunately) resort to a budget! Just now had my Gemini 10x give me fits when I put it into bit image mode. Just ordered a new NX-10 to replace it with. Yet another \$170! Next month however for MC!

Fm MISOSYS, Inc: If you think it is a fair trade, then so be it. At this point, we're pretty agreeable, and easy. No, MC doesn't include any functions to interface to the various hires graphics boards; however, I believe there still may be some on the forum that worked with LC (should be the same interface). Also, Harry Clayton has finally gotten around to submitting a set of functions to interface MC to GRPLIB (the graphics library which accompanies the RS hires board). That is in this issue of TMQ.

Fm Guy Mercier To MARK MUELLER: Now everything is clear! Actually, I still ain't got my 80-micro Nov. 87 issue (because of the Canadian postal strike I guess). That's why I didn't know about these big bad news. I used to read every issue cover to cover and I really couldn't explain how I could have missed this one! This is what we could call a real "punch below the belt" to all Model I/III/4/4P owners. Do you think it could be a good idea to let them (CW) know our disappointment? Maybe if their head office was buried under lots of disapprobation letters they would reconsider their decision?

Fm Daniel L. Srebnick To MISOSYS, Inc: Well, Roy, you may only consider it a support vehicle, but it does seem to be the only support left for TRS80 users. I certainly hope that you make enough money from Model 3 and 4 sales to justify keeping after the market.

Fm MISOSYS, Inc: Be serious. We can no longer justify spending our time actively on any TRS-80 product. That does not mean we won't continue to support what we already have. But don't expect any NEW development or changes to any of our existing products. We will still sell them. But even those efforts have to take a back seat because there are too few buyers left. Our energies have to be devoted to selling into and supporting a growing and buying customer base.

Fm MARK MUELLER To Pete Granzeau: Well, since the same company that owns them also publishes "Infoworld", I doubt that they care two squats about anything that isn't IBM compatible. The only thing worth reading in the last few issues had been Hardin's column and the ads. Really has gone downhill and its obvious they all have lost interest. But that's not going to stop me from writing a letter. My sub runs through July 88.

Fm ROBERT G STRICKLAND To Ray Pelzer: Well, there aren't any more Fords (human type), yet they still use that name, as is the case for most cars. Also, I don't see why they need to draw such a strict line. Many Tandy MSDOS types own TRSDOS equipment, so continuing to take TRSDOS ads and running an article now and then shouldn't be such a pain, or so it seems to me. After all, not everyone owns a hard drive, but they still take ads for them. Who knows!

Fm MARK MUELLER To ROBERT G STRICKLAND: I fully agree that a magazine is a profit-oriented enterprise. I also agree that it most likely isn't profitable to continue publishing a TRSDOS-only magazine. My beef is that they have simply walked away, cold turkey with not even a fond farewell. The editor's column came across to me as rather cynical and had a "so what" attitude that angered me more than the impending change. What I want to know is why not even a column on the Model 4, or a Q/A page, or continuing The Next Step, which, by the way, has to be the best-research and most informative technique and information column ever for any computer system. I just wish the 80-micro publishers had "thrown us a scrap or two" instead of just telling us all to get lost.

Fm Marc Nowell To Stephen Sutherland: I'm with you! Am canceling both subscriptions this week.

Fm Marc Nowell To ROBERT G STRICKLAND: It is true that the purpose of 80 MICRO is to make money for it's publishers, but it is MY right to withdraw a subscription to a magazine that drops all representation of the series of computers that it was created to support. That is capitalism in its pure form -- voting with your dollars. I don't need another MS-DOS magazine, because PC MAGAZINE, PC TECH JOURNAL, and DR. DOBBS JOURNAL all do such a great job for me.

Fm Adam Rubin To Gary Phillips: (I still haven't seen that issue yet!) That's a good point; I hadn't thought about 80-Micro's "demise" from the advertisers' perspective. Advertisements in TMQ could give a market to the advertisers, some revenue for the Quarterly, and -- well, some advertisements to us customers. Classified ads from individuals might be useful, too. Doesn't "Computer Shopper" have a few pages for the I/III/4 and CoCo?

Fm Daniel L. Srebnick To MISOSYS, Inc: I certainly can understand that. In my work, I customize applications that run under MS-DOS. But, it doesn't feel that same as a model 4 under LS-DOS. The PC compatible world may be where the money is, but is not my hardware of choice.

Fm MISOSYS, Inc: But MS-DOS is one of the choices left for those wishing to stay in business.

Fm ROBERT G STRICKLAND To MARK MUELLER: If tone of departure and lack of a "side altar" for TRSDOS is the discussion point, then I agree. I also think that they could carry a TRSDOS section of some sort of format without losing money or credibility. BTW, I think we tend to see our machines and interest in the warm glow of the tube (especially if its amber), while the business world, i.e. publishers, are much less so inclined. Might be related to the trade deficit.

Fm ROBERT G STRICKLAND To Marc Nowell: Couldn't agree more. If the shoe no longer fits, no point in keeping it in the closet. My observations were prompted more by what seemed to me to be a sense of editor-as-traitor. As I said earlier in the thread, 80MICRO ought to be able to throw us a bone or two without compromising what they see as their destiny. But

as also noted above, yankee capitalism can be rough, and our attachments to TRSDOS probably don't have a line item on the balance sheet.

Fm William chao To Stephen Sutherland: I will be writing to THEM shortly and I probably won't have a lot of nice things to say. One thing I would mention is they will need a name change comes Jan'88 since 80-MICRO will have NOTHING to do with the old TRS-80's. Also I will cancel my subscription and save it for another of TMQ, and I have sent a letter to The Alternate Source for info on Northern Bytes.

Fm William chao To H. Brothers: Hardin, maybe you can put in a quarterly article on assembly language. I have enjoyed your articles in 80-MICRO very much and I don't know what I'm going to do without it...SNIF

Fm William chao To MISOSYS, Inc: I just got the renewal notice for TMQ and by the time you read this, it should be in the mail. One comment I have to make about your reply envelope: it's too small for the check and the bill without folding it. To me, it's sort of annoying but you are excused if the larger size envelope is costing a lot of extra money.

I just got 11/87 issue of 80-MICRO and I'm sure you saw the editorial also. I'm wondering if you can get regular contributors like H. Brothers for some assembly language column or general feedback column.

Keep up the good work and you be assured to know that the refund I'm getting from 80-MICRO(I just renewed) will go to you for TMQ.

Fm MISOSYS, Inc: Envelope sizing is a compromise. The smaller one fits into the larger one. The reply can be easily folded. The alternative would have been to go to a much larger set of envelopes. Is that really necessary?

On the issue of regular contributors, they generally don't work for nothing, like I do with TMQ. I expect TMQ to prod some orders to come in - that's where I expect it to make some money. As a standalone "magazine", the subscription fees don't really cover the expense of publication. But even irregular hackers have something noteworthy to say. Take a look at the kinds of things discussed on this board. There are dozens of folks here with some intelligent and useful contributions. What we have lost out on in our society is people who have been taught how to WRITE. I

say, get them word processors out and practice. There are many useful small articles on construction which could be written up. Programming hints. I even think there is room for reviews of software/hardware. I'd like to see some input on uses for data base managers. What are you doing with PRO-WAM? There's lots that can be said from the masses. Any writer with a continuing series still needs to put bread on the table. But an occasional article from someone who has something to say can still work out. We'll even try to come up with some nominal stipend for a contribution.

Fm H. Brothers To William chao: Thanks for the kind words, William. As Eric mentioned in his column, they are considering a purely Model 4 newsletter/magazine/whatever. If that does come to be, you may see my byline there as well as attached to the new Next Step

Fm Gary Phillips To Adam Rubin: Last time I looked, Computer Shopper no longer had much of anything in the way of TRS-80 ads, even in the classified section. Virtually no vendor advertising, and just a dribble of CoCo material. The demise of Z80 publications was probably just as inevitable as today's market crash, but (in my humble opinion) the former is much more painful to deal with.

Fm Thomas Collins, Jr. To ROBERT G STRICKLAND: How much money are they going to make by covering a subject (MSDOS) that is covered by probably a dozen other mags (rags?). I give it another 12 months (and I'm be extremely generous) before it folds completely.

As to the others reading this thread, I have called in to cancel the remainder of my subscription, 18 months, and sent letters to the editor, sales manager, that big mouth Eric Bologna and a few others. And, I'd suggest they do the same.

Very Disgruntled, and glad I choose a Commodore C128D to replace my Model 1,3,4,100. (Which can be had very cheaply.)

Fm Nate Salsbury To All: In the current issue of '80', Eric Maloney invited letters regarding their termination of coverage for TRSDOS machines. I accepted. "Please cancel my subscription immediately and send me a refund for the remaining issues. I am sorry to do this because I have been a continuous subscriber since March '81. However, now that coverage of the Model III/4 is to be terminated, I have no reason to continue."

Fm Frank A. Boneno Jr. To All: I am upset to hear that 80Micro is stopping their support of our machines. I have been using a TRS-80 Model 4 for several years now and bought 80Micro for that reason. If they drop their support of the TRSDOS machines, they might as well change the name of the magazine to TANDYJUNK. The Model 4 is a great piece of hardware and I wouldn't change unless my machine totally died. It suits my purposes fine and is still one of the faster machines for home computing.

Fm ROBERT G STRICKLAND To Thomas Collins, Jr: Interesting predictions. We will see if they last or not. Should be an interesting scene at any rate. Take care.

Fm Thomas Collins, Jr. To Nate Salsbury: I did the same as you!

Fm Pete Granzau To Nate Salsbury: I used several paragraphs to say the same thing.

Fm Shane Dawalt To Nate Salsbury: I think I'll just let my subscription run out (it's up in March 88). I may find something interesting in the MS-DOS stuff.

Fm Adam Rubin To Gary Phillips: You're right; Computer Shopper's TRS-80 coverage has shrunk to almost nothing lately. (The current issue does review Enh-Comp, though.)

On the other hand, I'm not sure whether that's harder to deal with than yesterday's market crash. I was planning to pick up some extra money by selling my Tandy stock. Oh well...

Fm Michael Kushner To MISOSYS, Inc: I can't think of a better forum for Model III/4 people than this one (LDOS on CIS). I also recommend, to whoever listens, that THE MISOSYS QUARTERLY is their best source for information about our machines, at this time. I'm sure you will find that your subscription base is bound to increase, though it continues to astonish me how few people have heard of it, or LDOS 5.3 or LS-DOS 6.3!!

Fm MISOSYS, Inc: Don't forget that there are lots of folks with 4s who have never been in the mainstream of third party offerings. Those folks use their machines to run Tandy applications. When they run into problems, they get frustrated because Tandy can no longer help them. They also just don't know where to go. So they opt for an MS-DOS machine.

Fm Daniel L. Srebnick To MISOSYS, Inc: However, as a developer I am sure that you real-

ize a personal need for a better dos. If that was the case, would you ever have bothered to develop LDOS? TRSDOS 2.3 managed files on a disk, too.

Fm MISOSYS, Inc: As a developer under MS-DOS, I find no need to utilize a better DOS. True, there are some things I would prefer to be different. But it is not unworkable for program development. One of the things a business person must face is reality. And the reality of the situation is that MS-DOS is it. Once that is accepted, one finds ways of working with it rather than against it. As an aside, LDOS was developed for Lobo International (then Lobo Drives, International) to operate their IX-80 interface for the Model I CPU. It wasn't done just for the fun of it.

Fm MICK OCCHIUTO To MARK MUELLER: I agree. I have subscribed to 80-Micro and only missed a few of the first issues.

Fm ROBERT G STRICKLAND To MICK OCCHIUTO: Me thinks that there's only room for one TRS mag. My vote goes to the MISOSYS QUARTERLY. There seems to be several efforts to keep some sort of TRSDOS info flowing. If each of these separate efforts tries to fly, probably none will. Given that we are behind the barricades at this point, cooperation rather than competition seems to be the wise move. Roy has been with the system since the beginning and is personally qualified to be technical overseer. He is also the major vendor of software. I don't think that this constitutes a subsidy to MISOSYS; I do think it is the approach that has the best chance of succeeding. I'll be interested to see what happens. I hope that the TRSDOS users don't end up cutting their own throat.

Fm Charles A. Ainsworth: So Eric Maloney has finally come out openly with what TRS-80 users like myself could see and lived with for a couple of years, the practical abandonment of our line. It's no pain since most of what has been in his magazine for a few years has been junky, unreliable and I would say of little use to anyone, with a reduced number of exceptions such as Hardin's articles. They haven't even been capable of printing programs accurately, and sometimes one discovered the errors from corrections published about two months later after going crazy puzzling out why a program wouldn't work.

I join the protesters who have been disgusted at some of the inaccurate stuff that has been a regular feature of the magazine; and the re-

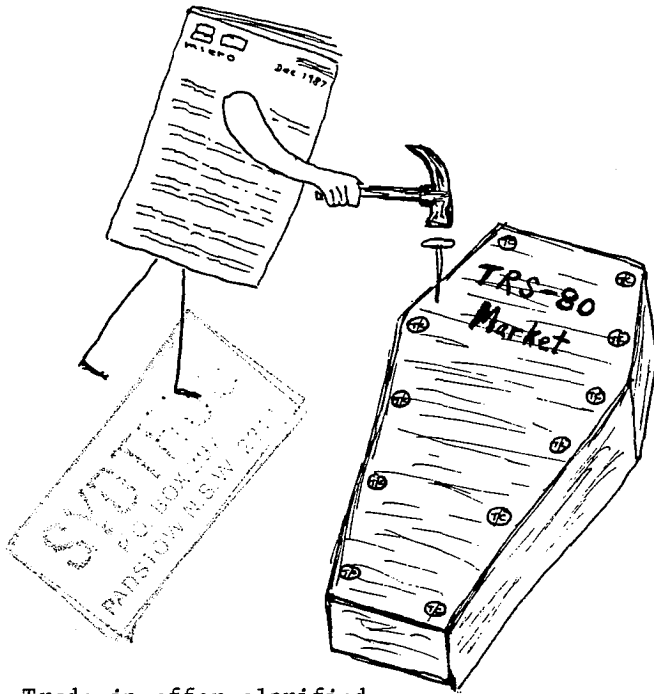
view of LDOS and the ensuing argument is nauseating. How any magazine can publish such trash is way beyond me! Doesn't anyone edit their articles, reviews and correspondence columns?

So now they will have two magazines devoted to MS-DOS machines, among a formidably competitive array of other much better publications. If they sink into oblivion out there in that wilderness, as they might well do unless they learn to produce good and accurate articles, I certainly shan't waste any tears about it. Hopefully, maybe they'll save their remaining readers (I won't be one of them!) the pain of Maloney's or someone else's mug shot every month, which may be great as the fellow's monthly ego trip but of no other use to anyone.

As a matter of fact, the only reason I didn't cancel my subscription to 80 Micro years ago was my interest in the advertising, which was really the only material I found worth reading (again, with few honorable exceptions). I find that the ads served me a useful purpose and, due to 80 Micro's change in direction, presumably advertising for the TRS world will practically vanish, leaving me, in some cases, not knowing quite which way to go if and when I need to buy anything further. I join those who have suggested that TMQ carry advertising which could fulfill a very useful purpose to those of us who have been dumped by 80 Micro.

Fm Donald J. Tysver: Several of your past issues have mentioned the possibility of advertisements from sources other than MISOSYS. I believe that this would be an important service to your readers, particularly since 80 Micro has decided to stop covering the TRS-80. Also, if there are any other publications still out there that support the TRS-80, it would be nice if they could be mentioned in THE MISOSYS QUARTERLY. The more information your readers have about the TRS-80, the better. And it might delay the eventual demise of the TRS-80 computer market for several months (years? -ed).

While I agree with the comments in the last TMQ about the diminishing quality of the TRS-80 coverage in 80 Micro, I am saddened to see that they are switching over to Tandy MS-DOS coverage only. I will miss Hardin Brothers' column and I will no longer know what companies are still supporting my TRS-80 Model 4 (unless TMQ picks up advertising). Hopefully, the following is not yet true.



Trade-in offer clarified

Fm Adam Rubin To MISOSYS, Inc: Speaking of the trade-in offer, a friend of mine had a question: Is it available to non-TMQ-subscribers?

Fm MISOSYS, Inc: It is available to all. That is our new policy. Same with the 30-day money back guarantee. Too bad there is no place left to advertise that. We may have Joe alter our bulletin here to insert that info.

MAC PAINT files in TMQ???

Fm Les Mikesell To JEFFREY KLINE: MACPAINT files have a 640 byte header followed by data with a simple compression technique. If the "index" byte is greater than 127, then repeat the next character (256-index) times. If the "index" byte is less than 127 then read the next (index+1) bytes. The decoded bytes should give you a 72x720 array. Depending on your hardware, you may see a transposition of each pair of 8-bit bytes, and you may need to adjust the bit ordering on output. On the Mac screen, the most significant bit appears on the left.

European access to Compuserve

Fm Ralf Folkerts 71420,1023: The DATEX Network here was the smallest problem for me. First I

had to get starter kit. That was almost impossible to get. Without Roy's help I wouldn't be here today ! When I got it the next problem was to get a charge card. As the 'normal' European I'm using EuroCheque cards or cash - and then I had to find out, where to get such a CC! I first looked a EuroCard, but found out that VISA was a lot cheaper (and has the smallest number of 'Vertragspartnern' here. Then came DATEX-P. That was the smallest problem (although DATEX-P isn't the cheapest). I have to pay for the DATEX - Use: DM 15,-- for the NUI (network user identification) per month; DM 0,23 per 12 minutes for the 'normal' phone connection to DATEX-P DM 0,04 per minute DATEX access - fees DM 0,06 per minute for the PAD (packing assembly disassembly facility) DM 0,05 for each connection DM 0,15 per minute (Zeitgebuehr) time fees + DM 0,013 for each segment transferred (1 segment are 64 Byte)! You can see, each typed letter has to be paid. A segment is transferred if it is filled (=64 Bye), if DATEX receives a <CR> or a <CTRL-Code>.

Tandy in the Federal Republic of Germany

Fm Ralf Folkerts: [Tandy's problems here are] not because there are no (potential) Computer customers, but because of Tandy's poor marketing strategy here. They had a few shops but made almost no advertising for their (really good, as we know) products. Tandy is only known by 'insiders' here. If you tell somebody that you own a Tandy Computer they say 'Ach sooo, Du sag mal, was ist das denn fuer einer' - almost nobody knows them. With such a grade of popularity it's no wonder that they had to close. But the 'last' Tandy Users here (most big companies don't use Tandy any more) ask ourself, where we will get the support now!!

Clock patches and Spelling errors

Fm Patrick H. Larkin to MISOSYS, Inc: If I read the information correctly in TMQ I.iv, you have included the patches for MSCRIPT. Are they compatible with POWERSCRIPT? Can they be split into partial patches to only use portions - e.g., to add default extensions, such as /TXT for ASCII files and /SCR for normal files?

In regard to some of my previous correspondence: (1) Requesting patch(es) to LDOS-530 to accommodate and utilize the ALPHA clock, there

are two PATCH files on the enclosed disk: A. LDOS53/FIX, and B. X63/FIX that I finally received from ALPHA Products in late May. I have also enclosed the JCL files originally provided by ALPHA, (the one for LS-DOS 6, I have modified to accept the "3" parameter to use X63/FIX for LS-DOS 6.3. I didn't bother to modify the LDOS13/JCL to accommodate LDOS 5.3, I just (1) patched SYSO, (2) ran LDOS3, and (3) SYSTEM (SYSGEN)ed. It now works like a champ. Also on the disk are several other files provided by ALPHA, including the original LDOS13/JCL and NEWCLK4/JCL and the several BASIC programs they included (in case you're interested).

As I have mentioned previously, I am, by training, an accountant (actually a CPA, but employed in the oil & gas industry), not a programmer, and frankly, I don't have the time to try to learn all the nuances needed to program in Machine Language - in fact, I sometimes have trouble writing JCLs for LDOS and LSDOS. In fact, if you look at my JCL file, modified to accept the "3" parameter (NEWCLK63/JCL), you can probably see that it really doesn't work properly for 6.0, 6.1, or 6.2 if one doesn't supply some parameter on the command line (possible to get an erroneous default). I am a firm believer in error-checking, (I use whatever checks I can in my use of SYMPHONY spreadsheets on the COMPAQ 286 I use at work), and I am a little surprised that the original JCL may not have been as efficient as it could have been in that regard, and I have perpetuated, nay, magnified, that lack of error-checking. This might be a call for a tutorial-type column on JCLs like the one published in the LDOS Quarterlys/Journals in the past -- to help those of us arriving on the LDOS5/LS-(TRS)DOS6 world after the earlier issues of the LDOS Quarterlys were sold out.

One last thing --- I note that in your README/TXT file, as well as frequently in TMQ, the word "pertinent" gets (mis-)spelled "pertinant" (it should have two "e"s and NO "a"s). Spelling errors tend to be a turn-off for me, and I'm sure, for others. Some people consider spelling errors as indicative of less than careful work in areas other than just grammar and usage.

Sorry for the long letter, but I wanted to communicate some of the questions and comments I'm sure others have been wanting to mention, but may not have gotten around to. Keep up the good work on LDOS - I'm sold on it! Also, thanks for taking the time to read this.

Fm MISOSYS, Inc: I have no idea whether our MSCRIPT patches are compatible with POWERSCRIPT; I've never used "ps" nor has anyone fed back that information to me. Although the MSCRIPT patches are broken out into three files, they may not necessarily be applied by particular function. Assuming they are commented, take a peek.

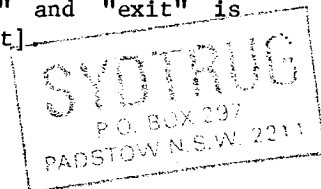
I'll probably note the Alpha clock fixes in a TMQ issue although I don't expect to waste printed space. Most likely, they will go on a DISK NOTES and be referenced in TMQ.

Sorry, but there really are more important issues today than re-printing stuff from long ago LDOS Quarterlies; I mean, we still sell Volume II and that goes back to January 1983. I have been thinking about consolidating some of the material that was printed as a series and offer a "booklet" external to TMQ. But there's no decision on that yet.

Lastly, someone such as myself, who is so overburdened with work, finds less importance in rooting out every last spelling error than getting the necessary content out to the folks. On the other hand, you have less to worry about; for now that I have switched over to MS-DOS, most text work is done using Microsoft WORD which comes with a spelling checker. Thus, automation comes to my rescue. Actually, some spelling errors get to me too; I can't stand to see "separate" spelled as "seperate". But then English is such a horrid language; how can we condemn misspellers when we have "mice" as the plural of "mouse" but use "houses" as the plural of "house"? Or the past tense of "emit" is "emitted" but the past tense of "edit" is "edited" and "exit" is "exited" [two t's versus one t].

Now for some TMQ praises

Fm Will I. Ramsey, Jr: Let me join in the chorus of praises for TMQ! As is the case with some other readers who have written to you in the past, much of the technical stuff in the publication is just over my head, but I'm trying to learn. Nonetheless, I eagerly await each issue and enjoy it immensely. I really do like the format and was especially impressed with volume II.1's use of large section headings to separate the different sections of the magazine. Since I have no MS-DOS machines and no interest in them at all, I can now skip over that section of the publication without



fear of missing something that might apply to me.

I was a bit distressed with the LSI Column this time around, though. First I had to get past the bilious invective of the author. Language such as "stupid", "idiots", "hair brained", and the like (as well as the general tone of the first two-thirds or so of the column) is very unprofessional and totally uncalled for. (Undoubtedly, this criticism will provoke more of the same from that guy!) Granted, software piracy is clearly a very serious problem, and obviously it has hit LSI where it hurts the most (the pocketbook), but is it really necessary to vent one's spleen publicly in this manner? I certainly do not recall anything of this nature coming from you, for which I am singularly thankful!

Anyway, once I got past all the nasty language, I learned that the present production level of LS-DOS is Level L, and it is stated that "[i]f you are not current check the patches in past issues of TMQ." Well, I did that, and the patches on page 46 of volume I.iv (which I assume is the most recent past issue) upgrade the DOS to Level K -- which is the level I presently have. Have I missed something? I realize that the world won't come to an end if I continue to operate at Level K for a while longer, but I'd hate to think that I could be at Level L but am not! Is the world passing me by or what?

I very much enjoyed the stuff on the HD64180 chip and the XLR8er board, although again some of it was a bit technical for my little brain. We purchased an XLR8er from H.I.Tech in August 1987. After some initial problems hooking it up due to a bad ribbon cable and after numerous panicked phone calls to Bob Jensen of H.I.Tech, I finally got the board up and running and have been very pleased with it so far. Apparently, one of the 150ns RAM chips which H.I.Tech sold me to go with the upgrade is bad, though, because in SCRIPSIT Pro the cursor sometimes tends to take on a life of its own. [For example, frequently, when using the left arrow key to move the cursor to the left, the cursor will start to go left and then just jump back to the extreme right. If it is not on the bottom line of text, it will also sometimes jump down. Also, the F2 key in the repeat mode sometimes somehow activates the F1 key. Sometimes if I leave the machine with an open text file, the cursor will start moving on its own, inserting lots of commas and insert-text indicators all over the file.]

I didn't mean to ramble on like that, particularly about a product I didn't purchase from you, but if it's any consolation at all, I would have purchased it from you if you sold it! I just wanted to join the many others in thanking you for your fine publication. I will support you in any way that I can as long as you continue to offer some support for LS-DOS and us Model 4 users. As you know, 80 Micro has joined Tandy and just about everyone else in totally abandoning TRS-80 users forever. We just canceled our subscription to it when that announcement was made in the November 1987 issue. I know that you can't be expected to service a "dead" market forever, and if you have to move on to MS-DOS exclusively to survive, I won't fault you for it. But you have the everlasting thanks of those of us who love our TRS-80s and refuse to let the market dictate what machines we will use.

Fm Rex A. Basham: Well, I just received my first copy of THE MISOSYS QUARTERLY and as has been the case with everything else I have purchased from you folks, it's superb! I got more info out of this one issue than a year's supply of 80 Micro. I'm going to get all the back issues but the money situation being what it is here, I have to opt for one at a time. My wife is threatening to remove important parts (of me!) if I spend any more money on my toy. The 4P is a hobby, I do IBM mainframes for a living, work for Northwestern Bell Telephone Company in the IBM Technical Support Section of their Information Systems Organization. I go into withdrawal symptoms if I don't get at least an hour a day on the Z80. I have some friends that swear they have to bring a crowbar to pry my fingers off the keyboard whenever they stop in for a visit.

The Family Update section was interesting reading, tell Brenda she did a great job. Jeri (my wife) and I have been enamored with each other for sixteen years; we have two children, Jason (15) and Angela (9); and last the pooch, Chingo (3 or so, spoiled rotten, and he thinks he's one of the kids!).

Thanks for an excellent publication in the Quarterly. With 80 Micro's decision to drop support, this and the LDOS SIG (CIS) are probably the last bastions of hope for us diehard Z80 people.

Fm R. C. Vohl: To me, TMQ has no dollar value. It is a necessary extension of the machine I enjoy. Yes, I have one of those clones on the other table. I use it when I have no other choice. MessDOS is nothing more than an en-

hanced version of CP/M. It has to be even more primitive than TRSDOS 2.0. My 4P is my Rolles Royce. Its operating systems (LS-DOS 6.3/LDOS 5.3) are far superior. Their designers teach the operating system in their documentation. They do not attempt to impress me with their computer science degrees. All I have to do is read the directions! Which is more than I can say for Heir Doktor Feldman of 80 Micro. You know, the magazine which split into another publication for the clone world but still contains MS-DOS columns, reviews, etc. Mine expires soon and I will miss it about as much as I miss Viet Nam.

Incidentally, my 4P has had an XLR8er since last March, and I have had no problem running any of my Model III or 4 software. I even run PROFILE III+HD out of the RAMDISK with no problem. In 4 mode, PRO-WAM, PRO-MC, PRO-HartFORTH, and utilities from your PRO-MACH2, PRO-ESP etc., run fine. In Model III mode I made a "boot" disk as the directions said and have had no problem. I am beginning to think that the only thing I have done different from some of the good folks who have called me about their problems with the XLR8er, is to Read the Directions.

Regards to Brenda and young'ens, keep the news about you and yours com'en. One other thing Roy, if you let some of those Alpha Hotels talk you into dropping the column about your family, I will personally fly out to VA and kick your Alpha!! Keep the news com'in.

Assistance needed, folks

I have a series of Basic Programs with Machine Language Modules that currently load at the top of memory, reserving about 8K to 10K for an inventory program. This interferes with the full utilization of many of the LDOS features. I have installed an additional 64K of RAM and have SET2RAM and memDISK, but am not sufficiently sophisticated in machine language to know how to relocate this inventory table and access it from the top 64K of new RAM, thus freeing up the usual high RAM to FFFF for LDOS Devices and features.

I would appreciate it if some skillful TMQ reader would undertake this project for me; naturally, I'm willing to pay for this service if the charge is reasonable.

Bob Arner, PO Box 4095, Rockford, IL 61110, (815) 963-4912, (815) 654-8326

Miscellaneous subjects of importance

Fm Lee C. Rice, Ph.D.: A number of my colleagues have written to 80 Micro to cancel their subscriptions in the light of their decision to ignore TRSDOS software in future issues. I have suggested to most of these that they consider subscriptions to The Misosys Quarterly, so I hope that your own subscription base will increase. I had recommended TMQ to some earlier, and also circulated copies of TMQ to them. Many are not programmers, and were reluctant to subscribe; but, now that you are the only game in town...

Nor are any of them "MSDOS haters." One of the things that most infuriates me about the 80 MICRO editorials is their continued suggestion that those who want continued TRSDOS coverage are political "reactionaries." I manage the Micro User Area in our department: we have several Zenith MSDOS systems, as well as nine Model 4s. Many of our faculty also have two systems, or use both types; and for them it is not a question of "either/or", but rather one of "both/and". We have found some things which the Model 4s do better than MSDOS (such as driving our laser printer under Allwrite), and others which MSDOS machines do better (PROLOG and memory intensive programs are one example).

A couple of suggestions, and a request or two.....

(1) There are vendors (both software and hardware) who wish to continue to support the Model 4 and TRSDOS/LDOS. You know who they are from the pages of 80 MICRO. Why not contact them for possible adverts in TMQ? It would not be merely a means of MISOSYS gaining some extra advert revenue (though that is certainly desirable!), but might be a service to readers who want to enhance their systems. Granted your circulation is smaller, but so are your rates (probably). And, above all, if you do get some extra adverts, remind readers who contact advertisers to let them know where they saw their adverts.

(2) Since TMQ may be the only game in town, you may want to broaden your article scope a bit if new readers come on board. That's not a suggestion: just something you may wish to ponder. TMQ is very attractive to programmers, but perhaps less attractive to non-programming end-users.

Sorry this letter is so long. I certainly don't expect an answer. Just tuck it away with

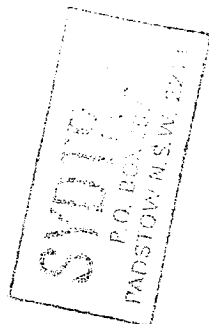
your "idea list," and with my assurances that I'm glad you're here.

You're most welcome to include any piece of this letter which might be of interest in your TMQ feedback section. I'm enclosing a disk version of it also (on an MSDOS disk, LeScript format for the file....).

Thanks for taking the time to read this letter. I'll be ordering MSDOS RATFOR shortly. It is performing flawlessly under TRSDOS, as no doubt it will perform equally well under MSDOS. Like (many!) 80 MICRO readers, I refuse to throw one machine away just because I have another...

Fm MISOSYS, Inc: I do appreciate the letter on disk, but apparently MS-DOS LeScript uses a file format different from TRSDOS LeScript - I couldn't accept the file without deleting all of the formatting control codes.

As noted previously, we are considering the acceptance (or search for) advertising. Also, since we internally are not in the magazine publishing business, I personally won't be spending time to broaden the scope of TMQ; however, we certainly encourage our readers to contribute. The more articles I receive from TMQ readers, the less I have to do (and the more time I can spend on programming).




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!

Only \$34.95

DOS Enhancements:

- Date support through December 31, 1999; time stamping for files.
- LDOS frees 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 files 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.
- PATCH supports D&F patch lines with REMOVE capabilities.
- DATECONV converts older disks to the new date convention.

BASIC Enhancements:

- Editing now includes 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 \$34.95 (+ S&H), the LDOS 5.3 upgrade kit includes a DOS disk and documentation covering the enhancements. Specify Model 3/4 or MAX-80. If you don't already own LDOS 5.1.4, get our USER manual for \$33 additional.



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



The Gobbling Box

The Gobbling Box, by Anthony Cosentino

Since the holidays are coming up, and everyone likes to get away from all of the serious stuff during the holidays, MISOSYS is re-releasing our last year's game program, The Gobbling Box. This fast-paced action arcade-type game was written by Anthony Cosentino to run on the TRS-80 Model I, III, and 4/4P/4D.

The GOBBLING BOX game generates a variety of special sound effects and music which complement the action on the screen. The action sounds are ported simultaneously through both the cassette port and the sound port of the Model 4/4p. The four arrow keys are normally used to control the movements of the GOBBLER in this game; although the game also supports the use of an Alpha Products joystick for "gobbler" positioning.

The object of the game is to have your GOBBLER eat up as many dots as possible, while trying to avoid the ZONKERS. The ZONKERS won't stop chasing your GOBBLER until one of them eats it or until the GOBBLER eats all the dots on the GameBox. As a reward for cleaning up the GameBox, the GOBBLER gets a new Box to play around in, with about 200 more dots to eat. To help you play longer, you'll get a BONUS GOBBLER. The GOBBLER can Tame the ZONKERS for a short while by eating one of the ENERGIZERS on the board. Then it's the GOBBLERS turn to chase, catch and eat the ZONKERS.

The game has two skill levels; the pace is fast; the sound is great; the action is continuous. For only \$19.95 plus \$2 S&H (\$3 Canada, \$5 foreign), you can't beat this bargain of a game. Even Stacey plays it!

Lair of the Dragon

Lair of the Dragon, by David Goben

Imagine yourself sitting down behind your computer, inserting a program disk, and as your fingers kiss the keys, a blast from the beyond suddenly rips you right out of your seat and transports you into a mythical land filled with mystery and magic? This is the stuff that interactive fiction is made of; for in place of just sitting back safely and comfortably in your easy chair as you read about the daring and exciting exploits of someone else, you instead read about the events that YOU YOURSELF create for the main character, who ALSO happens to be YOU! And instead of reading from start to finish a story that never varies in its content no matter how many times that you may read it, you can change the plot and vary the action within the story each time that you choose to go through it. YOU will decide how the main character will approach and overcome a problem. YOU will be the one who will have to defend yourself against an offending foe. You alone will decide when, how and where the daring deeds are to be done. Instead of reading about how the main character cleverly solved a puzzle or ingeniously navigated a mind-twisting maze, it shall be for YOU to test your skills at deducting and reasoning these courses. And if you fail to solve a puzzle or else get hopelessly lost, you will have only yourself to blame.

Here the hero or heroine is not safely shielded from death by being locked into a singular, never-changing storyline. Here they can get killed for a foolish misstep, an undiplomatic gesture, or by failing to watch out for an approaching danger. It is you who is responsible for exploring the lands and seeking behind the hidden corners. You are the one who will be solely responsible for the ultimate outcome of the adventure.

Let me tell you, fighting off an old geezer of a dragon is no easy task. Especially if it is a MEAN old geezer of a dragon. And ESPECIALLY if it happens that the battle is waged on the

fields of a MegAdventure. What would YOU do if you had to pit your wits against an ill-tempered dragon who has been terrorizing your village, raping your fields and starving your children? And how would you go about tracking it down in the mysterious northern mountains, the likes of which only heart-quickenning tales of terror and death are told?

MegAdventure is unlike virtually any other interactive fiction adventure that you have ever played, for it will more than just paint its pictures upon the canvas of your imagination - it will slap the sweat right onto your forehead!

If you truly believe that discovery is one of the finest points in life, if you would like to test your ability to think logically to the fullest extent of your ability, if you would like to take on the largest adventure ever written in the genre of interactive fiction, and if you have the guts to face that which would make any other mortal elf cringe in fear, then Lair of the Dragon, the first of the MegAdventure series, is your cup of poison; for reward is a hard-earned commodity here, not given easily to the timid and the faint-hearted.

So grab your flashlight, sack a lunch, and don't forget to pack your sneakers; for you are about to trek into an adventure whose sheer realism and vastitude will overwhelm you. Come on board to an exciting way to read a story; one that stars YOU as the daring deed-doer. And if this is your first journey into interactive fiction adventures, then WELCOME! You are in for a glorious challenge. And if you are an old hand at adventuring, then be prepared for a worthy opponent.

MegAdventure rips the door to adventure right off its hinges! Lair of the Dragon is available now for the TRS-80 Model 4 and Model III. Our MS-DOS version will be available in a few months. If you thought the TRS-80 was dead, think again. Lair of the Dragon, just \$29.95 plus \$2 S&H (\$3 Canada, \$5 foreign).

HQ: resident tools

HQ, by TEK Microsystems, Incorporated

Did you ever like a personally-used product so much that you decided to resell it to your own

customers? Well I have, and HQ is it. Here's an MS-DOS memory-resident desktop manager that blows Sidekick away! Infoworld rated it better than Lotus' Metro. Software Digest rated it the best among ten desktop products. I use it on my PC; HQ has become as essential as my right hand. Every day, besides using it for appointments and notes, I keep finding another feature that saves me the hassle of going to DOS to run other programs to get a job done - HQ pops up to get it done quickly!

Instead of just giving you the highlights of this excellent product, a 4-page flyer depicting HQ's features and specifications is reprinted in this TMQ. Let me just say that HQ will be absolutely the best piece of software for your money; and makes your MS-DOS machine do what you need it to do. You have my word on it. If that's not enough, HQ is covered by our no-questions-asked 30-day return policy. Buy it. If you are not satisfied, send it back. I'll refund your money (shipping and handling fees excluded). HQ is \$75 plus \$4 S&H (\$5 Canada, \$12 Foreign).

MC Library Update

MC Library update

Here's an update which all MC and PRO-MC users should want. The compiler remains the same, but the libraries have had some significant additions and alterations. First, we wanted to enhance those functions which had anything to do with the system date or the directory structure so that they would support the extended dating and time stamp of the x.3 OS releases. While that was being done, Rich felt that we should take the opportunity to include additional functions which are "state-of-the-art" for present day C compilers as well as other functions referenced in the current ANSI draft proposal and the IEEE POSIX standard. Here then is a list of the new functions as well as the new headers:

access()	test file accessibility
bsearch()	binary search
chmod()	change file modes
closedir()	close a directory stream
div()	provide quotient and remainder
difftime()	difference between two times
fattrib()	get/set file attributes
fgetpos()	return current stream offset
fsetpos()	set current stream offset

ldiv() quotient and remainder of long
 mktime() convert broken down time long time
 opendir() open a directory stream
 paint() fill in a bounded region
 readdir() pointer to next dir entry
 remove() same as unlink()
 rename() change the name of a file
 rewinddir() reset directory pointer
 sleep() delays program execution
 stat() obtain file statistics
 stime() set system time
 strcoll() behaves just like strcmp()
 strdup() duplicate a string
 strftime() time to formatted string
 strlwr() lowercase a string
 strstr() search for s2 in s1
 strtod() convert string to double
 strtok() break string into tokens
 strtol() convert string to long
 strtoul() strtol w/o leading sign
 strupr() uppercase a string
 uname() get name of current OS
 utime() set file time of last mod
 vfprintf() fprintf via arg list
 vprintf() printf via arg list
 vsprintf() sprintf via arg list
 yflag() return status of OS dating

ansi.h keywords, not implemented
 assert.h verify program assertion
 attrib.h used with fattrib()
 dirent.h used with directory operations
 limits.h specifies ranges for integer data
 malloc.h dynamic memory allocation
 memory.h memory functions
 stdarg.h variable argument access
 stddef.h common definitions
 stdlib.h general utilities
 string.h used with string functions
 types.h typedefs various data types
 unistd.h file access
 ustat.h used with ustat()
 utsname.h used with uname()
 varargs.h used with vprintf()

If you purchased your MC or PRO-MC after June 30th, 1987, the upgrade is free. If MC or PRO-MC was purchased prior to July 1st, 1987, the cost of the update is \$20 + \$2 S&H (\$3 Canada, \$6 Foreign). Please send both of your original MC or PRO-MC master disks and the update fee.

GreatBac-86

GreatBac-86, by Sellers & Company

If you have been procrastinating about backing up your MS-DOS hard disk, it's probably because you're frustrated with the tiresome and sluggish operation of BACKUP. Well it only takes a moment lose precious data. What then?

For those of you who have been living on the edge, GreatBac-86 is here to save your day, and speed the protection of your files. Here's some highlights:

- o GreatBac-86 is extremely fast
- o menu or command-line driven,
- o supports 360K, 720K, 720K in 1.2M, & 1.2Meg drives & supports 3.5" drives
- o works on PC/XT/AT and PS/2
- o can format floppies on the fly,
- o can back up to two similar drives
- o automatic disk change detection,
- o full or incremental backups

We use GreatBac-86 for our customer data base. And let me tell you, Brenda does not want to have to re-type 20,000 records! If you're still using BACKUP, then take my word for it, GreatBac-86 belongs in your machine. If that's not enough to convince you, try GreatBac-86 for 30 days; compare it to the others. If you are not satisfied with this product, send it back. I'll refund your money (shipping and handling fees excluded). GreatBac-86 is \$59.95 plus \$2 S&H (\$3 Canada, \$6 Foreign).

XLR8: memory & speed

XLR8 speedup/memory board available

With all of our emphasis on this board being a necessary enhancement of your Model 4, we have made arrangements to carry this product in stock. See how it can improve your Model 4 by reading our article on SPEED. When ordering, you must specify your exact machine configuration (4/4p/4d, hires graphics). Your cost is \$245 plus \$5 S&H (\$6 Canada, \$15 Foreign).

LDOS

Model I/III Information

Getting hold of a patchable FORTRAN I

Fm Gary Phillips To MISOSYS, Inc: In the process of hunting for a copy of model 4 Fortran (which I did finally get in spite of the Shack's efforts to dissuade me) I managed to locate a copy of the model 1 version. Recalling that there were patches to make the model 1 version work under LDOS in model 3, I bought the used model 1 software. Alas, the fixes in DL1 and on FixDisk 2 do not match the version I have in hand. I notice in an old LSI journal that LSI used to provide fixes for the various versions of F80/L80/M80 for a small fee. I'm sure you are too busy to provide such a service these days, but did you perhaps inherit any information about the differences between the various versions that would be of help to some of us? Could we induce you to share it?

I am fairly sure I can get F80 and L80 working by poking around in there myself and looking for the routines fixed by the standard patches (which are unfortunately X type patches so don't tell you what they are replacing). What I quail at is trying to reproduce the patch to FORLIB/REL. Can't disassemble that baby! (And MLIB barfs when I even try to map it.) The FORLIB/FIX uses D type patches without corresponding F lines. Do you (or anyone else for that matter) have the F information that I should be hunting for?

Fm MISOSYS, Inc: What LSI did was to recopy the patchable version onto your master disks. We will still perform that service. I doubt that you could patch FORLIB. And what do you mean that you cannot disassemble FORLIB/REL? Don't tell me you haven't heard of our UNREL package? It was announced three QUARTERLIES ago and has been in our language ad in 80 Micro. The solution to getting FORLIB/REL

loaded into MLIB (assuming it will fit), is to append a one-byte file to FORLIB/REL. The byte value is X'9E'. If FORLIB is too big to load, it needs to be split. The splitlib program in Fall 1986 TMQ is usable for that. UNREL also includes a similar facility (actually an enhanced splitlib). Does this solve your problems?

Fm Gary Phillips To MISOSYS, Inc: Yes, I had seen the information on UNREL, but don't have it, and up to this moment have had no use for it. I can see its applicability in this case, though. FORLIB is not too big for MLIB (at least, it is smaller than the MLIB buffer) but it hadn't occurred to me that the 9E ending byte could be missing... I'll take a look at that. Now, as for the F80/M80 upgrade, is it still on the same terms (\$10 for each product?). If I supply blank disks, can you provide the upgrade on the separate disk rather than overwriting the originals? (I realize I still have to send you the originals to prove I own them.) DSMBLR and MRAS I have and use extensively, but I'm not and do not expect to become a C programmer, so I don't have MC. I can see the utility of UNREL if I begin doing much work with other folks' /REL libraries, but I'm not sure how much of that I would ever do.

Fm MISOSYS, Inc: I'll handle your request for those same terms as LSI used to. UNREL would be quite useful in fooling with FORLIB; and yes, the 9E is necessary so that MLIB can pick up the end of the library correctly.

Fm Gary Phillips To MISOSYS, Inc: Aside on the MLIB/FORLIB problem. Checked FORLIB and found that it does already have x'9E' on the end, but in both the model 1 and model 4 versions the 9E value is separated from the preceding x'9C' (end of program) flag by several bytes of nulls. Tried patching the model 1 version to put the x'9E' immediately after the end of the last module, but MLIB still rejects it. When asked to load FORLIB/REL, MLIB reads for a while, and then reports an 'Invalid file format...' This happens with both versions of FORLIB. It shouldn't be size: MLIB's buffer is 40000+ bytes, and FORLIB is only 28K. Any other ideas why MLIB can't load FORLIB? (My MLIB is PRO-MLIB version 4.0, my DOS is 6.3, and the hardware is a 4P with XLR8er installed.)

Fm MISOSYS, Inc: My MLIB loads my copy of FORLIB. But then I'm loading a Model 4 FORLIB. That shouldn't matter; unless your Model I FORLIB is damaged. I don't have any other an-

swers without seeing your FORLIB and checking it out.

Fm Gary Phillips To MISOSYS, Inc: Well I figured out what I was doing wrong. You were right about the hex 9E, only I misunderstood the requirements. It IS present in the file, but not as the very last byte. The file ends on a 128-byte boundary and is padded to that point with nulls. (Looks like Microsoft must have used a CP/M utility or a Fortran program to build FORLIB with.) After using PATCH to change the last byte to a 9E, MLIB was perfectly happy. Now my next question: why doesn't MLIB object to the twenty or so bytes of 00 between the last 9C and the final 9E?

Fm MISOSYS, Inc: I don't know. It has to do with the way Rich coded his routine to scan through a REL library. It requires that the file's EOF be at a byte which is the 9E. The 9EH value, incidentally, is the special link item which is indicative of the end of the file.

Double-sided BOOT disks...

Fm Garry DeVol: I have a 128K 4p with 2 DSDD Tandon TM50-2 drives which I installed myself. They have worked well on TRSDOS 6.2.0 and 6.2.1 and on LS-DOS 6.3 in double-sided operation. I can get LDOS 5.3 to work once loaded, but I cannot get a double-sided disk to boot. The logo comes up, but the computer hangs up at that point, drive 0 tries to do something repeatedly and only hitting the reset switch is the only thing that will stop it. However, I can boot 5.3 with a single-sided disk and switch to the double-sided one with no problems. It is inconvenient, though. Is this a known problem? Is there a simple solution?

Fm Jim Beard To Garry DeVol: When the Model III/4 boots, it reads :0 using the ROM disk drivers, which only read one side. The DOS disk drivers are on SYS0/SYS. If you make a system disk by just "BACKUP \$/SYS:1 :0 (S)", the first file transferred will always be SYS6/SYS because of its privileged place in the directory. If SYS0/SYS is put on the back side of the disk when its turn comes, the ROM driver can't find it. The solution is to copy SYS0/SYS first with its own BACKUP SYS0/SYS:1 :0 (S) command. This is necessary for any DS system disk, regardless of DOS.

Model III CTRL key revisited

Fm Vince Pepe: I just got LDOS 5.3 and TED doesn't recognize the control-A (insert/overtyping toggle) key. I am using it on a 4P. Any ideas?

Fm MISOSYS, Inc: CONTROL key when operating in Model III mode is LEFT-SHIFT-DOWNARROW. If you read the README/TXT file as documented in the FIRST paragraph of the LDOS 5.3 docs, you will find a FIX to convert over to the RIGHT-SHIFT-DOWNARROW which is helpful if your 4P has a clustered keyboard. The ideal solution is to purchase the Model 4 hardware interface kit as discussed in the companion yellow flyer. It includes a KI4/DVR keyboard driver which makes use of the CTRL key, the CAPS key, and the Function keys. By the way, the discussion of the LEFT SHIFT vs RIGHT SHIFT was beat to death in a previous THE MISOSYS QUARTERLY.

Fm Vince Pepe To MISOSYS, Inc: Just received my Model 4 interface kit and it works fine. I read the readme file but somehow I missed that. I really don't use the 4P anymore as I do all of my work on my PS/2 model 50 but I set the thing up just to relive old times, so, I ended up upgrading to LDOS 5.3 (nice upgrade), getting the hardware interface kit, buying a new LDOS manual (can I get index tabs for it, I have lost my original manual).

Fm MISOSYS, Inc: Actually, we have hundreds of tab sets. The Blurb now advises how to get your "free" set.

SYSTEM disks, DATECONV, type ahead...

Fm Robert E. Cunningham: I recently purchased a copy of LDOS 5.3 for my TRS-80 Model 3. While I was initially quite happy with your product, I have some questions relating to five possible bugs I feel I may have discovered. I would appreciate any assistance you can provide me in determining the cause of my difficulties.

(1) Is it possible to create a 'system only' disk in one pass? If I follow the directions given in the backup section of my 5.1.3 manual, BACKUP :0 :1 (S). it will always copy all files (system plus all others) requiring me to do a PURGE (Q=N) to get rid of the extras.

(2) Is there a way to disable the DIR (A) type display? At times, I don't want to see all the extra information. The old LDOS manual pro-

vided a patch to SYS6/SYS, but the bytes for the new SYS6 don't match.

(3) How can I make a double-sided BOOTABLE disk? If I follow the directions for LDOS 5.1.3 (page 1-29), I am able to do a DIR, SYSTEM (SYSGEN), set filters/drivers, etc., but if I attempt to boot this disk, it will hang up shortly after it prints the big LDOS letters.

(4) Why, during the DATECONV process, does it give me a 'SYSTEM disk conversion requires a 5.3 or 6.3 disk!' error message even if it is a data only disk? I had created some of these disks years ago, but not all of these disks show this error. I cannot find a common thread as to what is different for some disks giving this message but others not giving it. Is the DATECONV utility looking at something in the directory that I could change with FED in order to complete the process?

(5) Occasionally, even with the KI/DVR (TYPE) set, the computer will drop letters during I/O or other functions. For example, if the screen is updating and I type DIR, it might show IR or DR which are not correct functions requiring me to re-enter the command. This was not a problem with 5.1.3. The only way I can get it to accept all the letters is to wait at least 1 second between keystrokes.

Fm MISOSYS, Inc: BACKUP was changed in 5.3 to work like LS-DOS 6.3 in connection with its parameters. If you want to make a "system only" disk, then issue a "BACKUP SYS:0 :1 (S)" command. BACKUP works by including additional files rather than by limiting. Thus, "BACKUP" does visible files, "BACKUP (I)" does visible and invisible, "BACKUP (S)" does system and visible. If you read the documentation on page 10 which came with the LDOS 5.3 upgrade kit, you will note this revision.

The patch to default DIR to the abbreviated listing appeared in THE MISOSYS QUARTERLY, Volume I, Issue iii. It was,

```
PATCH SYS6/SYS.SYSTEM (D08,92=00 00:
                      F08,92=FF FF)
```

You really should consider a subscription. At \$25/year for 3rd class mailing or \$30/year for 1st class mailing, it's a great value.

You can make a double sided system disk by following the procedure listed in the README/TXT disk on your disk; not those of 5.1.3. Have you read it?

DATECONV would give that error message when it detected the presence of SYS1/SYS on the disk. Double check those target disks; you will probably find a SYS1/SYS file on them.

Dropping letters will invariable occur if you have turned on the SMOOTH facility via "SYSTEM (SMOOTH)" and try to type while floppy disk I/O is in progress. With TYPE not set, you will always drop letters if you are a fast typist. With TYPE set, you shouldn't drop letters other than during disk I/O.

Beware of linking *DO to *PR

Fm Robert S. Arner: [When I LINK *DO to *PR], I get random underlinings, and sometimes, even unwanted form feeds. This is not too frequent a problem, but it happens often enough to be a nuisance! I do not see how a cursor control enters the picture, especially when I do things such as copy a sample LIST command from the documentation.

Fm MISOSYS, Inc: Let me see if I can shed some light on what is happening. To begin with, when you LINK the *DO device to the *PR device, any character code which is sent to the video will also be sent to the printer. This may have been okay in the old days when printers were dumb and responded only to control codes for line feed and carriage return, but today's printers do much more than that. Two of the most common control codes outside of carriage return (13d) and linefeed (10d) are codes which turn ON the cursor (14d) and turn OFF the cursor (15d). If you look at Appendix C in a Model III Radio Shack BASIC manual, you will see other possible video control codes: backspace and erase (8), tab (9), swap space compression/special chars (21), and various cursor movement codes (24, 25, 26, 27, 28, etc). Based on the behavior of your printing examples, I suspect that your printer is responding to a cursor OFF by enabling underlining. My DMP500, for example, uses the code 15D to "start underline" while code 14D "ends underline". If you really want to utilize your printer linked to the *DO device and your printer responds to typical video control codes, then you would need to insert a filter into the *PR device to strip off unwanted codes. This would then have a side effect that those codes could not be sent to the printer while such a filter is installed. Unfortunately, there is no way to easily install a filter between the *DO device and the linkage to *PR. The TRAP (in the case of trapping a

single code) or XLATE filters which we used to sell as part of the FILTER1 package or the MAXLATE filter part of FILTER2 package, would probably do the trick. We may bundle these two discontinued packages together for a year-end sale.

Problem with Aerocomp's 5 Meg HD driver

Fm Dr. Robert S. Arner, O.D.: The problem is that when I set up the hard drive with Aerocomp's HD5SSA3/DCT driver, and then tuck in KI4/DVR, LINK, KSM/FLT, and PR/FLT, I can barely protect the loaded table from my SALESORD program by setting HIGH\$ to 55545. In order to run SALESORD, which is a large program with a lot of memory reserved, I must RESET *PR and put nothing else into high memory. I'd like to use MiniDOS and SPOOL, etc., so I bought RAM expansion from Aerocomp only to find out that access is not so simple; hence I tried to use SET2RAM and MemDISK.

Exhibit III shows the problem of filling the screen with '9' characters with the hard drive activated. A simple floppy setup works. My first priority is to be able to use SET2RAM and memDISK with my hard drive, hopefully without buying more software. At your suggestion, I am enclosing a copy of the HD5SSA3/DCT driver on disk to help in your evaluation. This is the program you suspect is causing the aggravation with SET2RAM, when it won't permit JCL or "DO" commands, and won't copy from MemDISK.

Fm MISOSYS, Inc: Well, I was right, your problem was caused by the Aerocomp hard disk driver; but I do have the solution for you. Their driver zeroed the buffer pointer if a disk verify sector operation was requested. This had the unfortunate result of causing a sector of data to overwrite the first 256 bytes of memory; crash!!! I was able to come up with what I think is a patch to correct this problem. If you apply the patch,

```
PATCH HD5SSA3/DCT (D04,6F=26 38:
                          F04,6F=ED 62)
```

that should wind up using the region in the 3800H-39FFH address range for the verify buffer which is non-conflicting. Use whatever name you may have renamed their file to in lieu of "HD5SSA3".

Using the extended error flag

Fm Vince Domeraski: I spoke with you last week about my problem with installing 5.3 on my MAX-80 hard disk. I used the same installation JCL that works perfectly with 5.1.4, and ended up with a parity error during read when attempting to back up the system files to the hard disk.

Since we spoke, I purchased a 50-pin cable and an installation disk from Micro-Smart, and using the INIT206/JCL, found that the drive formatted and installed under 5.3 on a Model 3, without a hitch. The JCL file is virtually identical for the Model 3 installation except for the driver filespec. So, the problem may have something to do with the unique setup of the MAX-80 or the MAX driver.

Could I attach my MAX to the HD which is now formatted under 5.3 (via the Model 3), using the 40 pin cable, and somehow configure a boot disk top make the drive accessible under 5.3 on the Max?

Fm MISOSYS, Inc To Vince Domeraski: One of the limitations of abbreviated error handling which is exhibited when a set of I/O operations are in effect involving more than one disk drive and an error occurs, is that it is not necessarily evident on which drive the error occurred. Based on your printouts, it appears that the "Parity error during read" may have been detected when BACKUP was reading the source disk (the system disk). Since abbreviated error messages do not indicate the number of the involved drive, that is just an assumption.

You can use the system override to force extended error reporting for all calls to the @ERROR error message handler. You can affirm this override, enabling extended error reporting, by setting bit 6 of SFLAG\$ (at X'442B'). You can do this from command level by issuing a MEMORY (A=X'442B'), noting the value displayed, adding 64 to it, then issuing another memory command as above but with the "B=val" parameter added with val equal to the new value. Error messages will thereafter include the drive number, as well as the track and sector. That will give you an added clue. If its the source disk, then something on it is unreadable. Perhaps it was duplicated on a drive which is out of alignment with the zero drive used on your MAX. QFB the master from :1 to :0 in that case and try again. If its the hard drive that has the error, something else is screwy. Do you know where their driver in-

stalls? Issue a MEMORY command and you will get a high-memory map. I would hope that's where they put it. If they stuff it in low memory somewhere, it may conflict with other system code.

Lastly, you could connect the MAX HD formatted under Model III 5.3 but you would still have to install their driver. If you run their installation JCL but omit PART 5, are you able to access the hard drive with other commands such as DIR, FREE, etc? Find out where the error is occurring. I'll hold onto your listings in case you need to get back with me.

TED: Can the entire screen be refreshed?

Fm Patrick H. Larkin to MISOSYS, Inc: Has a (potential) "bug" been reported for TED on LDOS-530 having to do with the scroll-protected line separating text from the TED 1.1 copyright notice at the bottom? I have found that if I use the MiniDos {q} command at or near the bottom of the screen such that it over-writes the underscore line, when the screen is refreshed, whatever was on that line (1, 2, or 3 filenames) stays there (is not replaced by the underscore). {An example is enclosed. NOTE: In attempting to get the example for you, I discovered I don't always get the problem.} A minor distraction (cosmetic), to be sure, but not as clean as it could be.

Fm MISOSYS, Inc: The MINIDOS filter supplied with LDOS will disturb the video screen generated by any program. The only recovery of that is a program which allows you to redraw the entire screen; TED does not. A solution, albeit not a great one, would be to exit TED, re-enter TED using the "*" parameter, then do page downs till you reach the end of the text.

Model I: What do we do after 12/31/87?

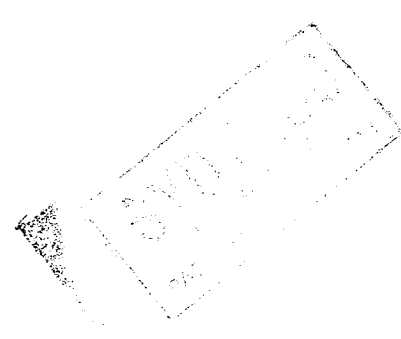
Fm Michael Riskin: I have just received TMQ Vol II, Issue I and I am worried that you may mean what you responded to David D. Brown regarding Model I 5.1.4 date extending. There are some of us still using 5.1.4 who do not use on-line services. I, for one, am relying on you for valid and useful information regarding updates and patches. Please do not forsake us to the whims of others. If there is to be a patch let it be of your quality and ability instead of from some unknown source. If there is not to be one from you, I will

turn off the date function rather than install a patch I can't trust.

Fm MISOSYS, Inc: Rest assured, we have taken care of the situation; although not necessarily with the response that some folks want. This issue provides a set of patches to Model I LDOS 5.1.4 which will (1) deactivate the dating which occurs in the directory, and (2) extend the system date support to 1999. There are five patches included in the set: BU/FIX, SYSO/FIX, SYS3/FIX, SYS6/FIX, and SYS7/FIX. A Job Control Language file called DATEFIX/JCL installs the fixes, painlessly. The following alterations are made to the DOS with the application of these fixes:

1. BOOTing and the DATE library command accept year entries in the range 1980-1999
2. DIR and PURGE no longer display the MOD Date
3. DIR, PURGE, and BACKUP no longer accept the DATE= parameter
4. @CLOSE no longer updates the date field in the directory
5. The DATE\$ storage used by ROM routines is correct to 12/31/99. This means that programs which call @DATE will be returned a valid date; this includes BASIC's TIME\$.

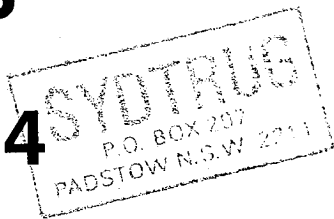
The patches and JCL file will be listed in The Patch Corner and will appear on DISK NOTES 2.2. They will also be part of FIX22.TXT on our CompuServe forum (PCS 49, LDOS).



LS-DOS

Model 4

Information



Programs which use too much system stack

Fm Anthony De Vito, Jr. To MISOSYS, Inc: I have noticed that Deskmate trashes my M4 system whether 6.2.1 or 6.3. I first noticed it when I exited Deskmate and tried to start up COM/DVR and couldn't. To check this problem out, I did "device (b)" and when it got past the disk drives it started scattering the display all over the screen interspersed with garbage characters. I discovered a foolproof way of creating this problem: turn on the alarm in deskmate. It doesn't matter if the alarm goes off or not or if you turn it off or not before exiting, and the rubbish displayed on the device command appears to not be uniform except that it only occurs if the byte option is specified.

Fm MISOSYS, Inc: That tells me that Deskmate has not set up its own stack area! That's ridiculous for a program of that stack usage. Since the system stack will overflow into the device table page, any program which overuses the system stack will corrupt the device table. Since it would be rare for any user to have all 31 device table entries active, the overflow into the DCB area was deemed the least harmful to the system when a program overuses the system stack. It's just good programming to use your own stack area.

ALTRES hazardous with LS-DOS 6.3

Fm Paul Rehberg: Have just installed 6.3. The ID command crashes my system if I run my normal boot up JCL, which includes Com driver, Altres (all modules installed), forms, and prowam 2.0. ID works on a 'clean' system, how-

ever. Please tell me my worst fears on your protection scheme have not been realized.

Fm LSI To Paul Rehberg: That's easy. Your worst fears on our protection scheme have NOT been realized. (and never will be!)

Now for the hard part. What is trashing your system? It must be something that your JCL is doing. Something there could be messing up a system file or maybe there is a spot of bad RAM just where one of the system files gets RES'd. The most likely suspect is Altres. REAL likely! Bill just talked to Roy and I think that Roy will be working on it. In the meantime, you could set up a memdisk with the SYS files.

Fm MISOSYS, Inc: ALTRES is the culprit. Since we have discontinued the PRO-ESP package, we may or may not be able to come up with the time to check into that. If we do, we'll post the result or publish it in TMQ.

Security for 5.3 & 6.3

Fm Peter Amschel To MISOSYS, Inc: What about 6.3? Can I use the DOS on my other two model IV's? Your documentation says no, so today I sent \$79 to Colorado to "key in" the OS so that they will work on all 3 machines. Knowing you, you probably have imbedded some sort of code which recognizes a particular machine somehow and then gets a counter going so that after a certain # of times using the one dos on the other machine, some sort of worm program launches forth on the data files in fulfillment of the prophecy set forth on your warranty, right? Well, send my "keyed in" DOSses back in a hurry because I am only running the new DOS on one machine.

Fm Marc Nowell To Peter Amschel: I think that you've got the wrong guy! Roy does LDOS and is located, with MISOSYS, in Virginia. LS-DOS 6.3 is published by LSI in Colorado. They are two separate firms.

As for the "anti-piracy" protection, all that we know about it is (A) We only have LSI's word for it that it exists, (B) We only have LSI's word for it that it won't cause problems for legitimate users in the future, and (C) The "customer support" guy at LSI ain't answering legitimate user's questions about it, instead taking a stand of "take the money and run, unless we can get more money out of them."

Fm MISOSYS, Inc: Neither Roy Soltoff nor MISOSYS have any documentation on 6.3 which state anything. 6.3 is NOT, I repeat, NOT our product. Please direct all questions of that sort to the proper company. You purchased 6.3 from LSI.

JCL blues

Fm Shane Dawalt: I must admit, I've been spoiled by the MS-DOS batch file. So, naturally, I tried some 'illegal' syntax stuff under LS-DOS's /JCL files. Well, as you probably figured out, it didn't work. Basically, I want to send a command to DOS from a JCL file. Now, this certain command, which is actually a C program and requires a redirection to the printer. The JCL compiler is getting stuck on the syntax "#*pr". The full command line is:

```
mcp pf:#dd" +o=#wd# #*pr
```

The JCL compiler thinks that I was typing a label and forgot to put the other "#" at the end of "#*pr". I can't see any way around this. Any body have any ideas?

Fm Jim Beard To Shane Dawalt: Use the JCL SET command to define arbitrary variables as troublesome strings.

Fm Shane Dawalt To Jim Beard: I thought about that, but I thought the compiler would still flag an error. I based this on the fact that if you place a substitution label within a compiler comment (and probably within a "regular" comment), the compiler will change it to the substituted value. Anyway, thanks for the suggestion, I'll try it.

Fm MISOSYS, Inc: The solution is to tell the DO compiler that you want a pound sign (octothorp) by coding two "#" characters. Thus, your line should read as follows:

```
mcp pf:#dd" +o=#wd# ##*pr
```

Note the use of the double pound sign which precedes the "*pr" string. Page A-22 of my TRSDOS 6.2 manual discusses the use of two "#" characters.

How do I unsysgen a hard drive?

Fm MISOSYS, Inc: You UNSYSGEN a hard drive startup disk by holding down the CLEAR key

after the date/time prompts. After that, you're clean. Permanent removal thereafter by "SYSGEN (OFF)".

Checking availability of memory banks

Fm JEFFREY KLINE: I am now writing my extended memory handler routines for a large program I am working on. Is there a way of determining if the machine has 128 k without using the @BANK call and testing each bank? What I am hoping is that I can look at an area of low RAM to determine if the machine is so equipped. If not then is there any reason that I couldn't use the area of x'2600' to x'2FFF' for these routines as they will only be needed once. My main program loads in at x'3000'. I am using the @DODIR SVC as well as the @CMNDR SVC, and if I remember, they use that area don't they? Also, can I swap banks 1 for 2 directly without having to return to 0 in between swaps? I plan on using something like HL to maintain where I am address wise in the bank that is enabled. This way I can ascertain if I am at the end of one bank and make a swap to the other. Incidentally, I am testing for all 8 banks and logging them to an 8 byte word of which by reading the bits that are set, you can find for the available banks. The program when done will work on 128 k machines but more is better!

One last item. I do now have something to write about for TMQ. How now do I go about getting it done and sent to you? What's involved in contributing to it?

Fm MISOSYS, Inc: Use @BANK to test for bank availability. What's the matter? Don't you have the time? Use of @BANK has nothing to do with 2600H-2FFFH; CMNDR does. You don't have to return to 0. You can install whatever bank and use it. Testing for all 8? To be totally effective in this day and age, you should test for banks 1-30. With the Alpha Tech memory board (banks 0-30) and the H.I.Tech XLR8er (banks 0-10), why not go the limit?

TMQ Article submission is best done by mail. You may use our normal address.

More on 3.5 inch drives

Fm Michael Kushner To DUANE SAYLOR: I noticed via the TMQ that you replaced your 5 1/4" floppy(ies) with the Toshiba drives. One

(possibly stupid) question - How do you go about extracting a disk from the drive? On the MAC, its a <CTRL><E> (more or less). Is there an equivalent method on the 4, or do these 3.5's come with some kind of drive door latch or manual ejection button? Am I making my question clear?

Fm DUANE SAYLOR To Michael Kushner: The 3.5 inch drive that I have (A NEC FD1035) has a pushbutton that ejects the microdisk. The Atari ST and Amiga use a similar drive.

Fm --jjkd-- To Michael Kushner: All the IBM 3.5 inch drive also have "real" eject buttons, I believe that the Apple Mac products are the only ones conceited enough to have software eject control.

Fm jeff brenton To --jjkd--: Actually, they DO have eject buttons on the Mac, but they don't have the extension on it to make it readily accessible. That little pin-hole under the drive is where you stick (for example) a straightened paper clip in to force an eject. They do this because FINDER can get seriously lost if you change disks on it randomly.

Fm --jjkd-- To jeff brenton: Unless my memory is misleading me, even that isn't a true mechanical eject button, merely a microswitch that forces the disk eject logic to hork out the disk. I and my bosses have been lucky enough in various battles such that there are absolutely no Macs at all present in my entire office complex (unlike much of the rest of Motorola), so I can't confirm this easily by trying to get a disk in and out with no power to the machine. Unfortunately I think I will end up having to acquire and support a Mac SE and LaserWriter sometime in the next few months. It isn't a total loss, the LaserWriter is a pretty useful device.

Removing protected files

Fm Dave Baldwin: I want to remove the unused files from my TRSDOS and LDOS system disks. By unused I mean files like "TAPE100/CMD" and "REPAIR/CMD" (etc ...). These are all password protected and I would like to know how to remove them from my system disks in order to free up some disk space. I don't have an editor/assembler and hope there is a way to do it from BASIC. I'd like to know what the passwords are so I could use the "ATTRIB" command to remove them. I don't feel comfortable at-

tempting to toggle the protect bytes on a directory entry.

Fm --jjkd-- To DAVE BALDWIN: You have the ability to do this (remove password protected files) right in the operating system, Dave. Simply use the command

```
PURGE $:0 (S,I)
```

You will then be prompted with the name of every file on the disk, and asked if you want to get rid of it. Things to get rid of on working disks are:

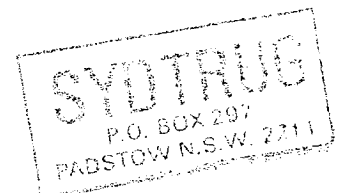
```
TAPE100/CMD
REPAIR/CMD
SYS5/SYS (if you don't use DEBUG)
SYS9/SYS (if you don't use DEBUG)
SYS13/SYS (if no User Command nor ECI)
MODELA/III (if you have a 4 instead of a
4P, or don't intend to use M3 mode)
```

ad nauseum. Ask if you aren't sure about any other names that come up.

Multiple file transfer with COMM downloads

Fm Alan Varga: Is there a patch to correct a problem with downloading more than one file at a time with COMM? I am running LS-DOS 6.3 and I can only download one file at a time. I reset *FR, close it and turn off handshaking. When I try to open a new filename and turn on *FR and handshaking I can't receive another file under the new filename. The only solution is to close everything up again, exit COMM, re-enter COMM and re-open handshaking, filename and device. Has anyone else had this problem?

Fm MISOSYS, Inc: When you first go into COMM, DUMP-TO-DISK is set ON. After you receive a file, d-t-t is automatically turned off when you do a RECEIVE FILE RESET. You have to turn DUMP-TO-DISK back on if you are going to receive a subsequent file. Incidentally, if you exit COMM without closing your send or receive files explicitly, COMM will do that for you automatically.



PERCOM 10Meg drive under LS-DOS

Fm Donald R. Arrowood: I just picked up a PERCOM 10 Meg HD, Model # PHDS10C2, I have the cable and the PDCHD-LDOS Winchester Disk Driver Version 5.1b. I have tried to format the HD with LSDOS 6.3 and RSHARD but it won't go, I seen in the MISOSYS QUARTERLY there is a mention that RSHARD supports the WD controller (1000)series) and when you run RSHARD it has the WD 1000 & 1010 controller mentioned, the controller in the percom is a WD 1001-05 and the only thing I have that will format it is Model 3 LDOS that is with the HD. I called that company about a driver for Model 4 mode and they said they don't have that any more just the Model 3 driver. Is a driver available through MISOSYS or is it not possible for a Model 4 driver in TRSDOS 6.2.1 or LSDOS 6.3. Or maybe a patch to modify the available drivers to work on the PERCOM. I now have LDOS 5.3 on the HD and I would like to format half of the drive for Model 3 and half for Model 4 if a driver can be got for the Model 4 mode or would there be a problem with this PERCOM drive to handle the dual format? Can the PERCOM HD have a larger bubble installed as easy as in the RS HD unites .

Fm MISOSYS, Inc: Donald, All that I can tell you is that RSHARD will not work the Percom drive; they used their own driver (I believe that it was done by LSI, but I don't think a Model 4 driver was ever done). Since I never had any involvement with the Percom work, I cannot add more than what I just said. You may need to resort to disassembling the existing driver and adapt it yourself to the Model 4 mode.

Fm --jjkd-- To Donald R. Arrowood: As I recall, the only two significant differences between the RS W series and the Percom drives were that the Percom used a different base address, and there was no WP signal. If I am recalling correctly, it would probably be a trivial thing to hack up RSHard to run on the Percom, given that you have a working Model 3 driver to use as a model (e.g. can look in it to check for sure). If you know assembler, you should be able to check, fix and confirm the whole thing in about a half hour with nothing other than FED II and Debug (E) as working tools.

Installing a bigger bubble in a Percom drive is even easier than in a RS box, as there is no WP signal to worry about. I seem to recall their power supplies looking a little small, though we never had one fail in service, I'd

probably try to pick a reasonably low power drive as a replacement. The Miniscribe 3650 seems to be a pretty good buy right now.

Fm Donald R. Arrowood To --jjkd--: The Percom Model 3 driver is a good working master copy. I have it up and running, is doing a fine job. I picked this bargain up at the Hamfest for \$160.00, just could not pass it up, got the guy down from \$215.00 he was asking for it. I am a little out of luck on knowing assembler, could maybe be something new to learn (just what I need more work). I am looking at a MiniScribe 40 Meg to put into this other RS 5 Meg unit, a couple of us are thinking about running a BBS in our area, we might have some support of several computer stores that seem to want to run ads on it - I'm not to sure about it. Nine months ago I did not own a HD now I have three for the Model 4, a 20 Meg, 10 Meg and a 5 Meg, and also two 20 Megs on the TIK with a total of 6 360K floppy drives - when will it (me) stop. It is lots of fun.

Fm --jjkd-- To Donald R. Arrowood: Congratulations on moving up to the hard disk world. As you have noticed, they're addictive. Once you get a small one, you need to go bigger and bigger, and buy more and more and MORE... Hard to remember how you accomplished anything with just floppies, eh? Les did the Percom Model 3 drivers, perhaps he'll chime in with a bit more advice.

Fm Donald R. Arrowood To --jjkd--: You're right about it being addictive. One has to think why they even make floppies, they don't even seem to be of any use after you get the HD burned into your brain. Bigger is better, working on the next one to be the Miniscribe 3650. Hope to get a little (little) help on the driver so I can make use of it in the Model 4 Mode, think I'll split the HD in half to make use of Model 3 and 4 Modes.

Fm Les Mikesell To Donald R. Arrowood: The only difference in the older Percom HD's (with WD controller) and the Radio Shack ones is the lack of hardware write protect and the base port address for the controller. If you make the following changes to the driver that came with TRSDOS 6.2, it will work for the old style Percoms. You won't be able to format under 6.x, but you don't need to either, since the 5.x formatter has already done the work. I like to set a drive up with 3 partitions: a single head for each DOS and the rest in a single shared drive.

Use FED to make these changes: (I'm too lazy to make it into a patch)

rec,offset	Percom	TRS
X'0000,02	X'50 44 43	X'54 52 53
X'0000,1C	X'50	X'48
X'0000,2A	X'00 00	X'D3 C1
X'0000,33	X'00 00	X'D3 C1
X'0000,37	X'00 00 00	X'DB CF E6
X'0000,3A	X'00 00 AF	X'80 20 E2
X'0000,3E	X'7E	X'CE
X'0000,42	X'7F	X'CF
X'0000,48	X'7D	X'CD
X'0000,4A	X'7C	X'CC
X'0000,4E	X'7F	X'CF
X'0000,65	X'79	X'C9
X'0000,73	X'00 00 00	X'DB C0 DC
X'0000,76	X'00 00 00	X'07 0D 20
X'0000,79	X'00	X'FC
X'0000,A5	X'7F	X'CF
X'0000,CE	X'7F	X'CF
X'0000,DD	X'79	X'C9
X'0000,E8	X'7F	X'CF
X'0000,EF	X'00 00	X'D3 C1
X'0000,F3	X'00	X'D3
X'0000,F4	X'00 7F	X'C1 CF
X'0001,0A	X'7F	X'CF
X'0001,13	X'7F	X'CF
X'0001,55	X'7E	X'CE
X'0001,58	X'7B	X'CB
X'0001,5B	X'7C	X'CC
X'0001,5E	X'7D	X'CD
X'0001,61	X'78	X'C8

When is 25.5K only less than?

Fm Adam Rubin: I've run into a little problem, or at least an interesting situation, with 6.3.0 on my 128K Model 4P. (The same problem occurs under 5.3.0 with the Model 4 Interface Kit, incidentally.)

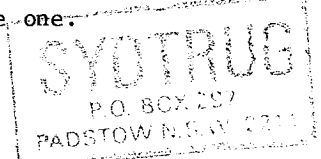
I created a double-density MemDISK as :7 using only bank 1, which gave me 25.5K of free space on that drive. However, when I tried to use BACKUP to move a 24K file onto the MemDISK, I got the message "File is larger than destination capacity -- BACKUP is bypassed."

Why won't BACKUP move this file? COPY had no problems with this file, and I still had 1.5K free on the MemDISK when COPY was done.

Fm MISOSYS, Inc: That's because BACKUP uses a simplified algorithm to calculate the target destination disk's free space by multiplying the number of cylinders (less one for DIR and one for BOOT) by the number of sectors per

cylinder; it doesn't use the GAT. Whereas COPY, will wind up using the DOS file system to attempt a WRITE of the highest file sector of the source file. If the DOS can allocate the space, COPY will proceed. The DOS will use any free space on the BOOT cylinder; BACKUP will during the file movement but not during the test for size.

Now then, why does BACKUP ignore free space on the BOOT cylinder? Why because some backups are going to use additional BOOT track space if the destination disk is going to be made into a system disk. BACKUP doesn't bother to test to see if it is or isn't - and you are aren't going to make one.



Grade Book Program

Fm Stephen Sutherland 71426,616: I have a series of programs that I have written that several people are using. The series began several years ago, and has grown into quite a large system. If there is any interest, I will try my hand at uploading to DLO after I get my instructions in order. Because of the size of the collective programs, I may have to send the complete documentation to you directly since the programs alone will take quite a bit of download time. (I think together they are around 70K.)

If after reading the following list of features you feel you would be interested, leave a message to me on the Compuserve LDOS forum. If there is little interest, I can make arrangements individually with those people.

- Uses a "weighted" grade system with optional score adjustment for low curves
- accepts up to 42 students per class
- keeps up to 100 grade "items" per marking period
- archives grading period data to a central disk
- students can be excused from any item
- return to a grade item to update the raw scores
- teacher selectable printer output: summaries, title info., graphs, charts, scales
- select all/some of grade items to include in final calculation
- print student results in 1-grade book order, 2-rank order, 3-Alph. order
- request parent signature on special form for return to teacher (w/ check off)

- create student roster on check-off form
- add or remove students during a marking period
- create up to 10 different grading scales
- view individual scores of students by raw score, or percentages
- handle DOS functions for backup, etc. from program (automatic)
- alphabetize words/names ignoring Upper/lower case
- create a word search puzzle w/ answer key (& select difficulty)
- place class list on blank paper to paste in grade book (w/ line size adjust)
- print stu. names on mail labels to create seating chart
- print final grade summary to paste in grade book
- create/maintain seating charts (MODEL 4 ONLY)
- retrieve archived data to active status

I have versions available for Model I (SS/SD), Model III (SS/DD), Model 4/4P, and Model 4D. All are running under LDOS/LS-DOS (the latest version for each). The large manual includes over 40 pages plus pages of output examples.

Resurrecting "lost" files

Fm Alan H. Pesetsky: From time to time I ROUTE my printer to a file for long downloads of BBS stuff, CIS messages, or whatever. Sometimes I forget to reset the printer to close the file before shutting down or my computer does a (seemingly) mysterious freeze-up (when the moon is full and Mars in in conjunction). Anyway I'm left with a file with no RECORDS, no EOF, no date but does have a FILE SIZE and of course is inaccessible. Any way to make such file accessible?

Fm jeff brenton To Alan H. Pesetsky: quick'n'dirty - first, find out how many sectors are involved (look at the allocation information in the directory). Then, in BASIC:

```
10 OPEN "R",1,"PRINTER/FIL",256
20 PUT 1,200 '<- replace '200' with # of
  records in file + fudge factor
30 CLOSE 1
```

this should let you access the file, but there will definitely be some garbage at the end, and you probably lost up to 256 bytes in an unwritten sector.

Why not hierarchical directories?


Fm Michael Rogers: Why doesn't LDOS/LS-DOS support subdirectories? It would be a boon for hard disk or extended memory users. I can only assume it must be impossible or too much of a compromise or someone would have tried to sell it to us years ago. Then again, if crappy Apple PRO-DOS can have nested directories, why not TRSDOS etc?

Every time I use MS-DOS or AMIGA-DOS, I really appreciate the features of LS-DOS 6.3, but when I have to manage a RAM disk of a hundred or so files, I long for nesting my files in neat directories and subdirectories all to themselves. Could this be the last programming challenge?

Fm H. Brothers: An MS-DOS disk has a root directory, which is at a set location and has a set size, and subdirectories which, as you say, are essentially files. None of the directories, however, have anything like either a GAT or HIT associated with them (MS-DOS would be a lot faster with a HIT-type structures, but since CP/M didn't have one). There is a special section at the beginning of a disk called a File Allocation Table (or FAT) that performs some of the GAT's functions. Subdirectories are easy to implement under MS-DOS because the FAT and directories are separate structures. The directory structure of LS-DOS would make true subdirectories much more difficult to implement without some major surgery to the /SYS files.

Fm MISOSYS, Inc: Subdirectories would be feasible from a DOS implementation standpoint - with a lot of reorganization. Unfortunately, the concept of path names also imposes restraints as to what is practical. If you recollect that when MS-DOS moved to version 2.0 when Microsoft introduced hierarchical directories, it also incorporated a new set of file access DOS calls. Why? The primary reason was that a file open was built around a file control block of 32 bytes in length. L(S)DOS also has a 32-byte FCB. This is significant in that the file specification must fit into that buffer length. No software already written has a larger area set aside for the FCB (unless it already maintained a 50-byte FCB buffer for TRSDOS 1.3 requirements). You can't rely on that. When MS-DOS 2.0 was released, no software existing at that time could take advantage of the subdirectory structure, paths in

particular, because of the new file access calls which dealt with what Microsoft called handles. I believe that's why WORDSTAR fumbled the ball - they didn't take advantage of paths soon enough. The same would be true with L(S)DOS. Were a hierarchical directory structure to be implemented, it would require all existing software to be re-written to use the new feature. Who is going to do that? To make such a revision to the DOS for the hackers to play with would be economic suicide. Your best alternative is to get diskDISK - if you have not already done so.



The LSI Column

by William Schroeder

First off, the latest version of LS-DOS 6.3 is "LEVEL-L" with file dates of July 1, 1987. This is the same as a Level-K if you patched your system instead of sending in for an update. Updates are still available for just \$5.00 if you send us your master LSI disk, and replacement master disks are available for \$10.00 (to registered owners only).

Well, there has been a lot happening since I last wrote for TMQ, 80-Micro has announced that they will end coverage of eight-bit machines (the Mod-4), the stock market has fallen 25%, the Dollar has plummeted against all major currencies, Fall COMDEX has come and gone and LSI is announcing a "SITE" license for 6.3 customers with several or more Model-4 machines.

That's right.... An LS-DOS 6.3 "SITE" or "USER" type license is now available. For users with many machines we are providing this special license. The "SITE/USER" license grants the holder the right to use the copy of LS-DOS 6.3 provided with the license on as many machines as the holder wishes, provided that the holder has all machines on one site and/or is the legal owner of all machines involved. For example if a user has 5 machines in his office the office can buy just this one license and does not have to buy 5 copies, or if your company has 17 branches each using Model-4s your company need only buy this one license (provided all machines are legally

owned by the same entity). To allow this, LSI provides a special LS-DOS 6.3 that has had the serializing and ID systems specially altered, more or less turned off. There is no single machine restrictions or limitations included in the "SITE/USER" licensed product other than as stated above.

This new offering is available for \$99.95 directly from LSI only and will be available for shipping on December 15, 1987. The "SITE/USER" license package is available at reduced cost to all REGISTERED LS-DOS 6.3 users. A present user need only pay the price differential from the purchase of his single copy license. If you already paid LSI \$39.95 then this package is available to you for \$60 as an upgrade. Note: You must have purchased LS-DOS 6.3 from LSI directly and you must be registered with LSI or provide proof of purchase to qualify for this upgrade price.

Also, I should remind potential customers of this license that LSI DOES NOT ACCEPT PURCHASE ORDERS!! Prepayment only for this package, LSI pays the shipping and handling. You can call (303) 243-7070 for additional information.

We have a couple of things that seem to be the most recurring topics for our customer service department. The first of these is the SYSGEN or environment setup of LS-DOS as it pertains to updating from 6.2.x. When you update what Tandy refers to as a "PROGRAM" disk with the 6.3 operating system, the procedure turns off any existing SYSGEN that may have been present on the PROGRAM disk. This is as it should be, as the operating system has been changed. Now, however, your program may not function the same or may not run at all. The reasons are simple, the program expects the environment to be set in a certain way, and now it is not that way. Most Tandy applications manual do address the setting up of the environment for the program involved, somewhere in the manual. This information is often found buried in the section on running the package with a hard drive.

This problem often becomes apparent with printed output that does not seem to "page" correctly, like it did before (under 6.2). The cure would most likely be to simply install the printer filter and SYSGEN the system. This is done at the LS-DOS Ready prompt by entering the following commands:

```
SET *FF FORMS <ENTER>
FILTER *PR *FF <ENTER>
SYSGEN <ENTER>
```

Many other configurations can also be set up and "remembered" by SYSGEN, but you will have to know what they are. To find out what a particular program disk has in its configuration you must first boot that program disk so that the configuration will come into the machine. Then you must get out of the program itself and get to the DOS prompt. Now you would do a DEVICE command with the "B" parameter on. This will show you how the system is presently set up. Note the information shown by DEVICE and then create the same environment on the 6.3 version of that program disk and SYSGEN it. If this all sounds a bit confusing you should read your manual under DEVICE, FILTER, ROUTE, SYSTEM and SYSGEN.

There is a short cut for setting up 6.3 that will work most of the time (but not always). Simply boot a known working, 6.2 version of the program disk. Then remove the 6.2 version from your drive 0 and replace it with a 6.3 version of the same program disk that you have already prepared by following the update instructions for system disks. As the 6.2 version of the program has booted and set up the system, the program is ready to run. Now by doing a SYSGEN, the configuration will be saved to the 6.3 version that is now in drive 0. Reboot after this and try the program to see that all is well before you begin working with your real data.

Another common area of customer service involves floppy drive reliability problems. The older Model-4s and the 4Ps have some serious problems in this area. Our LS-DOS 6.3 master disks are produced very carefully on the best drives available and are monitored closely for quality and accuracy of production. Yet many users seem unable to read the master disk when it arrives. We replace these disks and get the supposed "bad" disk back only to find it in perfect condition. The disk drives and/or controller in their computer is at fault. OK... so what... we would advise all Model-4 users that, rather than repair or even re-align one of these older drives (Tandons) they should be replaced with double sided TEAC drives. These are available from AEROCOMP in Dallas for about \$120 and a ready to plug in; even for a non-hardware person the switch is a snap. By the way these are DOUBLE SIDED drives and will be usable in the MS-DOS world in the future and will provide you with about 340K of storage now on your Mod-4.

Another interesting question is "WHY does FORMAT stop and ask for a password even if I use the ABS parameter?" Simple, that is what it is designed to do. Think about it, the ABS (ABSolute) parameter will go ahead and format a disk without any stopping if and only if the password on the disk is "PASSWORD"; this is as it should be. If you have put a "real" password on a disk, you should have put it there for a reason; you have important data on that disk. Should some other user of your machine, or yourself for that matter, need a disk to format, you could inadvertently wipe out that important data by formatting the wrong disk. Disks that you are not that concerned about should have the default password of "PASSWORD"; these the ABS parameter will function against as expected. Those with an explicit password are protected from use of the ABS parameter. Also remember the ATTRIB command will allow you to change the master password on a disk as the data it contains becomes less or more important.

There have been a few reported problems with the size of the gap between sectors as they are created by the FORMAT command. This can only occur with a user that is updating to 6.3 from a 6.2.0 or older system. The gap is the same on 6.2.1 as it is in 6.3. A problem can result if the drives on the offending machine were set to a higher rotational speed to overcome the problem of the heartbeat interrupt colliding with an attempted sector write. The cure would be to set the drive speed down to about 301 RPM (I'm told that the spec is 300 RPM +/- 2%). This will cure the problem by keeping the track from overwriting itself. If you are unable to solve this problem in any other way, LSI will provide you with a patch to revert the format on your disks to the "old" pattern (note that this is not an official patch, nor is operating with the "old" pattern supported or recommended by LSI).

We keep getting asked when we are going to stop supporting LS-DOS and the Model-4. To be quite frank about it, all of our future plans and products do not include Model-4 versions. I had just written several paragraphs about what LSI is doing and will be doing and then deleted them. I realized that this information is of little interest to Model-4 users. Let's just say that LSI will be entering a different software arena. We will not be offering products to the Tandy market place in a retail manner. We really don't know how long we can justify supporting the Model-4. Please don't expect us to continue much longer. If Tandy

elects to drop the Model-4 completely, we will then stop supporting it. Regardless what Tandy does, we probably will stop Model-4 support sometime in 1988. We hear groans whenever we say this but we will not lie about it, it just is not in the numbers. The cash flow has to be at a certain level to be profitable, and it's falling fast. If 1000 users were to each be willing to pay \$50 per year we could continue to keep a Model-4 customer service operation going. This \$50,000 would represent salary, equipment, stationary, printing, postage, maintenance, office supplies, insurance, etc. This is the minimum revenue required to operate a ONE person department. We can not believe from our experiences that the Model-4 users are ready to back up their requests with the needed funding. I could be wrong about this; if enough users are willing to join we would continue. Let us know what you think.

Let's not get the misconception that LSI is deserting the Model-4 users. We are committed to continue supporting LS-DOS 6.3 as long as the revenue justifies the effort. We are not Tandy, we are a small company with very limited resources, we do our best.

Some customers have sent or called in "Wish Lists" for 6.4 or 7.0, and a lot of our users have asked when and if a whole new LS-DOS will be available. I want to clarify this... We have NO PLANS WHAT SO EVER for a new version of LS-DOS. There will only be patches (if needed) to correct coding errors in 6.3. At this point we will not even consider patches that can be construed as design changes or enhancements. This could change if Tandy would decide to continue development and production of a MOD-4 like machine, but that does not seem likely.

Sometimes we are asked "Who exactly supports 6.3, LSI or TANDY?" The answer is simple... BOTH. As a general guideline, if you purchased your 6.3 from Tandy, go to them first; if you purchased from LSI, come to us first. However you can always request support from either of our companies at any time; the choice is yours. But please note that if you received your LS-DOS 6.3 from Tandy you will have to provide LSI with proof of purchase and your customer service number or we will have to ignore your request for assistance.

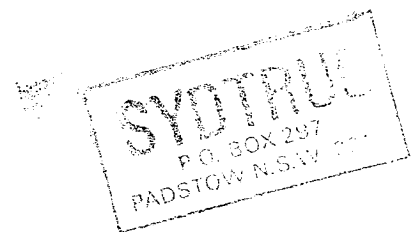
With the passing of 80-MICRO as the only other (TMQ is still with us) magazine left for Model-4 users, I was relieved. My personal opinion is that with the exception of the ads and a handful of articles over the last year

80-MICRO has been a worthless, inaccurate, boring, self serving, self defeating publication that was not worthy of continuing. After carefully going through PC-Resource, I don't think a whole lot more of that magazine. I believe you get a far better value from PC-World, PC Magazine, PC-Week, PC-Tech Journal, Infoworld, and Dr. Dobbs.

Speaking of places to get reliable information, check out the LDOS Forum on Compuserve. This SIG has become a lot more than just an LDOS board. There is plenty of dialog on the MS-DOS world also, with one of the best and most active SYSOPs on any SIG, Joe (jkdd). There are no requirements to become a member of this SIG (other than being a Compuserve subscriber), so just do a "GO LDOS" from any main prompt and sign in. Ask most any question about hardware or software of any type... You will be amazed.

A ways back I commented on the some of the PC Clones that are available. I said that I had a PCs LIMITED 286 and was well pleased with it. That was before the problems began and I came to find out how well (or poorly) PCs LIMITED is to work with. Well, I no longer use a PCs LIMITED machine. I was advised by Roy Soltoff before I got that machine to get an AST PREMIUM/286. I wish I would have listened in the first place. I am now very happy with my AST PREMIUM/286, 4 megs of FASTram, EVERCOMM 2400 baud modem card, EVEREX Microenhancer all mode video card, and MAGNAVOX multimode monitor. I have also added a 3.5" drive to my system and a EPSON LQ-2500 color printer. I now have everything I want in a system. The AST is super fast in all areas. If you are moving up to a 286 box... pay the extra and get the best, the AST is really a quality product. If you don't need the power of a fast 286 box, I would still favor the Turbo Clones from CLONE computers in Dallas. Most of you already know these folks as the owners of Aerocomp and Montezuma Micro. They are legit and they build a good machine at a very competitive price.

I have ordered a Tandy 1400LT portable and will mention how that seems in the next issue.



MS-DOS

Information

Migrating to MS-DOS

Fm MISOSYS, Inc: Most, if not all, of TMQ readers have been working and operating in the comfortable environment of LDOS, TRSDOS, and LS-DOS, with anything from a Model I TRS-80 to a Model 4 TRS-80. Prior to your first computer, you may have been playing with an "advanced calculator"; I know I was. I went through the technological advances with many acquisitions - each new acquisition replacing some outmoded piece of equipment. From the Bowmar BRAIN, to the Texas Instruments SR10, SR51, and SR52, I advanced to the Model I. Subsequently came the Model III, then 4.

Technology never stands still; nor does the needs and wants of those using this changing technology. The MS-DOS world represents larger addressable memory, faster machines (when you advance to 286 machines and beyond), and a greater selection of application software and programming tools; there just happens to be millions of installations rather than thousands of installations thereby providing a greater potential for "young" companies to grow and prosper.

Although there will always be folks resistant to change, and others fully satisfied with the capabilities of their current hardware and software, many of our readers are acquiring what I will call "PC CLONES". The brief history of the Intel 80x86 CPU world revealed that those machines which were not 100% compatible with IBM's machines soon found themselves, and their manufacturers, out of business, or altered to work just like IBM's. Thus, although we may generically say MS-DOS machines, what we really mean are machines which behave operationally as if they were PCs, XTs, and ATs (the trademarks of machine

classifications sold by IBM). An earlier terminology applied to competitive machines was "compatible". Even that proved to be insufficient. As many have learned, the operating system (and BIOS) interface to peripherals, specifically video I/O) proved so slow for reasonable programs, that most developers wrote their code to directly address the hardware of an IBM machine. Since a preponderance of software was tied to hardware specifics, users of those machines not quite 100% compatible to an IBM machine experienced frustrations with this piece of software or that piece of software. "Near compatibles" floundered and "clones" were born.

One of the great benefits of PC standardization to the consumer was that competition among manufacturers became keen and prices fell. Not to mention that the manufacturing process was able to benefit from volume production levels. The person now looking at acquiring a state of the art CLONE, has many choices which may appear bewildering. To assist those newly migrating to the world of MS-DOS (and others currently there who wish to expand), we expect to re-print some of the dialogue from our Compuserve forum and letters we receive which relate to what folks are using in the MS-DOS world and how they feel about their choices. Reviews you can find elsewhere: I want folksy chat that's helpful to your brethren. To begin a dialogue on this topic, let me relate what we here at MISOSYS are using.

Let me first state that we still have three model 4s and use them for various tasks - although not as primary machines. My machine is a standard IBM PC which we acquired from LSI. We expanded it to a 150 watt power supply, two half height floppies, one ST225 20 Meg hard drive, 640K of memory, and drive a standard MDA (monochrome) video. Because this machine was ungodly slow compared to our other machine (an AST Premium/286), we added an AST Hot-shot/286 speedup card. That made it almost as fast as our Premium/286. I also recently added a Logitech C7 mouse.

We acquired the AST Premium/286 (10MHz, 0 wait state) for use with invoicing, data base management, and eventually financials (ledger, payroll, etc.). Brenda uses that machine. It has 2Meg of memory and a 40Meg 28ms average seek rate drive. The drive is what sets that machine apart from my souped up PC.

We will soon be acquiring a laser printer to assist in the preparation of THE MISOSYS

QUARTERLY. I have narrowed the choice to the AST Turbolaser/PS or the NEC LC890. Both have exceptional specs; the AST is available for about \$400 less. You all will know by the next issue what was our final selection.

For software, we are using Microsoft's Quick-BASIC (a must for anybody and everybody - throw away GWBASIC), Microsoft WORD for word processing, Microsoft C 4.0 (soon to get 5.0), Microsoft FORTRAN 4.1 (to use with our RATFOR-86), the MACE utilities (for a bunch of things), and HQ for desktop management. We were so impressed with HQ, that we decided to take it on as a MISOSYS product. Elsewhere in this issue you will find a data sheet which discusses the product. We also recently acquired MASM 5.0 to use in comparing the next release of ED/ASM-86 which Phil Oliver has been working on for the past six months. We, of course, also make extensive use of LB86, our own data base manager, and FM86 for file management (the MS-DOS file management tools are pitiful). Our DED86 disk/file/memory editor has also saved the day a few times - it is the kind of tool that most folks should have on hand as insurance. We also will be acquiring an accounting package to replace the aging ledger we still run on a Model III. Now here is what others have to say.

Fm Ken Kane: I was at Software City yesterday looking for word processors and mentioned that I was interested in buying an AST machine via mail order. The salesman replied that his store was the local AST dealer, quoted me a price \$700+ higher than mail order for the AST Model 140. He then disparaged the mail order houses saying that they "strip down the machine and take out the AST parts" replacing them with inferior components. Is this a common practice? How to prevent against it? What to watch out for?

Fm MISOSYS, Inc: Check the ads carefully. If it is advertised as an "AST Premium/286 Model 140", then it has to be stock AST. If it is advertised as "AST Premium/286 40Meg EGA", then it is NOT going to be stock. Another giveaway is when they show a 40Meg with 512K. That notes it as a Model 80 with local upgrade; the Model 90 and Model 140 and up all have 1 Meg. I popped in another 1 Meg of chips from Microprocessors Unlimited for about \$165 (100ns chips). Back in the old Model I days, 1 Meg of RAM would have cost me about \$7000! Boy have times changed. And that wouldn't have been at Tandy prices of \$300 for 16K.

Fm --jjkd-- To Ken Kane: Yes and no. If they advertise an AST Model 140, it had better come with whatever AST puts in their Model 140. If they sell you a different model that doesn't come with whatever add-ons, then what gets put in is up to them as far as you let them go. I'd sure specify.

Fm Ken Kane To MISOSYS, Inc: That's a very useful caveat. One more question... will you please provide me with the phone number for PC Pros? I can't find a copy of Infoworld anywhere in Dayton OH. A clerk at our largest newsstand told me she thought they had folded. But I saw an ad for Infoworld in PCMAG or PCWORLD recently. Can't find an ad for PC Pros anywhere either.

Fm MISOSYS, Inc: PC Pros # in CHI is 312-810-1010. Actually in Darien, IL.

Fm H. Brothers To Ken Kane: There's nothing necessarily bad about mixing parts. I bought my AST from a local dealer who buys base models from AST and upgrades them as his customers want. My 80 Meg hard drive, second floppy drive, 1.5 Megs of memory, I/O board, graphics board, and 287 chip are all dealer-installed and non-AST. The only problem I had was related to the 287. When I first got the computer, every time a program tried to access the 287 the system would hang. I took the machine back and we swapped in a new chip -- same symptoms. Then we tried both 287's on another AST -- everything worked fine. AST wanted the computer back for 2-3 weeks to work on the problem. My dealer said, "Aw, sh*t" and simply swapped motherboards with another computer, so I was on my way the same day.

To H. Brothers: Thanks for chipping in with your experience. If I go for an MS-DOS machine it won't be because I need more power, speed or convenience than my ol' B&W Model 4. With 1/2 meg of RAM and a hard drive I have more than I need. Plenty more for me to learn in the LS-DOS world, too. And that's really what I use computers for... for the fun of learning and using structures. Still, I'll probably move into the MS-DOS world so that I can apply what I learn in a larger circle.

Fm H. Brothers To Ken Kane: I got started with MS-DOS because a client asked me to. It took me a long time to get over the limitations of the DOS; I kept thinking that everything would be easier on a Mod4. But the speed and power of this AST machine, especially with its fast hard drive and more-than-EGA card, are slowly winning my attention away from the Mod4.

Fm Ken Kane To H. Brothers: Can you recommend an EGA monitor to go with that AST more-than-EGA card? Do you leave your new AST powered up all the time? I have been leaving my Model 4 and R-S hard drive powered up months on end, on advice from somewhere on CIS.

Fm MISOSYS, Inc: We turn our AST on in the morning and turn it off when we close shop. Until I get an UPS on it, we also turn it off during thunderstorms. The drive in ours is an automatic shutdown; a 40Meg Micropolis.

Fm H. Brothers To Ken Kane: I use the Sony MultiScan monitor. I've run it next to the NEC MultiSync, and liked it better -- clearer print, cleaner colors, etc. I have a close friend with an NEC, and he looks at my Sony with envy and longing. On the other hand, InfoWorld's reviewers rated the Sony below the NEC monitor. I wouldn't want to use the Sony with normal CGA video generation -- the scan lines are too obvious and prominent. But it looks great with the "double-scan CGA" mode on my video board.

I turn the AST off at night but leave it on during the day -- the same as I do with my other computers. I've heard the arguments on both sides and so I take a middle road between them. No particular reason, actually, but the initial AST coldboot, when it does a thorough test of 2 MB of memory, is almost as slow as the bootup cycle of a Tandy 1000A, and much too long for someone like me who is used to an almost instant boot (my Zenith 150 boots without destroying memory, which has been invaluable for some development I've done). The AST warm boot is MUCH faster.

Fm MISOSYS, Inc: My AST doesn't seem to cold boot too slow. I also have 2 Meg. Perhaps I'll time the process and report back. I have the area above 640 configured as EMS with their driver. I then have the EMS split between a 16K spool buffer and the rest a BIG RAMdisk useful for the temporary drive when we re-sort our data base. I even have MACE's hard drive restore utility as part off the AUTOEXEC. But then, if you expect instantaneous and you get 10 seconds, that's enough to drive you up the wall. My IBM PC with 640K takes forever!

Fm H. Brothers To MISOSYS, Inc: I agree that it's all relative. I have my machine set up in much the same way. It's those slow numbers marching by while the memory above 640K is being checked which "feels" slow. As I said earlier, I was spoiled by my Zenith 150, which boots almost instantaneously since it doesn't

do a byte-by-byte memory check. Relative to other computers, the AST's bootup time is quite reasonable. Relative to my impatience, it seems to take forever.

Fm MISOSYS, Inc: Okay, I just cold booted my AST. It took 35.93 seconds. That included the time it took RXBAK to redo its directory files. A warm boot is faster.

Fm H. Brothers To MISOSYS, Inc: After a lot of thought, I decided that boot-up time is the wrong time for RXBAK. It makes more sense to me to run it at the end of a session instead of the beginning, so I run it every time I shut the system down and often after I've finished doing work on a major project. That saves a little bootup time, but once REMM finishes its memory check, everything else seems to run like greased lightning.

Fm MISOSYS, Inc: Putting it in the AUTOEXEC.BAT file on Brenda's machine (SHE has the AST!!!) makes it easy on her; she doesn't have to remember too much computer operation. Most of what she does is BAT-file driven. I've made use of the "QUERY" ditty to get her machine into different invoicing jobs. But yes, it does seem to make more sense to run the RXBAK after you have done a bunch of file activity. Of course, the trick is in remembering to do it. Some things work better when we don't have to remember to do it but rather when they are done automatically.

Fm Ken Kane To H. Brothers: The only ad with specifications says "CPD-1302 (PGA)", etc. About all I know about monitors beyond Langley-St. Clair, I learned from an article in PC Resource, and I didn't see the familiar "EGA" in the ad. Looking again at the PCR article... does PGA relate to PGC Professional Graphics Controller? And can the AST model 140 graphics card make use of the 900 H x 580 V display? Can common software packages access that resolution? Etc.

As you can see, my level of inquiry is ground floor or 1 step up. Just need reassurance that the Sony can utilize EGA mode; and I have that from your recommendation. Still, that Sony monitor is pretty steep.

Fm Ken Kane To --jjkd--: It looks like the new 101 key keyboard has made word processing harder for all word processors. Can you retrofit an old-style keyboard, (10 function keys on the left), to an AST machine?

Fm --jjkd-- To Ken Kane: Most "good" BIOS ROMs in ATs can handle either the old or new key-boards on a plug and chug basis. You'd have to ask AST to confirm if theirs fits into this category, but I would tend to think so.

Somebody recently mentioned that the keyboard supplied with NCR clones is a design with function keys on the side and more on the top, and is otherwise like the old AT in layout. Sounds like a winner if it is available separately and has a reasonable feel to it.

Fm H. Brothers To Ken Kane: The PGA generates analog output signals, while the EGA is digital through 6 color outputs. The MultiScan can work with both of them. I'm not positive about AST's board, since I use a Genoa video board in my AST computer. Maybe Roy can tell you. The nicest thing about the advanced modes for me is editing and programming with 43 to 65 lines on the screen. Not all programs can use the advanced modes, however. They have to be flexible enough to allow the user to set the screen size.

And yes, the Sony works very well in EGA mode (and monochrome, Hercules, and even CGA, although [did I say this before?] in CGA mode the dark lines between dot scans are too prominent. That's rarely a problem, since most "serious" programs that use CGA will work in the EGA's 8x14 mode without any trouble. [8x14 is the character dot box, not the screen size]

Fm Ken Kane To H. Brothers: Thanks for shedding light on the monitor scene. That analog mode, does that suggest that the Sony Multiscan is usable with the new IBM video standard? I could console myself about the high price if I know that I am buying future compatibility.

Fm Ken Kane To MISOSYS, Inc: Please comment whether standard AST video card can utilize PGA mode on the Sony Multiscan monitor. And do you recommend any particular color monitor?

Fm MISOSYS, Inc: The AST 3G+ card does not support PGA. It supports monochrome graphics adapter (MDA), color graphics adapter (CGA), enhanced color graphics adapter (EGA), and Hercules graphic card (HGC). It supports 640x350 in EGA, 640x200 and 320x200 in CGA, and 9x14 pixel in MDA. Handles up to 43 lines of 80 characters in HGC. Has 720x348 in HGC. I don't use an EGA monitor. I have an old CGA. I also have the Tandy 2000 hires color monitor which is compatible with nothing. So I can't recommend any EGA monitor.

Fm --jjkd-- To Ken Kane: The AST 3G card is an EGA card. The PGA mode is analog, 640x480 with 256 colors. There are some combo EGA/PGA cards, but they are in the \$1000 class. Some EGAs can go 640x480 digital, with 16 colors only, but it ain't the same thing.

For monitor, I'd go with the Sony. A bit pricey at around \$600 - \$650 discounted, but an excellent monitor. The NEC will run you about \$75 to \$100 less, but I think that the Sony is worth the price difference. The Sony will work on the PS/2 products, given the proper cable.

Fm Ken Kane To --jjkd--: Thanks for the video education. With your timely encouragement, I decided to buy the Sony to go with a new AST 140. Gawd, it's going to take a long time to make a new machine fit as well as this Model 4. But then, I never did get along with this clunky keyboard, etc.

Fm Ken Kane To MISOSYS, Inc: You may be interested to know that PC Pro's price for the AST Model 140 is \$ 2425 + \$ 38 S&H to Dayton, OH. I ordered one this afternoon along with a Sony CPD-1302 for \$ 599 + \$22 S&H. I appreciate your help in making this selection.

Fm MISOSYS, Inc: I am sure you will be most pleased. The machine will probably be all set up ready to plug in and boot. They put the DOS over on the hard drive. Depending on where you live, you may need to adjust the time on the hardware clock using the SETUP utility.

Fm Ken Kane To --jjkd--: To purchase DOS 3.3 for the AST 140, do I buy PC-DOS 3.3 sold by the local IBM outlet, or is there a more generic MS-DOS 3.3 that I should look for? I.e. do PC-DOS's work on AT clones in general and on the AST in particular?

Fm Ken Kane To MISOSYS, Inc: I am expecting to make a gradual transfer from the Model 4 to the MS-DOS world. (I have a B&W screen 4 at home and a 4P at work). It will take some time, (at a period when work demands are calling insistently), to learn about PATHs, JOINs and the joys and perils of multi-dimensional directories.

My reflex is to install the latest DOS before I go to a lot of effort to transfer files to the AST hard drive. I am confused about DOS 3.3 as reviewed by John Harrell in PC Resource Sept. '87. Would one expect PC-DOS 3.3 to run in a clone such as AST? Is a generic MS-DOS 3.3 to be expected, or already available? I

see that Harrell's review of DOS 3.2 in PCR June '87 listed three versions: PC-DOS, MS-DOS and (Tandy)-DOS and that the features varied a bit between the three. My main concern...what can I expect to RUN on the AST? I realize that a firm answer to this kind of question requires in-depth testing, etc., all I would like is your best guess.

Fm MISOSYS, Inc: I would phrase that question to AST. Good clones can run PC-DOS. But Microsoft should be releasing a version of MS-DOS equivalent to PC-DOS 3.3. Now I haven't had the time to get on the horn to AST to test their waters as to their upward support of newer DOS versions. But that is the path I would make - and the path I will make soon.

Fm LSI(Bill) To Ken Kane: The standard MS-DOS 3.2.1 will work perfectly in your AST and give you every needed feature to support 99.99% of all present languages and applications. For now STAY AWAY FROM version 3.3. Let 3.3.0 settle down to maybe 3.3.2 or so.

Fm --jjkd-- To Ken Kane: You could get a copy of "real" IBM DOS 3.3 and it should run as well on the AST as it does on anything else. Note the way that was carefully phrased. There are a number of creepie-crawlies in there. Interesting that this is the first DOS upgrade (from DOS 3.21 to DOS 3.3) that did not involve MicroSoft, word is that IBM did it on their own.

Personally, I'd stick with a copy of IBM DOS 3.21 or the latest version of MS-DOS (3.2 something, there is no MS-DOS 3.3 yet). I've heard that the latest Tandy version of MS-DOS 3.2 has some nifty added features, like a free disk defragger and a faster backup program.

Fm Ken Kane To --jjkd--: Thanks, just learning my way around. PC Resource had two articles by John Harrell that are informative about 3.2 and 3.3 But he didn't explain about the lack of an MS version of 3.3. Plenty to keep me busy with 3.2.1.

Fm Ken Kane To MISOSYS, Inc: Plenty to do with the MS-DOS 3.2.1 provided. I am very pleased with the manuals that come with the AST, (and obviously with the machine itself). Well, the Time of Day was pretty far off, so I had to learn about SETUP. Impressive display of impressive diagnostics.

Fm MISOSYS, Inc: SETUP is fairly easy to deal with. I had my machine opened up once, that

was to play with trying out different monitors. I said, "Well, looks like a machine!".

Fm LSI(Bill) To Ken Kane: I have also joined the AST-140 group (at ROY's prodding). Only had it one day but it is great so far and lots faster then the PCs-LIMITED it replaced. I had an EGA monitor (MAGNAVOX MULTI-SCAN) and am using same on the 140's 3G board, works great (in all modes). I looked at all the other multi-freq monitors and I personally liked the MAG the best, I even bought another one. Cost for the MAG is \$499 from Aerocomp. I've also added a 3.5" 720k drive and an ST-251 40meg as a backup device; the AST-140 swallowed each just fine.

Fm Ken Kane To LSI(Bill): Welcome to the club. Should we start an AST forum on CIS or just keep on meeting here on LDOS? This past evening I followed Roy's lead and plugged in the 36 256K RAM chips to bring 'er to 2 meg.

I still use my Model 4 for most things and will continue so for at least a few more weeks. As a newcomer to MS-DOS I don't own a scrap of software. Choosing a word processor is at least as important as picking a machine, I believe. I am inclined toward Microsoft Word but have a demo disk for Lotus Manuscript on order. I am resisting the urge to buy utilities and TSR's while I get a feel for what the bare MS-DOS can do. The DIRectory system really makes me regret leaving LS-DOS.

Fm --jjkd-- To Ken Kane: You don't want Lotus Manuscript unless the majority of what you do involves printing out math equations. WordPerfect and Microsoft Word are the two heavy hitters to choose from (I use WordPerfect myself), though XYwrite is probably survivable.

Fm Ken Kane To --jjkd--: Thanks for the steer. Does the current version of WordPerfect, (I believe it's 4.2), format automatically, or do you still have to go through a separate reformat procedure to put things in place on the screen? How good is the WYSIWYG. It is my impression that MS Word formats automatically as soon as you change margins, make an insertion, etc. Someone please correct me if I am wrong.

Fm H. Brothers To Ken Kane: WordPerfect 4.2 does the reformatting automatically and almost instantly on my AST. About the only thing that isn't WYSIWYG is headers and footers, but you can preview a document to see those if you want. It seems fast on the AST -- but then, almost everything seems fast on this machine.

Fm Ken Kane To H. Brothers: I just got the Manuscript evaluation package from Lotus... 4 floppies for \$10, very impressive. It permits importing graphics from 1-2-3 and Freelance and will print mixed graphics and text. It reminds me of ALLWRITE. The editor is completely separate from the format/printing program. The previewer is, I believe, a third program. The editor apparently keeps the text completely in RAM, so navigation is instantaneous, (nice!). The downside of Manuscript is that the preview module, which shows you where the graphics end up, page breaks etc., is slow... about four seconds for EACH page to set up. Longer if graphics etc. are being merged. The formatted page occupies half the screen and reminds me of those text preview modes provided by some of the Model 100 word processors... shriveled text! But you can quickly zoom in and read details if you want.

Since you JUST HAPPEN to have my EXACT graphics setup... can you use EGA color mode and get 43-line display? A salesman told me that WordPerfect supports 43-line but that you have to go Hercules to use it. I believe that Hercules means B&W, nict. wahr? (Spent last evening at the local Oktoberfest).

Fm H. Brothers To Ken Kane: WP 4.2 has no problem with 43 line EGA editing -- with colors, if you wish (I keep color to a minimum with word processors, but that is a personal choice). In fact, it has no problem with 60-line editing either, or anything else your video card can produce. The same goes for screen width -- I've used it over 100 columns wide with no problem. However, WP doesn't manipulate the video card to produce the screen setting. It expects to find the screen properly configured when it begins. That's really no problem -- I run a program called 43.COM as the first part of my WP batch file. The only real problem occurs if you Shell from WP to DOS, run a program that resets the screen to 25 lines, and then 'Exit' back into WP. It doesn't know the screen has been changed, so its prompt line is suddenly invisible. However, if you remember to reset 43-line mode before returning to WP, everything is fine.

Fm Ken Kane To H. Brothers: I gather that you can get in and out of this without changing the switches on the back of the EGA-Plus card? And I would have thought that you would have to take the cover off the AST to change the little switch over the RESET button to use monochrome.

Fm H. Brothers To Ken Kane: 43-line mode has nothing to do with switches or hardware. It is simply a manipulation of the capabilities built into an EGA card. Nor does it have anything to do with color or monochrome display. The EGA is a marvelously complicated little box and can do all sorts of neat tricks.

Fm Ken Kane To H. Brothers: I notice in the blurb I got from Quarterdeck re DESQview that they provide switching into 43-line mode etc. and back. DESQview is \$79 mail order and provides access to all that upper RAM via paged memory. Think I'll give it a try. Looks as if it provides Sidekick TSR features as well, so it really is quite a bargain. Also a communications pgm with XMODEM. (I'm still communicating on my model 4).

Fm LSI(Bill) To Ken Kane: My vote has to go to Word Perfect also, I've had experience with it and WORD and WP wins... (for me anyway). For comm work get PROCOMM (it's public domain). Hundreds of public domain software items are available for the MSDOS world many of them on CIS. You can look at these by doing a <GO IBMSW> at most any CS menu. If you get WP get the WP LIBRARY with it (well worth the extra \$50-60). For a modem I have both the EVEREX-1200 and the 2400, both work perfectly and are completely HAYES compatible, the 1200 is \$89 and the 2400 is \$189 (I recommend the 2400 within a year most services will be offering 2400 baud connects, be prepared). There is a DOS SHELL called "QDOS II", I love it, makes the long winded directory structures of MSDOS easy to understand and a snap to handle. QDOS comes from GAZZELE software in Utah for \$69 (I'd pay much more). I also like and use, QBASIC and both the "C" compiler and the assembler from Microsoft (took Roy advice the first time on these last two, glad I did). For an editor I am using "POINT", comes free with the LOGITECH C7 PLUS package (that's a 3 button serial mouse). I have also added the lmeg of extra RAM to my 140 and use it for a 64k spooler and a 1.35meg ran drive.

Fm Ken Kane To LSI(Bill): That about does it, I'll pick up WordPerfect. Two notes: 1) Over on the WPSG sig I learned that there is a release of WP 4.2 dated Sept. 9 which navigates faster through the document during editing. Claim top-to-bottom 25-30% faster, bottom-to-top 75-80% faster. I don't know how much of a problem this was in the past, but there it is. WP charges \$15 for the upgrade. 2) The latest PC Magazine has a collection of favorite macros for WP.

Fm MISOSYS, Inc: One of the things that gives a program "speed" in accomplishing a given task is the "apparent" speed of the operation. Microsoft WORD had the best speed in top-to-bottom, and bottom-to-top because they maintain those blocks in their buffers (sort of like a cache). WP finally saw the light.

Fm LSI(Bill) To Ken Kane: On your AST-140, you won't get your hand off a key before ANY function in WP has completed your request (unless you have over a 300k file in). Either version will work at a very pleasant clip.

Fm LSI(Bill) To H. Brothers: But it all depends on your uses, your style, whether you want to use a mouse, and how much you want to spend. BRIEF has a hundred "bells and whistles" of which I personally would find use for but a few. Point is nifty and simple, pretty fool proof, well debugged and very reasonably priced. With editors it's each to his own.

Fm H. Brothers To Bryan Headley: Version 2 of BRIEF is (1) faster, (2) easier to install, (3) has much better docs, (4) is much faster, (5) allows easy keystroke redefinition, (5) explains writing macros very clearly, (6) is more than worth the upgrade price.

Fm H. Brothers To LSI(Bill): Using the Logitech mouse with Brief is no problem. The automatic compilation inside the editor, with the cursor moving to each error line (and support for all compilers except those idiotic things from Borland that don't allow external compilation) is worth the price of Brief for me. The rest -- and there is a lot -- is all gravy (auto restore of a previous editing session, for example). But as you said, with editors it is definitely a matter of taste.

Fm Bryan Headley To LSI(Bill): Yeah, but adding mouse support is trivial. There was a cutie mouse driver in one of the "PC" magazines that converted mouse movement to arrow keys. That works fine under WordPerfect and Brief. Still, agree with your parting comment - it's all a matter of what you are used to and what you like. Although those that prefer EDLIN get my disdain!

Fm --jjkd-- To Ken Kane: WP's on-screen presentation does not have the ability to show different fonts as fonts (point size, italics, etc.). Word will do this, but at the expense of running in graphics mode (slow!). All reformatting is automatic/dynamic, for both lines of text and pages. No "repagination" like Model II SCRIPSIT or MultiMate.

Fm --jjkd-- To Ken Kane: I use DESQview very heavily, and find it very useful. I'd say my AT is up under DESQview 95% of the time. Note that I think you are merging together two different products there: 1) DESQview 2.0, \$129 list, \$80 discounted 2) DESQview Companions, \$100 list, \$?? discounted, \$75 direct from Quarterdeck. The second product is what gets you the sidekick-like stuff and the communications program, they aren't included with regular DESQview. Make sure that you get DESQview 2.0, it has major improvements over the 1.x products, and I seem to recall the upgrade as being expensive compared to the discounted price of the product.

Fm Ken Kane To --jjkd--: Got them both on order, \$75 each. Sounds as if The Companions provide most of the Sidekick TSR-type features. (But it probably doesn't include the Export-to-DOS that I have grown used to in PRO-WAM. That feature is hard to come by. Have you found a way to do that?

Fm LSI To Ken Kane: WP will not function very great as an "EDITOR" per-say. It can save in a sort of ASCII format but you really will need a straight forward editor also. A simple text editor is built into QDOS by the way. Their are also several simple editors available on the IBMSW SIG. They are also (\$00.00) other then download time.

Fm --jjkd-- To Ken Kane: DESQview has a built in screen mark and transfer function, along with a reasonable key re-definer. They'll talk to just about any reasonable program.

Fm LSI(Bill) To --jjkd--: The tech at PC PROS said that I should only use 100ns rams on my fast cards, I told him I was using 80ns. He seemed to think this might cause some problems. I have always been under the assumption that in general you can always use a ram chip that is one step faster then the speed specified (ie. 120 vs 150 when 150s are spec'd). Is there any reason that a problem could occur when doing this. I'm going to call AST and see what they have to say, just for fun.

I ordered DESQview yesterday and another 2meg FAST RAM card. I also had them send me a DESQview companion (whatever that is). I've got yet another 2meg AT EEMS ram card (but it's not FAST RAM) should 4.3 meg of EEMS fast ram create a pleasant DESQview set-up or should I through in another couple of meg.

Fm LSI(Bill) To Ken Kane: As long as you have WP you might as well get the "WP LIBRARY" it

HQ[®]

FEATURES & SPECIFICATIONS

HQ makes you more productive.

Whatever you use a personal computer for, this powerful memory-resident software makes you more productive by:

- providing fast, easy-to-use tools for organizing your time and work;
- making programs you run easier to work with and more useful;
- simplifying the DOS operations required to manage your PC.

HQ includes eight modules: Appointment Book, Calculator, Disk Manager, Lock Up, Note Pad, Organizer, Phone Book, and QuickKey.

Plus, an HQ:Configure module allows you to choose the colors (for a color monitor) and position for any HQ window.

You can set up HQ to provide the modules that fit both your needs and the available memory in your computer. When set up to use at least 92K bytes of memory, HQ can run any of its modules.

HQ resides in memory with whatever other program you may be using. When a special key combination is pressed, <Shift> and <Alt> plus the letter of the module you want, HQ puts your main program on hold and takes over.

HQ calls in additional information, if required, from disk. It decides how to use memory most efficiently to minimize disk access.

When you are through using HQ, you can return to the software you were using, exactly where you left off.

HQ Main Menu

Hold Shift and Alt keys
Press key for desired function

* Appointment Book	Set appointments, use calendar
Calculator	Do calculations, with tape and memory
0123456789=	Retrieve numbers from Calculator
Disk Manager	Access files and DOS commands
\	Retrieve filename from Disk Manager
Lock Up	Lock up your computer
Note Pad	Type notes and short text files
Organizer	Organize projects, to-do lists, etc.
Phone Book	Reference phone book, dial numbers
* QuickKey	Define and execute keyboard macros
* HQ:Configure	Configure windows, clock and printer
- +	Extract/Recall screen information
Esc	Quit all modules
F1 - Help	Help with HQ

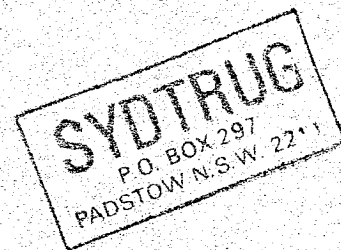
Screen Transfers

HQ has a powerful data transfer capability. You can transfer information displayed in any application program or HQ module to another application program or HQ module.

HQ allows you to re-format information when you transfer it. You can designate up to four characters to be played out at the end of each line as it is recalled.

Help System

Instant reference screens along with a full context-sensitive system of help screens support each module. General HQ help is also available as well as Help for each error message.



Appointment Book

With the Appointment Book, you can schedule and review your appointments easily. An unlimited number of appointment books can be maintained.

You can move quickly to any date on the continuous calendar display. Alongside is a page of appointments for any day you wish to view.

Commands in the Appointment Book allow you to:

- Return directly to today's date.
- Mark important dates.
- Print appointments for a day or a range of weeks.
- Insert an appointment for any time.
- Edit, move, or erase appointments.
- Restore an erased appointment.
- Find a string of characters in the current month's pages, e.g. a name.
- Set alarms to remind you of appointments.
- Change the time interval displayed between appointments.
- Select different appointment books.

Calculator

The Calculator lets you perform 12 digit calculations including addition, subtraction, multiplication, division, sub-totals, and percent.

For checking calculations, a tape display records all operations. Ten memory cells are available. The Calculator runs in either algebraic or RPN mode.

In the Calculator, you can:

- Fix the number of displayed decimal places.
- Set the mode to Algebraic or RPN.
- Erase tape or memory cells.
- Print tape or memory cells.

Numbers in the display register and memory cells can be transferred directly to other HQ modules or applications program.

All current numbers are retained when you leave the Calculator and are there when you return.

Disk Manager

The Disk Manager gives you fast and easy access to DOS operations.

You can list any directory and can move quickly to a filename on a directory list by typing the first few letters of the name.

A filename can be transferred directly to another program.

The Disk Manager also allows you to:

- Change the default directory for a drive.
- Make or remove directories.
- Sort a directory listing by name, extension, size, or date.
- Print the displayed directory as sorted.
- Display information on the number of disk files and disk space.
- Copy, rename, or erase files.
- Display a file's contents on the screen.
- Locate a file anywhere on a disk.
- Format a floppy diskette.

Lock Up

Lock Up allows you to lock your PC with a password of your choice, preventing viewing or entry of information while you are away.

You can secure your work in a program at any stage and return to it exactly where you left off.

Appointment Book						
S	M	T	W	T	F	S
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30						
Dec	1	2	3	4	5	6
1986	7	8	9	10	11	12
	14	15	16	17	18	19
	21	22	23	24	25	26
	28	29	30	31		
Jan				1	2	3
1987	4	5	6	7	8	9
	11	12	13	14	15	16
	18	19	20	21	22	23
	25	26	27	28	29	30

Monday December 15, 1986	
7:00a	
8:00a	meeting with design group
9:00a	
10:00a	
11:00a	
12:00p	
1:00p	lunch - George Smith
2:00p	
3:00p	budget review
4:00p	
5:00p	
6:00p	

Help:F1	Commands:F10	Day # 349 of 1986	12-15-86	3:45:14p
---------	--------------	-------------------	----------	----------

Note Pad

With the Note Pad, you can quickly create and edit notes, memos, or even short documents. You can scroll text up and down in any of ten Note Pad pages to view or enter information as required.

Total Note Pad size can be easily configured to meet your needs.

An unlimited number of Note Pad files can be created and edited. Word wrap is automatic; Insert or Overwrite mode is selectable. Batch files can be edited.

Data is retained in memory when you Quit from the Note Pad and is there when you return.

In the Note Pad, you can:

- Find a string of characters.
- Cut and recall a block of text from the page.
- Insert the current date and time.
- Insert a memo header.
- Erase the current page or all pages.
- Print the current page.
- Import and export ASCII text files.

Organizer

The Organizer gives you all the advantages of an electronic index card file. Each file contains up to 100 index cards and each card holds up to 30 lines of information.

A list of all cards in the current file is presented and can be arranged easily by moving any card to its desired position in the list.

In the Organizer, you can:

- Find a string of characters in the list or on the cards.
- Move a card on the list or a line on a card.
- Make a copy of a card.
- Limit operations to cards in a designated category.
- Erase a card or a line on a card.
- Print a section of the list of cards, or whole cards.
- Access different Organizer files on a disk.
- Cut and recall a block of lines from a card.
- Insert a blank line on a card.

Phone Book

The Phone Book is a powerful tool for recording and accessing names and phone numbers.

An alphabetized list of names is displayed for the set of cards in the current phone book. As you move down the list, the associated card is displayed to the right of the list.

You can move quickly to a name on the list by typing the first few letters in the name.

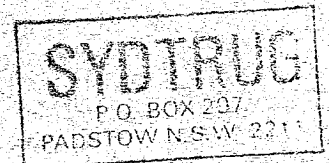
You can auto-dial either of two numbers on a card if you have a Hayes-compatible modem.

A Phone Book file holds up to 5000 cards. An unlimited number of phone book files can be created. Two phone book files can be merged.

The Phone Book commands allow you to:

- Find a string of characters in the list, or in the cards.
- Dial either phone number on a card. (You can set up dialing prefixes, communications port, modem baud rate.)
- Copy a card.
- Limit operations to cards in a designated category.
- Erase a card.
- Print full cards, labels, or a directory.
- Access different Phone Book files on a disk.

Phone Book	
BCS Bulletin Board Bedford Information Systems Carleton, George Carlson Office Supply Davidson, Sam Hudson Realty IBM Service Ingraham, Shipley, & Frank Jacobsen HVAC Kopley Computer, Inc. Marsten Photo Copiers *TEK Microsystems, Incorporated*	Name: TEK Microsystems, Incorporated Address: 2067 Massachusetts Avenue Cambridge, MA 02140 Phone 1: (617) 497-1200 Phone 2: Category: software support Comments Please call us if you have questions.
[Total cards: 42]	
Help:F1 Commands:F10	12-15-86 3:49:59p



QuickKey

Using QuickKey, you can create and edit a variety of time-saving keyboard macros, using simple but powerful procedures.

The macros currently in memory are presented on a list, ordered by the keys which invoke them. Each macro can have a description and a "Context"—a single program in which the macro is active. A macro does not have to be attached to a key. Macros can be protected to prevent accidental erasure.

Macros can be created or edited on an edit screen or by means of an interactive (learn) mode. Special characters are represented in brackets and are treated as single characters in editing, reducing the chance for errors. Edit changes can be cancelled prior to re-entering a macro.

Menu-macros can also be created. A menu-macro presents a list of other macros for the user to select from. The Menu-macro window can be moved anywhere on the screen. Each menu can be given a title.

Playback modes include Fast, Slow and Step. Macro playback can be turned on or off in applications, HQ, or both.

A macro file can contain up to 250 keyboard macros. The total amount of memory allocated for macros can be set by the user.

QuickKey also allows you to:

- Recall last 100 keystrokes into a macro.
- Place a macro in a mode which allows its transfer into a worksheet in standard 1-2-3 macro format.
- Append one macro file to another.

You can enter special functions in macros to:

- Accept user input during a macro.
- Restart a macro automatically.
- Print the date or time.
- Cause the macro to wait a specified time.
- Set the Caps, Num, or Scroll states on the keyboard.

QuickKey		
[Context: DOS]		
Key	Description	Context
=	load Lotus 123	DOS
]	swap with] key	NONE
]	swap with = key	NONE
{Alt-1}	example of QuickKey functions	
{Alt-A}	company name and address	
{Alt-C}	toggle HQ Clock On/Off	
{Alt-D}	demonstration boilerplate	
{Alt-F}	Format decimal places in range 123	
M {Alt-M}	Menu of DOS operations	DOS
{Alt-S}	reassign LPT1 to serial port	DOS
M {Alt-T}	Menu of text choices	
{Alt-W}	weekly appointment entry	HQ
{Alt-X}	go directly to ASCII table	
{AltF1}	type current date	

Help:F1 Commands:F10 12-15-86 3:49:27p

Other HQ Features

- File menus always presented for choice of data files.
- Unlimited number of data files with any module.
- Reminders to save data.
- Instant peek under all HQ windows to application and under a Help screen to an HQ screen.
- Pause or cancel during printing.
- Full table of ASCII character codes presented in General Help screens.
- Control character sequences can be sent directly to printers.
- Screen clock can be customized for display in application programs.
- DOS time and date can be set directly from HQ.
- Fast exit from all modules in use.
- HQ easily unloaded from memory.
- Setup utility for easy configuration of memory sizes.

Requirements

To run HQ you need the following hardware and system software:

- IBM PC, XT, or AT, or fully compatible computer.
- To run all HQ modules: 92K of RAM memory, over and above what your application software and DOS require.
- At least one floppy disk drive (hard disk recommended).
- DOS 2.0 or later.

HQ works with all IBM compatible display adapters in text mode. For graphics mode operation, HQ supports IBM CGA, IBM EGA, and Hercules Graphics Card.

HQ works with any printer that accepts ASCII characters and the control codes CR and LF.

HQ is not copy-protected and has a suggested retail price of \$79.

HQ is a registered trademark of TEK Microsystems, Incorporated.

The following are trademarks of the companies listed. IBM, IBM PC, PC XT, PC AT : International Business Machines Corporation; 1-2-3 : Lotus Development Corporation; Hayes : Hayes Microcomputer Products, Inc.; Hercules : Hercules Computer Technology.

contains lots of goodies, that in general function with the WP command key format. The price is \$50 - 60 and worth every penny, best deal from the WP people. It include a complete ASCII editor called PE (programmers editor) same as the programmers at WP use, not bad. QDOS also has a built-in editor that handles files to 64K and it a snap to learn (5 minutes, worst case) all menu driven.

Fm Ken Kane To LSI(Bill): Right now I'm snowed under with two new software packages: WP and DESQview. DESQview Companion member Notebook provides a simple text editor. And DESQview has many features that overlap with WP library, such as macros. But I'll give a closer look at WP library. It was reviewed in detail in PC Magazine a few weeks ago.

Fm Marc Nowell To Ken Kane: I use BRIEF exclusively! I edit all of the messages here on the SIG with it for off-line reply composition. "Is BRIEF suitable for this?" takes on new meaning when you realize that if a desired function is not IN BRIEF, you can simply add it! I've done that to the point of having a perfect editor, at least from my viewpoint -- which is the whole idea of a configurable editor! There are some excellent macro routines written to enhance BRIEF's performance as an off-line message editor. If you get the package, I'd be glad to forward them to you.

Fm Ken Kane To Marc Nowell: It will be a while (if ever) before I try out BRIEF, however. I am half drowning (grin) in DESQview, which includes (in the extra DV Companions package) a very nice editor for CIS messages etc. DESQview Companions also is (are?) my one LINK with the modem world of CIS and such. The LINK communications program is very easy to use. DESQview is topheavy with menus but provides infinite LEARN macro capability from any level. So it is easy to bypass the menus. Most programs can be run within DESQview so the LEARN facility is generally available.

DAC Easy Payroll 2.0: Not Easy!

Fm Nate Salsbury: Do NOT be fooled by DAC's advertising campaign and APPARENTLY low prices. The dollars spent on their buggy software will be enormously leveraged into HOURS of your time: Attempting to unravel their manuals and screen messages; Trying to reach an intelligent person to talk to on the phone regarding non-performance of the software to the manual's description; Revising and re-

entering data in an attempt to salvage some value from the time and \$\$s already spent.

I have spent seven (that's right.. SEVEN) weeks attempting to get a working version of Report 2.0 and Payroll 2.0. I sent letters, screen printouts, disk files with examples of non-functioning data, made phone calls... all to no avail. I received one replacement disk for Report which still contained over 1/2 the bugs I complained of originally. Their software manager personally promised a "fixed, final" version on August 13. I am still waiting... but NOT expectantly I returned both programs for credit - per their advertisements - on August 20. I have not received a refund NOR an acknowledgement.

They charge \$60 for an hour of "consultation". The advice received for this is invariably 'workarounds' for their program's errors or limitations. They will NOT talk to you - even for bugs in THEIR software without your paying that fee. They do NOT answer letters.

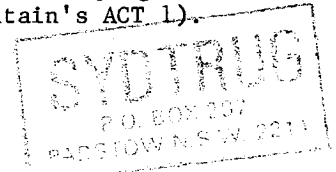
Buy your accounting software programs elsewhere (I like Cougar Mountain's ACT 1).

W-2s on disk for IRS

Fm Nate Salsbury: Our wonderful IRS has decreed that any company which files over 200 W-2 forms for a year MUST do so on 'magnetic media'. I do work for such a company. They are using the DAC-EASY Payroll 1.0

DAC has now abandoned all the early customers who bought the 1.0 versions. They will NOT discuss or support it. They refuse to sell the program which could extract data into an ASCII file. Therefore, I have unraveled their data file structures and written a program (in GW-BASIC) to extract the data demanded by IRS. The program then re-writes this info in the very specific file sequences mandated by IRS including headers, frequent sub-total records, summary records etc.

If you would like this program (on MS-DOS DSDD diskette), send \$ 5.00 to Salsbury Associates Inc. 610 Madam Moore's Lane New Bern NC 28562. The program will include a complete fielded breakdown of the 642 byte data record used in the DAC program. You can use this info for submitting data to Employment Commissions or other government agencies who are starting to demand magnetic media submissions.



The VAL function

Fm Nate Salsbury To All BASIC-bashers: Type this little program into GW-BASIC and run it. I would be very interested in anyone's explanation for the output when 'A' is 28, 29 or 31! See if you can guess what will happen before you run the program.

```
10 INPUT "ANY numeric value e.g. 123 or
45.7 or -17.6."; T$
20 FOR A = 1 TO 31
30 PRINT A;VAL(T$ + CHR$(A)),
40 NEXT A
50 PRINT:PRINT:GOTO 10
```

This 'feature' is NOT present in M3 or M4 BASIC.

Fm Marc Nowell To Nate Salsbury: Wow! That is WEIRD! No explanation that I can think of, though... I ran it on my 1400LT, and couldn't believe such a strange output.

Fm --jjkd-- To Marc Nowell: Well, I know what is causing that to happen, but not why that's happening, if you know what I mean. Probably not a design feature, but a result of Microsoft's propensity towards reusing/overlapping code.

Altering GWBASIC Function Keys

Fm --jjkd-- To Ken Kane: No GeeWizBASIC I know of supports configurable storage for Fkeys. A quick peek with FED86, DED86 or DEBUG should pop up the answer. My guess is that they would be stored as variable length, null terminated strings and so patching with a longer string would be a problem.

How about changing the name of the BASIC, setting up a batch file that brings up BASIC and runs a program that sets the function keys to be what you want.

Fm Ken Kane To --jjkd--: Right, I have just such a solution, but was looking for a cleaner way. I set up GWBasic to run within DESQview which has impressive macro capabilities. I set up a DESQview macro which automatically changes the key definitions first thing after installation.

(Those keys remind me of my trusty Model 100. Saving altered key definitions on the M100 is not a problem because you never turn off the mos memory!).

How can I redirect printer output to a file?

Fm Lee C. Rice, Ph.D.: Despite all of the utilities which I have bought and written for my MS-DOS system, I can't find anything like your TRSDOS Z-SHELL utility (which I use all the time on the Model 4). Using standard MS-DOS redirection, it is easy enough to send OUTPUT from a program to a FILE (although it is easier still under TRSDOS, and also faster). What you can't do is to redirect PRINTER output to a FILE instead. Of course you don't even need ZSHELL under TRSDOS/LDOS to do this, thanks to the ROUTE and LINK commands; but ZSHELL makes everything smoother and faster. MS-DOS seems to leave no room for printer redirection at all. Nor can I find any commercial utility for MS-DOS which does printer redirection (and I have several spoolers). And it is not because I have not looked...

So, if you are looking for some MS-DOS ideas, how about authoring (and SELLING!) an MS-DOS ZSHELL utility?

Indeed, a general comment. Since MISOSYS is entering the MS-DOS market too, how about some small one-track utilities - the kind which you had available in great quantity for TRSDOS/LDOS, but which are virtually unavailable for MS-DOS. There are large numbers of big, kludgy, cumbersome utility "packs" available usually for equally big dollars. What is missing are the small single-track utilities which TRSDOS/LDOS users have already grown accustomed to using. Heck, I might even consider an MS-DOS diskdisk; since the MS-DOS subdirectory structure really wastes disk space on a hard drive.

Fm MISOSYS, Inc: Here is some insight into MS-DOS and I/O redirection. First off, one major point of concern in MS-DOS is that it is not re-entrant. What this means is that anything trapping a DOS service function cannot request a DOS function. There is little guarantee that trapping any given BIOS interrupt will deliver the knowledge that the original requester was not DOS. Thus, an attempt to write a TSR which would filter the BIOS 17H printer services and send such output to a disk file (requiring DOS services) could indeed totally crash the sys-

tem. MS-DOS is not re-entrant because the STACK segment:offset registers are saved in a fixed memory location on a DOS call. You can understand the rest.

MS-DOS achieves I/O redirection of stdin and stdout by duping the file descriptors associated with the first two handles always established. Note that only handle input and output is affected. A program that goes to the BIOS for I/O is not affected by command.com's redirection support.

It would have been easy for MS-DOS to support redirection of stdprt (another standard handle for printing) by supporting some other redirection character. Likewise, some folks have wished for easy redirection of stderr. Here's what some folks have done (according to one source). If you write a brief C program, say routepr, which accepts a command line composed of a file specification and the command line which would have invoked the program you want to run with printer redirected, then routepr could freopen the file to stdprt and spawn a process to run the subsequent program. Spawned processes use the handles established by the parent; in this case, stdprt output goes to the file you specified. When the spawned process terminates, control passes to the parent, routepr, which closes up and goes back to DOS. All of this will only work with programs using handle output for printing. I would venture to say that's a small quantity.

This is not to say that redirecting the printer at the BIOS level is impossible. Indeed, it is only necessary to determine at the TSR level when the state of the machine is not in execution of a DOS function. This is not a trivial task, but it is possible. Many existing TSR's already do it. In the simple sense, it is only necessary to trap INT 21H and set a in-dos flag within the trap code. Then other portions of the TSR which have trapped other BIOS interrupts could determine whether the in-dos flag was set. If it wasn't, any characters buffered from printer calls could be dumped to the file. You would have to set up a buffer to deal with buffering characters until the in-dos flag was reset.

Lastly, although I have no personal knowledge of its operational characteristics, I have come across an ad from SOFTWARE RESOURCE (a distributor) noting a product called "DJ" from Revolution Software, Inc. The package is a collection of 10 utilities: one of which is "REDIRECT sends printer output to a file for later printing".

On the other hand, a diskDISK86 certainly falls within the realm of things doable under MS-DOS (perhaps practical, to boot. HMMMMM...).

Microsoft WORD and the Daisy Wheel II

Fm MISOSYS, Inc: Last issue I commented about my experiences with WORD and the Radio Shack Daisy Wheel II printer. I discussed a bug in the printer driver supplied with WORD and how I corrected it. Here's some feedback on that topic.

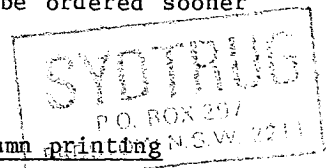
Fm Mark P. Fishman: You might be interested to know that Microsoft WORD 3.1 not only supports the Daisy Wheel II, it supports proportionally spaced printing with the DWII. I also had to discover and learn how to fix the improper initialization of the driver, but you seem to have done it so I won't go into it here.

The CUBIC wheel, which you used on the old NOTES from MISOSYS, is also supported; this is Madeleine.

Fm MISOSYS, Inc: While working on this issue of THE MISOSYS QUARTERLY, I received the 4.0 version of WORD. Now I really didn't want to install it until after the issue was completed - lest some problem interfere with the TMQ editing and printing. But I did take the opportunity to thumb through the documentation supplied with the 4.0 release. Needless to say, I think Microsoft solved the DWII problem; they don't provide a printer driver for it! I guess that I'll have to pour over the Printer Information manual to see if the older DWII.PRD driver will work with release 4; I do know there are some additional sections of information in the release 4 drivers. Perhaps that new laser printer will be ordered sooner rather than later.

Microsoft WORD and mixed column printing

Fm MISOSYS, Inc: Here's a little background: The first four issues of THE MISOSYS QUARTERLY were done using PROSOFT's ALLWRITE operating on a TRS-80 Model 4; output was on a Radio Shack Daisy Wheel II printer. We shifted over to Microsoft WORD running on our PC (and subsequently AST/286) primarily to gain operational experience with the MS-DOS environment. WORD also provided us some benefits; online spelling checker, style sheets, multiple file



editing through windows, and whatyouseeiswhatyouget (wysiwyg) editing, which we prefer to downstream formatting.

One of the formatting capabilities we used in ALLWRITE was its ability to provide multi-column printing on a page. In fact, it supported the capability of mixing column formats ON THE SAME PAGE. WORD supports multicolumn formats with good wysiwyg editing, but WORD does not allow mixed column formats on the same page. So I drafted the following letter to Microsoft Customer Support.

"I have a problem that, perhaps, you may be able to solve. We have been making extensive use of WORD to produce software user manuals. We also recently shifted over to using WORD in the preparation of our QUARTERLY publication. THE MISOSYS QUARTERLY averages about 100 pages per issue. We format this in two columns but occasionally shift to single column when inserting program listings. Therein is the difficulty.

We previously had been using a powerful word processor called ALLWRITE; this being done on a Tandy Model 4, of all things. One of the capabilities of ALLWRITE was the ability to shift between single and multiple columns ON THE SAME PAGE. I prefer WORD, now that I have gotten used to it because I am soon shifting to a Laser Printer and I appreciate the wysiwyg interface. In any event, WORD appears to NOT enjoy the same power of dual page formatting.

I have enclosed a few samples of previous issues of TMQ which show (1) shift from two column to one column, (2) shift from one column to two column, and (3) shift from two to one to two to one to two column (whew!). I have tried this technique with WORD. The best I was able to do was to split a column via a DIVISION BREAK COLUMN so that the two-column "spread" was balanced but then the single-column text which followed used the left margin of the previous second column. As I see it, the problem comes in because none of the division break formats apply. PAGE would break the page and use the new format on the next page - but I want to continue the new format on this page! CONTINUOUS doesn't start the new format until the next page but fills up the current page with the new division text - but I want the new division single column on the current page. COLUMN doesn't do it; neither does EVEN nor ODD! What I need is something like MULTI which would continue the current page with the new format using text from the new division.

Now ALLWRITE also allowed two ways of specifying multi-column pages. They were either BALANCED or UNBALANCED. A balanced multicolumn format would spread all of the columns such that the columns in the last page would end as evenly across the page as possible. An unbalanced multicolumn format would use the full vertical space on a page for a column prior to starting a new column; WORD's behavior with multicolumn is de facto unbalanced.

Now perhaps I am overlooking something. On the other hand, I am guessing that this columnar behavior may be missing from WORD's powerful repertoire. I have read in the industry press that Microsoft has been testing its 4.0 release of WORD. If such is the case, I am hoping that this multicolumn feature has been incorporated and that any day now, my mail will reveal the upgrade notice.

How about it? Do you have any help for my formatting problem?"

Here's what Microsoft reported in their response to my letter: "Thank you for writing us concerning your difficulty with WORD.

You are right in suspecting that the program does not support different column formats on a single page. Unfortunately, I must tell you that WORD 4.0 does not either.

WORD does offer the running head feature which allows you to put a single column at the bottom or top of a page but, after looking at the sample documents you included I can see that this feature will not be a complete solution to your problem."

Applications for the User

Focus on speed

Over the past five months, a number of things have occurred which place a greater emphasis on machine speed. There appears to be continued interest in the H.I.Tech accelerator card known as XLR8. There is a great deal of interest in the new breed of MS-DOS based 80386 machines. And at MISOSYS, we acquired a fast 80286 machine - the AST Premium/286 clocked at 10 MHz, 0 wait state memory access.

To inaugurate this heightened interest in speed, we present excerpts from a thread on the LDOS forum on Compuserve. This thread, started by Adam Rubin, begun as a simple test of the Model 4 raw processor speed. It got pretty involved with the intricacies of the TRS-80 Model 4 hardware - and the specifics surrounding the Z80 processor cycles. As all things which ramble, the "speed" topics touched on MS-DOS and PC machines as well. It may get kind of techie, but bear with it; you may just learn something.

We follow up with a suite of C-language machine speed test programs gleaned from the July 1987 issue of *BYTE* magazine, and one assembler speed program derived from Adam's initial program illustration. Because it was derived from his work, I have named it the "Rubin's" test. I also re-coded it in 8086 assembler, keeping the code stream as close to a one-to-one correspondence, so as to derive an equivalent speed test for PC Clones. I ran these tests on four different Model 4 computers here at MISOSYS: one was one of the first Model 4's manufactured; another was a more recent non-clustered arrow keyboard machine; a third was a white screen non-clustered keyboard Model 4P, and the last was a green screen clustered arrow 4P equipped with an XLR8er board. I also ran them on three different PC-class computers: a real genuine IBM PC

both at 4.77MHz and with the AST Hotshot/286 card running at 10 MHz; the old Leading Edge 8088 machine at 8 MHz; and my new AST Premium/286 machine identified above. All of the results are posted in a table which appears following the discussion on speed.

The suite of programs are also shown. They may give you some rough ideas as to what kinds of operations are being evaluated. Let's jump right in. If reading our Compuserve dialog turns you off, skip to the test results.

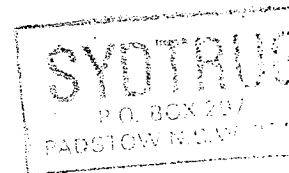
LDOS Forum thread on speed

Fm Adam Rubin: Here's this week's entry in the "interesting statistics" category. I created the following short program:

```

                ORG      6000H
BEGIN          LD        HL,6020H
                SVC      @TIME
                LD        DE,0100H
LOOP1          LD        BC,0000H
LOOP2          DEC      BC
                LD        A,B
                CP        C
                JR        NZ,LOOP2
                DEC      DE
                LD        A,D
                OR        E
                JR        NZ,LOOP1
                LD        HL,6030H
                SVC      @TIME
                RET
                END      BEGIN

```



Admittedly, this is a pretty useless program, but it does take time to run. Under LDOS 5.3.0, with SYSTEM (FAST) and type-ahead active (for KI4), it took a bit under 111 seconds (as measured by the system's real-time clock), and under LS-DOS 6.3.0 took 112 seconds. SYSTEM (TYPE=NO) knocked about one second off both times, but SYSTEM (SLOW), with type-ahead active, added about two seconds more than twice the time for the faster speed.

I calculated how long the looping portion of the program ought to take (i.e. not including the SVC @TIME), and came up with 436215547 T-states. At the 4P's specified clock rate of 4.05504 MHz, it came out to 107.57... seconds, which means that most of the remaining few seconds were used for updating the clock, blinking the cursor, and all those other necessary things. Comments, anyone?

Paul Bradshaw To Adam Rubin: I just ran that program on my OLD model 4 equipped with an XLR8er running at less than full speed (one wait state added back), and came up with an execution time of about 72 seconds. Not bad! And for those of you contemplating the purchase of an XLR8er, when compiling a 625 line program with ALCOR (TRS80) PASCAL, my compile time went from 4 minutes 40 seconds to less than 2 minutes 10 seconds (and that's NOT at full speed). Very interesting!

--jjkd-- To Adam Rubin: Yessirreeebob, you have just created a honest-ta-gawd benchmark, surpassing in applicability even that bastion of the MS-DOS user, Norton's SI.

A wait-state is typically the insertion of one additional time slice equal in length to a t-state for whatever the microprocessor clock is. A wait-state may be inserted on every memory access, this is most common, but not always.

The Model 4 systems that have wait-states generally have them on M1 only. M1 is the first memory access for the execution of an instruction, also called the opcode fetch. Any additional bytes fetched with memory cycles during the execution of that instruction would proceed without wait-states. Thus, the effective speed of the system will vary with the instruction mix.

Other forms of synchronizing the processor to slower memory or peripherals exist that result in the equivalent of variable duration wait-states. The device signals when ready, as opposed to waiting a fixed time. Of course, the CPU will turn around and quantize the time by re-syncing to the system clock, so the result is a variable multiple of a t-state.

Adam Rubin To --jjkd--: The "variable duration wait-state" is what the Model 4 FDC board uses, though other devices (e.g. printer) might be better handled by polling.

I have one question about the wait states for the Model 4's memory access. If the memory is slow enough to need one or two additional T-states when fetching the first byte of an instruction, how can additional bytes be successfully read without wait states?

Shane Dawalt To Adam Rubin: The Z80 treats an op-code fetch differently than a memory fetch. Op-code fetches require the memory to be able to provide it's data during t-state 2 after the system clock's falling edge. For a memory

read or write (not op-code fetch), the memory has until the falling edge of t-state 3 to "do it's thing". Hence, for memory read/write, the memory has an additional t-state. Why is the op-code fetch treated differently from memory read/write? ... because of automatic refresh. During t-states 3 & 4 of an opcode fetch, the microprocessor is decoding the op-code just read. This is "lost time", so during t-states 3 & 4, a 7 bit refresh address is placed on the lower 7 address lines and the *RFSH signal is asserted. This is used by the internal PALs (or gate arrays for the newer models) to refresh the DRAMs. This only takes 2 t-states! Why not make the op-code fetch 5 t-states? Because a machine cycle is only 4 t-states. You can't have 1 machine cycle spanning 5 t-states! Incidentally, any wait states are inserted between t-states 2 & 3 for all machine cycles, even I/O port requests. The Z80 automatically generates 1 wait state for I/O port access too.

Ray Pelzer To Shane Dawalt: Actually, THAT could explain why some simple memory tests which just do memory read/write/compares will pass a chip yet the execution of a program in that memory will fail (such as the so-called "worm" tests which execute a block of code whose last instructions move the block of code a byte higher and then re-execute)... that extra available t-state permits marginal memory to hide its flaws!

--jjkd-- To Adam Rubin: Remember that an instruction consists of one or more bytes. The first byte, the opcode fetch, requires more strict timing constraints. The remainder of the bytes in that instruction may be fetched with more relaxed timing constraints. As soon as you get to the next instruction, the whole process starts over again. So, it makes sense that you might need a wait state on M1. What doesn't is needing two wait states for M1 and none for other memory cycles, as some Model 4s are configured. If you need two wait states for M1, you probably should have one wait state for other memory cycles.

Shane Dawalt To Adam Rubin: I have the (I think) oldest manual going from Zilog; it's copyright date is 1977. And the cover certainly looks like it's that old ... the "well used look" is what I call it. Read the description of M1 CAREFULLY. I missed it the first 5 times I read it too. The signal is asserted DURING op-code fetches ONLY. For multiple op-codes, it is asserted multiple times. NOTE that the entire instruction is NOT the op-code. Only the instructions dealing with

the CPU. The other part is the operand and is considered data. The op-code is accessed with the Op-code machine cycle while the operand and all memory read/writes are accessed with the Memory Read/Write machine cycle.

External hardware asserts the wait states. Specifically, the hardware looks for the *M1 signal and the *MREQ signal. Those two signals tell the hardware the CPU is fetching an op-code which you know has a "shortened" memory access cycle. From the above info, you can see that these wait states will only be generated when the CPU is fetching op-codes and at no other time since *M1 is not asserted during normal memory read/writes. And during interrupt acknowledge, *M1 is asserted, but *MREQ isn't ... again, no wait state is generated.

Adam Rubin To Shane Dawalt: I think I see where some of my confusion is coming from... as I mentioned, I have Zaks's book, and when he wrote the chapter on all this the internal timing for the Z80 wasn't available, so his book uses Intel's 8080 for the discussion of M states and T states, and ALL the 8080 instructions have a one-byte opcode, plus data. I think another fifteen minutes in the library and I'll understand it all.

Adam Rubin To --jjkd--: So timing is different even when the instruction itself is more than two bytes? (Things like "BIT b,r" and "SBC HL,rr").

--jjkd-- To Adam Rubin: No, only the first byte ever counts for M1. Thus the instruction

```
LD HL,0 [21 00 00]
```

has three bytes, requires three memory cycles, one of which is a M1 cycle. Having one wait on M1 would make this instruction take one more T-state to execute than the Zilog documentation says. If we had one wait state on all memory cycles, this instruction would take three more T-states than the doc says.

A two byte instruction like

```
LDIR [ED B0]
```

has two bytes, but still only one M1 cycle. The M1 cycle, or opcode fetch, gives enough information for the CPU to "know" the format of the remainder of the instruction/data, which is what takes the extra time during M1.

Given two programs, one of which is heavy on one byte instructions, and the other is heavy

on multi-byte instructions, running on a machine with one wait state on M1. Assume that you didn't know about the wait state. The first program will run markedly slower than would be predicted based on the CPU clock speed and counting instruction T-states. The second would also run slower than predicted, but not as much as the first.

Shane Dawalt To --jjkd--: Each OP-CODE generates an M1 cycle, i.e. a wait state is generated for EACH op-code read during an instruction read. M1 is cycled as many times as required to read in an instruction. Of course, the data in the instruction (i.e. addresses, offsets, etc) are read in using normal memory reads, not an M1 cycle.

On your example with LDIR (ED B0), that would require 2 M1 cycles, but you would still call that 2 machine cycles since, in reality, there are 2 machine cycles involved. This is because BOTH bytes are opcodes.

--jjkd-- To Shane Dawalt: You are entirely right. For those multi-byte instructions that are multiple opcode, (e.g. most of the Z80 additions) the M1 signal is asserted on both opcode fetches. In this case, the name "M1" or "Machine cycle 1" is a misnomer.

Shane Dawalt To --jjkd--: One thing with bothers me about this idea is the BIT MANIPULATIONS on the INDEX REGISTERS. If you notice, the instruction has two opcodes, a data area (offset byte) and another byte of information. I have wondered if the CPU sees the 4th byte as an opcode (I would guess so) or a data byte (I think not). My Z80 manual doesn't say (although it gives info on every other instruction, practically). I can't believe it would read two opcodes, then a data byte then another opcode since that would be going. I believe, it is more appropriate to say that while the Z80 is reading opcodes, it is "stuck" in the M1 cycle. You may disagree by saying, but it's executing a new machine cycle each time an opcode is fetched. Yeah, I know ... but from the hardware view, it is looping in the M1 cycle. (At least the microcode is.)

--jjkd-- To Shane Dawalt: If the doc you have breaks down the instruction T-states by machine cycle, check the number of T-states for each byte. Machine cycles generally run 3 T-states for a "data" fetch and 4 T-states for an "opcode" fetch, when other operations aren't clobbering things up.

To see if M1 is actually asserted on that byte fetch, you could use a logic analyzer or a bit of creative programming. You "know" how many T-states the instruction is supposed to take in a zero wait state machine, execute a million of 'em and see how long it really takes on a machine with one wait on M1. I hate to say this based on my recent track record, but I don't think that the Z80 has microcode in the sense of firmware and an execution unit. I believe that the Z80 was one of the last true implemented-from-gates-and-stuff CPUs.

Shane Dawalt To --jjkd--: For a BIT b,(IX+d) [where b is the bit number and d is the displacement], my docs show 4 bytes (which makes sense), 6 machine cycles (which makes sense, 4 for reading one to fetch the byte to modify from memory and one to rewrite it) and finally, 23 T-states. With 4 T-states in an op-code fetch and 3 in memory I/O, that won't add up. Note that my docs say an op-code fetch can be anywhere from 4 to 6 T-states long. (I wonder how the logic determines when to make the access longer?)

Your track record is safe. You are correct. Microcode isn't the appropriate name for the combinational logic internal to the Z80. It's micro-(something).

--jjkd-- To Shane Dawalt: The four and three T-state numbers come from "simple", "whole byte" instructions. Bringing in the barrel shifter takes extra T-states I'm sure.

Adam Rubin To Shane Dawalt: My Nanos Z80 Reference Card says BIT b,(IX+d) takes $4+4+3+5+4=20$ T-states. SET b,(IX+d) takes $4+4+3+5+4+3=23$ T-states; I'd guess it's the same as BIT, plus an extra 3 T-states to write the byte to memory.

It looks like the only instructions where the opcode fetch takes more than 4 T-states are PUSH AF, BC, DE, and HL (5+3+3); LD SP,HL (6); DJNZ if condition false (5+3) [JR cc if condition false is 4+3]; RET cc (5+3+3 if true, 5 if false); and RST (5+3+3).

Anyway, I gather that M1 is asserted only on the opcode fetch, which is the first byte (and only the first byte) of every instruction, and the only byte that is read from memory before T3. Do I at last understand this? <grin>

Shane Dawalt To Adam Rubin: OF COURSE! I forgot about that extra 3 T-states required to place the modification back into memory [I'm talking about SET b,(IX+d)].

Your reply also points out an important answer to a question I asked jjkd. The displacement (3rd byte) within the SET b,(IX+d) is seen as data and not as part of the opcode, however the 4th byte is seen as an OPCODE. That means it cycles as OPCODE, OPCODE, memory cycle, OPCODE. That's interesting. One would think it would read all the opcodes first, THEN do data reads to the end of the instruction.

No. You are not correct. If there are 3 opcodes in 1 instruction, there will be 3 OPCODE fetches and the M1 is asserted 3 times ... one for each OPCODE fetch. Does this make any sense? That's why I was wondering about SET b,(IX+d). Although doing 2 opcode fetches, one data fetch and then a third opcode fetch isn't profound, it wasn't what I had expected.

--jjkd-- To Adam Rubin: There are a number of Z80 (non-8080) instructions for which the first two bytes are fetched under M1 conditions. I believe that Shane's question was "In the instructions BIT b,(IX+d) and SET b,(IX+d) we know that there are at least two M1 cycles as the first two bytes are fetched. Is the fetch of the (I think it is) fourth byte also treated as an M1?" Note the number of T-states and the way the instruction is coded.

Adam Rubin To Shane Dawalt: In the Z80, all the 8080 instructions can be identified by the first byte, so the only M1 cycle occurs when the first byte is fetched. The extended non-index-register instructions need two bytes to be identified, so M1 is asserted (i.e. NOT M1 goes low) for the first two M cycles.

For any operations using either index register, the first byte indicates an index register is used, so two bytes (two M1 cycles) are needed to identify the instruction. If the instruction would ordinarily reference (HL), the third byte is the offset (b) for (Iz+b), and is read as data (no M1). For BIT/RES/SET (Iz+b), the fourth byte specifies the operation and which bit, and so M1 signals the reading of this byte. So, if SET 5,(IX+3) (DD CB 03 EE) is executed, this happens: M1 4T Fetch x'DD' (3 T-states). Determine (1T) this is an index register instruction; another byte will be needed to determine which one. M1 4T Fetch x'CB' (3T). Determine it's rotate, shift, or BIT/SET/RES instruction. Next byte will be offset. M2 3T Read offset, x'03' (3T). No "decisions" needed. M1 5T Fetch x'EE' (3T). Determine it's SET 5 (1T), and add offset to copy of IX (1T). M2 4T Read (IX+3) (3T). Set bit 5 of value read (1T). (For BIT,

set Z/NZ.) M3 3T Write new value to address (IX+3) (3T).

Let's see, that adds up to 23 T-states. Therefore, a Model 4 with no wait states would actually take 23 T-states to execute this, or 23 x 1/4.05504 MHz. A Model 4 with "1 wait state" would add an extra T-state on each of the three M1 cycles, so it would take 26 T-states for this instruction, and "2 wait states" would mean 29 T-states. (Out of curiosity, that's effective clock speeds of 4.05504, 3.587, and 3.216 MHz respectively for this instruction.)

Shane Dawalt To Adam Rubin: You are 100% A+ correct in your message. Your statement about T3 and all is correct ... as you probably already knew and didn't need me to confirm it. (I'll try to slip my brain back into gear as soon as I can get the clutch to work!)

BTW, your mapping of each Machine cycle in the SET b,(IX+d) instruction was quite nice. I liked that! Isn't it disappointing to calculate out the T-state timing and determine that the Model 4 (which is shown to be running at 4MHz in the specs) is only running at around 3.587 MHz? I would like to get my greasy mitts inside this machine and pull it's WAIT wire from the PAL which generates the WAIT for each M1 assertion. Actually, after looking closely at the old M4 schematics, I don't see an explicit relationship between WAIT, M1 & MREQ. As a matter of fact, WAIT goes to the external bus and to the video timing circuitry. There is always a wait generated if the RD/WR lines are active and if the video is being accessed for refreshing of the screen. The only connection I see with the M1 signal is in the timing PAL which generates the processor clock. I can't see why they would do this, but I suppose they know why...

Adam Rubin To Shane Dawalt: I took a quick look at the Model 4 schematic in the "old" (8.5 x 11) tech ref manual, and didn't see any obvious connection between the M1 and WAIT pins. Either it's hiding in there somewhere, or was omitted from the schematic. I do remember a suggestion from someone that the easiest way to get rid of the wait on M1 was to bend up the M1 pin on the Z80. I have no intention of trying it, though, since all 4Ps run with no wait states anyway.

I followed jjkd's suggestion, and came up with the empirical method of determining how many M1 cycles a SET instruction takes. Make sure you're using SYSTEM (FAST) to run it, and

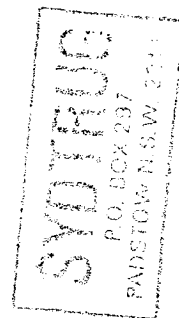
you'll probably want to re-set the time when you're done. Note that it takes between five and seven minutes to run. Here's how to interpret the results:

Wait States			
M1 cycles in SET	0	1	2
1	5:23	5:48	6:13
2	"	5:56	6:29
3	"	6:05	6:46

As predicted, I measured 5:23 on my 4P. Here's the program described above.

```

ORG      3000H
;Give user 5 seconds to get ready
START   LD      D,5
DELAY   LD      BC,0
        LD      A,10H      ;@PAUSE
        RST     28H
        DEC     D
        JR     NZ,DELAY
        LD      B,7
        LD      A,68H      ;@SOUND Beep!
        RST     28H      ;Stopwatch on
        DI
        LD      IX,DATA
        LD      DE,0200H
        LD      BC,0
LOOP    SET     0,(IX+0)
        DEC     C
        JR     NZ,LOOP
        DEC     B
        JR     NZ,LOOP
        DEC     E
        JR     NZ,LOOP
        DEC     D
        JR     NZ,LOOP
        LD      B,7
        LD      A,68H      ;@SOUND Beep!
        RST     28H      ;Stopwatch off
        RET     ;to DOS
DATA    DEFB   0
        END     START
    
```



--jjkd-- To Shane Dawalt: Sounds like they are using the same method to generate CPU memory access delays that LOBO used in the MAX80 - they stretch the CPU clock itself instead of going to the wait input.

Shane Dawalt To --jjkd--: EXACTLY! I found that to be quite a surprise since the Z80 has stringent duty cycle requirements on it's clock input. Boy, I saw that in the timing docs and about fell off my chair.

Shane Dawalt To Adam Rubin: My stopwatch tells me 6:29 which corresponds to 2 wait states. What is the "M1 cycles in SET" column for? I

don't quite follow it. Why I'm showing 2 wait states isn't clear, but it will probably dawn on me! Anyhow, now the question is, how to change that to 1 or 0 Wait states? I would probably have to obtain 120ns DRAMs for 0 Wait states (if that can be achieved) but my machine already has 150ns DRAMs which I'm confident will handle the deletion of one wait state. Of course, before I really stick my foot in my mouth, I'd have to do some timing analysis with DRAM spec sheets in hand.

I was looking through the timing data supplied, and figured out how Tandy is getting by without using the WAIT to slow the processor during memory accesses. Those little sneaky devils play with the CLOCK going to the CPU. When it is in 4 MHz mode, the M1 line is used to gate out a clock cycle (which is during an opcode fetch). This is, in effect, what the processors WAIT line does, but this is a brute force method. I wonder about their reasoning for this setup since a video wait cannot occur simultaneously. (How is it going to output to the screen while fetching from the memory?) I suppose a simple solution would be to trace cut the M1 signal to U3 (the culprit) and assert it high through a 2.2K pull-up resistor. I wonder if the XLR8 generates it's own timing and ignores the hardware WAIT request?

A neat idea to fix this video wait problem (which isn't active all the time, but still, it's the principle of it) would be to change the current 2K x 8 static RAM w/ 200ns access to a 100ns access chip (which can be bought for around \$7 now) and disable the ACCESSINPROG signal to the *PWAIT generator. Thanks for the your timer program.

Ray Pelzer To Shane Dawalt: Well, what I did with my unit was a little sneaky. The original design of the Model 4 board (black&white monitor) allowed you to pull a pin from a PAL chip (for some reason, U6 comes to mind) and then the WAIT line would never be asserted to the Z-80 chip. This was a commonly recommended technique at the time. I, however, got the first green-screen unit with clustered arrow keys (not YET the Gate Array CPU board, though), and THAT CPU board also snuck an extra input to the WAIT pin from somewhere else on the board, so pulling the pin on the PAL chip did nothing. What I did was bend out the M1 pin on the Z-80 itself. The RAM chips I had in the unit were of adequate speed it appears, because the unit has run flawlessly in this state for the last couple of years.

Adam Rubin To Shane Dawalt: Thanks for running that program. It looks like we now have answers to two questions. Remember we weren't completely sure how many M1 cycles a SET instruction used? Well, when I was calculating the timing on that program, I realized that the number of M1 cycles in the SET instruction would cause (number of M1 cycles) * (number of wait states per M1 cycles) wait states for each execution of the SET instruction, so the actual time to run that program depended on two things: how many wait states were inserted after every M1 cycle, and how many M1 cycles were needed for the SET instruction. I figured out the time needed for 0, 1, and 2 wait states per M1 cycle, and 1, 2, or 3 M1 cycles for SET, and came up with that table. The number of M1 cycles for SET is constant for all Z80s, so it looks like the SET instruction actually uses only two M1 cycles, not three as we'd assumed. It also looks like your machine's inserting two wait states after every M1 cycle.

You're absolutely right about how the M4 generates those wait states on M1. I finally found it in the Tech Ref Manual. That does sound kinda kludgy. Incidentally, the 4P Service Manual specifies the memory as MCM6665, 64K DRAM, 200 ns.

I'm not sure if you'd want to change the video wait "problem". That's there for a different reason. Remember, the Z80 and the CRTIC can't both access video memory at the same time, but the CRT (and hence the CRTIC) keeps going whether it has any data or not. I gather that if the CRTIC can't get its data, it just displays "white" on the screen, and so without video waits, you'd get a faster, but "snowy", display. See page 6 of the CRTIC spec sheet, or Bradshaw's file PORTS.DOC in TRS80PRO's DL4.

Adam Rubin To --jjkd--: Are there any advantages to that method of simulating wait states? (I assume Tandy did it that way to cut costs, right?)

Shane Dawalt To Adam Rubin: Well, that's certainly funny ... the M1 cycles in SET I mean. Everything you said makes sense! Why my machine is inserting two waits is not! The timing diagrams for the docs shows only one; perhaps I should reread it WITH MY SPECS ON.

The XLR8er board's HD64180 has programmable wait states within it.

My notation uses "*" to represent NOT or negative logic (be forewarned). As far as the

video idea; I've been looking over the schematic again. And it still seems that *PWAIT is being asserted regardless of the CRTCLK status. I assume the CRTCLK is being clocked slower than the rest of the system. RS didn't think of providing a timing diagram of CRTCLK (the clocking signal for the CRTCLK). I see the normal operation of the video wait circuitry like this: since we are accessing the video, *VIDEO becomes asserted and releases U31's clear signal. We are doing memory access so *RD/*WR will be asserted for the appropriate operation. U32 combines *RD and *WR to generate the data signal (D) for U31. The formula for this would be:

$$D = *1 + **RD + **WR$$

but ** is a double NOT, and a double NOT cancels each other. So the latter can be written:

$$D = *1 + RD + WR$$

also, *1 is always 0 but, 0 ORed with X is always X so throw away 0. Finally,

$$D = RD + WR$$

It can now easily be seen that $D = 1$ when either *RD or *WR are active (we are now in positive logic expressions so an active *RD = logic 1 [isn't this fun]). This is all fine and dandy, but so what? Well, this insures the data input (D) of the flip-flop is always 1, which will be important later on.

Now, output of U31, ACCESINPROG (AIP), is zero since U31 was held at clear until *VIDEO was asserted. U16 ORs *VIDEO (which is asserted) with AIP and obtains *PWAIT. Since 0 OR 0 is always 0, *PWAIT is asserted as logic 0 and hence, a wait state is generated, regardless of who is accessing the video memory.

Ok, how do we get out of the wait state? Well, *PWAIT can be reset only two ways. If *VIDEO goes high or AIP goes high. *VIDEO can't possibly go high if the CPU is in a wait state since processing is suspended! Thus, CRTCLK must occur which passes the D signal (logic 1) to the AIP signal. Since 1 OR 0 is always 1, *PWAIT is released and the CPU continues on it's merry way ... releasing *VIDEO no doubt! (This is why the logic 1 for the D signal was important, it disables the *PWAIT generation.)

It appears, from this, that *PWAIT is asserted for no good reason during any memory access. Did I make a logic error here somewhere? It sounds valid to me. I can't see where CRTCLK

memory access is checked, unless it is decoded internally within U4 (with the help of U5). This could adjust CRTCLK before it reached the *PWAIT circuitry and thus, keep the CPU in suspense a bit longer during a *PWAIT.

--jjkd-- To Shane Dawalt: You do know there is a reason why that wait state is there, Jon't you? Yanking the M1 waits may not seem to cause a problem at first, but the waits are there for a reason -- reliable operation. Note that there are some Model 4's that will crash and burn if you try to run the spooler using the back bank memory under TRSDOS 6.2 and later. The reason? I've never been able to confirm it, but I very much suspect a PAL improperly programmed to not generate M1 waits when running code in the back banks. That is, M1 waits when running in main memory only. Now, that does justify one M1 wait state on machines that were designed to run that way. It certainly doesn't justify two M1 waits and no waits on normal memory cycles. That configuration doesn't make sense.

The reason for the video wait circuitry has nothing to do with memory access cycle time. The wait generator there is specifically to delay a memory cycle until the next horizontal or vertical sync interval. This allows writing to the display ram asynchronously and the video wait generator will sync things up to prevent video hash.

You could probably program a new PAL to go to one M1 wait from two. I don't think that eliminating the wait entirely would be a good idea, even with faster RAM chips, as I believe that the problem is with the design of the supporting circuitry, not the RAM cycle time.

--jjkd-- To Adam Rubin: One advantage is that the restriction on how and when a wait condition is achieved is then determined by the external hardware. The Z80 only samples/recognizes the WAIT input at certain times during the machine cycles.

Shane Dawalt To --jjkd--: Those model 4s you are speaking of probably doesn't include mine. One PAL chip takes care of the M1 generation, and it is regardless of the memory setup (memory setup of banks that is. Of course, it depends on the Model mode setup.)

I've never had problems with the spooler even in TRSDOS 6.0.0. (Tandy engineers tried to get tricky and outsmarted themselves.) HA! Oh, now that wasn't nice was it? Ahem. I was going to state that the extra wait state could have

been from DOS flashing the cursor on the screen (the waits just added up thru the testing time), but beings the interrupts were disabled during the test, that can't hardly be the reason. And I'm not sure that the cursor, flashing at the rate it does, and for only 7 minutes would make that much difference. Without any firm calculations, however, I could have already put my foot in my mouth.

The PALs can only be of two or three levels, thus propagation delays are down about 20 to 30 ns. A 4 MHz clock cycles every 250ns so certainly I would think the circuits could handle that (at room temperature of course). (My machine is stuck in a humidity controlled basement which keeps a pretty good average temp of about 75 in the summer and 68 in the winter.) Of course, the other chips have delays too, so that would probably add up, but I don't believe it would be so bad as to throw the system into a tail spin. Besides, if it doesn't work, yank out the fix! What's one cut trace that a small piece of 30 AWG won't fix.

Adam Rubin To Shane Dawalt: Well, whaddaya know... for some reason I opened the manual for RS's ALDS (don't have PRO-MRAS yet!), and, lo and behold, the appendix says, "Note that during execution of 2-byte op-codes, *M1 is generated as each op-code byte is fetched. These two-byte op-codes always begin with CBH, DDH, EDH, or FDH."

I studied your description of the video wait circuitry, and I think I see a mistake. (I make no claims to being infallible, though!) As I see it, your logic is correct, but your reasoning is flawed. You say, "since we are accessing the video memory, etc.," and describe how *PWAIT is asserted in that case. What I don't see is how you conclude, "*PWAIT is asserted...during any memory access," since you're describing the situation when video memory is accessed by the CPU.

I found the "missing" piece of information in the CRTC spec sheet. CRTCLK is the clock input to the CRTC, and "in alphanumeric terminals, this signal is the character rate." It's this line that determines whether the system's address bus or the CRTC's address lines access video RAM. (It's U59, on page 2, that determines when the address bus is pointing to video RAM.) Thus, one phase of CRTCLK means the Z80 may safely access video RAM, and the other phase means the CRTC is doing so, and the Z80 should wait until the CRTC's done.

This also explains what CRTCLK is. It's the frequency at which characters are displayed on the screen. Working backwards from the CRTC initialization table in 6.3's BOOT/SYS, it should be 1.584 MHz in Mod 4 mode.

Adam Rubin To --jjkd--: It looks to me like the video wait only lasts until a byte from video RAM has been read into the latch for it, so a video wait would last no more than half the time needed to display one row of dots for one character. I think that's $1/2 \times (1/1.584\text{MHz})$, or less than one clock cycle for a 4 MHz CPU clock.

Adam Rubin To --jjkd--: I get the impression that the engineers who did the hardware design of the 4 and 4P generally knew what they were doing.

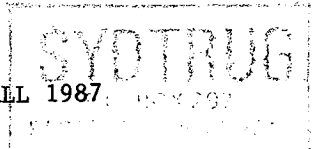
Adam Rubin To Ray Pelzer: That's what the *WAIT input is for. *M1 tells the rest of the system there's an opcode fetch, and *M1 and *IORQ together indicate an interrupt acknowledge (needed for interrupt mode 0).

--jjkd-- To Ray Pelzer: Nope. The Z80 asserts the M1 signal when in opcode fetch mode. S'matter fact, that's how the Z80 family detect interrupt ack, the Z80 asserts M1 and IORQ at the same time, something that would never otherwise happen.

--jjkd-- To Shane Dawalt: The problem with the spooler back bank problem wouldn't show up in 6.0 or 6.1, the 6.2 release was the first that ran as much code as it did in the back bank. Previous spooler problems were almost exclusively the fault of the BASIC interpreter.

--jjkd-- To Ray Pelzer: Check the Z80 tech ref from Zilog. M1 has nothing to do with wait generation internal to the Z80. M1 is asserted by the Z80 so that external logic can recognize M1 to insert a wait state if desired, and for interrupt acknowledge.

Fm --jjkd-- To Adam Rubin: Yes, your analysis is correct for the Model 4 video hardware. I haven't looked at the generation of video waits since the Model 3, where the wait was generated and lasted until the next video blanking interval. These included both horizontal and vertical sync intervals, and the blanking lines between character rows. The Model 4 video is a far faster design with video waits on. On the Model 3 you could disable video waits for fast screen I/O, on the Model 4 the potential wait is much shorter and disabling them would make little difference.



Fm Shane Dawalt To Adam Rubin: RE: my error in saying "*PWAIT is asserted ... on any memory access". Poor choice of words. What I was meaning is that *PWAIT is asserted on any video memory access by the CPU. Meaning that whether the CPU is reading or writing, it is asserted.

After looking at your reply and looking at the CRTC data and looking back over the CRTC/CPU multiplexer circuit, I can see exactly what is (and isn't) happening. Your right, unless I want a nasty looking display, leave the waits where they are!

According to National Semiconductor specs (which should be a good enough source) those multiplexers (U33-35) are switched to the A inputs when CRTCLK is low (CPU access) and to the B inputs when CRTCLK is high (CRTC access). With that known, the previous description only makes more sense. CRTCLK is sent to U31 (the flip-flop) as it's clock. But CRTCLK is inverted so U31 sees *CRTCLK. Assuming the CPU is requesting I/O with the video memory and CRTCLK is high for CRTC access, this should place the CPU into a wait state, via *PWAIT, until CRTCLK goes low. Indeed, it does. When CRTCLK goes low, *CRTCLK goes high. U31 (the flip-flop) passes it's data (which is logic 1 since we are reading/writing video memory) to it's outputs only at the positive transition of it's clock input. This means ACCESSINPROG goes high and *PWAIT becomes inactive.

Another interesting idea: what if the CPU already has access? Well, the *VIDEO signal is asserted and the CLEAR input is released on U31. U31 generates ACCESINPROG which is now low and so a *PWAIT is generated! Because U31 is positive edge triggered flip-flop, *PWAIT cannot possibly be cleared until CRTCLK goes low then back high again. This certainly does synchronize the CPU video memory request with the horizontal character writing, but it is strange to wait on EACH character. I know, how many times do you write to your screen?!

I'm almost sure the last "idea" would occur since CRTCLK and PCLK are phased together. I know it is in the gate array version, and I'm sure it is in the non-gate array version. The gate arrays use a neat little PLL circuit. I suppose the non-gate array uses internal PAL gates and signals developed by U5 (the counter) to synchronize CRTCLK with the main 20MHz clock.

Fm --jjkd-- To Adam Rubin: BTW, to find a system where a video wait lasts until the next vertical sync interval, merely look for the nearest IBM PC with a color/graphics display adapter. Because of the way this board is designed, you can't write to the display RAM without causing hash on the display (ala the Model 3/4), and there is no hardware assist as there is on the Model 3/4. Most MS-DOS software that accesses the display does so via direct memory writes, and includes a software loop that checks for the sync pulses and delays accordingly.

To safely update video memory on the widest variety of IBM look-alike hardware without "chroma-blizzard", you can write exactly one byte during horizontal sync, and something like 160 (I don't recall exactly) during vertical sync. To simplify things, most software waits for a sync pulse edge for each byte written to the display RAM, slowing things down tremendously. It is still far faster than using the BIOS.

Fm Shane Dawalt To --jjkd--: There should be a straightforward way to stuff the RAM while the CRTC does it's thing. If not straightforward, then some strange action to read and write asynchronously from/to the RAMs.

Fm Shane Dawalt To --jjkd--: Dual ported RAM was my first thought. But, there isn't too many of them on the market. Probably would have to be developed before it could be used. Anyway, the CRTC interface in your Tandy 2000 sounds rather interesting!

Fm --jjkd-- To Shane Dawalt: Dual-ported RAM is commonly available from TI and other vendors. Most are dynamic implementations rather than static, however, and as such are generally intended for somewhat more complex video implementations that those we are discussing.

Fm Adam Rubin To --jjkd--: Ah, so THAT's why some of the Model 4 docs list a bit for "Video Waits On/Off" when the corresponding pin in the schematics is marked "NC".

Fm Adam Rubin To Shane Dawalt: The CRTC spec sheet lists four approaches for solving contentions between the CPU and CRTC for video memory, and I think we've mentioned all of them already. One, CPU always gets priority (causing "hash"). Two, CPU synchronized by interrupt to perform accesses only during retrace. Three, CPU synchronized with wait state(s). Four, synchronize with M6800 CPU as described by jjkd.

Fm --jjkd-- To Adam Rubin: Direct video writing on the IBM is indeed as clumsy and slow as that (syncing writes to the horizontal and vertical sync pulses).

Going through the BIOS is even worse because there is no "string" output, the overhead on each call for every byte is a killer, and you will never get more than one byte per sync pulse, whereas if you are careful you can get one byte per horizontal sync and over a hundred on vertical sync, if you are ready with that many bytes to go at once and you are doing it yourself.

ANSI.SYS is nasty too because it turns off the video when it does the block move to scroll the screen. Causes the most awful flashing on the display.

Fm Adam Rubin To Shane Dawalt: I'm not sure if there's any way to totally remove the video waits without getting hash (unless you have both the CRTIC and CPU using the same clock), though a separate processor for the display could keep the "main" CPU from waiting. (Why not have the display CPU handle the keyboard, too?) As I calculated before, though, on the M4 a video wait last about half a clock cycle, so I think I'll leave it the way it is for now. The really lengthy wait on a M4, of course, is for floppy disk I/O, which is about 32 microseconds in DDEN (64 in SDEN) between bytes, but most software assumes the I/O is done when the DOS call returns. As (I think) Roy pointed out a while ago, you could probably best increase the display speed by using the 64180's hardware multiply and divide in the *DO driver.

I know little about the inner workings of MS-DOS, but bear in mind that not all programs use the fastest possible method of writing files to disk. (Compare BASIC in 6.2 and 6.3, for example.) That's probably why some programs that create a .BAK file take less time to save a document than some that don't.

Fm Adam Rubin To --jjkd--: About all I can say to that is, double yech! The Model 4's video waits sound like the best technique for a non-6800 system. I'll have to try ANSI.SYS on an MS-DOS system... sounds, er, interesting. Question for discussion: If the BIOS routine for video output had been faster, would we be facing quite as many ill-behaved software packages today?

Fm H. Brothers To Adam Rubin: If the BIOS routine had been faster AND if it had been more

useful (the EGA and AT video BIOS routines are a partial step in the right direction).

Fm --jjkd-- To Adam Rubin: If the IBM BIOS routines for video display (INT 10) and serial communications (INT 14) were far better, yes, there would be a lot less in the way of ill-behaved programs out there today.

The set that TRSDOS 6 allows via @DSP, @DSPLY and @VDCTL would have been a better start, along with mods to cover the attribute and graphics functions.

Fm Bryan Headley To --jjkd--: Yeah, providing that this imaginary @vdctl for PC-compatibles can be made to work with a user-defined area on the screen.

Fm --jjkd-- To Bryan Headley: Oh, no, I don't think that anybody believes that it would be practical to "re-tread" this kind of feature into the IBMs now, it is far too late. The house has long since burnt to the ground.

The statement was that "if from day one, IBM had better BIOS routines, including features such as we have been discussing, more existing IBM programs would currently be well-behaved." That is, the need for direct hardware access would have been considerably lessened.

Results of tests

Results of speed test programs

The following table summarizes the results of each test applied to each machine. Columns 1-8 describe the various machines as follows:

1. Very old model 4
2. Newer Model 4, non-clustered
3. non-clustered, B&W Model 4P
4. clustered, green, 4p e/w XLR8
5. IBM PC, 4.77 MHz
6. IBM PC e/w AST Hotshot/286
7. Leading Edge Model M, 8 MHz
8. AST Premium/286, 10 MHz

	Speed results (all times in seconds)							
--test--	--1---	2---	3---	4---	5---	6---	7---	8--
Rubins	149	131	112	70	144	25	115	27
dhamp	189	166	152	101	42	11	41	6
fib	965	863	773	536	758	159	628	129
float	1877	1657	1461	921	352	64	168	29
quick	2316	2053	1800	1186	212	39	173	32
savage	2812	2498	2221	1375	267	52	213	38
sieve	636	568	509	338	106	89	16	15

A few comments are in order. C optimizers under MS-DOS are highly advanced. BYTE had a follow up article which addressed the issue of questionable results caused by optimization. They recommended that the optimizer be turned off to avoid altering sequences of code carefully designed to be executed. This was extremely evident on the float test. Thus, all MS-DOS C programs were compiled with Microsoft's MSC version 4.0 without optimization (use of switch /Od).

We compiled the Model 4 versions with PRO-MC. Since our optimizer is far from the sophistication of the one supplied with MSC, these versions were compiled with optimization. Of course, double precision floating point operations are a little slow compared to those implemented under MS-DOS; thus, the test results may be skewed for those C programs with intensive floating point operations. In particular, it may prove useful to compare the times of these programs revised for BASIC interpreter processing and compare the results with Model 4 BASIC versus Microsoft's BASICA or GWBASIC interpreters. I leave that as an exercise for our readers; please report your results.

All times were calculated by taking the difference in system time between the start and end times; the standard C function time() was used in the case of the C programs. The examples derived from BYTE used a function called difftime() which returned the difference expressed as a whole number as a double (i.e. no fractional part). For the Model 4 testing, I wrote such a function and put it into the math.h library; MSC includes such a function. This function is also printed here.

All C programs were directly compiled using either PRO-MC or MSC without alteration of any source. The use of the #ifdefs provides this capability for any differences between machine environments (the only use I had to make of it was to enable the math library to be searched under PRO-MC).

Now for some brief analysis. Certainly the differences between Model 4s (excepting the XLR8er case) prove that all Model 4s were not created equally. In fact, they weren't. Model 4s varied from an effective 3 MHz machine to a 4MHz machine - depending on when it was manufactured. Also, the RUBINS test shows that a Model 4 equipped with an XLR8 board can be much faster than a standard PC (or turbo) depending on the mix of code being executed. Of course, none of these tests evaluate I/O,

which is somewhat necessary in any kind of operating environment. I make note that a pokey old PC can even be improved to compete quite favorably with a speed demon such as the AST Premium/286 which runs at 10MHz and zero wait states. Of course, I achieved that with the Hotshot card, also manufactured by AST, which runs at 10 MHz and incorporates a 16K memory cache (I purchased my card from Moneysworth for about \$325).

If you are going to continue using your Model 4 for a few more years, the XLR8 board should prove to be an excellent investment. It provides an additional 256K of memory and the much faster 64180 CPU.

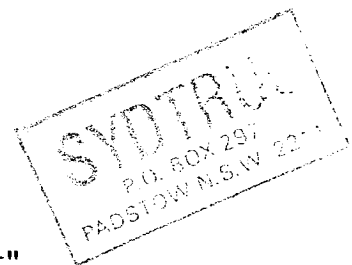
Test Programs

Speed test programs

```

=====
/* difftime.ccc */
#include <time.h>
double difftime(finish,start)
time_t finish, start;
{
    return (double) (finish - start);
}
=====
/* dhamp.c */
#include <stdio.h>
#include <time.h>
#ifdef dos
#include <math.h>
#endif
#define FIB          24
#define TINY         100
#define MAXINT       179
#define LITTLE       1000
#define SMALL        9000
#define PRECISION    .000001
#define FILENAME     "zyxw.vut"
#define NUMTEST      6
#define ILOOP        1
#ifdef ERR
#define ERR          -1
#endif
struct {
    int      cresult;
    int      irect;
    int      cprresult;
    unsigned  irect;
    long     lresult;
    double   dresult;
} results;

```



```

main()
{  char buf1[TINY],buf2[TINY];
   int i = 0;
   unsigned fib();
   long square, sq();
   double dmath, sroot(), dply();
   time_t start, finish;
   printf("Start of %d iterations\n",ILOOP);
   time(&start);
   while (i < NUMTEST ) {
       switch(i) {
           case (0):
               results.cresult =
stest(buf1,buf2);
               printf("\ncresult =
%d\n",results.cresult);
               break;
           case (1):
               results.cresult = intest();
               printf("\niresult =
%d\n",results.iresult);
               break;
           case (2):
               results.uresult = fib(FIB);
               printf("\nuresult =
%u\n",results.uresult);
               break;
           case (3):
               square = 0L;
               results.lresult = sq(square);
               square = sq(results.lresult);
               printf("\nlresult =
%ld\n",results.lresult);
               printf("\n square = %ld\n",
square);
               break;
           case (4):
               results.dresult =
sroot((double) results.lresult);
               printf("\ndresult =
%f\n",results.dresult);
               dmath = dply(results.dresult);
               printf(" dmath = %f\n",
dmath);
               break;
           case (5):
               results.cprresult = mcopy();
               printf("\n copy =
%d\n",results.cprresult);
               break;
           default:
               break;
       }
       ++i;
   }
   time(&finish);
   printf("\ndone using %f
seconds\n",difftime(finish,start));
}
long sq(big)

```

```

long big;
{  int i;
   static long j = 1L;
   if (!big)
       for (i=0; i < SMALL; ++i) {
           big += j;
           j += 2;
       }
   else
       for (i=0; i < SMALL; ++i) {
           j -= 2;
           big -= j;
       }
   return (big);
}
double sroot(num)
double num;
{
   double temp1, temp2, abs();
   temp2 = num / 2.0;
   temp1 = num;
   while (temp1 > PRECISION * temp2) {
       temp1 = (num / temp2) - temp2;
       temp1 = abs(temp1);
       temp2 = ((num / temp2) + temp2) / 2.0;
   }
   return (temp2);
}
double abs(x)
double x;
{  return (x < 0 ? -x : x);
}
double dply(x)
double x;
{  int i = TINY;   double y;
   while (i-->0) {
       y = x * x * x * x * x * x * x * x;
       y = y / x / x / x / x / x / x;
       y = y + x + x + x + x + x + x;
       y = y - x - x - x - x - x - x;
   }
   return (y);
}
unsigned fib(x)
int x;
{  if (x > 2)
   return ( fib(x-1) + fib(x-2));
   else
   return (1);
}
int stest(b1, b2)
char *b1, *b2;
{  int i, j;   void mstrcpy();
   for (i=0, j=0; i < SMALL; ++i) {
       mstrcpy(b1,"0123456789abcdef");
       mstrcpy(b2,"0123456789abcdee");
       j += mstrcmp(b1,b2);
   }
   return (j);
}

```

```

int mstrcmp(c,d)
char *c, *d;
{ while (*c == *d) {
    if (!*c)
        return (0);
    ++c;
    ++d;
}
return (*c-*d);
}

void mstrcpy(c,d)
char *c, *d;
{ while (*c++ = *d++)
    ;
}

int mcopy()
{ FILE *fp, *fopen();
char buf[TINY];
int i, j;
mstrcpy(buf, "Disk I/O test");
if ((fp = fopen(FILENAME, "w")) == NULL) {
    printf("Cannot open file");
    exit(ERR);
}
i = 0;
while (++i < LITTLE)
    for (j=0; buf[j]; ++j)
        putc(buf[j], fp);
fclose(fp);
return (i);
}

int intest()
{ int i, j, k, sum;
for (i=0; i < LITTLE; ++i) {
    sum = 0;
    for (j = 0, k = 1; j < MAXINT; ++j) {
        sum += k;
        k += 2;
    }
}
return (sum);
}

```

```

=====
/* fib.c */
#include <stdio.h>
#include <time.h>
#define NTIMES 100 /* iterate # times */
#define NUMBER 24 /*biggest with 16 bits */
main() /* compute Fibonacci value */
{ int i; time_t start, finish;
unsigned value, fib();
printf("%d iterations: ",NTIMES);
time(&start);
for (i=1; i<=NTIMES; i++)
    value = fib(NUMBER);
time(&finish);
printf("Fibonacci(%d)=%u.\n",NUMBER,value);
printf("Execution time was %f seconds\n",
difftime(finish,start));
exit(0);
}

```

```

}
unsigned fib(x) /* compute recursively */
int x;
{ if (x>2)
    return (fib(x-1)+fib(x-2));
else
    return (1);
}
=====
/* float.c simple benchmark for testing
* floating-point speed of C libraries;
* does repeated multiplications and divisions
* in a loop that is large enough to make the
* looping time insignificant
*/
#include <stdio.h>
#include <time.h>
#define CONST1 3.141597E0
#define CONST2 1.7839032E4
#define COUNT 10000
main()
{ double a, b, c;
int i; time_t start, finish;
a = CONST1;
b = CONST2;
time(&start);
for (i=0; i<COUNT; ++i) {
    c = a * b; c = c / a;
    c = a * b; c = c / a;
    c = a * b; c = c / a;
    c = a * b; c = c / a;
    c = a * b; c = c / a;
    c = a * b; c = c / a;
    c = a * b; c = c / a;
}
time(&finish);
printf("Done in %f
seconds\n",difftime(finish,start));
}

```

```

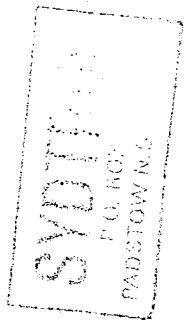
=====
/* quick.c - sorting benchmark
* calls randomly the number of times
* specified by MAXNUM to create an integer
* array of long integers, then does a
* quicksort on the array of longs. The
* program does this for the number of times
* specified by COUNT.
*/

```

```

#include <stdio.h>
#include <time.h>
#define MAXNUM 1000
#define COUNT 100
#define MODULUS ((long) 0x20000)
#define C 13849L
#define A 25173L
long seed = 7L;
long random();
long buffer[MAXNUM] = {0};
main()
{ int i,j;

```



```

long temp;
time_t start, finish;
printf("Filling array and sorting %d
times\n",COUNT);
time(&start);
for (i=0; i<COUNT; ++i) {
    for (j=0; j<MAXNUM; ++j) {
        temp = random(MODULUS);
        if (temp < 0L)
            temp = (-temp);
        buffer[j]=temp;
    }
    printf("Buffer full, iteration %d\n",i);
    quick(0,MAXNUM-1,buffer);
}
time(&finish);
printf("Finished %d iterations in %f
seconds\n",COUNT,difftime(finish,start));
}
quick(lo, hi, base)
int lo, hi; long base[];
{
    int i, j;
    long pivot, temp;
    if (lo < hi)
    {
        for (i=lo, j=hi, pivot=base[hi]; i<j;)
        {
            while (i<hi && base[i]<=pivot)
                ++i;
            while (j>lo && base[j]>=pivot)
                --j;
            if (i<j)
            {
                temp = base[i];
                base[i] = base[j];
                base[j] = temp;
            }
        }
        temp = base[i];
        base[i] = base[hi];
        base[hi] = temp;
        quick(lo,i-1,base);
        quick(i+1,hi,base);
    }
}
long random(size)
long size;
{
    seed = seed * A + C;
    return (seed % size);
}
}
=====
/* savage.c - floating-point speed and
* accuracy test. C version derived from BASIC
* version which appeared in Dr. Dobb's
* Journal,Sep 1983, pp 120-122
*/
#include <stdio.h>
#include <time.h>
#define ILOOP 2500

```

```

extern double tan(), atan(), exp(), log(),
sqrt();
main()
{
    int i; double a; time_t start, finish;
    printf("Start of %d iterations\n",ILOOP);
    time(&start);
    a = 1.0;
    for (i=1; i<=(ILOOP-1); i++)
        a = tan(atan(exp(log(sqrt(a*a)))))+1.0;
    printf("a=%20.14e\n",a);
    time(&finish);
    printf("done using %f seconds\n",
difftime(finish,start));
}
=====
/* Eratosthenes Sieve Prime-Number program
* from BYTE January 1983
*/
#include <stdio.h>
#include <time.h>
#ifndef TRUE
#define TRUE 1
#endif
#ifndef FALSE
#define FALSE 0
#endif
#define SIZE 8190
char flags[SIZE+1];
main()
{
    int i, prime, k, count, iter;
    time_t start, finish;
    printf("100 iterations\n");
    time(&start);
    for (iter=1; iter <= 100; iter++)
    {
        count = 0; /* prime counter */
        for (i=0; i<=SIZE; i++)
            flags[i]=TRUE;
        for (i=0; i<=SIZE; i++)
        {
            if (flags[i]) /* found a prime */
            {
                prime = i+i+3; /* twice index + 3 */
                /* printf("\n%d",prime); */
                for (k=i+prime; k<=SIZE; k+=prime)
                    flags[k]=FALSE; /* kill all */
                /* multiple */
                count++; /* primes found */
            }
        }
    }
    time(&finish);
    printf("%d primes found in %f seconds\n",
count,difftime(finish,start));
}
=====
;RUBINS/ASM
DOS5 EQU @@1
DOS6 EQU .NOT.DOS5
IF DOS6
@DSPLY EQU 10

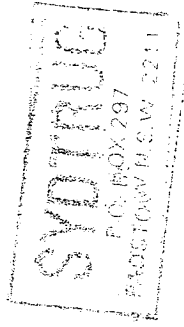
```

```

@TIME EQU 19
@HEXDEC EQU 97
@MUL16 EQU 91
SVC MACRO #NUM
    LD A,#NUM
    RST 28H
    ENDM
    ENDIF
    IF DOS5
@DSPLY EQU 4467H
TIME$ EQU 4217H
@MUL16 EQU 444EH
@DIV16 EQU 4451H
SVC MACRO #NUM
    CALL #NUM
    ENDM
    ENDIF
    ORG 6000H
BEGIN:
    IF DOS6
    LD HL,BUFF$
    SVC @TIME
    EX DE,HL
    ENDIF
    IF DOS5
    LD DE,START
    LD HL,TIME$
    ENDIF
    PUSH HL
    LD BC,3
    LDIR
    LD DE,0100H
LOOP1 LD BC,0000H
LOOP2 DEC BC
    LD A,B
    OR C
    JR NZ,LOOP2
    DEC DE
    LD A,D
    OR E
    JR NZ,LOOP1
    POP HL
    LD DE,FINISH
    LD C,3
    LDIR
    LD HL,START+2 ;Normalize
    LD DE,FINISH+2 ; the
    LD A,(DE) ; hours
    SUB (HL) ; to
    JR NC,$+4 ; avoid
    ADD A,24 ; overflow
    LD (DE),A
    LD (HL),0
    LD HL,START ;Start
    CALL HMS2S ; seconds
    PUSH HL
    LD HL,FINISH ;Finish
    CALL HMS2S ; seconds
    POP DE ;Finish
    OR A

SVC HL,DE ; minus
LD DE,MSG$2 ; start
SVC @HEXDEC
LD HL,MSG$1
SVC @DSPLY
RET
HMS2S PUSH HL
    POP IX
    LD L,(IX+2) ;Hours
    LD H,0 ; times
    IF DOS6
    LD C,60 ; 60
    ENDIF
    IF DOS5
    LD A,60
    ENDIF
    SVC @MUL16 ; to HLA
    ADD A,(IX+1) ;Plus mins
    JR NC,$+3
    INC L
    LD H,L
    LD L,A
    IF DOS6
    LD C,60 ;Times 60
    ENDIF
    IF DOS5
    LD A,60
    ENDIF
    SVC @MUL16 ; to HLA
    ADD A,(IX+0) ;Plus secs
    JR NC,$+3
    INC L
    LD H,L
    LD L,A
    RET
    IF DOS5
    LD B,5
    LD A,' '
    LD (DE),A
    INC DE
    DJNZ HEXD1
    PUSH DE
    HEXD2 DEC DE
    LD A,10
    SVC @DIV16
    ADD A,'0'
    LD (DE),A
    LD A,H
    OR L
    JR NZ,HEXD2
    POP DE
    RET
    ENDIF
    BUFF$ DS 8
    START DS 3
    FINISH DS 3
    MSG$1 DB 'Rubin''s rating = '
    MSG$2 DB 'xxxxx seconds',13
    END BEGIN
=====

```



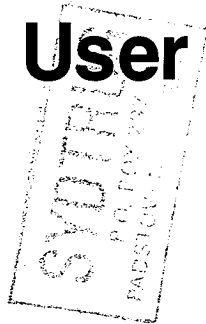
```

;RUBINS86.ASM - For ED/ASM-86 (MS-DOS)
@DSPLY EQU 9
@TIME EQU 44
@EXIT EQU 76
TIME STRUC
SECS DB 0
MINS DB 0
HOUR DB 0
TIME ENDS
SVC MACRO NUM
    MOV AH,NUM
    INT 21H
ENDM
ENTRY JMP BEGIN
START TIME <>
FINISH TIME <>
MSG1 DB 'Rubin',27H,'s rating = '
MSG2 DB 'xxxxx seconds',13,10,'$'
BEGIN: SVC @TIME
        MOV [START+HOUR],CH
        MOV [START+MINS],CL
        MOV [START+SECS],DH
        MOV CX,0100H
LOOP1 MOV BX,0000H
LOOP2 DEC BX
        MOV AL,BH
        OR AL,BL
        JNZ LOOP2
        DEC CX
        MOV AL,CH
        OR AL,CL
        JNZ LOOP1
        SVC @TIME
        MOV [FINISH+HOUR],CH
        MOV [FINISH+MINS],CL
        MOV [FINISH+SECS],DH
        MOV SI,OFFSET START+HOUR
        MOV DI,OFFSET FINISH+HOUR
        MOV AL,[DI]
        SUB AL,[SI]
        JNB NOROLL
        ADD AL,24
NOROLL MOV [DI],AL
        MOV BYTE PTR [SI],0
        MOV SI,OFFSET START
        CALL HMS2S
        PUSH AX
        MOV SI,OFFSET FINISH
        CALL HMS2S
        POP DX
        SUB AX,DX
        MOV DX,AX
        MOV BX,OFFSET MSG2
        CALL @HEXDEC
        MOV DX,OFFSET MSG1
        SVC @DSPLY
        MOV AL,0
        SVC @EXIT
;
; Routine to convert hours, mins, secs to secs
; SI => pointer to time structure
; AX <= value of time in seconds
;
HMS2S PROC NEAR
    MOV AL,[SI+2] ;Hours
    XOR AH,AH ; times
    MOV BX,60 ; 60
    MUL BX
    MOV BL,[SI+1] ;Plus
    ADD AX,BX ; mins
    MOV BL,60 ;Times
    MUL BX ; 60
    MOV BL,[SI] ;Plus
    ADD AX,BX ; secs
    RET
HMS2S ENDP
;
; Routine to convert 16-bit binary to ASCII
; DX => 16-bit number
; BX => pointer to 5-byte buffer
; bx <= buffer + 1
;
@HEXDEC PROC NEAR
    MOV CX,5
    MOV AL,' '
HEXD1 MOV [BX],AL
        INC BX
        LOOP HEXD1 ;until CX=0
        PUSH BX ;set
        MOV AX,DX ; dividend low
HEXD2 DEC BX
        XOR DX,DX ;dividend hi
        MOV CX,10 ;divisor
        DIV CX ;dx:ax by cx
        ADD DL,'0' ;remainder to
        MOV [BX],DL ; ASCII
        CMP AX,0
        JNZ HEXD2
        POP BX
        RET
@HEXDEC ENDP
END BEGIN
=====

```

Applications

for the User



HIRES Graphics for MC

Hi-Res Graphics for MC with a GRPLIB Interface

by Harry G. Clayton, Jr.

I got this interface working in October 1986 and as I write this it's now September '87 -- Why the Heck didn't I submit this sooner? Well..., you see it's like this: this interface is really an interim solution until I have time to attack GRPLIB with UNREL and put it back together as a self-contained MC Hi-Res Library (already in progress); and besides, just how many people out there of the 900 or so that subscribe to TMQ are really interested in this? After reading TMQ issue III the answer to that question was at least two. So, after much procrastination, here goes.

GRPLIB is a Hi-Res graphics library in MicroSoft REL Library format. This library is for use with MicroSoft's FORTRAN-80 to support the Radio Shack Hi-Res Graphics Board for the Model 3 or 4. GRPLIB and it's documentation are provided with the graphics board.

The objective is to provide a means to access the FORTRAN SUBROUTINES contained in GRPLIB as C functions so that we can write all of our code in C. The first step is to read the

manuals. Pages 5-26 and 5-27 of the MC manual describe how function arguments are passed to the function. Appendix C, pages 254-256 of the Model 4 FORTRAN manual describe how parameters are passed to a subroutine.

Since I knew you wouldn't read the manuals, I will now summarize. In a C function invocation, the argument values are pushed to the stack. For non-void functions (which return a value), the return value is returned in the primary register for the value type. Since the only non-void functions we will have are of type int, the value will be returned in the HL register pair. The order in which the values are pushed is best demonstrated by the following pseudo-code which summarizes the code that MC would generate for a function invocation having three arguments.

```

; whatever_type a, b, c;
; int          z, function();
; z = function(a,b,c);
    push c          ;the number of bytes
                    ; on the stack
    push b          ; that are occupied by
                    ; an argument
    push a          ; depends on the
                    ; argument type.
CALL FUNCTION      ;function leaves value
                    ; for z in HL

    pop
    pop
    pop
LD    (&z),HL

```

A FORTRAN subroutine call places the addresses of the parameter values in the registers, i.e., the registers will contain pointers to the values being passed. When the number of parameters is 3 or less, HL will point to parameter_1, DE will point to parameter_2, and BC will point to parameter_3. If the number of parameters is greater than 3, HL and DE will be used the same way, but BC will point to a table containing pointers to parameter_3 through parameter_n. In other words, BC points to an array of pointers. The following table may be helpful.

for CALL SUB(A,B,C) for CALL SUB(A,B,C,D,E,F)

HL points to A	HL points to A
DE points to B	DE points to B
BC points to C	location (BC+0) points to C
	location (BC+2) points to D
	location (BC+4) points to E
	location (BC+6) points to F

The routines in GRPLIB will be expecting parameter pointers in register pairs, and the C function invocation will be providing the parameter values on the stack. All we need to do in order to interface the GRPLIB routines with MC is to write some assembly language that figures out the parameters' locations on the stack and puts the addresses into the appropriate register pairs and then calls the routine in GRPLIB.

The file HRFUNC/MAC defines the MC Hi-Res functions. This was written to be assembled by M80 (Macro-80) and uses the MACRO's contained in the file HRMACS/MAC. Each function defined in HRFUNC is the C equivalent of the FORTRAN SUBROUTINE contained in GRPLIB. Notice that the name of the function must be different from the name of the routine in GRPLIB. In most cases I have simply prefixed an 'h' to the GRPLIB name. These functions perform the parameter translation, call the GRPLIB routine, and place the return value in HL (where appropriate). The GRPLIB documentation describes the function and use of each of the routines so I will not.

HRFUNC is written to be concise. I got tired of pulling out the HRG manual just to look up the order of parameters, so I use this file as a 'cheat-sheet' whenever I'm writing Hi-Res Graphics code in FORTRAN or C.

I kept HRFUNC small by using the macro \$HARG to perform the parameter translation. The major reason that this is written for M80 is that I knew I could write this as a M80 macro, and I am still not up to speed with MRAS macro's. \$HARG is in the file HRMACS/MAC. When invoking \$HARG to perform the parameter translation, it is given the same number parameters as needs to be translated. The \$HARG parameters inform \$HARG of the function argument types: I for int, F for float, P for pointer. For example, the function hcircle() has as its argument types - hcircle(int,int,float,float,float). These are translated by: \$HARG I,I,F,F,F

I will make no attempt to explain \$HARG, read the M80 manual if you really want to know. I use M80 macro's a lot, and I'm thoroughly convinced that they are some of the most deviously unreadable programming constructs ever devised. But I keep using them anyway (and Dr. Wirth thinks programming in BASIC causes irreparable brain damage).

The easiest way to see what \$HARG does is to look at the file HRFUNC/LST, the assembly

listing. The code that does the argument translation can now easily be seen. The only thing at all tricky involves the arguments of type float. MC, as any K&R C, defaults all floating point to double precision, while the GRPLIB routines expect single precision. After carefully checking that the floating point formats used by MC and FORTRAN are the same (thank you Rich), I decided that I could do a simple double to single conversion by ignoring the four least significant bytes of the mantissa. The double placed on the stack by MC is an eight byte value. A pointer to the double actually points to the least significant byte of the mantissa. By adding four to the pointer, it will then point to a four byte single precision float.

NOTICE to MRAS users -- Instead of attempting to implement an MRAS version of \$HARG, use HRFUNC/LST to create MRAS source without macro's. In fact, you will find that there are some unnecessary instructions that can be removed, and some ways to accomplish the same results with fewer machine cycles. It's not easy to write a macro which can produce optimal code for each of the many possible situations it needs to support.

Assembling HRFUNC will produce HRFUNC/REL which can then be linked with the compiled C program which uses the Hi-Res functions.

*****IMPORTANT*****

The /REL modules must be linked in a particular order. There are label conflicts between MC Libraries and GRPLIB and FORLIB. This will not prevent getting a working object module, but you will not get through a link without some warning messages. The MC installation library contains functions such as line() which have names identical to routines contained in HRGLIB. HRGLIB requires some routines from FORLIB and FORLIB contains routines such as \$MEMORY with the same names as routines in the MC libraries.

The proper order to link is: The C program main first, then any other modules of the C program, then link HRFUNC. Next, search the libraries in the following order: LIBC, MATH, LIBA, GRPLIB, IN, and FORLIB. The file MCHR/JCL is a revised MC80 to automate the compile, assemble, and link of an MC program which uses HR functions. I have not attempted to use MRAS/MLINK with HRFUNC/REL, but I see no reason for it to not work with them.

Since linking HRFUNC loads all of the functions, even if you only use two or three of them, with large programs you may need to assemble a smaller version of HRFUNC which contains only the functions used by your program.

The next step would be to place the HR functions in a library. I decided against this step primarily because of the label conflicts. Also I don't like having to search FORLIB because of its' additional label conflicts. My conclusion was to use HRFUNC as it is for MC Hi-Res support without wasting any more time on it, and instead divert my attack to a different angle.

Instead of simply interfacing to GRPLIB, why not modify it to be directly compatible with MC? This would allow me to eliminate the label conflicts, and the modules for FORLIB could be incorporated making the FORLIB search unnecessary. With the availability of UNREL this approach became even more attractive, and that is what I am now working on. I had hoped to have that project finished by now, which is why I hadn't sent HRFUNC to TMQ sooner.

Just a quick description of the few remaining files I sent.

DECHRUNC/H is a header file containing the declarations for all the functions in HRFUNC.

HRGFUNC/CCC is the C source for some more Hi-Res functions that are composed of functions in HRFUNC.

BALLBOX/CCC is the C source for a program that uses many of the functions in HRFUNC. The program draws a border around the screen then draws ten random rectangles which may not intersect one another. A 'pong' ball is then let loose to bounce where it may.

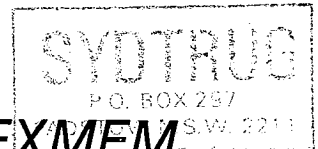
HRGTEST/CCC is the C source for program which uses all of the functions in HRFUNC. This is a C version of the FORTRAN HRGTEST that was provided with the Radio Shack Hi-Res Board. I did tweak it a bit.

STDH/H is my standard header file which is needed to compile BALLBOX and HRGTEST.

[Editor's note: Due to the extent of the files accompanying this article, it was deemed

prudent that the files would only appear in DISK NOTES 2.2. We may also get them uploaded to Compuserve in an ARC file]

Additional Hardware Note: I was having intermittent problems with my Hi-Res graphics board. I noticed that the female connector on the Hi-Res Board has gold plated contacts, but that the dual row header on the CPU board that the Hi-Res board mates with is typical Tandy tin. I spent a couple bucks on a gold plate dual row header. Then spent an hour or so unsoldering the tin plate piece of crap (with much muttering of 'death to Fort Worth bean counters'), and installing the gold plate header. Problem solved. If you have never unsoldered dual row headers before, I wouldn't recommend learning on your Micro. Find a friend who is an electronics technician. Headers fit their holes in the PC Board rather snugly even before being soldered, and getting them unsoldered and out of the board without wrecking the clad a bit is an acquired skill. I also highly recommend Gold-Plug 80 connectors for Hard Disk users. I installed them on the CPU card edge where the Hard Disk Cable connects and where the printer cable connects. I had been having problems with the Hard Disk freezing up. Installing the Gold Plugs is not too hard. I saw a reference to the 'Pink-Pearl fix' on the LDOS SIG. I threw my Pink-Pearls away after I installed Gold Plugs on my Model I. I was hoping that the Model 4 would not need them, but Tandy has an inexhaustible supply of ignorant bean counters. I wonder if the Mac II has gold plated connectors.



BASIC <-> @EXMEM

BUFDATA/BAS - by Edwin Ketola

I would like to let you know that I appreciate your EXMEM in THE MISOSYS QUARTERLY Volume I.iii. You mentioned that you would like to know how people are using your EXMEM. I have always had the problem of running out of memory space, even when programming on main frames. I try to save memory by removing comments, putting multiple statements on one line, shortening variable names, and chaining sections of program code. Your EXMEM allows me to essentially chain my data arrays.

Because other people may be interested in using EXMEM to store data from BASIC programs,

I have provided (1) a short article of how I use EXMEM for data storage, and (2) a listing of my BASIC program,

BUFDATA/BAS: A BASIC application program to store data in extended memory

@EXMEM/CMD can be called from BASIC in order to store data in extended memory. BUFDATA/BAS shows how this can be done. Almost every program line contains a comment, therefore this article will not go into detail on how BUFDATA/BAS works. All variables are defined either in the main program (in lines 130 to 180) or in the Subroutines (in lines 4100 to 4250). Lines 10 to 810 contain a dummy main program that should be replaced by your program. Lines 4000 to 4550 contain two Subroutines that actually call EXMEM.

The main dummy program generates data using RND to fill up an array, A(X1%,X2%). In practice, data may be obtained from a data base on disk. The Subroutines are set up so that memory space will be used most economically if the first dimension, X1% of the array is larger than the second dimension, X2%. X1% is increased until it is a multiple of 64 single precision numbers (256 bytes) so that the Subroutines can work with one set of X1% numbers at a time. The data array is set up in such a way that as much as possible resides in Bank 0: A(X1%,M%) with M% being the largest number less than X2% that will fit in Bank 0. The additional data (M% to X2%) is accessed by inputting one set of X1% numbers into the last position, M%, of the array.

To check that the subroutines are doing their job, the main program calculates the sum of all X1% for each X2%. Before the data is output into extended memory, SUM(X2%) is calculated. In practice, any other iterative mathematical calculations would be performed in place of SUMCHK(X2%). Of course, any mathematical calculations that need only to be performed once for each X2% should be done without EXMEM by setting up A(X1%) and laying each set of X1% on top of the previous one.

The first time Output (GOSUB 4270) is called, it offsets the address for A(X1%,M%) by 8 bytes. The explanation for this is in the TRSDOS 6.2 manual page 2-175 which states that when using VARPTR (array variable), the first element in the array is preceded by a sequence of 8 bytes.

There are two error checking routines in BUFDATA/BAS that have not been tested, because the errors have not occurred:

(1) In BASIC, mathematical calculations using floating point numbers usually do not produce exactly identical answers. Lines 670 to 740 check for identity between two floating point numbers and may have been expected to give answers that were very close but not identical.

(2) In lines 4420 and 4530, the Subroutines check for the successful completion of the SVC call. Just in case of a rare failure, T% should allow the SVC call to be tried twice before aborting the program.

The following information is given to help you write your own main program. As written, the Subroutines expect that your main program will:

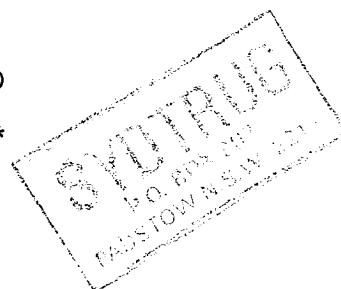
- (1) Use OPTION BASE 1.
- (2) Dimension the array, A(X1%,M%) and E%(6).
- (3) Set S%=0 to initialize the Subroutines.
- (4) Use J% as the pointer to the second dimension of array with J%>=M%.
- (5) Use X1% as the size of the first dimension of array with X1% a multiple of 64 single precision numbers.
- (6) Call the Output Subroutine first.

Your main program should expect that the subroutines will output (GOSUB 4270), or Input (GOSUB 4460) one set of X1% numbers in the M% element of the second dimension of the array.

```

10 '***** MAIN PROGRAM *****
20 'Main Program is a dummy program that uses EXMEM from THE MISOSYS QUARTERLY.
30 'Bank 1 is used a data storage buffer for an Array 'A'
40 'PROGRAM SEQUENCE: Generate pseudo data using RND; Save data in Array 'A';
50 'Store in Bank 1 that part of the data that will not fit in Bank 0;
60 'While saving data, calculate SUM as the sum of the all the elements of
70 'the first dimension of A;
80 'After data has been saved, go back through data calculating SUMCHK in the
90 'same way as SUM was calculated;
100 'Check SUM and SUMCHK to see that they are identical.
110 'If SUM and SUMCHK are identical then Program correctly used EXMEM
120 'INTERNAL VARIABLES:
130 '   ER% = True if SUM and SUMCHK are not identical
140 '   I% = Pointer to first dimension of A ; should be between 1 and OX1%
150 '   OX1% = Original X1, When using FOR NEXT loops, OX1% can be endpoint
160 '   SUM(X2%) = Sum of all Xi of A(Xi,J%) ; Calculated before data outputted
170 '   SUMCHK(X2%) =Same as SUM except calculated after data read from Bank 1
180 '   X2% = Desired 2nd dimension of Array. M% to X2% will be put in Bank 1
190 CLS'           Clear screen
200 CLEAR'        Clears all variables and closes all files
210 OPTION BASE 1'  OPTION BASE 1 does work with EXMEM
220 ON ERROR GOTO 770'  Error Handler
230 S%=0'         S%=Subsequent Subroutine call Starts as false
240 PRINT @ 16,"Use EXMEM to Store Data in Bank 1"
250 INPUT"Enter the size of the first dimension of Array; X1 of A(X1,X2) ";OX1%
260 IF OX1%<1 THEN PRINT"X1 can not be less than 1":GOTO 250
270 'Assume OX1 is much greater than 64 so that if X1 is made a multiple of 64
280 'single precision numbers (256 bytes), only minimal memory will be wasted.
290 X1%=(OX1%\64+1)*64'   X1% is a multiple of 64 that is >= OX1%
300 INPUT"Enter the size of the second dimension; X2 of A(X1,X2) ";X2%
310 IF X2%<1 THEN PRINT"X2 can not be less than 1":GOTO 300
320 '           M% is the Maximum size of Array that will fit in Bank 0
330 M%=(MEM/4-12-2*X2%)\X1%-1'   -2*X2% is needed only for SUM and SUMCHK
340 IF M%>=X2%THEN PRINT"Extended Memory is not needed for this Array":GOTO 250
350 IF X1%*(X2%-M%)>32767/4 THEN PRINT"Need over 32K Extended Memory":GOTO 250
360 DIM A(X1%,M%),E%(6),SUM(X2%),SUMCHK(X2%)
370 PRINT"Please be Patient. It may take 5 minutes to check out EXMEM."
380 '***** Put as much of Array into A as possible *****
390 FOR J%=1 TO M%-1'         The last position M% will be moved to Bank 1
400 SUM(J%)=0'             Initialize each element of SUM as zero
410 FOR I%=1 TO OX1%'       Only fill Array to Original Dimension ; OX1%
420 A(I%,J%)=RND(100)'     Array will contain numbers between 1 and 100
430 SUM(J%)=SUM(J%)+A(I%,J%)'  SUM(J%) will be the sum of all Xi of A(Xi,J%)
440 NEXT I%,J%
450 '***** Put that part of A into Bank 1 that will not fit into Bank 0 *****
460 FOR J%=M% TO X2%'      M% to X2% will be moved to Bank 1
470 SUM(J%)=0'           SUM and SUMCHK are used to check program
480 FOR I%=1 TO OX1%
490 A(I%,M%)=RND(100)+1'   In practice Array could be filled from a
500 SUM(J%)=SUM(J%)+A(I%,M%)'   disk data base.
510 NEXT I%
520 GOSUB 4270'           Put one set of X1 numbers into Bank 1
530 NEXT J%
540 '***** Calculate SUMCHK *****
550 FOR J%=1 TO M%-1'     The first loop is for data already in Bank 0
560 SUMCHK(J%)=0
570 FOR I%=1 TO OX1%
580 SUMCHK(J%)=SUMCHK(J%)+A(I%,J%)
590 NEXT I%,J%
600 FOR J%=M% TO X2%'     The second loop is for data stored in Bank 1

```



```

610 GOSUB 4460'           Input one set of X1 numbers
620 SUMCHK(J%)=0
630 FOR I%=1 TO OX1%
640 SUMCHK(J%)=SUMCHK(J%)+A(I%,M%)'All X1% sets are stored in M% of A(I%,M%)
650 NEXT I%,J%
660 '***** Check to see that SUM(J%) equals SUMCHK(J%) *****
670 ER%=0'               If ER% is True then an Error occurred.
680 FOR J%=1 TO X2%
690 IF SUM(J%)<>SUMCHK(J%) THEN ER%=-1
700 IF SUM(J%)<>SUMCHK(J%) THEN PRINT J%,SUM(J%),SUMCHK(J%),SUM(J%)-SUMCHK(J%)
710 NEXT J%
720 IF ER% THEN PRINT"Error found in using Extended Memory"
730 IF ER% THEN PRINT"Errors smaller than ";OX1%;" may be due to rounding"
740 IF NOT ER% THEN PRINT"Extended Memory Checks out OK"
750 END
760 '***** ERROR HANDLING ROUTINE *****
770 IF ERR=6 THEN 800'   OVERFLOW Error
780 ON ERROR GOTO 0
790 '***** Handle Overflow Errors *****
800 PRINT"Overflow Error Occurred."
810 GOTO 200'           Start Over
4000 '
4010 '***** SUBROUTINE: Output one set of X1% single precision numbers *****
4020 'Under "LS-DOS Ready" Type EXMEM and press ENTER to Install.
4030 'Subroutines are written for a 128K Model 4 running under LS-DOS 6.3 BASIC
4040 'Instead of a separate buffer, pointer(FNJ%) tells EXMEM where data starts
4050 'PROGRAMMER: Edwin Ketola
4060 '           4178 South 72nd Street
4070 '           Greenfield, WI 53220
4080 '           Phone 414-545-7952
4090 'VARIABLES SUPPLIED BY CALLING PROGRAM:
4100 '   A(X1%,M%) = Array; Subroutines will Output or Input 1 set of X1% data
4110 '   X1% = Maximum size of First dimension of Array
4120 '   M% =Maximum size of second dimension of Array that will fit in Bank 0
4130 '   J% = Pointer to second dimension of A. Should be between M% and X2%.
4140 '   S% = Subsequent calls to Output ; Initialize subroutine only once
4150 'INTERNAL VARIABLES:
4160 '   C = Bank Number. Subroutines assume that Bank 1 is adequate
4170 '   E%(1) = SVC Number ; Returns with Register AF
4180 '   E%(2) = Register HL ; Relative offset in bank (0000H to 7FO0H)
4190 '   E%(3) = Register DE ; Address of 256 byte Buffer
4200 '   E%(4) = Register BC ; B=Function ; C=Bank Number
4210 '   E%(5) = Register IY ; Not Used
4220 '   E%(6) = Register IX ; Not Used
4230 '   EM% = The Extended Memory in bytes that is needed for X1% numbers
4240 '   OS% = The OffSet for each 256 byte section of X1% numbers
4250 '   T%=Try SVC Twice. If Z flag is not set, make a second SVC call
4260 '***** Initialize Subroutine *****
4270 IF S% THEN 4330'   S% is True for all Subsequent calls
4280 'FUNCTION: Calculate the Extended Memory Offset in bytes for this J%
4290 DEF FNJ%=(J%-M%)*X1%*4
4300 C=1'               Register C= Bank Number. Bank 1 may be enough
4310 EM%=X1%*4-256'     EM% = Extended Memory needed for X1% numbers
4320 '***** SUBSEQUENT ENTRY POINT *****
4330 FOR OS%=0 TO EM% STEP 256' Offset starts at 0 and goes to EM%
4340 T%=0'              T% is True if last SVC call was unsuccessful
4350 E%(1)=108'         SVC Number
4360 E%(2)=OS%+FNJ%'   Offset for Bank 1 is OS% plus Offset for J%
4370 IF NOT S% THEN OS%=8' First time Output is called OffSet must be 8
4380 E%(3)=VARPTR(A(1,M%))+OS%' Address of 256 bytes to put into Bank 1

```

```

4390 IF NOT S% THEN OS%=0:S%=-1'   Reset OS% for FOR NEXT loop to increment
4400 E%(4)=&H400+C'               B=Function 4 (Output Data) ; C=Bank 1
4410 X=USR11(VARPTR(E%(1)))'      Call SVC 108 (EXMEM)
4420 IF(E%(1)AND 64)=0 THEN PRINT"Z FLAG NOT SET ON OUTPUT":IF T% THEN END ELSE T%=-1:GOTO 4350
4430 NEXT OS%
4440 RETURN
4450 '***** SUBROUTINE: Input one set of X1% single precision numbers *****
4460 FOR OS%=0 TO EM% STEP 256
4470 T%=0'                         Try SVC Twice if no success the first time
4480 E%(1)=108'                    Must reset E%(0) to SVC Number
4490 E%(2)=OS%+FNJ%'              Offset for Bank 1 is OS% plus Offset for J%
4500 E%(3)=VARPTR(A(1,M%))+OS%'   Input data into A offset by OS%
4510 E%(4)=&H300+C'              B=Function 3 (Input) ; C=Bank 1
4520 X=USR11(VARPTR(E%(1)))'      Call SVC 108 (EXMEM)
4530 IF(E%(1)AND 64)=0 THEN PRINT"Z FLAG NOT SET ON INPUT":IF T% THEN END ELSE T%=-1:GOTO 4480
4540 NEXT OS%
4550 RETURN

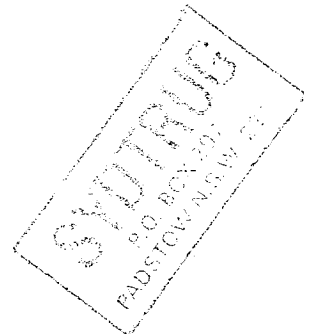
```

Input SUB for QuickBASIC

```

'           A L L   I N P U T   S U B R O U T I N E   V E R - 1.0.8
'
'           Written 1987 by William D. Schroeder of Logical Systems, Inc.
'           to be considered public domain software without restrictions.
'
'           This routine is written to be used as a "SUBROUTINE" or a "SUB-PROGRAM"
'           All Variables are global in nature, but all internal variables can be
'           defined as local depending on the usage of this routine.
'
'           The "Front end" that is present here is for demonstration and testing
'           and should be remove when the routine is actually used.
'
'           The routine is written for QUICK BASIC 3.0 or later, but should function
'           with little or no change on TURBO BASIC.
'
'           NOTE: You must set EVENT TRAPPING to ON to compile this routine.
'
'           This routine will take any type of input or edit, allow editing of any
'           string value that is sent to it. Insert/Delete/Right Arrow/Left Arrow/
'           Backspace/Enter/Escape are the active keys.
'
'           <ESCAPE>      : When hit at first input position will return with flag set
'                       : at other then first position will clear input and restart.
'
'           The use of all other keys should be obvious. The cursor will change to a
'           large block when in the insert mode. Regular cursor is typeover.
'
'           ENTRY VALUES : row,col= desired input location
'                       : chars= Number of characters to allow 1 to 79
'                       : type= Input type allowed
'                           : 0= Y/y N/n or YES or NO or yes or no
'                           : 1= all characters
'                           : 2= alpha only
'                           : 3= alpha upper case only, convert on fly
'                           : 4= numbers only plus +,-
'                           : 5= dollars, (fix 2 decimal if not entered)

```



```

'           : 6=
'           : 7=
'           : 8=
'           : 9=
'           : high.num= highest number to accept
'           : low.num= lowest number to accept
'           : letter$= string of letters to accept, i.e. "EQNFLS"
'           Note - This IS case sensitive
'           : car.ret= 1 - return if field filled
'           0 - return only on C/R
'           : field.char= ASCII value of field character
'           : edit$= if not null then will set up to edit edit$
'           : forg,backg= Foreground and Background for COLOR f,g
'           : Prompt$= If not null will be placed in front of field
'           : force.ret= ascii value of key to cause immediate return
'
' EIXT VALUES : inp.ret$= response string
'           : escape= 0 - did not return with ESC key
'           1 - did return with ESC key
'           : letter.pos= word or letter position from string(s)
'           : key$= value of last key entered
'
'           : inp.error= ERROR RETURNS CODES
'           0= NO ERROR
'           1= ROW selection out of range
'           2= COLUMN selection to far left
'           3= COLUMN selection to far right
'           4= CHARS selection out of range
'           5= TYPE selection invalid
'           6= EDIT$ longer then CHARS
'
test.main:
  cls:locate,,1,6,7
  print "INPUT SUBROUTINE version 1.0.8"
  input "ROW      :";row
  if row=0 then row=16
  input "COLUMN   :";col
  if col=0 then col=15
  input "CHARACTERS :";chars
  if chars=0 then chars=50
  input "TYPE      :";type$
  if type$="" then type=1 else type=val(type$)
  input "LETTERS   :";letter$
  input "FIELD CHAR :";field$
  if field$="" then field$="."
  field.char=asc(field$)
  line input "EDIT$   :";edit$
  line input "PROMPT$  :";prompt$
  if prompt$="" then prompt$="TEST INPUT : "
  input "CAR.RET 0/1:";car.ret
  input "HIGH.NUM  :";high.num
  input "LOW.NUM   :";low.num
  forg=1:backg=1
  input "FOREGROUND :";forg
  input "BACKGROUND :";backg
  if forg=0 then forg=10

gosub input.sub

  color 2,0

```

```

print:print:print "INP.RET$=";inp.ret$
print"CHARACTERS=";len(inp.ret$)
print"LETTER.POS=";letter.pos
print"ESCAPE=";if escape then print " YES" else print
print"YES/NO=";if inp.yes=1 then print " YES"
if inp.no=1 then print " NO"
if inp.yes=0 and inp.no=0 then print
if inp.error>0 then color 10,0:print"ERROR =";inp.error:color 2,0
print:input"PRESS <ENTER>";a$
if a$="" then goto test.main

end

```

```
input.sub:
```

```

' INPUT.SUB Version 1.0.8
'   written 1987 by William D. Schroeder of Logical Systems, Inc.
'   this is public domain software without restrictions.

```

```

'Turn on the arrow keys and special keys
Key (12) on:Key (13) on
Key 15,chr$(0)+chr$(&h47)
Key 16,chr$(0)+chr$(&h4f)
Key 17,chr$(0)+chr$(&h52)
Key 18,chr$(0)+chr$(&h53)
Key (15) on:Key (16) on:Key (17) on:Key (18) on
on key (12) gosub left
on key (13) gosub right
on key (15) gosub home
on key (16) gosub lineend
on key (17) gosub insert
on key (18) gosub dlete

```

```
error.checks:
```

```

if row<1 or row>25 then inp.error=1:return
if col-len(prompt$)<1 then inp.error=2:return
if col+chars>79 then inp.error=3:return
if chars<1 or chars>79 then inp.error=4:return
if type<0 or type>5 then inp.error=5:return
if len(edit$)>chars then inp.error=6:return
inp.error=0

```

```
inp.reentry:
```

```

key$=""
gosub restart

```

```
input.return:
```

```

if type <>0 then goto type.not.zero
yes.or.no$=left$(inp.ret$,1)
if yes.or.no$="Y" or yes.or.no$="y" then inp.yes=1
if yes.or.no$="N" or yes.or.no$="n" then inp.no=1

```

```
type.not.zero:
```

```

if type <>5 then goto type.not.dollars
if instr(inp.ret$,".") then goto check.pos

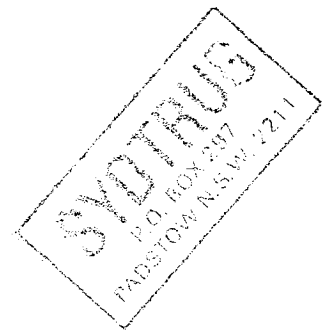
```

```
pad.left:
```

```

if len(inp.ret$)<2 then inp.ret$="0"+inp.ret$:goto pad.left
left.ret$=left$(inp.ret$,len(inp.ret$)-2)
right.ret$=right$(inp.ret$,2)

```



```

inp.ret$=left.ret$+"."+right.ret$

check.pos:
  if instr(inp.ret$,".")<len(inp.ret$)-1 then goto move.right
  inp.ret$=inp.ret$+"0":goto check.pos

move.right:
  locate row,col:print string$(chars,field.char);
  locate row,col+chars-len(inp.ret$):print inp.ret$;

type.not.dollars:
  if high.num=low.num then goto all.done
  inp.value=val(inp.ret$)
  if inp.value<=high.num and inp.value>=low.num then goto all.done
  goto inp.reentry

all.done:

  key (12) off
  key (13) off
  key (15) off
  key (16) off
  key (17) off
  key (18) off
  locate ,,1,6,7
  ins.flg=0

  return

restart:
  if key$=chr$(27) and field.pos=1 then escape=1:return
  locate row,col-len(prompt$):color forg,backg
  print prompt$;string$(chars,field.char);
  Locate row,col,1,6,7
  field.pos=1
  inp.ret$="":escape=0:inp.yes=0:inp.no=0
  if edit$>"" then inp.ret$=edit$ else goto no.edit
  field.pos=len(inp.ret$)+1
  if field.pos>chars then field.pos=chars
  print inp.ret$;:locate row,col+field.pos-1

no.edit:
  string.cleanup#=#fre("")

get.key:
  key$=inkey$
  if key$="" then goto get.key
  if key$<" " then goto control
  gosub check.key
  if key.bad then goto get.key
  if field.pos=len(inp.ret$)+1 then gosub add.char else gosub type.over
  if len(inp.ret$)=chars and car.ret then goto car.ret
  goto get.key

control:
  key.asc=asc(key$)
  if key.asc=8 then gosub back.spc
  if key.asc=13 then goto car.ret
  if key.asc=27 then goto restart
  goto get.key

```

```

back.spc:
  if field.pos=1 then return
  if field.pos=2 then left.ret$="" else left.ret$=left$(inp.ret$,field.pos-2)
  field.pos=field.pos-1
  if field.pos>len(left.ret$) then left.ret$=left.ret$+right$(inp.ret$,len(inp.ret$)-
field.pos)
  locate row,col+field.pos-1
  print right$(inp.ret$,len(inp.ret$)-field.pos);chr$(field.char)
  inp.ret$=left.ret$
  locate row,col+field.pos-1
  return

add.char:
  if len(inp.ret$)=chars then return
  inp.ret$=inp.ret$+key$
  field.pos=field.pos+1
  print key$;
  return

type.over:
  mid$(inp.ret$,field.pos,1)=key$
  field.pos=field.pos+1
  print key$;
  return

left:
  if field.pos=1 then return
  field.pos=field.pos-1
  locate row,col+field.pos-1
  return

right:
  if field.pos=chars then return
  if field.pos=len(inp.ret$)+1 then return
  field.pos=field.pos+1
  locate row,col+field.pos-1
  return

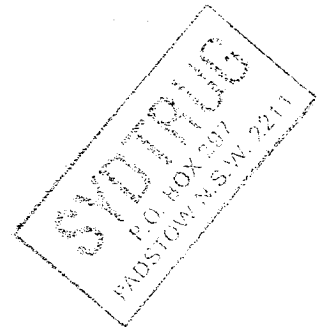
dlete:
  if field.pos=len(inp.ret$)+1 then return
  left.ret$="":right.ret$=""
  if field.pos=1 then right.ret$=right$(inp.ret$,len(inp.ret$)-1):goto dsp
  left.ret$=left$(inp.ret$,field.pos-1)
  if field.pos<len(inp.ret$) then right.ret$=right$(inp.ret$,len(inp.ret$)-field.pos)

dsp:
  inp.ret$=left.ret$+right.ret$
  print right.ret$;chr$(field.char);
  locate row,col+field.pos-1
  return

lineend:
  field.pos=len(inp.ret$)+1
  locate row,col+field.pos-1
  return

home:
  field.pos=1
  locate row,col
  return

```



```

insert:
    if ins.flg then goto insert.reset
    return insert2

insert2:
    ins.flg=1
    locate ,,1,0,12

ins.key:
    key$=inkey$
    if key$=chr$(13) then goto car.ret
    if key$="" or key$<" " then goto ins.key
    if len(inp.ret$)=chars then goto ins.key
    gosub check.key
    if key.bad then goto ins.key
    left.ret$=left$(inp.ret$,field.pos-1)+key$
    right.ret$=right$(inp.ret$,len(inp.ret$)-(field.pos-1))
    print key$;right.ret$;
    field.pos=field.pos+1
    locate row,col+field.pos-1
    inp.ret$=left.ret$+right.ret$
    if len(inp.ret$)=chars and car.ret then goto car.ret
    goto ins.key

insert.reset:
    locate ,,1,6,7
    ins.flg=0
    return get.key

car.ret:
    return

check.key:
    key.bad=0
    if asc(key$)=force.ret then return car.ret
    on type gosub all.ok,alpha.only,upper.case,numbers,dollars,date,date,date
    if type=0 then gosub yes.no
    if letter$="" then goto all.ok
    letter.pos=instr(letter$,key$)
    if letter.pos=0 then key.bad=1

all.ok:
    return

alpha.only:
    if key$>="a" and key$<="z" then return
    if key$>="A" and key$<="Z" then return
    key.bad=1:return

upper.case:
    gosub alpha.only:if key.bad then return
    key$=chr$(asc(key$)and &h5F)
    return

numbers:
    if key$>="0" and key$<="9" then return
    if instr("+-/," ,key$) then return
    if key$="." and instr(inp.ret$,".")=0 then return
    key.bad=1:return

```

```

dollars:
  if key$>="0" and key$<="9" then return
  if key$="." and instr(inp.ret$,".")=0 then return
  key.bad=1:return

date:
  gosub numbers
  if key.bad=0 then return
  if key$="/" then key.bad=0 else key.bad=1
  return

yes.no:
  if instr("YESNOyesno",key$) then return
  key.bad=1:return

' END OF INPUT.SUB

```

Do you want your ad here?

MISOSYS

SOFTWARE FOR YOUR P.C.

Order Now 800-MISOSYS (647-6797)

LB86™

- A useful and flexible data manager

Easily used by anyone for managing their data. It's menu driven for ease of use; absolutely no programming needed. Why settle for expensive overkill with data managers you have to program when economically priced LB86 will do mailing labels, customer lists, registration data.

- Store up to 65,534 records per data base
- Up to 1,024 characters per record
- Up to 64 fields per record
- Up to 254 characters per field
- Nine field types for flexibility
- Select and sort on up to 8 fields
- Keep 5 different indexes for data access
- 10 input/update screens per data base
- 10 printout formats per data base
- Extensive on-line help always available

LB86 [L-86-510] . . . \$74.95 + \$5 S&H

DED86™

- A utility disk that won't leave you puzzled.

A powerful sector-oriented disk/file editor and a page oriented memory editor. Search your entire disk while you tag sectors or clusters to be "kept"; direct DED86 to write the sectors as a disk file. Great for file recovery without fussing over FATs. DED86 gives you important features:

- Position by cylinder/head/sector
- Position by logical sector or cluster
- Jump about subdirectories
- Keep sectors/clusters/memory
- Edit bytes in hexadecimal or ASCII
- Search strings in ASCII or hexadecimal
- DOS subshell available for your use

DED86 [M-86-020] . . \$59.95 + \$5 S&H

MISOSYS, Inc.
 PO Box 239
 Sterling, VA 22170-0239
 703-450-4181 MC, VISA, CHOICE

VA residents add 4 1/2% sales tax.
 S&H: Canada add \$1;
 Foreign use S&H times 3

MISOSYS

Products'

Tidbits

Assembly: EDAS, MRAS

What do I need, PRO-CREATE or PRO-MRAS?

Fm Guy Mercier: I own a TRS-80 Model 4P 128K microcomputer with TRSDOS 6.02.01 (and LS-DOS 6.3.0) and I'm actually considering the possibility to learn how to program using assembly language. I often saw assembly language program listings in 80-MICRO requesting Editor/assembler such as PRO-CREATE 4.3A or PRO-MRAS, both of which are part of your product line. I'd like to know what are the main differences between these two packages. Which one should I order from MISOSYS? Which one would do the best job? Please forgive me my ignorance. I really am a novice in machine language programming.

Fm --jjkd--: Either Pro-Create or Pro-MRAS will work as excellent machine language development tools. The basic difference is that Pro-MRAS is a relocating macro assembler, whereas Pro-Create (aka EDAS) is a non-relocating macro assembler. What's the difference? The latter is better suited for larger and/or more complex tasks, though the former is certainly still adequate for major projects (Both LDOS and TRSDOS 6, all versions, were built with EDAS/Pro-Create).

So, Pro-Create is going to be a bit easier to learn, merely because there is less "overhead" involved in getting started. That said, I'd like to recommend Pro-MRAS. Why? Because, Pro-MRAS supports the use of the Pro-MC 'C' compiler, whereas Pro-Create does not. That is, you can add Pro-MC later and have a complete Z80 asm and 'C' development system.

Fm Guy Mercier: Thanks a lot for this info! It really helps me to make a choice. I'm really glad to know Pro-MRAS is a better choice to support 'C' compiler too. Actually, I'm also looking for a 'C' compiler package for my Model 4, but my Radio-Shack computer dealer told me the Radio-Shack 'C' compiler (product no. 26-2230) was discontinued and no longer available. Even the customer support service, at Fort Worth, told me exactly the same thing. So I will order this MISOSYS product, Pro-MC 'C' compiler, with my Pro-MRAS order.

About orders: Is it possible to order MISOSYS products directly, using Compuserve Email and your personal CIS Id. no.? Of course, it would be a lot faster this way, instead of regular mail, since I live in Canada! If I can, I will specify all necessary details, such as credit card number, shipping address, phone number, etc...

Fm MISOSYS, Inc: We can accept orders via our forum in a message saved as PRIVATE.

Fm H. Brothers: Thank whatever lucky stars shine upon you that there wasn't an RShack C package available. The C compiler they sold is bad enough that it chased a lot of folks away from the C language forever.

Fm Bob Schindler To --jjkd--: Excuse me for being an innocent nitwit, but what is it that makes Pro-Create (I still love that name! <grin>) get ill over the 'C' compiler's output. It's just asm, isn't it?

Fm --jjkd-- To Bob Schindler: Well, 'tisn't that the assembler "gets ill", just that the structure of Pro-MC output is specifically designed for a relocating macro assembler with linkable output modules (and, of course, a linker to put them together), which is what Pro-MRAS is. It certainly is possible to turn 'C' into executable code without a relocating macro assembler, but the job is a somewhat different one, and much harder to do "efficiently". The predecessor to MC/Pro-MC, specifically LC, used EDAS instead of MRAS.

Fm Bob Schindler To --jjkd--: Is it possible to judiciously construct the 'C' source file so that the output is compatible with other assemblers? I ask this only as a point of curiosity, not as a practical problem.

Fm MISOSYS, Inc: It is impossible to revise MC so that it would support other assemblers. You actually would not have too much of a problem with MC per se, since it's code output is straightforward. It also references a bunch of macros which could be adapted. The kicker is the extensive libraries. You would have to UNREL the entire set of libraries and revise them to some other LINK format. That's where the trouble is and that's why we refuse to support something other than Microsoft's REL format which is available in both M80 and MRAS.

*RADIX doesn't work as expected

Fm Rex A. Basham: If I'm reading the documentation correctly, the directive '*RADIX 16' at the beginning of a program should force all numeric values to base 16 or hexadecimal. Why do I get an error during assembly for numeric values like OAF, 2C, and 3F, while values such as OAO, 3), OFFOO, 55, etc., all assemble correctly? After I append an 'H' to the constants ending with the characters A, B, C, D, E, and F they also assemble without errors. Correct me if I'm wrong, but I thought the whole purpose of defining a radix of 16 was to force all numeric constants to hex without having to append the 'H' character. Did I miss something in the documentation or is this a bug in the assembler?

Fm MISOSYS, Inc: Concerning the RADIX assembler directive of MRAS, you didn't miss anything. RADIX is a funny animal; it really is not an effective facility. The reason that the use of a RADIX 16 specification still requires the postfix number type override of 'H' when a hexadecimal number ends in 'A' through 'F' is that MRAS does not inhibit the test for a postfix override if a RADIX directive has been detected. Thus, the ASCII to binary conversion routine still checks for a non-digit alpha terminator. If it finds one, it will flag an error if it is not one of the accepted specifiers. Since 'B' implies binary and 'D' implies decimal, a hexadecimal number ending in those 'letters' will be acceptable but the conversion value will be wrong.

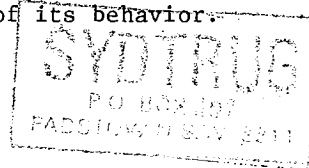
One solution would be to inhibit MRAS from checking for a postfix override specifier if a RADIX directive was in use. Since MRAS has been in the field for a few years, the possibility exists that some folks may be taking advantage of the way in which MRAS currently operates. Thus, I feel that at this point in time, I should not change the manner in which MRAS deals with a RADIX directive. Rather, let me just make folks aware of its behavior.

MLINK and IRL libraries

Fm Jeff Page: I have some questions regarding use of libraries with MRAS and MLINK. Is the search of /IRL faster than /REL? Is there a form of *REQUEST that will cause MLINK to search an /IRL library? The -S=FILENAME command will search /IRL if no /REL is found but, *REQUEST FILENAME returns "FILENAME/REL >File not found" although FILENAME/IRL exists. Lastly, is there a switch (or a patch) that instructs MLIB to default to one or the other so that the prompt "REL or IRL format (R/I) ?" does not need to be answered so often?

Fm MISOSYS, Inc: Only empirical analysis would prove whether IRL library searches are faster than REL library searches because it depends on many factors. Here's some information which may give you some insight.

L80 will search a library once, sequentially. This means that a *REQ or command line -S will cause the linker to read through the entire library. If the library is disorganized, you may have to force a repeated search of that library. Disorganization means that a module positioned at "n+k" has an external reference to a symbol declared public in module "n" and module "n" has not been referenced by other modules until module "n+k" is linked. You can see that after one pass through the library, "n+k" remains unlinked. A second search must be invoked. Extend this scenario to some other symbol declared extern in "n" which was previously unresolved would require still more sequential searches. The library is read until its end. Certainly a totally disorganized library can be most inefficient. MLINK, on the other hand, performs automatic researching of the library until it makes a pass whereby no new modules are linked in. That implies a minimum of two sequential searches through the library. Of course, MLINK also does not search a library if no symbol is currently undefined.



An IRL library maintains, at its head, an index of all public symbols and pointers to the module where each is declared public. Thus, MLINK need only read the index, locate the module by direct positioning which defines the extern'd symbol, and then read the module. This procedure should dramatically speed up the linking of large disorganized libraries depending on the average seek speed of the drive which contains the library. The faster the drive, the better the improvement of IRL over REL. Of course, if the REL library is totally organized, accessing it should probably be faster. The libraries provided with MC are totally organized. In fact, Rich put together a small program to test whether a library is or is not optimally organized. Maybe I'll get that into TMQ one of these days.

Lastly, I have investigated your report of *REQ being unable to "find" an IRL library. That is true; it used to work but we broke it when we applied MLK66/FIX. Here's how you can un-break it:

PATCH MLINK (DOA,18=00 27:FOA,18=20 29)

The *REQ originally used IOBUF for its parsing of the request filename. But that caused a problem when a REQ was used in the root of an overlay generation and the command line containing the -O switch also contained additional linker commands (IOBUF contained the command line). We had patched the *REQ handler to use FCBREL for the parsing forgetting that the REL/IRL open routine would need to recover an untouched filespec to add the /IRL default extension if a REL library was not found. This patch changes the REQ handler to use the REL-BUF region which would be non-conflicting. MLK66 was installed back in November of 1985; shows you how many folks use IRLs. Incidentally, there is no patch that we have developed to have MLINK default to one or the other saving you from responding to the Rel or Irl? query.

C Language: MC

Beware of function library organization

Fm RICHARD WATKINS: I am using Pro-MC and MRAS packages with my model 4. I wrote a function called getscrn and put it in a library. Tonight I wrote another function, called up_scroll, which calls getscrn. When I write

a separate program to use the up_scroll function the program compiles and assembles but returns an undefined symbol error when MLINK tries to link it. The symbol is GETSCRN. Why is this? Do I have to declare getscrn as An external from up_scroll? Neither return any values but were not declared as void.

Fm Shane Dawalt To RICHARD WATKINS: You have found the problem with libraries and calling functions within them. When a library is searched, it is searched from beginning to end and all symbols required by the program being linked are extracted (along with the code associated with them). Now, this assumes that any symbols being used by functions within the library are declared after the function that uses them. You can look at it like this, if you don't know what to include before hand, how can you include? I assume your library has GETSCRN followed by UP_SCROLL. UP_SCROLL requires GETSCRN, but the linker has already searched past GETSCRN. Since nothing else required GETSCRN, it is not linked into the program. The linker doesn't remember previous symbols, only those which are required does it remember. So, the linker scans the entire library for GETSCRN which it will never find since it occurs before UP_SCROLL. Fortunately, there's an easy way to overcome this problem. Simply rearrange your library. Place UP_SCROLL before GETSCRN. The linker will see GETSCRN in UP_SCROLL, then it will find GETSCRN in the library and link it into your program! This is a partial reason for the two libraries in PRO-MC, LIBA and LIBC.

Random file access in C

Fm RICHARD WATKINS: I am trying to do random file access using 'C' (Pro-MC) and having problems. Using fseek I can position the file pointer to any place in the file. Is there anyway to quickly determine where the end of the file is or the length of the file in bytes so that I can tell how many records are in the file? Evidently if I position the file pointer past the end of the file and then do a fread, the computer fills the space with garbage. I don't understand how I can check the file position with the end of file?

Fm jeff brenton To RICHARD WATKINS: Use something like this:

```

long int where_eof(fp)
FILE *fp;
{ long int temp, endof;
/* save current position */
  temp = ftell(fp);
/* get to the end */
  fseek(fp,0L,2);
/* what is the end of file? */
  endof = ftell(fp);
/* return to original place */
  fseek(fp,temp,0);
  return(endof);
}

```

the above is relatively non-destructive.

Fm RICHARD WATKINS To jeff brenton: I tried the program that you have written to find the end of file. I either don't know what I am getting at the end or I have done something else wrong. I opened the file using fopen and "r+". For some reason where_eof(fp) always returns 19039 even for an empty file. Got any suggestions?

Fm jeff brenton To RICHARD WATKINS: Did you declare where_eof() as returning a long int before using it? If not, the significant portion of its return value will be lost, and you will get the same number back for any file under 64K long. A proper [declaration] would be:

```
long where_eof();
```

In addition to my earlier reply, there is the fstat() function in MC, which fills a structure with lots of useful information, including the size of the file in bytes. see page 4-87 of the manual.

Fm RICHARD WATKINS: Help! I am using PRO-MC and have been trying to figure out file I/O for about 2 weeks now. No luck. I don't understand the difference between text and binary. If I intend on reading and writing complex structures with fread and fwrite that would be binary wouldn't it? If so, do I open the file with fopen and use fseek to position the file pointer? If that is true how do I find the end of the file. I tried the program the Jeff left several messages back and I can't get it to work. I get so weird answers. Can anyone enlighten me?

Fm MISOSYS, Inc: There is no such thing as "text" and "binary" files under TRSDOS-type operating systems. Files are files. We don't have the "enjoyment" which the MS-DOS and CP/M worlds experience when they have to deal with files; they have to specify "text" or

"binary". You can use either the stream routines [fopen(), getc(), etc...] or the block I/O functions [open(), read(), ...]; it's your choice. Sounds to me like you don't have any books on C. There are dozens of good books on C which would help you along. Most folks these days use the stream I/O functions.

Fm RICHARD WATKINS To MISOSYS, Inc: I have a book but is geared towards an MS-DOS environment. I've been using the streams but I can't seem to figure out how to find the end of the file without reading through the whole file. Jeff's routine used fseek to position the pointer to the end and then used ftell to get the value. When I try this I always get the same answer no matter what the file length.

Fm MISOSYS, Inc: Are you sure you properly declared whatever values are needed as "long"? That's where you can make a big error. You must have a declaration of

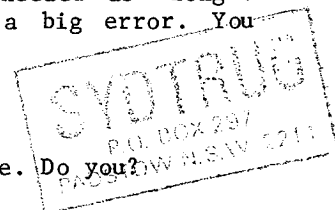
```
extern long ftell();
```

in your code prior to its use. Do you?

Fm Shane Dawalt To RICHARD WATKINS: What answer do you get? MS-DOS is different from TRSDOS (boy is it different). Like what you found, MS-DOS needs 'text' and 'binary' definitions while TRSDOS doesn't seem to care. That's only because MS-DOS uses two characters for the end of line ('\n' newline in C) where TRSDOS uses only one character. If you are reading text, the C compiler must know to include the code to fetch the two characters for the newline, lest one would be left in the input stream and everything would be screwed up! In binary, you don't want the C compiler eating two codes up every time it sees a "suspected" newline, therefore you must tell it 'binary'. It is weird. I couldn't understand it until I opened a file up incorrectly on the Z-248. Boy did I get strange results!

Fm MISOSYS, Inc: MS-DOS "text" files also return EOF when a CTL-Z is read (I believe).

Fm David Huelsmann To RICHARD WATKINS: Don't get confused by references to text or binary in MS-DOS "C" reference manuals. For your purposes in TRS/LS-DOS, they are handled exactly the same for file I/O. Your problem in getting the same value returned from your ftell() may be related to a failure to realize that ftell() returns a long value. The following is a very simplified example that should work.



```

main()
{ FILE *fp; long bytct;
  fp = fopen(filename,r);
  fseek(fp,OL,2);
  bytct = ftell(fp);
  printf("The number of bytes in this
file is %ld\n",bytct);
}

```

I would normally include some error checking for file existence, etc., but this should make a clear example and should return proper values for you.

Fm RICHARD WATKINS To David Huelsmann: Do you have to open the file in "r" to make ftell work? I am using "r+". Does the r need to be enclosed in quotes. I have set up the function to return a long integer but I still can't seem to get it to work.

Fm --jjkd-- To RICHARD WATKINS: The difference between text and binary modes is only important in the UNIX or MS-DOS environments, where a certain character or group of characters is translated to a logical "newline".

I'd make that

```

#include <stdio.h>
main()
{ FILE *fp; long bytct;

```

That missing STDIO.H will kill you, as you will not have the proper function return types. [Also] "r+" should be ok, and it must be in quotes.

Fm Les Mikesell To --jjkd--: Unix does not have the problem of binary vs text modes on files since the newline character is a single linefeed.

Fm RICHARD WATKINS To MISOSYS, Inc: I have to declare ftell() as extern long? I guess I know what the problem is now.

Fm MISOSYS, Inc: When you're coding a program, any function external to your program and any forward-referenced function which returns something other than an int MUST be declared so the compiler knows its TYPE. Thus, if you are going to access a function, func(), which returns a value of type long, you need to declare it via the statement,

```
extern long func();
```

If you have a function in your source file which returns something other than an int,

then you have to declare it too, but not as extern. In that case, it would be declared something like,

```
double functionreturningdouble();
```

A compiler has no way of knowing the type of the function's return value unless you tell it. It always assumes int unless told otherwise. That should fix problems you're having.

Fm David Huelsmann To RICHARD WATKINS: As Joe pointed out, you do need to #include <stdio/h> (which I assume you must have done or there should have been some complaint over the FILE declaration). I use AZTEC C and not MC so my ftell() is declared in the stdio/h file which I would also expect to be the case in the MC header file as well. Since the problem appears to be entirely related to the long value that needs to be returned and assuming that you included the %ld in the printf() function (the l is important), I have only one guess left. I found that a library function I was using in AZTEC wasn't returning a long once and it related to AZTEC failing to re-compile the function once they corrected the return value in the library they sent me. Fortunately, I had the source code and was able to correct the problem. Unless some other C wizard can see something missing, I would refer the problem to Roy Soltoff along with a sample of the code that doesn't work and your version (including all patches) of MC that you use.

Fm MISOSYS, Inc: We don't put those declarations in the stdio.h header. If we did, M80 users would find every library function so declared winding up in their program. The type of every function is documented in the library section of the MC manual. A piece at the front end of chapter 4 alludes to the necessity of declaring the type of functions. On the other hand, a quick glance at the manual does not reveal a statement of the form, "all functions which return other than an int must be so declared"... I suppose that it is such a common need that really spelling out that requirement was overlooked.

Fm David Huelsmann To MISOSYS, Inc: Ah, that may explain his problem then. I am so used to have those streams declared in stdio and it is so automatic to include it that I certainly would have missed that as the problem.

Fm MISOSYS, Inc: We had a lot of restrictions when our premise was that we had to cooperate with M-80. Like that meant we couldn't have a third segment type for uninitialized data

which would be zeroed at runtime. That's why the "-z=y" switch was put into my linker. The topic was beat to death in TMQ II.1.

Fm Les Mikesell To MISOSYS, Inc: Ummm... Why should "long ftell();" generate any assembler code that would force m80 to link the function? Such declarations should only tell the compiler what kind of return value to expect if the function is ever called.

Fm MISOSYS, Inc To Les Mikesell: MC will generate an "EXT identifier" statement anytime it sees either a "long func();" or an "extern long func();". Due to space restrictions, it was decided to not keep a flag which would indicate that a particular function was or was not referenced to suppress the EXT statement when it was not. Also, if it was suppressed when a function was not referenced, there would be no way to "automatically" force MLINK to load in the unreferenced module when a root segment was being linked (when a root and overlays were being constructed). Thus, MC always generates the EXT. Now anytime that M80 sees an EXT statement, it generates a null length chain external even if the symbol is never referenced. It's that EXT statement which causes M80 to generate the chain and subsequently L80 links it in. MRAS would not generate a null length chain external unless the -FE switch was on. The decision was that MC was big enough without having to add more overhead to deal with function declarations when the function was not referenced.

Fm Les Mikesell To MISOSYS, Inc: Function declarations typically do not include the extern modifier, since the compiler has to be able to figure that out anyway, and should not generate references to functions which are declared but not actually used.

Fm MISOSYS, Inc: We won't be changing the behavior of MC. The end result is that M80 will always generate the chain external record on an EXTRN even if the symbol is not used anywhere whereas MRAS does not unless specifically requested to do so by a switch.

Fm RICHARD WATKINS To --jjkd--: I think I have got the file I/O licked. Roy Soltoff enlightened me about declaring all functions that don't return integers as extern. I am building a library of commonly used functions. I may actually have a completely working program by the end of the weekend. I am now advancing to sorting. I have a good reference that shows lots of examples on the sorts and

searches and I seem to be able to follow the source code.

Fm MISOSYS, Inc: Actually, what I said was that all functions you have in the file being compiled which do not return an int should be declared explicitly (i.e. long myprog();) Now all functions external to the module being compiled which return something other than an int should be explicitly declared via an extern type func(); statement. That tells the compiler what to do for processing of the returned value and that the "extern'd" function is not in the current file. Also, don't forget that MC includes a QuickSort function.

BAD BLOCK in PRO-MC compiled programs

Fm Shane Dawalt To Roy Soltoff: I just encountered a BAD BLOCK error from PRO-MC, MC/CMD v1.5b. This BAD BLOCK is a problem of my own nasty programming, but I noticed that the error isn't written to stderr. I redirected stderr and stdout to two different disk files. The BAD BLOCK message was written directly to the screen, thus it must not have been written to stderr or stdout. I attempted to list both of the files, however, this couldn't be done because neither file was properly closed when the error was generated. Should PRO-MC do this?

Fm MISOSYS, Inc: Is this during an MC compile? Or is this from a program you compiled? On the other hand, MC compiled programs should do exactly that when you issue a free() with a bad pointer. The reason it is forced to the screen is due to the assumption that a free() with a bad block pointer means that memory is probably corrupted. Since a puts() with that error message may be dealing with a file pointer which uses allocated memory space, it would not make sense to use the standard stream functions to output such a message. Now how does MC know that you have passed a bad block pointer? Why it's smart! I think that the memory integrity allocation code we put into MC's dynamic memory allocation functions was just one more thing that really was never highlighted but which sets MC apart. How many times would you have a debugging nightmare when a program frees a "block" using a bad pointer when the effect doesn't get noticed until much later after your program crashes.

Fm Shane Dawalt To MISOSYS, Inc: This is during running of the compiled program. Ok, I never thought of the fact that the memory may

be corrupt. (Actually, in one particular instance, my program doesn't allocate any memory, but when it goes to free up the linked list ... BOOM goes the program.) Thanks for the reply. I suspect the corrupt memory reasoning is why the redirected files are not closed.

Fm MISOSYS, Inc: No, the reason why the files are not closed is that since the underlying assumption of the dynamic memory handler is that memory is corrupt, it will refuse to utilize the stream I/O functions since they would be suspect as well. Thus, the "BAD BLOCK" error message is sent direct to @DSP and an @ABORT is done. That can't close any open files since the function of cleanup is done by the exit() handler. Make sense?

Fm Shane Dawalt To Roy Soltoff: Yeah, that makes sense. (Actually, I was thinking more in terms of the file control buffers being corrupted other than the code. One is about as bad as the other though.)

Memory space and MC

Fm Lee C. Rice: I subscribe to several programming journals, the best of which is certainly Dr. Dobbs. Many of the C programs and utilities offered in its issues compile without problem using PRO-MC. [That, parenthetically, is why I don't purchase much applications software: with FORTRAN, C, RATFOR, etc., I can usually produce my own.] I presently have many UNIX utilities running under TRSDOS6.3.

One problem with PRO-MC and the Model 4 is memory space. With many a program from Dobbs, one gets the all-too-frequent "out-of-memory" message. I'm used to this, having spent several years recompiling International Library of Scientific Subroutines Fortran subroutines on both the Model 3 and Model 4 (redimensioning arrays is a way out in Fortran).

Your PRO-MC compiler provides for overlays, which are probably more efficient than my kludge approach to Fortran. The problem is: I have never been able to get overlays to work with PRO-MC. That's not your problem, and it's certainly not a bug in PRO-MC. I'm not an experienced C programmer, and I have no previous experience with overlays.

So, in your copious free time (a joke!), you might want to include some editorial or examples or commentary in TMQ about use of overlays in MC. Or perhaps you could suggest that some of the TMQ readers who are experienced C wizards could send some examples.

AGAIN: this is NOT a complaint about PRO-MC, or your documentation. It is a cry for help from a novice in C. I suspect that I am not alone in my ignorance...

Fm MISOSYS, Inc: The "out of memory" error from PRO-MC is because a big compiler handling all of K&R just happens to be big. Certainly one could have written the compiler in assembler, but the development time would have been prohibitive. I also imposed the requirement that it be implemented in C to ensure portability - if there ever was to be an MC on another machine (not too prudent a task at this time). Don't forget, you also have the "MP" JCL option in MC/JCL to force the multi-pass version of the compiler. Compilation is slower, but you gain about 6.5K or so more of usable space. Certainly that increased space should handle much larger source files. You can also separate a large source file into multiple files and use the techniques of separate compilation.

Ah, overlays. I'm not sure anyone has attempted the use of them. It would be good to query my TMQ readers on that issue. We have tested out the concepts using the examples discussed in the appendix. They are relatively easy to follow. More complex usage would just be based on an extension of the basic concepts. If you have some definitive applications in mind, perhaps we can discuss some plan of attack.

On converting doubles to ints

Fm Richard J. Edgar: I've recently started using your PRO-MC compiler to port a few programs that I wrote on a UNIX system, which I've also ported to a VAX-C (under VMS) and Microsoft C 4.0 (MSC) on a PC clone. I'm very impressed by the quality of this product, and by the great lengths you've gone to to make LS-DOS look like UNIX (even to accepting dots instead of slashes in filenames). I'd like to pass along a list of bugs, minor annoyances, and a wish.

(1) A minor annoyance. The symbols EDOM and ERANGE are #defined in both math.h and

errno.h, so that any program that #include's both of these header files causes MCP to issue warnings (so I have to sit here at the computer while it compiles). Perhaps something like this could be done:

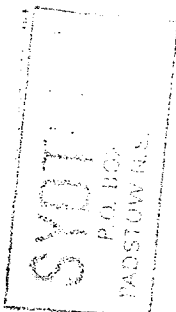
```
#ifndef EDOM
#undef #EDOM
#endif
#define EDOM etc..
```

and likewise for ERANGE.

(2) A bona fide bug (or two). I'm enclosing a little program that misbehaves; it seems that if y is a double and i is an int, that doing either i=y or i=(int)y [which should be equivalent, and are] sometimes produces incorrect results. First, i never comes out -1. If y has an integer value (though it's a double; for instance the result of a floor() call), and negative, i comes out (y-1). This could be a result of the limited precision of the machine; perhaps negative integers are really a tad less than the advertised value (like "-1" being -1.000000000001 or something), so they round down to the next lower int (-2 in this case). Here are the programs and the results from PRO-MC and MSC 4.0:

```
#include <stdio.h>
#include <math.h>
main()
{ double y; int i;
  for (y = 1; y > -2.5; y -= 0.5)
  { i = (int) y;
    printf("y = %f, i = %d\n",y,i);
  }
}
```

y value	PRO-MC i	MSC i
1.000000	1	1
0.500000	0	0
0.000000	0	0
-0.500000	0	0
-1.000000	-2	-1
-1.500000	-2	-1
-2.000000	-3	-2



(2) Also, if I feed a number into a double via y=atof(argv[1]) with the argument = -99999999 (that's 8 or more digits, all nines), y comes out -0.5. Hmmm...

(3) Another small annoyance. All the other C compilers that I've used define a constant or something called HUGE, the largest double allowed. PRO-MC defines HUGH_VAL instead, which is no problem, but not very portable. MSC de-

fines both, but some UNIX compilers (DEC Ultrix and 4.2BSD at least), define only HUGE. So I stuck '#define HUGE HUGE_VAL' into my copy of math/h. Others might like to do this, too.

(4) Just a wish. It's sometimes useful when writing code to be ported to be able to tell which of several compilers and/or machines the code is being compiled on, so that machine-specific features can be put in. For example, MSC defines MSDOS, VAX-C defines VMS, and so on. So I inserted the line '#define LSDOS6' into stdio/h, and then when I need an LS-DOS feature (or hardware limitation), it can be put into the portable code inside an '#ifdef LSDOS6/#endif' block.

Again, let me say that I'm impressed by this product, which makes porting code to the Z80 world a reality (at last).

Fm MISOSYS, Inc To Richard J. Edgar: Concerning some shortcomings of our PRO-MC product, your letter came at the right time since we are preparing a minor update to take care of extending the date support in the Unix-compatible time functions as well as some tweaking in other functions.

You are right on about EDOM and ERANGE. We apparently never ran across that conflict before if both math.h and errno.h are #included. We can make that change in the upcoming release.

Your suspected bug illustrated by the test.c program was indeed a bug. I was able to trace that to the \$d2i routine in liba/rel. This routine is used when converting a double to an integer. The bug occurred when the argument was negative and an exact power of two [-1.0, -2.0, -4.0, ... fall into this category]. The error was also in the \$f2i routine as well.

Now to further add fuel to the fire, take a look at K&R page 184 (copy enclosed). They state that "the direction of truncation of negative numbers varies from machine to machine". As an exercise, I ran your test program on my MSC 4.0; the results were the same as yours. On the other hand, I adapted the program to BASIC and ran it on my Model 4 (Microsoft's BASIC), on my IBM PC using BASICA (Microsoft's) and QuickBasic (Microsoft's). Here's that program (using single precision since Microsoft's BASIC doesn't permit doubles as the index of a FOR-NEXT loop):

```

10 FOR Y!=1.0 TO -2.5 STEP -0.5
20 I%=Y!
30 PRINT Y!,I%
40 NEXT

```

The results were not uniform. In particular, Model 4 BASIC converted 0.5 to 1 and -0.5 to -1 (Model III BASIC is different still). After further investigation, I decided that Microsoft is wrong in Model 4 BASIC as well as MSC, and the MS-DOS BASIC's I tested. Specifically, I propose that the conversion of floats to ints should be the same as what floor() should return; thus, converting -0.5 to an int should produce -1. That means that Microsoft's changes in MBASIC 5 (what's on the Model 4) introduced a bug [cint(0.5) is not equal to 1!]. Similarly, MSC, BASICA, and QuickBASIC 3.0 also have a bug [cint(-0.5) is not equal to 0 but -1 according to their treatment of -1.5, -2.5, etc.] I altered my \$d2i to be the same as floor(arg).

In the other example of passing "-.99999999" to atof(), that was also a bug traced to the \$d2f [conversion of a double to a float]. The bug was introduced anytime a number stored as "?? ?? 8? FF FF FF ??" was converted. The rounding upward would cause the exponent to be incremented but the exponent change wasn't picked up in the new number. There was also another error which we discovered in this routine: under those exact set of conditions, a positive number would get converted to -0.5! This has been corrected. On the other hand, you really should have been using the atod() function; not the atof() function.

Now you are right about HUGE and HUGE_VAL. The UNIX System V manual noted that HUGE_VAL was the future direction. We took that to mean HUGE would be dropped. Your "fix" was suitable. I did note that Microsoft '#defined HUGE_VAL HUGE' in my copy of 4.0.

Now the wish for automatic system defines is obtainable quite easily. You may have overlooked the documentation of the ones MCP defines on page 2-39. In particular, the MC pre-processor pre-defines "DOS" and either "DOS5" or "DOS6" depending on the version - MC or PRO-MC.

Thanks for your reports. As good as we think MC is, there's always room for slight improvements; major improvements are no longer justified.

Printer status from C

Fm Dirk Vandebossche: On page 54 of THE MISOSYS QUARTERLY summer 1987, you publish a letter by Jerry Wagers, concerning the opening of the *PR device. In the listing 1 I enclosed, you see a routine I have been using successfully for some time. This function makes use of the @CTL SVC and is consequently not very portable code. For those who are interested, in listing 2, another function to open files is given.

```

/* listing 1 */
#include <z80regs.h>
#define CTL 5
#define GTDCB 82
printon()
{
    union REGS reg;
    /* get pointer to DCB printer */
    reg.E='P'; reg.D='R'; call(GTDCB,&reg);
    reg.DE=reg.HL; reg.C=0;
    /* get status of device */
    if (call(CTL,&reg)){
        fprintf(stderr,"Printer not
ready\n");
        return (0);
    }
    return (1);
}

/* listing 2 */
#include <stdio.h>
FILE *openfile(file_name,mode)
char *file_name, *mode;
{
    FILE *fopen(), *fp;
    /* open stream */
    fp = fopen(file_name,mode);
    /* printer ? yes -> */
    if (strcmp(file_name,"*PR")==NULL)
    /* printer not ready? */
        if (!printon())
            fp = NULL;
    /* did open fail? */
    if (!fp){
        fprintf(stderr,"Can't open file
\"%s\" for %s\n",file_name,
(*mode=='r') ? "reading" :
((*mode=='w') ? "writing" :
"appending"));
        exit(-1);
    }
    return(fp);
}

```

Fm MISOSYS, Inc: Thanks for the input. Perhaps some folks will find these functions useful. One caution I will note is that the printon() function uses the @GTDCB service call. That SVC is handled by a DOS overlay (SYS2/SYS).

This presents absolutely no problem or operational slow down when `printon()` is used immediately following the `fopen()` which uses `SYS2/SYS` as well. On the other hand, if one calls the `printon()` function prior to the output of every character to that stream, it could introduce substantial overlay thrashing depending on what else was being requested from the DOS. A way around that would be to separate the `"call(GTDCB,®s)"` from `printon()` to obtain the printer device pointer once, then keep that stored in a static variable for further use. Still another would be to utilize the address in the file control area of the stream which maintains an image of the device control block. Unfortunately, that is one area we decided not to document in order to ensure that we would not be locked in to the scheme employed in MC when it was released. Since there is no expectation to change the scheme at this point, it may prove useful to discuss that in a future TMQ.

Is an MC86 in the works?

Fm J.R. Jacques Baril: How about either a C compiler and or a library for MS-DOS that would be 100% compatible to MC. I would then buy PRO-MC (and its MS-DOS brethren) and convert all my source code to one compiler's/library syntax (I am not sure that is the right word but you probably get my drift). I presently use Aztec as I was hoping for that compatibility but found so many routines lacking in their Model 4 library that the LS-DOS/MS-DOS portability is a bit of a chore.

Fm MISOSYS, Inc: From the previous dialog and others which have appeared in TMQ, you should be aware that we have devoted maximum attention to making the MC libraries as close to the standard UNIX System V library as we could, considering the DOS environment. Since PRO-MC's libraries are as compatible to UNIX System V as can be, you will experience little difficulty in porting programs compiled under MC to other systems. Our users report minor changes in porting to/from MS-DOS (using a good MS-DOS C compiler - like Microsoft's, MSC) or Xenix. The changes invariably originate in the handling of file specifications, i.e. things inherently different in the various OSes. Most folks plan ahead and put system-specific code into conditionals (i.e. use `#defined` identifiers surrounding system-specific blocks). It makes it easy when MC pre-defines "DOS" and either "DOS5" or "DOS6". MSC pre-defines "MSDOS". Thus, you will have

the least problems in porting code to/from MS-DOS (and UNIX) compared to the other two full C compilers available for the Model 4.

Now the reality of the situation is that the MS-DOS marketplace is flooded with C compilers; there is absolutely no sense in trying to implement MC under MS-DOS and competing in that arena - even though I like the sound of "MC86". In fact, I predict that in a few short years, the MS-DOS C compiler competition will narrow the field to perhaps 2-3 vendors (Microsoft, Borland, and ???). I don't think any other firm has the financial resources to continue butting heads with those two. Perhaps what we may need to do, though, is make available the PRO-MC installation library functions for MS-DOS. Comments, anyone?

diskDISK

diskDISK and Aerocomp 20Meg HD continued

Fm Ron Ungashick: I received updated drivers from Montezuma Micro to correct the incompatibility problem with diskDISK. The following is the letter I received from MM.

"Dear TRSDOS/LDOS User: It has been brought to our attention that there is an incompatibility with our driver and the DiskDISK utility sold by Misosys. In attempting to get around the TRSDOS/LDOS limitation of 404 cylinders per drive we have apparently generated an inconsistency in the drive table that causes this program to malfunction. Since the driver has been in use for more than a year before this problem was first reported it seems likely that this is not a problem for any other software.

The enclosed disk contains an updated driver for TRSDOS, LDOS, or both depending on your requirements. To overcome the deficiencies of the operating systems these drivers have been modified to make the 20 meg drive appear to have twice as many heads and half as many cylinders. Thanks to Roy Soltoff for offering this kludge.

Please accept our sincere apologies for this problem, and our thanks for your patience and cooperation while we corrected it. Best Regards, Jesse Bob Overholt, Software Wizard"

I have not tried the new drivers yet as it appears I will have to backup and reformat the HD. The letter is dated August 25, 1987. I sent my diskettes in several weeks ago when I reported the problem to them.

Fm STEVEN JERKINS To MISOSYS, Inc: The letter that Aerocomp sent with the new drivers gave you credit for the "kludge" (their term) to make the drivers work with DiskDISK. The result does not work at all.

Fm Steven Jerkins To Ron Ungashick: I received the drivers and they will not work on my machine. The drivers will not sysgen as the old drivers would and when a diskdisk is installed the system locks up when you go to access the diskdisk. If they work for you please let me know as I may have received a bad copy of the drivers (if they work for you). Anyway, let me know the results of your try with Aerocomp hard disk and DiskDISK.

Fm Ron Ungashick To STEVEN JERKINS: The LDOS driver works correctly for me. I am able to sysgen it and create and access all the formats of diskDISK. I have the problems you have with the TRSDOS driver. I called Montezuma Micro and they told me that there was a problem with that driver. Something to do with overlaying the drives? Anyway they had already mailed to me, and about four other people, the corrected driver. They said I would probably receive it on Tuesday. (I called them on Friday) You can probably expect to receive a new driver from them this week. Do you have problems with the LDOS version? I will let you know when I receive the new driver.

Fm Ron Ungashick To STEVEN JERKINS: I received new drivers yesterday from MM. The LDOS driver works with diskDISK but I can't SYSGEN it. I have had no luck with the TRSDOS driver. The system locks up when I attempt access diskDISK drives and when I boot from a sysgened disk I receive endless ERROR 11H (?) messages. Are you having any better luck than I am? I tried calling MM a couple of times today but got put on hold. I wish that Roy would write some drivers. I will attempt to call MM again on Thursday. Let me know if you have had any success.

Fm --jjkd-- To Ron Ungashick: Sounds like they still aren't totally up to speed. The limit is 203 logical cylinders with up to 256 sectors per cylinder. Their conflict was with the use of the double bit to indicate two physical cylinders per logical cylinder (406 physical tracks per surface max, still no more than 256

sectors per cylinder). They weren't using it right. Interesting that they call this "deficiencies of the operating system". Sounds like Jesse Bob. Well, as long as they fixed their problem I don't think I'll complain too loudly.

Fm Ron Ungashick To --jjkd--: I think that the "deficiencies" Jesse Bob referred to was the limitation of 404 cylinders per drive. At least that was the way I read it. Anyway they have succeeded with the LDOS driver. It works fine now. The TRSDOS driver still does not work after two replacement disks! They want me to return the disk again with an explanation. This is getting rather frustrating. But, I guess they are trying to fix the problem. I just wish they would test their fixes. Booting from a Sysgened disk does not work and diskDISK does not work.

Fm Ron Ungashick To Steven Jerkins: I stated previously that I could not SYSGEN the driver under LDOS. I was wrong. When I sysgened it before, I had a diskDISK active which is a no-no. I'm still having problems with the TRSDOS driver. (the second replacement disk). I called MM today and talked to John Long. I requested that I return the disk with a description of the problem. So I guess it will be a while longer.

Fm --jjkd-- To Ron Ungashick: You can indeed sysgen with diskDISK active. This should work, and I did it all the time when working with a fifteen meg drive and the Radio Shack TRSHD6 drivers. Note that this can be very dangerous, as if the diskDISK file moves, it's all over. Matter of fact, I was sysgened not only with diskDISK active, but with a type 2 diskDISK acting as drive zero!

Fm Mark P. Fishman To MISOSYS, Inc: I am puzzled, however; just how different mechanically are the 20 Meg and 30 Meg Aerocomp disks? I have the 30, and have not had a problem with their drivers -- not having dD, I am not sure what to try to make it fail (not that I am eager to have it fail). Does anyone out there know what the differences might be between these two hard disks/drivers?

Fm MISOSYS, Inc: The problem was specifically related to the setup of their 20Meg driver. Since that driver was unique to their 20Meg drive, their 30Meg driver apparently did not have the problem. Personally, I think it kind of wrong to not make a universal driver in this day and age. It was one thing when TRSHD3/DCT first came out. Another when

TRSHD5/DCT came out to support up to the Tandy 15Meg. But my RSHARD driver for Tandy drives supports up to 8 heads and 1024 cylinders. Gives a little more flexibility to the hard drive owner in case he/she wants to upgrade the drive later.

Fm Mark P. Fishman To MISOSYS, Inc: Maybe if you do suggest that we ADE users up(?)grade to diskDISK, I'll report back on whether the 30 Meg driver has the problem. Meanwhile, I'm happy. My wallet keeps yelping, though, as I want a few more things from MISOSYS and the wallet wants to be fed first.

What are the problems in using RSHARD on a Tandy drive and then buying a third party bubble (say, Miniscribe) to put in the box? Will RSHARD work so long as the controller is unchanged? So, in principle, one could write a version of RSHARD to use the Adaptec controller instead, and get the benefits of properly done code. I'm not suggesting you do this, Roy, just seems like a nice project for our copious free time.

Fm MISOSYS, Inc To Mark P. Fishman: RSHARD interfaces the Tandy controller - regardless of what drive you connect to the controller. It supports up to 8 heads and 1024 cylinders. True, we could write an Adaptec driver, but why bother? Thanks for the wishes.

Fm Ron Ungashick To All: After the third attempt to fix the TRSDOS driver for the 20 meg drive, the people from Montezuma Micro got it right. I had returned their second try along with a note explaining the inability to boot a SYSGENed disk. Jesse Bob Overholt said that in his haste to correct the driver he never did a SYSGEN. The 2nd LDOS driver I received as well as this, the third TRSDOS driver work correctly. I am able to create and use all formats of diskDISK as well as SYSGEN them. I know of no more problems with the drivers. I would like to thank Roy Soltoff and the other members of the forum who contacted Montezuma Micro about the problem with drivers. I am sure it sped up their fixing the problem. Now, maybe after 6 months of owning a hard drive, I can REALLY use it.

Correction to diskDISK patches

Fm Ron Ungashick To Roy Soltoff: Roy, I think there are a couple of typos in two of the patches [TMQ II.i]. In DDFORM63/FIX the first line reads

D02,A5=63;F02,A5=52

I believe the 52 should be 62. In patch DD6BH2/FIX both lines begin:

D02,2A=
F02,2A=

After searching with LSFEDII it appears that the 2A should be 8A in both lines. Thanks for uploading the patches.

How to use diskDISK

Fm --jjkd-- To Shane Dawalt: You should be able to set that drive up as two logical partitions of about ten meg each (the limit is about thirteen meg for a logical drive) when using RSHARD-style drivers. I would probably set it up as two, and then make a setup like this:

- 0) System on memDISK
- 1) Slot for diskDISK
- 2) Slot for diskDISK
- 3) Slot for diskDISK
- 4) Large HD partition
- 5) Large HD partition
- 6) Floppy
- 7) Floppy

The OS would be on a type two diskDISK, which would then be used to generate the memDISK during boot. If you want to use the memDISK for something else, I might then slice the HD into three pieces, with a small one for the OS. This costs you one place to put diskDISKS. "Where's my third floppy?", he says. "Do you really need it in all the time?", I responded.

Fm Shane Dawalt To --jjkd--: I suppose it is out of the question to install DOS on one partition and use the other partition without DOS on it too? Or can that be done? I'm not sure how these partitions would be used with diskDISK. I'm assuming these partitions are set up like the discussion has been going on this forum, that each partition is like a subdirectory. Is it absolutely necessary that the DOS be on it's own partition ... or is that housekeeping in action?

Fm H. Brothers To Shane Dawalt: There's a zillion ways you can organize the slots with diskDISK, a hard drive and a ramdisk. I have at least a half dozen setup /jcl's depending on the job at hand. Usually I have DOS and editors on the ramdisk as 0, compilers and

assemblers on diskDISK as 1, a diskDISK somewhere for source code (I've learned never to put the working copy in a ramdisk), and other hard drive, diskDISK and floppy slots that move around as the job requires. It is particularly easy to swap a slot back and forth between diskDISK and floppy drive as needed. On my hard disk, I have one partition for the DOS and working tools, another very large one that holds nothing but diskDISKs and a third smaller one that gets miscellaneous "stuff". But that's partly because I put together the hard disk and assigned partitions before I had diskDISK and I've been too lazy to change it.

The most important thing is to understand how diskDISK works, to know how big your ramdisk is, and then to analyze your own working requirements. That should give you a good idea of how you want to organize things.

Fm --jjkd-- To Shane Dawalt: DOS can be installed in any, all or no partitions. The only problem this will buy you is that only a partition that has DOS in it can serve as drive zero. The "large" partitions are used to hold diskDISK files, which are then "installed" in logical drive slots.

It is not absolutely necessary to keep DOS by itself, you are right, that's Good Housekeeping in action. Generally I keep DOS and all of my commonly used utilities, FED, LED, SAID, etc., in the same partition.

Fm Shane Dawalt To H. Brothers: You brought up a point which I hadn't expected. I thought once you had your system setup, you don't mess with it until something breaks or needs updating or whatever. But, you state you have JCLs to handle multiple cases. Wouldn't all applications run under a particular setup?

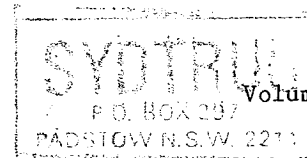
Fm H. Brothers To Shane Dawalt: Well, there are a lot of things that need to change from one setup to another. I have a Basic program that I use a lot for some specialized record keeping that needs all the memory possible, so I have to boot up without Pro-Wam to use it. If I'm programming, I don't dare put my source code in the ramdisk (especially when I'm working in a-1) because of the possibility of a lockup, but I want the source code to be in as low a slot number as possible. However, if I'm writing, I don't really care what slot the text files go in and often want other tools in low-numbered slots. Sometimes I need to have both floppy drives available; other times, those slots are better devoted to diskDISK pseudo-drives. Sometimes I want a lot of

space available for downloading -- other times, I just log onto CompuServe to read and answer messages (which I do online) and only need to make sure that Orchterm is available. It all depends.

Fm Shane Dawalt To H. Brothers: Perhaps I'm still mislead about diskDISK. To access a hard drive, it must be partitioned into individual logical drives. I know that (at least). I was under the impression that diskDISK used one slot per partition. I don't see how you can just blindly go and add or delete diskDISK slots. That would be like erasing a partition from existence. Maybe there is something else which fiddles with partitions. Maybe diskDISK doesn't go by partition arrangements. (Sorry if this doesn't make total sense. I'm not sure how hard disk utilities work with hard disks.) I am trying (perhaps incorrectly) to like an HD on a M4 with what I have learned while setting up Z-248 systems under MS-DOS with a 20M and 40M HD. On those, you set the partitions and forget them. If you change them, you run the risk of loosing data. If diskDISK is programmed such that each slot is one partition, wouldn't deleting one slot delete a partition and thus loose data in that partition?

Fm H. Brothers To Shane Dawalt: Let's start over from the beginning: You get a hard disk for a model 4 -- let's say it's 20M -- and a driver for it. Okay, let's say you decide to split it into 3 partitions of 5M, 5M, and 10M (there are probably better arrangements). Now, we both know that TRSDOS/LS-DOS limits the number of directory entries, so even on your 10M partition you can have no more than 256 files. If you have only very large files to store, you're all set. If your files on the 10M partition average 40K, you have no reason to go further.

BUT -- maybe you have a whole flock of small files. How are you going to store them? The DOS and hard disk drivers don't provide any answer. diskDISK, however, comes to your rescue. There are two parts to it: DDFORM/CMD and DD/CMD. DDFORM creates a file on a physical disk with an extension of /DSK. This is simply a file -- nothing fancy -- except that it looks like a physical disk: boot sector, directory, GAT and HIT tables, etc. You can choose to make the file look like a 5" single- or double-sided disk, like an 8" disk, or you can specify 1 or 2 sectors per gran. You can also specify the number of total grans (up to the DOS limit) in the "disk."



DD/CMD installs a small driver in memory which, using such a file, makes the file look and behave like a physical drive. It gets assigned one of your 8 DCT slots, has a drive number, etc. You can swap DD disks by simply running DD again and installing a new "disk" in the same slot. Or you can uninstall DD and then turn the slot back over to a hard disk partition or a floppy drive.

Even if you always leave your 3 hard disk partitions visible, you can use DD, a drive-swap utility, and the DOS Floppy program to move things around in a large number of permutations and combinations. I reserve one hard disk partition exclusively for /DSK files; it has about 30 "disks" on it. If you really want to get creative, you can even create a /DSK file inside of a /DSK file, although I have never found a reason to do so.

Does this muddy the water more or make things clearer?

More Problems with Aerocomp Drives

Fm John Tollini To All: Are there any special tricks to setting up a type 2 diskDISK on a HD for use as drive 0. I can make it work from a floppy or ramdisk but not from the HD. My system is: 4p w/128k XLR8er Aerocomp 5m HD LS-dos 6.3.

Fm --jjkd-- To John Tollini: Worked fine for me last time I tried it, it went something like this:

- (1) DDFORM SYSTEM:5
Answer the necessary questions, type 2 and 96 cylinders
- 2) DD :4 SYSTEM
- 3) BACKUP SYS/SYS:0 :4 (S,I)
- 4) SYSTEM (SYSTEM=4)

What exactly was the problem you ran into?

Fm John Tollini To --jjkd--: When I did the SYSTEM (SYSTEM=6) the system responded "Press enter when SYSTEM disk in drive 6". One difference I see is the cylinder count I only used about 20 cyls. NOTE: I took a SYSTEM/DSK File that would run as drive 0 from RAMDISK and copied it to the HD, Then it wouldn't work No problem using diskDisk's on the HD as long as I don't try to make it drive 0.

Fm --jjkd-- To John Tollini: Let's take this one from the top: What is your exact hardware

configuration? Computer Internal Hardware? Upgrades? What brand, style and size hard disk? Which drivers are you using? What version of DOS, number and level? What version of diskDISK?

Fm John Tollini To --jjkd--: Joe, here are the answers to your questions plus another question: Computer: 4p; Extra Hdwr: XLR8er board & 128K; HD: Aerocomp 5m; Driver: Aerocomp HD5SSA4; DOS: LS-DOS 6.3 Level K; PRO-WAM v2.01b; DISKdisk: V1.0.1

Also after I left the last message something else weird happened. Using Allwrite (V1.12) with a DD as a data disk, I had several files farked. When a file was saved to disk pieces were overwritten on other files within the DD. I've been real busy so I haven't had time to isolate it except that I've had no problems without DD.

Fm MISOSYS, Inc: John, If you send me a copy of the MM driver you are using, I may get a chance to peek at it and advise you if the thing has the same problem as their 20Meg driver. Also send a hardcopy of your Drive Control Table for the hard drive slots.

Fm --jjkd-- To John Tollini: My guess would be that there might be a conflict with the Aerocomp five meg drivers as there was with the Aerocomp twenty meg drivers. The twenty meg drivers were "broke" in a fairly subtle way such that the flaw only showed when using diskDISK. Does your 5 megger have more than 203 physical tracks per surface (the actual physical drive, not the "logical" drive)? If so that's another point towards the Aerocomp drivers being broke.

Fm John Tollini To --jjkd--: My drive is supposed to be 4 heads 153 cyls. You do raise an interesting question about the driver though. (I just finished TMQ II/i). Any suggestions on checking things out? One thing I don't think I mentioned in my previous message: my partitions are: :0 HDISK1 (1 head/153 cyls); :4 HDISK2 (1 head/153 cyls); :5 HDISK3 (2 heads/306 cyls); :6 Floppy #1; :7 Floppy #2

Fm --jjkd-- To John Tollini: Uh oh. That display for :5 looks very, very suspicious. If your drive is indeed a four head, 153 cylinder drive, that is impossible. If your problem was with a diskDISK file stored on drive 5, and the 306 cylinder report is from a DEVICE command, I would expect the problems you have been having. A workaround would be to reformat as four logical drive on the HD, and the

driver seems to be ok for a single head partition.

Fm John Tollini To --jjkd--: Joe, I just double checked. Device shows 306 cyls for 2 head partition. Roy also responded & I'm sending a copy of the driver & DCT table to him.

Fm --jjkd-- To John Tollini: Yep, that's it then. Same as with the twenty megger; they are setting the double bit incorrectly. Too bad good ol' boy Jesse Bob didn't ask if he wasn't sure.

Fm MISOSYS, Inc To John Tollini: I went through your previous messages and derived only the following. You created a type 2 diskdisk with 20 cylinders and then used that to backup the DOS. I noticed that you did that on a memdisk first then copied the /DSK file over to the hard drive. Copying /DSK files can usually be done; however, the number of extents permissible to contain the /DSK file is limited (I believe the number is 12). You must make sure that the copied file does not exceed that. I had no problem in generating and using a type 2 diskdisk of 20 cylinders as a system drive yesterday in testing. I also noticed a message where you claimed problems in writing ALLWRITE files to a diskdisk. Now, what has been the host partition? Where all of these diskdisks placed on drive 5? Drive 5 is your 2-head partition. Your drive is 4 head, 153 cyls.

After close inspection of the stuff you sent, I now know that the problem is indeed the Montezuma Driver, AGAIN!!! The problem will occur only if you select a partition having more than one head. I can't believe these guys!

Here's what THE PROGRAMMER'S GUIDE TO TRSDOS 6 says about the DBLBIT (that's DCT+4, bit-5), "When the DCT is associated with a hard drive, this bit may be used to indicate that a logical cylinder represents two physical cylinders thereby providing support for twice as many cylinders as limited by the Granule Allocation Table". Again, MM has the DCT set up wrong in the HD5SSA4/DCT driver if you choose more than one head for a partition. In the case of your logical drive 5, the data shows DBLBIT set, 2 heads, 32 sectors per track (SPT), 4 granules per cylinder (GPC), 8 sectors per gran (SPG), 153 cylinders. The number of sectors per logical cylinder is calculated by either $GPC * SPG * DBLBIT$ or $SPT * HEADS * DBLBIT$. The former results in $4 * 8 * 2 = 64$ sectors per cylinder (correct for 2 head partition on a 153 cylinder drive); the latter results in $32 * 2 * 2 =$

128 sectors per cylinder (totally INCORRECT for a 2 head partition on a 153 cylinder drive).

In looking at a disassembly of the driver (boy what contortion they went through), my hunch is that the author was so totally wound up in the data structure of the SASI controller that he/she completely misunderstood the easy translation of the DOS data structure to the SASI data structure. These SASI controllers do not deal with cylinder or head data for sector access. Those characteristics are uploaded to the controller at initialization time (@ICNFG). Thereafter, a sector access is requested by passing a logical sector number to the drive as part of a command data string. A logical sector address is a 24-bit number. I surmise that MM considered that a 2-head partition had 306 tracks, thus they set up the DCT with the DBLBIT set. Your complaint has to be with them, and I suggest that you get on their case. In the interim, if you partition that 5 Meg drive into 4 pieces, your driver will correctly set up the DCT.

Fm John Tollini To MISOSYS, Inc: Thanks Roy, on Joe's suggestion I have already reformatted my drive 1 head/partition. That does seem to solve the problem. I think I may still call MM and see if they will fix it.

Fm MISOSYS, Inc: With all of the problems with MM's drivers, I sure wish I had the time to adapt my VRHARD driver to their drives. Sure would save a lot of these headaches and frustrations.

Incidentally, I just found another problem with their Model III 5Meg driver, HS5SSA3/DCT. If you are using it on a Model 4 in III mode with our SET2RAM installed, it crashes. Seems they use a verify buffer at address 0 (crunch). If anyone else out there has that driver, I believe a patch at X'5683' to 26 38 from ED 62 will work. The D location is D04,6F. I changed the SBC HL,HL (which sets the buffer pointer to zero if the operation is verify) to a LD H,38H.

Miscellaneous problems with diskDISK

Fm Robert S. Arner: Concerning the question of whether or not you have to disable a diskDISK drive before loading another, it would appear that my exhibit shows that I can repeatedly load different diskDISKs into the same drive slot with no apparent ill effects.

Next question. Contrary to your documentation, if I disable both diskDISKs, the \$DD and \$DL diskDISK format and linkage programs are not deleted from memory, and nothing whatsoever was done to write to high memory between. The formats are not the issue. Of course, if I install just one diskDISK, then upon disabling the one, all of the high memory is cleared.

You will note that there is an extraneous character appearing between the "=" and the "32/58" in the header [when I do a FREE or DIR command]. This only happened when I did a backup by class and a mirror image backup from the diskDISK to a floppy. You will note, of course, that the diskDISK file itself has 181 granules, while the floppy has only 180.

Final note. If you do it in TMQ, then you can also skip this paragraph. I am 55, an optometrist, Amway distributor, Black Belt in Karate, and ocean sailor. I have a 42 ft ketch, at great personal sacrifice, on Chesapeake Bay, which my wife and I are going to sail down to North Carolina this week. My eventual goal in the near future is to sail across the ocean again, and be a sailboat bum for a while. If I don't stop spending money for computer hardware and software, the dream will surely be delayed.

Fm MISOSYS, Inc: You are correct in that DD can install a diskDISK into an active drive slot. In fact, DD permits you to install the linkage into an active drive slot even if it is not in use by diskDISK already but by some other drive (say a floppy or hard drive).

On the other hand, I am correct about diskDISK removing itself from memory - providing you do it "correctly". Here's the problem. When you first install diskDISK, it loads a base module (DD) then the linkage to the /DSK file (DL0). If you install another diskDISK, it only installs another file linkage (DL1). If you now disable the first installed diskDISK, it must retain the DD module. It can get rid of the DL0 linkage but note that the DL0 piece is trapped in memory by the DL1 linkage. Now if you disable the second installed diskDISK, it can remove the DL1 linkage but the DD module is now trapped by the DL0 linkage. DD is not smart enough to detect that it is trapped only by a /DSK linkage which is no longer in use. A minor shortsightedness in its design.

The difference in file size of GAMES/DSK on HDISK3 (181K) vs the diskDISK size of 180K stems from the need for diskDISK to have a header sector prefixing each /DSK file so that

it can determine the logical configuration of the diskDISK when installed. That sector is transparent to your operation; but DD needs this information.

I don't know why you get the extra "." in the header listing of the type 2 diskDISK only after it is backed up. A type 1 is a funny animal. I'm not really up to a thorough investigation, especially when it would involve chasing straws. I did create two type 1 33 track diskdisks and could not duplicate your result. If you get into the same situation again and can duplicate it repeatedly, copy the /DSK file from the host drive to a floppy and sent the floppy to me with a note. Perhaps we can then isolate it.

Fm Dr. Robert S. Arner, O.D.: Thanks for your letter, and the phone explanation a few days ago. I used to feel great anxiety when I would call you the first few times; but have enjoyed the last conversation or two. Either you are mellowing, or I'm learning the language...probably the latter.

Now for the part you can skip...just got back from moving or 42ft sailboat from Baltimore MD, down the bay, through the intercoastal waterway to Minnesott Beach, NC, just west of Beaufort and Morehead City where we will stay for the winter. Had gale force winds from the Northeast for four days which emptied the water out of the waterway!! It was kind of exciting to tool along in a 39,000 lb boat with the keel dragging through the mud! In the spring, I will go out there, do a little work on it and then move it down the ICW towards Florida, the Bahamas, and eventually the South Pacific. Its exciting and a bit scary to contemplate, but then, that's the fun of it.

Regarding the "." in the type 1 diskDISK, it does happen repeatedly with the enclosed file. It does not seem to happen when I do other diskDISKs. Obviously not a big problem in terms of function, just aesthetics, to see that big blob on the header.

Regarding the extra file size for a /DSK, i.e. 181 rather than 180, this presents a problem if you want to backup the /DSK as a file to one floppy, rather than as a contents backup. Since the floppy only holds 180K there is a problem. This came about because I wanted to remove the MOD flag from a /DSK file by doing a backup. In addition, it would be nice to flag the /DSK file with a MOD flag if one of the contained files was modified. In this way, one would know whether or not to install the

/DSK file and do a backup, or if the 181/180 problem didn't exist, just backup the /DSK file. Obviously, one way around this is to create a type 5 /DSK with only 179K so that it would come out to 180K with the header. Again, this observation is gratuitous if you don't want to fool any more with LDOS and Model III/4 programs.

Fm MISOSYS, Inc: Reference your third problem, the type 1 diskDISK. I couldn't duplicate your results until I received your file. Here's why. LDOS 5.3 recognized your UTILITY diskDISK as a SYSTEM disk because a particular bit in the configuration byte stored in the directory was reset. This bit is used under LDOS 5.3 (and all releases of TRSDOS6/LS-DOS6); set implies a data disk, reset implies a system disk. Under this concept, a system disk always reserves 14 file slots in the directory for system files. Your UTILITY diskDISK had 43 files plus 1 for the BOOT/SYS and one for the DIR/SYS entries. Since the directory on a type 1 DDEN diskDISK is 8 sectors long, it can hold $6*8=48$ files. But both the DIR and FREE commands calculated the available file slots by subtracting $45+14$ from 48. This resulted in a negative number which was incorrect and unsuitable for the binary to ASCII conversion routine used by DIR and FREE. That's why the display was garbaged. The solution to the problem is to correct DDFORM for use under LDOS 5.3. This was done in THE MISOSYS QUARTERLY Issue II.1. A copy of the page containing the relevant patch is attached. Once applied, all future type 1 diskDISKs created with DDFORM will be correct (even DATECONVed). Now to correct the UTILITY diskDISK already in use, change the byte I have circled on the attachment from 47H to C7H (relative byte CD in the first sector of DIR/SYS). Use FED for that by editing DIR/SYS.SYSTEM.

There is also no way around the 180K vs 181K difference. DiskDISK requires the header sector. True, you could use 39 cylinder diskDISKs; but is that really necessary? It is quite easy to attach a /DSK then use a mirror image backup to a floppy. There is also no easy way to update the directory of the /DSK file on the host drive to use it's MOD flag as a "supervisor" of the diskDISK's directory.

Lastly, I did not skip over your paragraph which related your sailboat adventures and upcoming journey. Brenda and I went on a day sail once; on our 4th anniversary vacation last year. We vacationed at St Thomas for a little under a week. While there, we took one

of the one-day sails on a 60 footer. There were six of us plus the two-person crew. Really a great experience. We do a little boating with our 16-ft outboard, but that's nothing compared to the quietness and serenity of a sail on a larger craft. I wish you well on your trip; maybe you'll take along a laptop.

DiskDISK and memory usage

Fm John H, Mercer: My principal use (so far) for LS-DiskDISK has been with cataloguing and archiving exchange newsletter articles. The word processor I use is ALLWRITE by PROSOFT. As you are no doubt aware, ALLWRITE has the facility to edit three different documents residing in memory at the same time, if using a Model 4 with 128K of memory. I have noticed on a couple of occasions, if I have two or more LS-DiskDISK files active, when I try to invoke Area 2 from ALLWRITE I get a message 'Memory not available' (or words to that effect). Does this mean that LS-DiskDISK is using some of the extra bank(s) of memory for something and this is preventing ALLWRITE from gaining access? If this is the case, are you aware of any procedure which will enable LS-DiskDISK to utilize a portion of memory, not located in banks 1 or 2 so that these banks are available for ALLWRITE? I feel sure that LS-DiskDISK is not actually using very much memory, but perhaps you could enlighten me on this.

Fm MISOSYS, Inc: LS-DiskDISK does not use any of the memory in the alternate memory banks. On the other hand, I believe I have an answer. If HIGH\$ is too low, ALLWRITE will not allow you to use Area 2 or 3. It probably needs a minimum amount of standard memory in order to switch banks. For instance, if I have PRO-WAM loaded (which takes about 2.5K), I can't switch to Area 2 or 3. If you take a look at what's in memory, you'll find that diskDISK takes very little. You may even be able to fit all of it in low memory unless you have other things like a hard disk driver, forms filter, etc., installed. If you can reduce your high memory usage a bit, you'll probably be able to switch areas even with two diskDISKs active. If you check directly with PROSOFT, they'll be able to tell you the exact figure for HIGH\$ which would limit area switching.

Data Management: LB

LB and memory constraints

Fm Pete Granzau To MISOSYS, Inc: I was wondering about LB. My normal configuration (with all filters, etc, installed) leaves HIGH\$ somewhere about X'F320'. I remember some comments about LB not obeying HIGH\$, or maybe it wouldn't accept relatively low values. Any information?

Fm MISOSYS, Inc: I think that is a tad lower than what LB will live with. I just checked the INSTALL code, it checks on HIGH\$ below X'F800' as too low. LB itself checks for HIGH\$ < X'F7FF' as too low. Has to do with the buffering requirements, etc... I expect to look into reducing the memory requirements some time in the future; but it's down on the list of things to do.

Fm --jjkd-- To Pete Granzau: Little Brother observes HIGH\$ as much as possible, but will refuse to run with HIGH\$ below X'F800' (in the original version, anyhow). Roy was discussing re-doing LB with Pro-MC, the original version was done with Aztec's C80, so this could change.

Unsolicited comment on LB

Fm Bill Schaper To ALL: I'd just like to drop a note to comment on the Little Brother program from MISOSYS. As an Accountant, I decided to implement a time study program. Sure it required all to submit their work (per pre-coded activity schedule) on a continuous basis. The information is based on calendar month, activity code, time (start/stop) and date worked on. The Little Brother program can sort on employees, months, activity ... you name it. The program is fantastic! It allows different input screens (user defined) and different print screens (user defined). I would be hard pressed to find a better data base for the Model IV (and for the MS-DOS arena).

My associate, who is in the insurance/annuity field, thinks enough of it to put it to the test to flag birth dates, renewal dates, type of policy dates and other information, be it mailing labels or what-ever.

After the brief time I have used the program ... I can only give it praise. The sort method/time is one of the best I've used and the Index/Select option allows printing of ANY selected field desired. It is more than worth the REASONABLE (small) price asked.

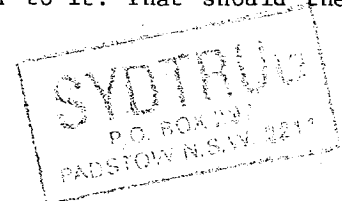
LB86 Configuration problem

Fm Robert Hardie To MISOSYS, Inc: I am using a Tandy 1000SX, 2 disk drive, 384K computer with 3.2 MS-DOS. I have been unable to load LB86 onto a formatted system disk because of insufficient disk space. Is this because I am using MS-DOS 3.2 rather than MS-DOS 2.x? Also, I do not have CONFIG.SYS. Can I use MODE.COM? I have determined that I can configure DEVICE=ANSI.SYS by using COPY or a .BAT file.

Fm MISOSYS, Inc: Because DOS 3.2 takes up more space than DOS 2.x, all of the LB86 files cannot fit onto a DOS 3.2 disk. That's an easy fix. Just use a data disk. All that is needed after you are booted up is to have COMMAND.COM on the disk containing the LB86 program files. You can easily do that by taking a DISKCOPY of the LB86 disk and copying COMMAND.COM to it via the COPY command.

You also appear to be unaware of the MS-DOS environment. You should take some time to read over your MS-DOS user manual. CONFIG.SYS is a file you create; it contains certain configuration commands optionally used by MS-DOS. There should be at least one section of your MS-DOS user manual which discusses CONFIG.SYS, what it is used for, and how to create it. It should specifically discuss the ANSI.SYS driver. Since EDLIN is a line-oriented editor which is included with MS-DOS, you should at least get familiar with that. You can use EDLIN to create and/or edit CONFIG.SYS.

With that out of the way, under 3.2, you would need to boot an MS-DOS system disk which has ANSI.SYS on it and the CONFIG.SYS file which includes the line, "DEVICE=ANSI.SYS". After booting that disk, you can remove it and insert the LB86 work disk which has had COMMAND.COM copied to it. That should then be usable.



SAID

SAID and tabs: Problems? No!

Fm Shane Dawalt: I have a question (or bug report) on Model 4 PRO-SAID, vl.1 with all of the TMQ patches installed. The trouble: SAID seems confused on tabs which occur at the end of a line. At times, I place one too many tabs at the end of a line which causes a line wrap. No problem, delete them, right? Wrong. I'll let you see the problem. [editor note: the example line is not printed here as it would require an 80-column line to illustrate] Enter the line in your SAID and place the cursor (in overtype mode) over the space between "integer" and "*/". Type <CLEAR><RIGHT ARROW>. The cursor should be on the "*" on the next line. Press <BACKSPACE> and the cursor is placed at the beginning of the line, not at the end of the previous line as would be expected (since that's where the tab originated at). Press the <BACKSPACE> again and the cursor is positioned over the "r". That's understandable since the last character was a tab. But the question remains, how to position the cursor after the "r" to delete the tab. Turns out, it can't be done. The best that can be done is to delete "r", then delete the tab which then appends "*/" to the word "integer". Then, "r" must be inserted. I tried just deleting the tab character (just using backspace once instead of twice like I said previously). That deletes the tab fine, but it still doesn't append the remaining contents to the end of "integer". In any case, the "r" must be needlessly deleted to force SAID to append the next line contents to "integer". Is this an oversight, a bug or the way it is supposed to work? I think it a bug.

Fm MISOSYS, Inc To Shane Dawalt: You seem to be making a mountain out of a molehill. I did exactly as you said. I placed the cursor in overstrike mode over the SPACE between "integer" and the asterisk. Then I entered a TAB. Correctly, SAID noted that it had to "wrap" the tab expansion spaces to the next line. Since the tab expansion is logical rather than an actual insertion of spaces, SAID has to make some decisions as to where to position the cursor for forward and backward positioning across a TAB expansion. You are wrong to infer a bug is present. If you are positioned on the asterisk and depress BACKSPACE, SAID moves the cursor to beginning

of the line; that happens to be an arbitrary location. If you want to delete the TAB, or overstrike it with another character, that's where it is. The fact that you can replace it with a SPACE and the line doesn't close is also not a bug; the line will close back up when the screen is redrawn (hit BREAK and you will see). The behavior of SAID where a TAB is at the end of a line is as expected.

Fm Shane Dawalt To MISOSYS, Inc: I didn't know that BREAK "redraws" the screen. I though there had to be some way of regenerating the display for situations just like this. I have ran across this many times, and it finally prompted me to write. Thanks for your time.

I rather expected SAID to be positioning the TAB at the beginning of the next line rather than simply padding spaces across the line boundary. My concern was with SAID not recognizing that a TAB had been deleted and that it should have automatically appended the line onto the previous (given that the current line was wrapped by a tab from the previous line). I'll try BREAK.

Fm MISOSYS, Inc: BREAK does more than regenerate the screen. It also closes up spaces which were inserted due to an INSERT-LINE operation - provided the cursor is currently positioned within the "line which was opened up". That action then requires that the screen be redrawn.

Fm J. R. Jacques Baril: I am glad I got that SAID editor with PRO-CREATE. It has become my favorite text editor. The only improvement I would like, and I am not complaining or even asking hard, is for a less awkward keystroke combination to tab.

Fm MISOSYS, Inc: As far as SAID's "insert TAB" character, don't forget that SAIDINS allows you to define that to be whatever YOU choose. It's not as if we force you to use <CLEAR RIGHT ARROW>; it's just that the Model 4 is kind of sparse on discrete keys. It would have been great if Tandy had a TAB key which was different from the RIGHT ARROW key. You can't have both cursor movement and tab insert with the same key!

PRO-WAM

PRO-WAM and word processing

Fm Peter Amschel To MISOSYS, Inc: We are enjoying Pro Wam 2 very much on LeScript. My secretary does not even know that Pro Wam will import and export data to certain word processing programs which do not grab the keyboard drivers, or whatever it is. Which word processing program is it? Do you think it would be practical to use 2 word processing programs, such as the LeScript for big, long court pleadings and briefs and then the other program for letters? The letters need the data import and export function more than the pleadings anyway.

Fm MISOSYS, Inc: PRO-WAM can export to SCRIP-SIT, ALLWRITE, and I believe "Lazywhatever". With Lynn Sherman's new CTL255 filter available in the DL (our Compuserve forum), PRO-WAM can export to Superscripts as well (although I personally would not recommend SS). You may be able to get by using two WPs, that is if it won't confuse your help.

Fm Lynn Sherman To MISOSYS, Inc: Are you familiar with LESCRIPT? Do you think it might access the keyboard via the @CTL 255 call? I'd be curious to know if the filter would work with it. At one time I had the Demo version of LESCRIPT, but I guess I decided to use the disk for something else. Wish I'd kept it now.

Now for the question you'll wish I hadn't asked - I really like the PHRASE/APP. Its the perfect way to store all those boilerplate kinds of things that I use in wordprocessing (like signature blocks, address blocks, distribution lists, etc.). Now that I can export to Superscripts, I expect to make frequent use of PHRASE. It sure would be great if it could export strings of characters longer than 78, say 254. I imagine that this would require a major rewrite of PHRASE and is therefore not in the cards. Is something like this possible? How about a special character at the end of a PHRASE/TXT line that indicates that the next line should be exported as well? Hope you don't mind my throwing "wish list" ideas at you at this late date - the value of something like this didn't occur to me until I starting exporting to Superscripts.

Fm MISOSYS, Inc: Of course, most things are doable. Actually, my first desire was to try that kind of implementation. Unfortunately, the import/export interface was too tied in to rectangular blocks that a major change would have been needed. I really didn't want to get to another terminating character to indicate premature termination. That would be needed to avoid a bunch of trailing spaces on the last line (if it wasn't filled). I took the best compromise that I could.

Fm Lynn Sherman To MISOSYS, Inc: Yes, I can see the problem with blank spaces. In fact, I could use CARD/APP to create the boilerplating, but wind up with blank spaces as you say. PHRASE is a very useful and versatile application anyway. I was just wishing out loud. I may play with some ideas myself, as time permits. I hope the new release sells at the high volume it deserves. I'm very pleased to have been of any help by Beta testing.

Fm MISOSYS, Inc: It's not impossible to install a programmable export terminating character; just not practical at this point. But yes, I think PHRASE can be a very exciting application. My hint on adding it to a KSM key should not be overlooked. That can be useful to emulate additional memory resident applications. If you move the library to a ram disk or operate off of a hard disk, the access time is minimal. KSM can get you into an application with one keystroke (two keys).

Fm Ken Kane To Peter Amschel: I just dusted off my copy of LeScript and found that (a) PROWAM imports from the LeScript screen with no problem; (b) attempts to export from PROWAM CARD/APP to LeScript leave no trace!

My vote, buy ALLWRITE for documents long, short and in between. Move around your document, search and replace almost instantly (all in RAM). Jump tall buildings with a single bound, (provided they are less than 32276 bytes long). But ALLWRITE links document segments so smoothly that it is not a problem.

My copy of LeScript is available, say for \$25. I bought it because it can handle documents of indefinite length, if you have time to wait while it plods through text.

Speaking of Word Processors, can anyone recommend a MS-DOS WP that can cut and paste between files as ALLWRITE does?

Here's some PRO-WAM uses from our readers

Fm Ken Kane: My favorite new application for PHRASE/APP in PROWAM is for browsing the DL's. I set up a KSM key to invoke PHRASE as per Roy's instructions on pg. 112. Last night I did a search for ink jet printer reviews on several SIGs. My PHRASE/TXT file included:

```
S1BRO/KEY:DICONIX <chr127>
S2BRO/KEY:HEWLETT <chr127>
S3BRO/KEY:INKJET <chr127>
S4BRO/KEY:JET <chr127>
```

I didn't find much, but was my searching fast! I searched about a dozen DL's on several SIGs in about 2-3 minutes. True, I could have used separate KSM keys for each BRO/KEY: but this is less disruptive. Since the first PRONTO release I have used CARD/APP to keep track of message numbers to reply to, (composing replies off-line, etc).

Fm Ken Kane To MISOSYS, Inc: Well, as for PHRASE/APP a la PRO-WAM uses... I used to keep two keys tied up under KSM to log onto CIS at 1200 to download, and the other at 300 baud et alia for XMODEM. I now have other ways to use my DTERM communications program including experiments logging on at 2400 via TELENET, and at 2400 to a local BBS. Well, all these different baud rates and script files are stored in PHRASE/TXT, very accessible yet very easily changed. PHRASE/APP has shortened the export function by several keystrokes.

I have always loved KSM functions, being a better button-pusher than typist. PHRASE/APP greatly expands the KSM function only a little less accessible. But very easy to change and to expand. One more reason to hang in there with the ol' Model 4.

Fm MISOSYS, Inc: If you have Mister ED, then you can even use TED/APP to alter the PHRASE/TXT file on the fly. That's akin to dynamic alteration of the macro text. I think that PHRASE will wind up with more uses than I ever intended. The ability of coupling it with a KSM invocation to cut down on keystrokes makes for a very useful environment.

Fm Ken Kane To MISOSYS, Inc: Agreed. I hadn't thought of using TED/APP, always go back to DOS and use the word processor. But I'll use TED for that now.

Fm MISOSYS, Inc: That's a perfect use for TED/APP. You can use it right in the middle of

doing something else; assuming you have a spare memory bank.

Fm Ken Kane To MISOSYS, Inc: Roy, Yep, I'm off to fatten up WAMO/APL by adding TED. (A quick on-line check using DOS through PRO-WAM shows that WAMO is already 48K!. But I know that what's in there is what I want there.

Fm MISOSYS, Inc: Don't forget that you can access any /APL from the PRO-WAM menu. If TED was in, say, WAM1/APL on drive :1, then it could be accessed via the spec of "ltd:1".

CAL - Another small problem fixed

Fm John Tollini To MISOSYS, Inc: I ran across something that may be a bug in cal/app. I have release 2.01a of PRO-WAM. The bringup flag in cal displays off 1 day (early) in cal but shows up on the right day in bringup. I almost forgot, this only happens in 1988. I haven't seen anything about this before so I thought I'd leave a note.

Fm MISOSYS, Inc: Per your phone call, I took a look at the code. There is a bug. I need to shuffle around a little code because an ADD HL,BC instruction does indeed change the CF. Also, there is another bug if the day marked is the 31st of a month. I will be working up a patch soon. At least you see a STAR!

Fm Lynn Sherman To MISOSYS, Inc: I checked John's comment on CAL. He's right, CAL does mark BRINGUP activities one day early in 1988. In my setup only the activities in January and February are offset by one day. From March on they are correct. However, my first BRINGUP entry for March is on March 1st. I don't remember this occurring before the patch to correct the change in Today's date when changing years, but certainly might have missed it.

Fm MISOSYS, Inc: It did occur, just nobody noticed. We'll get it fixed. Already started to work on a patch. I need to squeeze a little to fix the problem if the date is the 31st; that bug hasn't been reported - I picked it up in checking John's report.

Where or when is the Programmer's Toolkit?

Fm Bob Haynes To MISOSYS, Inc: Received PRO-WAM 2.0 and LB the other day; thanks for the quick service. I've resisted the impulse to immediately load the software, instead reading

the docs as I should (first time in years, he said!). Umm, since my fresh new 2.0 was an upgrade rather than a new purchase, would appreciate clarification of a couple of questions:

Are the 1.0 programming tools now rendered obsolete by the upcoming toolkit mentioned in the docs? Should they be scrapped as part of the upgrade agreement? May they be retained for purposes of writing old style /APPS for the benefit of 1.0 users? Or should upgrade owners use the old tools but use the "PROWAM" (nee PRONTO) header for 2.0 applications, and not worry about the toolkit at all? Sorry, maybe these issues are obvious to you. They're not to me, and I do want to play by the rules.

Will any of the current MISOSYS/TRS-80 product docs (MC, MRAS, EDAS, etc) be changing over to the new smaller format? If so, would you consider selling fresh docs to REGISTERED owners? If you're printing them anyway, perhaps that could be another source of income for you. I'd love it! (And how about the GUIDE? Naw, now I'm really pushing.)

Fm MISOSYS, Inc: The programming tools under PRO-WAM release 1 will suffice with one exception which I will cover in a future TMQ. Has to do with @WCLOSE. Briefly, register setups are changed.

C=0 for close, no export;
 C=1 for close with manual export;
 C=80H for programmed export; CR at EOL;
 C=81H for programmed export: NOCR at EOL.

Under programmed export, HL holds the upper left corner (absolute to 23x79) and DE holds lower right. This differs from 1.0 where C<>0 implied export.

Also, @WEXEC is function 12. HL points to the spec string. The WAMLIB and PRUN utilities will accept either the old "PRONTO" or new "PROWAM" strings so coding an /APP with either will work. The new module name of the high memory module is "WAM"; it was "PRONTO". Got it? We also published a patch to the WINLINK driver in the last TMQ so that it checked for this new module name.

We don't foresee sufficient business in the TRS-80 market which would necessitate any re-printing of user manuals. We have enough of most except MRAS which we are now duplicating on our copy machine; we have no justification to print any manual in the volume sufficient for offset printing. Thus, there will be no revisions to the existing manuals to downsize

them. At least not until we see a preponderance of orders; difficult to imagine that happening since it has never happened in the past. We also have at least 500-600 LDOS 5.1.4 manuals. That's why you won't see a revised 5.3 LDOS manual. Not enough buyers to justify the work. We were contemplating writing a combined LDOS/LS-DOS manual in three sections: DOS, BASIC, and Technical - but I just don't see a rationale for that any more.

Fm Bob Haynes To MISOSYS, Inc: Thanks for your several responses, Roy, especially the one re PRO-WAM 2.0. Appreciate having that info. Will very likely want the 2.0 toolkit, I think, if for nothing else but the docs. Keep us posted.

Fm MISOSYS, Inc: Will do, but I don't foresee the toolkit coming out too soon. Looks like more past the end of the year than anything. Too much else to deal with and I don't think too many folks will need it, or want it.

How about Mister Ed's TED?

Fm John Tollini To MISOSYS, Inc: Are there any fixes ;for ted/app? The last README/TXT entry on my MISTER ED was 6/6/86 ;file date for ted/app 12/2/86. I know there were a few for ted/cmd ;but I don't see any for ted/app.

Fm MISOSYS, Inc: We never worked up any fixes for the few problems which are common to both TED/APP and TED/CMD. One day we may discover a way to get some patch space in the /APP.

CARD and CARDX must have ONE record

Fm Dirk Vandebossche: When I create a new CARDx/DAT file, the program immediately puts you into the 'Add' mode. The ^F to file; BREAK to abort prompts are displayed, but the BREAK function seems not to work.

Fm MISOSYS, Inc: That's how CARDX is supposed to behave. CARD does the same thing. They both require that at least one record is in the card file once it is created. Thus, when you first start a new file, the /APP goes into ADD automatically. BREAK won't abort since it finds 0 records. Simply do a ^F to file the record (even if blank). Then you can exit.

TODO/DAT only creates on drive 0!

Fm Dirk Vandenbossche: It seems impossible to create the TODO/DAT file other than on the 0 drive.

Fm MISOSYS, Inc: Here's a fix for the TODO problem. You were the first to notice that.

PATCH WAM1/APL (D5C,56=08:F5C,56=0B)

How do I edit DIALER macros?

Fm Stephen A. Torkko: In the DIALER application, I cannot get the MACRO option to respond. I have read the manual, and it just does not work as detailed. After invoking MACRO from the menu and getting the prompt:

Edit macro (@-0) ?

no matter what I key in from "@" to an alpha character A-0 followed by an <ENTER>, the cursor arrow (==>) simply returns to the directory without "the string (being) presented for (my) editing convenience" as indicated on page 101 of the manual or anything else happening. The dialer function works with my autodialing modem and I would like to send the "ATDT" command to switch it to tone dialing from the defaulted pulse dialing. Also I would like to be able to set up a macro to access long distance networks. What am I doing wrong?

Fm MISOSYS, Inc to Stephen A. Torkko: The MACRO editing in DIALER works; you just were too quick on the draw with the <ENTER> key. When you get the "Edit macro (@ - 0) ?" prompt, just enter the single letter of the macro you wish to edit. <ENTER> will terminate the editing.

PRO-WAM, DOS Spooler, and ALLWRITE

Fm Stephen A. Torkko: When I have the spooler module of DOS installed in bank 2, and I am using my ALLWRITE word processing software, if I invoke PRO-WAM the whole system locks up and I have to reboot. I am getting around that problem by yanking the spooler out of memory except when I absolutely need it for a long document print. I need PRO-WAM much more frequently than I do the spooler.

Fm Claude E. Hunter: I received PRO-WAM Version 2 several days ago and find the package

very interesting. I think I may have discovered a bug with the program. My system is a Model 4P with a 15 Meg HD and two printers. I've used the import/export function of PRO-WAM successfully when I ran ALLWRITE, entered PRO-WAM, executed a DIR command and used export to send the filename to the ALLWRITE filename prompt. Everything worked fine until I attempted to spool my printer with the following command before entering ALLWRITE:

SPOOL *PR TO SPOOL/BUF:2 (B=2,D=300)

When I attempted to export a filename, the system froze. I have attached a copy of my memory configuration.

Fm MISOSYS, Inc: The problem with the LS-DOS spooler relates to the same issue as that fixed by the BOOTFIX. We just learned of this the other day. Here's a patch you can install to LS-DOS which will correct that.

PATCH SYS8/SYS.LSIDOS
(D21,E4=40:F21,E4=60)

If anyone is interested, see page 415 of THE SOURCE - Library Files (line 14520).

HEAD/APP: Great but...

Fm Stephen A. Torkko: Would it be possible to patch the Address and Head modules so as to allow either for a longer first name in place of the Data1 space or the incorporation of the Data1 information in the Head import directly before the first name. Especially in my profession [Stephen is the pastor of a church], there is often need for a title to be prefixed to a name in an address like "Bishop" or "Rev." or "Dr." or "Pres." and it would be so very handy if that were in the address file; the few keystrokes involved in Head would result in an address name like "Bishop Lowell Knutson". If the first name is short, I can "cheat" by including the title as part of the first name, but as with the example given, "Bishop Lowell" is three letters too long for the allowed field length. I don't expect you to spend significant time on this request since you do need to be about the money making projects of your firm, but tuck it away somewhere as a "challenge" to your creativity.

[By the way,] when using the video preview feature of ALLWRITE with the Screen Saver module installed, if I pause the listing, the system locks up and I have to reboot. I have

been getting around that problem by not pausing the previewing of the document.

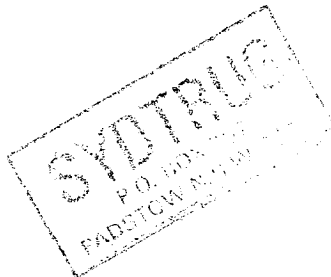
Fm MISOSYS, Inc: The field structure of ADDRESS (and hence HEAD) is quite dependent on the assembler coding of those applications. It may be possible to patch ADDRESS for some slight variation in fields; but it would be rather difficult. It would also be easier to rewrite HEAD with custom specifications than to try to patch that application. Maybe this is a good time to learn to tinker with assembly language programming? Of course, in your profession, you probably have enough to do. I had always wanted to work on a user-definable field structure application like ADDRESS, but the time never materialized. At this point in time, I don't foresee any economic return on Model 4 software development; thus, I don't expect such a thing to be written.

Lastly, I can't address the third problem you are having of system lockup using PAUSE of ALLWRITE's preview feature when your PD screen saver utility is installed. What does screen saver do? Is that something which blanks the screen when there has been no keyboard activity for a period of time? I can't even guess at what that problem could stem from. I see from your device printout that SS filters the keyboard device before PRO-WAM is installed. Have you tried it the other way around? These are just suggestions; no need to respond.

PRO-WAM and Job Control Language

Fm W. J. Russell: Please find enclosed my PRO-NTO disk for upgrading. It has been useful but not as much as I would have liked. There seems to be some conflict with JCL files and LeScript. This is the file:

```
. START/JCL
. PROGRAM 1 MTERM
. PROGRAM 2 LESCRIPT
. PROGRAM 3 LESCRIPT with PRONTO
//KEYIN Select program 1, 2, or 3....
//1
MTERM/CMD:0
//EXIT
//2
LESCRIPT/CMD:0
//EXIT
//3
PRONTO (ACTIVE=170)
LESCRIPT/CMD:0
//EXIT
///
```



Although the calculator and calendar worked, the filer and dialer would not. I hope the upgrade can give me a system with better compatibility. Certainly the Bringup would be very useful.

Fm MISOSYS, Inc: Neither PRO-NTO nor PRO-WAM will work properly or as expected when you are executing a program driven by Job Control Language. The difficulties generally relate to the operation of JCL. You must realize that JCL fetches its input from the disk file whenever an @KEYIN service call is invoked; JCL ignores @KEY requests. LeScript won't generate @KEYIN requests, but it's entirely possible that something in use while you are using PRO-WAM may. That means the request would be satisfied by a line from the JCL file. Since the line after the LeScript command is "//EXIT", the operation will terminate. If you change the "//EXIT" to a "//STOP", then invoking something from PRO-WAM which uses an @KEYIN call will function as expected; on the other hand, when you exit LESCRIPT using //STOP without PRO-WAM intervention (or none which uses an @KEYIN request), then you will not get the "DOS Ready" prompt. Our last THE MISOSYS QUARTERLY on page 73 discussed an issue of PRO-WAM and JCL which may work to your benefit. We did alter Release 2 so as to be able to work with JCL in some instances. Without further details, I can't speculate on what you mean by, "although the calculator and calendar worked, the filer and dialer would not."

Entering PROTECTED characters

Fm Dirk Vandenbossche: Some time ago, I received the PRO-WAM program. As everybody should know, the best product there is for the Model 4. In this letter, I'd like to pass on some of my experiences.

One rainy evening I was experimenting with the CARDX Filer and CARDXF Forms applications. I was creating a database form with CARDXF (you know, fields for names, addresses, etc...). Except for the specific input fields, the whole window was in reverse video. Afterwards I could use CARDX to introduce the data.

When I wanted to save a certain record, I mistakenly pushed <CLEAR><F> in stead of <CTRL><F>. I saw on the screen the letter 'F' was displayed in reverse video, and it wasn't possible to erase it (that is normal). Immediately I saw some new possibilities. Indeed, every time I pushed the <CLEAR> key together

with another key, the character was displayed in reverse video. So I got the possibility to enter data in reverse video in a blank field, and in that way protected data were created. As CARDX makes use of reverse video character strings to protect them from editing, you get a protected field and protected data (e.g. to store a birth date, which normally never changes). Caution: If you type the wrong letter, you can't modify it, but you can push <BREAK> and start anew.

Maybe in the USA you can't do this, because you don't use the French AZERTY keyboard and TRSDOS 6.2.F (F=French). I have never had any problem with American Software and this keyboard, except for LS-DOS 6.3. Because I want to use the 6.3 version, I tested MAXLATE to filter the *KI device of my American version of TRSDOS 6.1 and this seemed to work. Although I presume it's not the best solution, I intend to buy LS-DOS 6.3 to try this out. But there seems to be some hope for Belgium. Recently Tandy Belgium has announced an international version of LS-DOS 6.3 (for the French and German keyboards) will be released in October 1987.

Is it normal that, when a part of the screen is in reverse video, and you call the CHARSET application, the characters from 128d to the end are also displayed in reverse video! At that moment, the Toggle option seems to work neither. I also suspect that DOLOAD and DOSAVE doesn't work with reverse video.

Fm MISOSYS, Inc: What an interesting observation. Yes, all keyboards can generate "extended" characters, those with bit-7 set, by using the <CLEAR> key in combination with another key. That's a feature of the DOS6 keyboard driver. Since CARDX accepts any non-control character for editing, once you insert an extended character, you can't delete it. Actually, you could eliminate it in a round about way. Go to another card record which is "blank" and go into EDIT mode; invoke CARDX again and position to the record with the "protected" character(s); then export the card contents to the "blank" card. Only the unprotected characters will be exported.

Now it is certainly normal that CHARSET will display its bottom half window in reverse video of the top half if reverse video is in effect. That's because the Model 4 is not capable of displaying both reverse video and extended characters. Actually, this can be fixed up by a patch to CHARSET so that it always

disengages reverse video, if it is ON. Here's a patch to do that.

```
. CHARSET1/FIX - to WAM1/APL of PRO-WAM 2.x
. keeps reverse video OFF while in CHARSET
. Apply via, PATCH WAM1/APL CHARSET1
D41,20=CD 04 2D
F41,20=21 E1 29
D46,04=0E 1C 3E 02 EF 21 E1 29 C9
F46,04=00 00 00 00 00 00 00 00
. Eop
```

Reverse video is turned OFF by sending a HOME character value (28d) to *DO via the @DSP service call. That's what the above patch does.

Now both DOLOAD and DOSAVE work by copying the exact screen contents; thus, they are impervious to reverse video - once you understand the theory behind reverse video. A program used to process DOSAVED files would need to ascertain the meaning of character values greater than 127. They would be reverse video of their corresponding ASCII character if reverse video was in effect; or extended characters if not. We did not put any intelligence into those two utilities; perhaps we should have.

XLR8er, PRO-WAM, and low memory squeeze

Fm Walter J. Gabriel: I wrote H.I.Tech about their board for my Model 4P. Bob of H.I.Tech phoned me that it was not compatible with PRO-WAM. It's been some time since TMQ wrote about the H.I.Tech board. Do you anticipate that it can be made compatible with PRO-WAM?

Fm MISOSYS, Inc: H.I.Tech misinformed you if they stated that their board was incompatible with PRO-WAM. What they should have stated is that the programming of the interface code was so bloated that it required all of available lower memory and thus denied any other program the use of low memory; PRO-WAM needs about 45 bytes - H.I.Tech takes about 512 for their two interface modules. So it has nothing to do with the board, per se. I, in fact, use a XLR8 board in my 4P. Of course, I can get away without having to use the slowdown filter since my 4P model is fast enough. One of these days, someone may find the time to rewrite H.I.Tech's interface modules to shorten their low-memory RAM requirements.

PRUN: Invoking TED from it

Fm John Tollini: I'm still trying to isolate the conflict, but there seems to be a conflict when I try to invoke ted using prun. (ted is in wam0/ap1). Result is system lockup. Also, back a while there was some discussion about fixall/flt (required or not). How can you tell if it is?

Fm MISOSYS, Inc: I personally have never tried running TED via PRUN since PRO-WAM release 2. On the other hand, since you can run TED/CMD from DOS ready (which is where you are at to run PRUN), why not use that for the time being?

I took a quick peek at invoking TED from PRUN. It only took a quick look. The problem is that PRUN sets the stack pointer (SP) to be 256 bytes past its end. This is OK for every PRO-WAM application except TED/APP. That bugger is quite clever.

I suppose not too many folks ever thought about how I was able to get a 32K text buffer from a PRO-WAM application. Well what TED/APP does is swap bank-0 32K of memory starting from 3000H to a spare bank unless no spare exists or HIGH\$ is less than 3000H+32K. It swaps so it can recover it later, if necessary. Now with the SP somewhere at 33??H, you can see what happens. Well I scratched out some code to work up a patch to have PRUN first check if HIGH\$ can be changed. If so, it puts the stack up there and lowers HIGH\$. Otherwise, it will be forced to use its own stack (HIGH\$ should not be frozen at that time). After PRUN regains control when your APP completes, it releases its HIGH\$ allocation. Check out The Patch Corner for this little adjustment.

Altering PRO-WAM window borders

Fm Michael Rogers: PRO-WAM 2 (and probably 1) users may be interested in making the following patch which will give the windows solid borders made up from standard graphics characters. With this modification, windows stand out from background programs and they have a more solid look. However like most jiggery-pokery which attempts to improve on a program written by a vastly superior being there is a trade off. If reverse graphics are enabled, by CARDX for instance, the wonderful new borders become less attractive as the graphic characters become reverse ASCII and non-ASCII characters. You may feel though that this trade

off is worth it overall, I do. The patch to PROWAM/CMD is as follows:

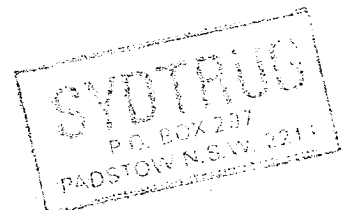
```
. PRO-WAM solid border for windows patch
D00,C4=97 AB 83 95 AA 20 B5 BA B0
F00,C4=2B 2B 2D 7C 7C 20 2B 2B 2D
. Eop
```

Now those wanting to give a more pleasing result when using programs (such as PFS-file) that enable reverse graphics, may wish to experiment with the following. Since I find PRO-WAM an invaluable utility when using PFS-file, I have a separate version of PRO-WAM that I have renamed PROWAMRV that uses solid window borders that are acceptable when reverse graphics are enabled. The patch to PROWAM/CMD is as follows:

```
. PRO-WAM reverse graphics solid border
D00,C4=DF DF DF 82 82 20 DF DF DF
F00,C4=2B 2B 2D 7C 7C 20 2B 2B 2D
. Eop
```

Note: This patch is for the original, not my first patched version. Also remember that this is intended only for programs that enable reverse graphics, otherwise results will not be as aesthetic.

Any Model 4 user reading this who does not own PRO-WAM is missing out on an invaluable utility/application which is both a great time saver and a program which could be used to revitalize interest in your Model 4, should it be flagging. All users should appreciate its facilities and the imaginative will find a myriad of applications. I have recently used its brilliant export-import facility to save retyping a database I wished to transport from one program to another. Besides reducing typing, it was fun - giving that satisfying, 'my computer is saving me time' feeling.



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

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.

If you use the FIXES?/TXT file from the DISK NOTES corresponding to this issue, please separate out the individual fixes which you need by use of any text editor you find convenient to use.

```

/*=====
datefix/jcl
. DATEFIX/JCL - Procedure to alter Model I LDOS 5.1.4
. Deactivates directory dating and extends DATE$ dating to 1999
. Invoke via: DO DATEFIX (DRIVE=d)
.   where "d" is the drive which contains the disk to patch
//if -drive
//. must enter drive=d parameter
//quit
//end
patch sys3/sys.system:#drive# s3
patch sys0/sys.system:#drive# s0
patch sys6/sys.system:#drive# s6
patch sys7/sys.system:#drive# s7
patch backup.rrw3:#drive# bu
. Disk alteration complete...
//exit
/*=====
bu/fix
. BU/FIX - Patch to Model I LDOS 5.1.4 BACKUP - 10/12/87
. at 59AA
D04,8B=18; WAS 28
. at 60C8
D08,28=64; WAS 44
. at 60D0
D08,30=64; WAS 44
. at 66DE
D10,7B=18; WAS 28
. Eop
/*=====
s0/fix
.S0/FIX - Patch to Model I LDOS 5.1.4 SYS0/SYS - 10/12/87
. Extends DATE$ dating to a range 1980-1999
. at 4EB5
DOD,6F=A8; WAS A7
. at 4EC3
DOD,7D=14; WAS 08
. at 4EE4
DOD,9E=32 07 43 3D BE 30 C6 78 32 66 44 00
. WAS 3D BE 30 C9 1A 07 07 07 B0 32 07 43
. at 4EFD
DOD,BB=3A 66 44 F5 E6 03 CD C6 50
. WAS CD C6 50 F5 0F 0F 0F E6 1F
. at 4F23
DOD,E1=00 00; WAS E6 07
. at 4F31
DOD,EF=07; WAS 03
. at 4F88
DOE,46=3A 44 40 CD D2 50 67 22 A5 50 21 A1 50 CD 67 44
. WAS 21 A1 50 CD 67 44 3A 07 43 E6 07 C6 30 CD 33 00
. at 50A5
DOF,B6=78 78 03 1E 44 61 74 65
. WAS 38 03 1E 44 61 74 65 20
. at 50C6
DOF,D7=21 BB 50 36 1C 20 01 34 3A 07 43 C9
. WAS 3A 07 43 4F E6 03 21 BB 50 36 1C 20
. at 50D2
DOF,E3=D6 20 2E 38 FE 3A D8 2C D6 0A C9
. WAS 01 34 79 C9 00 00 00 00 00 00 00
. Eop

```

```

/*=====
s3/fix
. S3/FIX - Patch to Model I LDOS 5.1.4 SYS3/SYS - 10/12/87
. Patch deactivates updating of MOD date on file close
. at 4E5F
D00,9F=18; WAS 28
. at 4E67
D00,A7=18; WAS 28
. Eop
/*=====
s6/fix
. S6/FIX - Patch to Model I LDOS 5.1.4 SYS6/SYS - 10/12/87
. correct DIR command
. at 5266
D03,4E=18; WAS 28
. at 55B7
D06,AB=18; WAS 28
. at 59B0
D0A,B4=20 20 20 20 20 20 20; WAS 4D 6F 64 20 44 61 74 65
. at 5948
D0A,4C=64; WAS 44
. at 5950
D0A,54=64; WAS 44
. Eop
/*=====
s7/fix
. S7/FIX - Patch to Model I LDOS 5.1.4 SYS7/SYS - 10/12/87
. Correct DATE command
. at 521A
D01,6E=14; WAS 08
. at 5250
D01,A4=32 07 43 78 32 66 44; WAS 07 07 07 B0 32 07 43
. at 52C9
D02,1E=07; WAS 03
. at 531C
D02,71=D6 20 2E 38 CD D4 53; WAS D6 50 C6 30 32 85 53
. at 53CD
D03,26=4E 6F 44 61 74 65 0D; WAS 44 61 74 65 20 6E 6F
. at 53D4
D03,2D=FE 3A 38 03 2C D6 0A 67 22 84 53 C9
. WAS 74 20 69 6E 20 73 79 73 74 65 6D 0D
. Correct PURGE command
. at 525B
D21,24=18; WAS 28
. at 53C2
D22,8F=18; WAS 28
. at 5701
D25,BB=64; WAS 44
. at 5709
D25,C3=64; WAS 44
. EOP

```

```

/*%=====
helpp1/fix
. HELPP1/FIX - Patch for PRO-WAM's HELPP facility - 11/04/87
. Corrects HELPP(EXPORT) display & references to PRO-NT0
. Apply via, PATCH HELPP HELPP1
D31,81=2D
F31,81=29
D31,AB=20 20 20 20
F31,AB=01 02 11 2C
D32,5E="WAM"
F32,5E="NT0"
D32,A8="WAM"
F32,A8="NT0"
. Eop
/*%=====
mlk512/fix
. MLK512/FIX - Patch to MRAS MLINK/CMD - 10/27/87
. Corrects search for /IRL libraries broke by MLK56/FIX
. Apply via, PATCH MLINK MLK512
DOA,OC=00 53
FOA,OC=30 55
. Eop
/*%=====
mlk612/fix
. MLK612/FIX - Patch to PRO-MRAS MLINK/CMD - 10/27/87
. Corrects search for /IRL libraries broke by MLK66/FIX
. Apply via, PATCH MLINK MLK612
DOA,18=00 27
FOA,18=20 29
. Eop
/*%=====
prun1/fix
. PRUN1/FIX - Patch to PRO-WAM's PRUN/CMD
. Allows PRUN of TED/APP
. Apply via, PATCH PRUN PRUN1
D00,41=85;F00,41=0B
D00,43=C3 46 31;F00,43=31 AE 33
D01,4D=C3 70 31;F01,4D=31 00 00
D01,89=31 AE 33 22 6B 31 21 00 00 45 3E 64 EF 54 5D 25
F01,89=00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D01,99=3E 64 EF 22 78 31 20 0C 23 36 18 23 36 FE 23 73
F01,99=00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D01,A9=23 72 EB F9 21 00 00 C3 07 30 21 00 00 45 3E 64
F01,A9=00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D01,B9=EF 11 00 00 B7 ED 52 20 05 EB 24 3E 64 EF 31 00 00 C9
F01,B9=00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
D02,1A=62;F02,1A=61
. Eop
/*%=====
rsh52/fix
. RSH52/FIX - Patch to RSHARD's RESTORE5 - 10/20/87
. Corrects acceptance of parameters
. Apply via, PATCH RESTORE5 RSH52
D00,90=C3 20 5B
F00,90=32 EA 57
D07,80=32 EA 57 E1 E5 C3 4F 54
F07,80=00 00 00 00 00 00 00 00
. Eop

```

```

/*%=====
rsh62/fix
. RSH62/FIX - Patch to RSHARD's RESTORE6 - 10/20/87
. Corrects acceptance of parameters
. Apply via, PATCH RESTORE6 RSH62
D00,7B=C3 92 38
F00,7B=32 C2 35
D06,EE=32 C2 35 E1 E5 C3 3A 32
F06,EE=00 00 00 00 00 00 00
. Eop
/*%
sys7h/fix
. SYS7H/FIX - Patch to LDOS 5.3 PURGE
. Apply via, PATCH SYS7/SYS.SYSTEM SYS7H
L72
X'54DC'=C3 92 57
X'5792'=CD 97 57 AF C9 3E E4 EF
. Eop
/*%
todo2/fix
. TODO2/FIX - Patch to PRO-WAM WAM1/APL TODO application - 11/05/87
. Allows TODO/DAT file on any drive
. Apply via, PATCH WAM1/APL TODO2
D5C,56=08
F5C,56=0B
. Eop

```



<h1 style="margin: 0;">MISOSYS</h1>	<h2 style="margin: 0;">SOFTWARE TO STRETCH YOUR TRS MODEL 4</h2>	
<p style="text-align: center;">PRO-WAM™ Release 2 Window and Application Manager</p> <p>Our applications turn your 128K Model 4 into a sophisticated business or personal machine rivaling the best of them. Because easily installed PRO-WAM comes with many useful and powerful menu-driven time savers and work organizers. PRO-WAM is accessed with a single keystroke; its EXPORT/IMPORT functions allow you to move data across windows between programs.</p> <ul style="list-style-type: none"> • Address CARDS, LABELS, and new HEADINGS for display and export • Improved BRINGUP tickler file; new PRINTING and sorting • Improved CALENDAR flags BRINGUP items visually on screen • Ten 3 x 5 CARD files with FORMS and FIELDS using reverse video • New virtual PHRASE access for export • New TODO list manager with "who does it" <p>PRO-WAM [M-51-025] \$74.95 + \$5 S&H</p> <p>Upgrade your old PRO-NT0/PRO-WAM to release 2; return your master disk with \$24.95 + \$5 S&H</p>	<p>MISOSYS has been supplying the TRS-80 community with professional quality software since 1978; that's over eight years of experience captured in a host of other software products ranging from language compilers and assemblers, fine crafted utilities, other applications software, and operating systems. We also publish a magazine, THE MISOSYS QUARTERLY, which is available on a subscription basis. Call or write us for a complete catalog of our product line.</p>	<p style="text-align: center;">LB —A flexible data manager</p> <p>Easily used by anyone for managing their data. It's menu driven for ease of use; absolutely no programming needed. Requires a Model 4 with 128K or a hard drive. LB86,™ an MS-DOS version is also available.</p> <ul style="list-style-type: none"> • Store up to 65534 records per data base • Up to 1024 characters per record • Up to 64 fields per record • Nine field types for flexibility • Select and sort on up to 8 fields • Keep 5 different indexes for data access • 10 input/update screens per data base • 10 printout formats per data base • Extensive on-line help always available <p>LB [L-50-510] \$74.95 + \$5 S&H</p>
<p>MISOSYS Inc. P.O. Box 239 Sterling, VA 22170-0239 703-450-4181</p> <p><small>VA RESIDENTS ADD SALES TAX. S&H: CANADA ADD \$1; FOREIGN USE \$4H '3</small></p>	<div style="border: 1px solid black; padding: 5px; width: fit-content; margin: 0 auto;"> <p style="margin: 0;">ORDER NOW 800-MISOSYS 800-647-6797</p> </div>	



Your Model 4 computer
may not speak XZ#M%S
But with MISOSYS
language products, she will
speak ASSEMBLER, BASIC,
C, 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 + \$4 S&H

PRO-DUCE — A 2-pass labeling Z80 disassembler from disk or memory with screening input for data areas. Generates /ASM files. \$34.95 + \$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. \$89.95 + \$4 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. \$49.95 + \$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 + \$4 S&H

LS-TBA — A structured BASIC translator. Labeled statements. Conditional translation, pseudo global and local variables, 14-char var names \$39.95 + \$4 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

RATFOR-M4 — A professional implementation of RAtional FORtran. Provides structure and greater portability to FORTRAN programs. Fully documented with tutorial user manual. Requires FORTRAN compiler. \$99.95 + \$5 S&H

Note: Model 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 sales tax. S&H: US \$2, Canada \$3, Foreign \$6.

MISOSYS

SOFTWARE FOR YOUR P.C.

Order Now 800-MISOSYS (647-6797)

ED/ASM-86™

- 80X86 Assembly Language development

An integrated editor, assembler, linker, and debugger for 8086, 80186, 80286, and 8087 assembly and disassembly. All features are in a single EXE file.

- a full-featured line EDITOR with intra-line editing, block move and copy, partial save and load.
- a full-screen turbo-type editor is supported on CGA-equipped systems.
- the ASSEMBLER supports structured coding; can directly generate a COM or EXE file; assembly to memory for faster debugging.
- The DEBUGGER includes symbolic disassembly of arbitrary code to disk.

ED/ASM-86 [M-86-030] . . \$124.95 + \$5 S&H

RATFOR-86™

- Rational FORTRAN translator

A professional implementation of RAtional FORtran by James Beard, Ph.D. Provides structure and greater portability to FORTRAN programs to reduce your programming time and effort dramatically over that required when writing directly in FORTRAN. Includes tutorial reference manual.

- compiler has extensive macro support
- "arith" macro for binary arithmetic
- short form READ and PRINT
- enhanced switch case support
- Requires a FORTRAN compiler

RATFOR-86 [M-86-73] . . . \$99.95 + \$5 S&H



MISOSYS, INC.
PO Box 239
Sterling, VA 22170-0239
703-450-4181 MC, VISA, CHOICE

VA residents add 4 1/2% sales tax.
S&H: Canada add \$1;
Foreign use S&H times 3