THE MISOSYS QUARTERLY

In this issue:

- pfsFILE data file structure revealed,
  by Roy Soltoff
- A PRO-WAM Help Displayer,
  by Matthew Kent Reed
- LB Data Manager, A Review,
  by Ken Strickler
- SYSFLEX: The Flexible /SYS file Loader,
  by Matthew Kent Reed
- LB Data Manager Version 2.2 released

A smorgasbord of programs is available for your order

Volume Vi.ii **$10** **Winter 1991/92**

PRICE LIST effective February 1, 1992

TRS-80 Software

Product Nomenclature	Mod III	Mod 4	Price S&H
AFM: Auto File Manager data base	P-50-310	n/a	$49.95 D
BackRest for hard drives	P-12-244	P-12-244	$34.95
BASIC/S Compiler System	P-20-010	n/a	$29.95 B
BSORT / BSORT4	L-32-200	L-32-210	$14.95
CP/M (MM) Hard Disk Drivers		**H-MM-???**	**$29.95 B**
CON80Z / PRO-CON80Z.	M-30-033	M-31-033	$19.95
diskDISK / LS-diskDISK	L-35-211	L-35-212	$29.95
DISK NOTES from TMQ (per issue)			$10.00
DoubleDuty		M-02-231	$49.95
DSM51 / DSM4	L-35-204	L-35-205	$49.95
DSMBLR / PRO-DUCE	M-30-053	M-31-053	$29.95
EDAS / PRO-CREATE	M-20-082	M-21-082	$44.95 D
EnhComp / PRO-EnhComp	M-20-072	M-21-072	$59.95 D
Filters: Combined I & II	L-32-053	n/a	$19.95 B
GO:Maintenance	n/a	M-33-100	$39.95 B
GO:System Enhancement	n/a	M-33-200	$39.95 B
GO:Utility	n/a	M-33-300	$39.95 B
Hardware Interface Kit	n/a	M-12-110	$24.95
HartFORTH/PRO-HartFORTH	M-20-071	M-21-071	$39.95 B
LDOS 5.1.4 User Manual	L-40-020	n/a	$15.00 D
LDOS 5.3.1 Mod1 Upgrade kit	M-10-133	n/a	$39.95 B
LDOS 5.3.1 Diskette - M3	M-10-130		$15.00
LDOS 5.3.1 Mod3 Upgrade Kit	M-10-333	same	$39.95 B
LED / LS-LED	L-30-020	L-30-021	$19.95
LB Data Manager-M4 (Ver 2.2)	n/a	M-50-510	$99.00 D
LS-DOS 6.3.1 Upgrade Kit - M4	n/a	M-11-043	$39.95 B
LS-DOS 6.3.1 Diskette - M4	n/a	M-11-243	$15.00
LS-DOS 6.3.1 Upgrade kit - M2/12/16		M-11-002	$39.95 B
LS-Host/Term	n/a	L-35-281	$39.95
LS-UTILITY	n/a	L-32-150	$24.95
MC / PRO-MC	M-20-064	M-21-064	$79.95 D
Mister ED	n/a	M-51-028	$39.95 B
MRAS / PRO-MRAS	M-20-083	M-21-083	$59.95 D
PowerDot (Epson or Tandy)	P-32-21?	n/a	$19.95
PowerDraw	P-32-220	n/a	$19.95
PowerDriver Plus (Epson).	P-50-200	P-50-200	$17.95
PowerMail Plus	P-50-003	P-50-004	$39.95 D
PowerMail Plus TextMerge	P-50-100	P-50-100	$15.00
PowerScript	P-50-142	P-50-142	$24.95 B
PRO-WAM	n/a	M-51-025	$74.95 D
PRO-WAM Toolkit	n/a	M-51-225	$29.95 B
Programmer's Guide DOS 6.	n/a	M-60-060	$20.00 B
QuizMaster	L-51-500	n/a	$19.95
RATFOR-M4		M-21-073	$59.95 D
RSHARD - R/S HD driver	M-12-013	same	$29.95
ST80-III	P-35-300	n/a	$39.95 B
SuperUtilityPlus	P-32-132	P-32-104	$44.95 D
SuperUtilityPlus CMD file diskette	P-32-832	P-32-804	$20.00
Supreme HD Driver (PowerSoft-RS)	P-12-113	P-12-113	$34.95
TBA / LS-TBA	L-21-010	L-21-011	$19.95 D
THE SOURCE 3-Volume Set	n/a	L-60-020	$40.00 D
Toolbox/Toolbelt	P-32-203	P-32-245	$24.95 B
UNREL-T80	same	M-30-054	$29.95
UTILITY-I	L-32-070	n/a	$19.95
XLR8er Interface Kit (M3 mode)		M-12-X10	$20.00 B

MSDOS Software

LB Data Manager 2.2		M-86-510	$99.00 D
DED-86 [Disk/Memory sector editor]		M-86-020	$29.95 D
RATFOR-86		M-86-073	$59.95 D
HartFORTH-86		M-86-071	$59.95 D
SAID-86 [Text Editor]		M-86-040	$29.95
FM-86 (File Manager)		L-86-050	$29.95

TRS-80 Game Programs

Bouncezoids (M3)	M-55-GCB	$14.95
Crazy Painter (M3)	M-55-GCP	$14.95
Frogger (M3)	M-55-GCF	$14.95
Kim Watt's Hits (M3)	P-55-GKW	$9.95
Lair of the Dragon (M3/M4)	M-55-021	$19.95
Lance Miklus' Hits (M3)	P-55-GLM	$19.95
Leo Cristopherson's (M3)	P-55-GLC	$14.95
Scarfman (M3)	M-55-GCS	$14.95
Space Castle (M3)	M-55-GCC	$14.95
The Gobbling Box (M3/M4)	M-55-020	$19.95

MSDOS Game Programs

Lair of the Dragon	M-86-021	$19.95

Hardware

Floppy Disk Controller M3/M4	H-MM-FDC	$45.00	F
Double Density Controller (DDC) M1	H-MM-DDC	$45.00	F
RS232 Serial Card M3/M4	H-MM-SPC	$45.00	F
RS232 Serial Card Kit M3/M4	H-MM-SPK	$50.00	F
TeleTrends TT512P modem (M4P)	H-4P-512	$74.95	E
Floppy drives (5.25" 360K 1/2 ht)	**H-FD-360**	**$75.00**	**D**
Floppy drives (3.5" 720K 1/2 ht)	**H-FD-720**	**$85.00**	**B**
Floppy Drive Case (2-1/2 ht drives)	H-FD-2SV	$60.00	F
MSCSI HD kit e/w clock, 20Meg M3/M4	**H-HD-020**	**$450.00**	**?**
MSCSI HD kit e/w clock, 40Meg M3/M4	**H-HD-040**	**$575.00**	**?**
Aerocomp HD - 5 Meg M3/M4	**H-MM-005**	**$250.00**	**?**
Aerocomp HD - 20 Meg M3/M4	**H-MM-020**	**$400.00**	**?**
Aerocomp HD - 40 Meg M3/M4	**H-MM-040**	**$500.00**	**?**
Hard drive joystick port option	H-HD-JSO	$20.00	
Hard drive: Kalok KL320	R-HD-020	$200.00	F
Hard drive: Seagate ST251-1	R-HD-040	$320.00	F
Hard drive: Seagate ST-157N (SCSI)	R-HD-S40	$350.00	F
Cable: dual floppy extender	H-FD-2EX	$18.00	
Cable: 4Ft floppy (1 34EDC each end)	H-FD-C04	$12.50	
Cable: 4Ft M3/M4 printer	H-RC-PM4	$20.00	
Cable: 4Ft Radio Shack hard drive	H-HD-CT4	$20.00	
Cable: 4Ft MISOSYS hard drive	H-HD-C04	$22.50	
Cable: 26-1069 internal floppy	H-FD-2NG	$20.00	
Cable: 26-1069A/26-1080 internal floppy	H-FD-2GA	$20.00	
Cable: 26-1080/A internal floppy	H-FD-24P	$20.00	
Cable: drive power Y	H-HD-CPY	$5.00	
Cable: XT hard drive set	H-HD-CXT	$5.00	
Cable: Custom IDC ribbon (M3/M4/M2)	?-??-???	varies	
Standby Power System: 200VA	**R-PS-200**	**$199.00**	**?**
Standby Power System: 450VA	**R-PS-450**	**$399.00**	**?**
HD Controller: Adaptec 4010A	H-HD-CA4	$75.00	D
HD Controller: Xebec S1421A	H-HD-CX2	$75.00	D
T80 to SCSI host adaptor	H-HD-MHA	$75.00	D
ZOFAX 96/24 **Fax/Modem** (PC XT/AT)	R-Z1-FAX	$125.00	G
Infochip Systems Expanz! (PC)	R-IC-EXP	$150.00	G
DJ10 Tape Backup (PC)	R-TD-D10	$199.00	G
DJ20 Tape Backup (PC)	R-TD-D20	$265.00	G
AB10 Tape Adaptor (PC)	R-TD-A10	$75.00	
KE10 External tape adaptor/case (PC)	R-TD-K10	$110.00	D

The Fine Print

Freight codes: A = $3.50; B = $4.00; C = $4.50; D = $5.00; E = $5.50; F = $6.00; G = $7.00; H = $12.00; ? = varies; All unmarked are $3.00 each; Canada/Mexico add $1 per order; Foreign use US rates times 3 for air shipment. Virginia residents add 4.5% sales tax. We accept MasterCard and VISA; Checks must be drawn on a US bank. COD's are cash, money order, or certified check; add $4 for COD.

MISOSYS, Inc.
P.O. Box 239
Sterling, VA 22170-0239
703-450-4181; Orders only: 800-MISOSYS (800-647-6797)

The MISOSYS Quarterly is a publication of MISOSYS, Inc., PO Box 239, Sterling, VA 22170-0239, 703-450-4181.

THE MISOSYS QUARTERLY

subscription rate information

Each issue of TMQ has information on MISOSYS products, programs and utilities, patches, significant messages from our CompuServe forum, and articles on programming. Not only that, TMQ will keep you up to date with information, news, and announcements concerning our entire product line and related machine environments. Subscription cost varies by rate zone as follows:

A = $25; United States via 3rd class bulk mail
B = $30; Canada, Mexico, via 1st Class
C = $32; Colombia, Venezuela, Central America via AO Air
D = $35; South America, Europe, & North Africa via AO Air
E = $40; Asia, Australia, Africa, Middle East via AO Air

TMQ Toolbox

The MISOSYS Quarterly is published using the following facilities:

The hardware used to produce the "camera ready" copy consists of an AST Premium/386 computer (20 MHz) with 9 Megabytes of RAM, a Seagate ST4096 80M HD, ST251 40M, Expanz! card; a CMS DJ10 tape backup, a NEC Multisync II monitor driven by a Video Seven VGA card, an AST TurboScan scanner (Microtek MS300), and a NEC LC-890 PostScript laser printer.

Text is developed, edited, spell-checked, and draft formatted using Microsoft WINWORD Version 1.1; Submissions on paper and letters are scanned and converted to text using ReadRight optical character recognition software by OCR Systems. Final page composition is developed using PageMaker 4.0 by Aldus.

Table of Contents

List of Advertisors

List of Patches in this Issue

Points to Ponder

Joint ventures seem to be the latest offerings in the computer world. Perhaps the biggest may be pairing of IBM and Apple to develop an operating system for the future. Gee, I thought that was supposed to be OS/2!

Tandy also has their joint venture; they have paired up with Matsushita Electric Industrial to form a new company called PTCC Inc., 50% owned by each. PTCC will be in the business of making laptop and notebook computers with production in Fort Worth. Production capability levels are expected to approach 100,000 units of 32-bit and 16-bit computers per month.

Tandy is also now carrying LANtastic - the popular peer-to-peer network. LANtastic's the reason that big Red, Novell, brought out their low-end network.

Remember the size of the TRS-80 Model I floppy disk drive? Fujitsu's latest in the small hard drive class is a 2.5" 2-platter unit with 90 megabytes of capacity. JVC also has an 86 megabyte model just 19mm high and weighing in at 180 grams. Areal Technologies of San Jose drove right past the imports with its 181 megabyte 2.5" drive which is 3.45" long and 0.59" tall; it's platters are made of glass!

Guess who's now big in software? Mickey Mouse, that's who. Disney's Computer Software Division recently released a major breakthrough in a digital sound reproduction peripheral for PCs; it's called "The Sound Source". Jointly developed with Electronic Speech Systems and Phoenix Technologies, it produces AM-radio quality - at a very small price compared to other speech add-ons.

The Blurb by Roy Soltoff

TMQ Schedule

I try to **target** mailing *THE MISOSYS QUARTERLY* approximately every three months. After dragging my feet with the last issue because I had too few subscribers to warrant publishing in August, I ran up against the winter holidays. That's why this issue appears late.

Note that your mailing label usually has the expiration date of your subscription. For instance, **those with "92/05" complete their subscription with this issue.** The renewal fee to continue with the next two issues is 50% of the subscription rate shown on page 1.

TMQ advertising

If you are interested in reaching a **dedicated** TRS-80 audience, consider *THE MISOSYS QUARTERLY*. If you have a TRS-80 Model III or 4 related product to sell, you can reach these buyers by placing your advertisement in our publication. Current space rates are as follows (reduced from previous rates):

Full page	$75
Half page	$40
Quarter page	$25
Ninth page	$15

I compose the ninth-page ad layout so you have no artwork charge. Just submit your text. Ads for our inside covers are printed in the same color as the cover (TMQ alternates between PMS colors: green 354, purple 266, blue 293, and red 199). If you would like to place your ad in *THE MISOSYS QUARTERLY*, send it in. You have two issues left.

Corrections to last issue

A few errors popped into TMQ 6.1. Note that the first code line of SIMIL on page 25 should read:

```
SIMIL: $GA DE,BC
```

and not DE,HL

The DISK NOTES was correct as I caught the error early. I made some changes to the code after I shifted the file to TMQ and hand edited the magazine copy after testing the code copy. I overlooked editing that line.

Note also that the blank half page on page 47 was supposed to contain an advertisement for TRSTimes. I apologize for that; things have been a little hectic here.

Speaking of TRSTimes, if you are looking for an excellent eye-pleasing bi-monthly magazine which covers the TRS-80 Model I, III, and 4 environment and which has not reverted to half catalog, you ought to give TRSTimes a tryout. Check out their ad in this issue - honest, it's in here!

PD Software Librarian

Vic McClung has volunteered to be the librarian for the collection of TRS-80 public domain diskettes. Henceforth all requests and contributions be directed directly to him at:

Vic McClung
914 Crescent
Sikeston, MO 63801
USA

Note that if you upload a "public domain" file to our CompuServe forum [PCS-49], and want it to receive general distribution, please also mail a copy on disk to Vic. There is no legal provision for downloading files from Compuserve and re-distributing them, unless you were the uploader. Some of our readers who do not have access to our forum have an interest in those submissions. So if you want to help out the most numbers of fellow users, don't limit your submissions to just one source.

MISOSYS Forum

MISOSYS sponsors a forum on CompuServe. You can reach some "experts" on TRS-80 and MS-DOS subjects by dialing in. The forum is reached via GO PCS49, or GO LDOS. If you have any questions concerning access, get on and leave a message to SYSOP. Joe Kyle-DiPietropaolo will get to you. Please don't call me here at MISOSYS because I cannot answer any questions as to its operation.

The forum contains a great deal of programs which you can download, as well as enter into the lively discussions which thread through the message system. If you do programming on a PC, the forum also contains the listings from *Programmer's Journal*. If you want to direct a message to me, my user ID is 70140,310. Post a message in private if you don't want it "broadcast"; some folks even send me orders via a PRIVATE message.

DISK NOTES 6.2

Each issue of *THE MISOSYS QUARTERLY* contains program listings, patch listings, and other references to files we have placed onto a disk. DISK NOTES 6.2 corresponds to this issue of TMQ. If you want to obtain the patches and the listings, you may conveniently purchase a copy DISK NOTES priced at $10 **Plus S&H**. The S&H charges are $2 for US, Canada, and Mexico, $3 elsewhere.

Older TMQ's available

Volume I and Volume II are out of print issues; they are no longer available. The price for back issues is $4 + S&H (minimum order of $15). S&H for a single issue is $2.75 in the U.S. and CANADA; $5.50 zone D; zone E is $6.50. S&H for four issues is $5 (US), $6 (CAN), $14 (ZoneD), $20 (ZoneE). Here's a synopsis of past issues:

IV.i Cataloging files with a word processor; Page display PRO-WAM application; File undating with FUNDATE; Array load routine for BASIC; XLR8er and the GT-180 graphics board.

IV.ii Printing from BASIC without cutting words; LOAD100 for Model 100; Generating date/time stamp; Favorite recipes; Some BASIC routines.

IV.iii Fast in-memory sort using XLR8er RAM; Using XLR8er RAM as graphics video RAM; Upgrade your 4P with external floppy drives; Doubling of files solved; SuperScripsit document file format' FELSWOOP PRO-WAM export utility.

IV.iv Five Twelve K: A better way; Multi-Command; Touch/ccc; Fixes for LS-DOS 6.3.1; DoubleDuty Version 2.6.0 released.

V.i 300 Dots on the TRS-80; Tandy 16/6000 Hard Disk Drives; NXWAM PRO-WAM application; A review of M.A.D.'s XROM; a MIDI interface for your TRS-80.

V.ii Image processing on the TRS-80 Model 4; A MAKE utility for MC; New XLR8er patches for LS-DOS 6.3.1; FORTH: A language for every application.

V.iii It's rude not to interrupt; A Model 4 mouse driver; Profile 4+ to filepro 16/dBASE III; and a complete map to *Lair of the Dragon*.

V.iv Fill low memory; Internal HD for 4P; Roll your own on the XLR8er, Profile 4" printer codes, Boot LDOS 5.3 from HD Model 4; Memory: How much and why; 300 Dots: An update; and Lair of the Dragon hint sheets.

VI.i The final solution to the XLR8er question; Pattern matching - including SIMIL.ASM; ELEMENTS, a PRO-WAM application; LDOS 5.3 XLR8er Interface Kit.

LB Data Manager 2.2.0 Released

I have completed an LB database conversion utility which will be bundled with LB release 2.2. The utility creates an LB database directly from any of the following: pfsFILE4, Profile4, DIF, dBASE II, dBASE III, and fixed fielded records using signed/unsigned integers, Microsoft format single and double precision, strings, and IEEE format single and double precision (for MS-DOS version). The utility also directly generates the following types of database files or output from an LB database: DIF, dBASE II, dBASE III, tab delimited, comma delimited, and ASCII strings. The LBCONV database conversion utility's menu structure and operation is similar to other LB commands, so you do not have to learn another interface.

In the conversion from the. dBASE II and dBASE III database structures to LB: the first 'D' field will be converted to '\'; others to 'L'; 'C' fields will be converted to 'L'; 'N' fields will be converted to 'N'; and 'L' fields will be converted to 'A'.

For conversion from Data Interchange Format (DIF), all fields are defined as type 'L'. To maximize the automatic conversion effort, LBCONV supports LABEL and SIZE titles for use in pre-determining LB's field sizes and names.

For the conversion from pfsFILE-4, all fields are defined as type 'L'. LBCONV will automatically scan the pfs data file to calculate the field size of every field; LB fields will then be fixed according to the maximum fields sizes found; field names will be extracted as the rightmost 19 characters of the screen description.

For the conversion from Profile-4, field names and lengths will be extracted from the MAP file. Field types will be initially set at '*' (which is used by Profile to indicate an alphanumeric field); then all screen files will be scanned for field designations. The first non-'*' found for a given field will establish that field's type. Types '<', '?', '/' are established as "must fill" Literal; type '!' established as 'E'-protected Literal; types ')', '(', '*' established as Literal; type '>' established as must fill 'N'; types '#', '+', '-' established as 'N'; type "'" as must fill 'D'; type '.' as 'D';and type '&' and '@' as '\'. An edit mode allows you to over-ride these default types as well as inhibit any Profile field from being converted to the LB database.

LBCONV will directly convert an LB database to a. DIF file; LABEL and SIZE titles will be generated. Field types F, D, R, and N will be generated as numeric values; all others as strings.

In the conversion to dBASE II and dBASE III: field type '\' is converted to D; types ABLU to C; and types FDRN to N. Field names will be truncated to 11 characters.

In the conversion to TAB or Comma delimited files, trailing spaces will be stripped from all fields. LBCONV also can convert to a file composed of ASCII strings, which are comma delimited with field data in quotes.

The LB 2.2 release provides a number of small improvements - and a few big ones - to the data manager; here's what has been changed:

1. The structure of the index file has been changed to store field numbers of all fields used in the attachment string. This requires all old index files to be re-generated before being used with this version, but it allows for additional features.

2. A **Find** command has been added to the sort/select module; this is used to find potential duplicate records. The functions requires an index file and will automatically use the fields by which the index file was sorted as the comparative strings.

3. A validity check has been added to Define File which requires a valid data path before allowing a continuation.

4. The Dup function in the LBMANAGE utility was changed to not continue if the index entry was "bad".

5. LB's field input editor now accepts a backslash character on alphabetic character fields.

6. The Find command now displays the second key name if an index is active which uses more than one field and the primary key is A, B, L, or U. With this enhancement, Find can use key1 and key2 of a sorted index file. Strings are entered as, "first\second". Note that the first or key1 string must be sufficiently unique in the ordered sequence of records to satisfy the ability to find the second key; and the string-entry requirements are based on the field type of the primary key.

7. The Print module's output has been converted to a line buffer. This allows a TAB control in print definition screens so you can force a tab to a column. The tab can be anywhere on the print line.

8. There's now a Print command module sub-function in the MSDOS version to print a CR-LF or just a CR. This is needed if an output file is to be used for auto-input.

9. LBMANAGE was changed to compare only field types and field lengths excluding field names and print flag.

10. The TRS-80 version was changed to use fputs(HOMECLR,stdout) in lieu of an internal syscls() to clear screen on exit (and reset inverse video).

11. Ensured that Select module has introduced use of CKBRKC for TRS-80 version in three submodules, and added CKBRKC for T80 version in print engine.

12. I added a check for an output device error for printing. If an error is detected, LB will now prompt for ESC/BRK or

RET. Also, the output device will be issued a cleareof() to reset an end-of-file condition. This is useful if you are directing output to a printer which goes off-line.

13. I changed Delete in the Edit module to zero the number of active records and the number of deletes if the last record used is deleted. Also the index file, if any in use, is detached.

14. I added a command in the Define screen sub-menu, to automatically generate an Edit/Update/Delete screen. Use "^G". It uses the following procedure: (a) if the length of any field name and its field length exceeds 78, auto gen is inhibited. (b) if the maximum field name length plus the maximum field length exceeds 78, the screen is generated on a fully left alignment, horizontally centered in the screen area. (c) Otherwise, the screen is column aligned so that all field entries start in the same column. The longest field name+length is centered. (d) Field names precede field entry; a colon-space is used as a separator. (e) A maximum of 21 fields can be displayed; one per screen line. If there are less than 20 fields, a centered title (the name of the database), is inserted. (f) If a calculated field is being handled, the normal screen prompts are generated for entry of calculation string and format. (g) Once complete, the user remains in screen define mode - use <F3> as normal to initiate escape to the define command menu so that the screen can be <S>aved - or continue to edit the screen definition.

15. A last minute request from one user resulted in an additional function in field editing. The <HOME> key under MS-DOS or the <SHIFT><F1> key under LS-DOS will move the cursor to the start of the field being edited; the mode will revert to overstrike.

If you are an existing LB 2.1 user and wish to move up to the new release 2.2, return your existing two disks with $10 for the update; if you purchased your LB 2.1 after December 31, 1991, the.update is free until March 31, 1992 Existing LB 1.0 users can still upgrade to version 2.2 by sending in the Table of Contents page from the LB User Manual with the $40 upgrade fee plus appropriate shipping.

I have started a new policy for trade-ins of equivalent software products for a MISOSYS software product. Previously, you had to return the complete package; that's no more. Just send in an original Table of Contents page with the trade-in fee which is 50% of the price of our product. So for LB 2.2, trade in any other database product and you can purchase LB for $49.50 plus S&H. How's that for a deal?

International LS-DOS 6.3.1

With my major bungling out of the way, I discovered that MISOSYS did indeed have the international keyboard driver for LS-DOS. Thus, I have been able to prepare the French and German versions of LS-DOS 6.3.1 (F & G). If you have purchased a 6.3 version in the past, you need to order only a replacement disk - but you must specify which version you want - F or G. The cost is $15 plus S&H. If you are starting with 6.2 or earlier, order the upgrade kit for $39.95 + S&H.

MRAS/PRO-MRAS

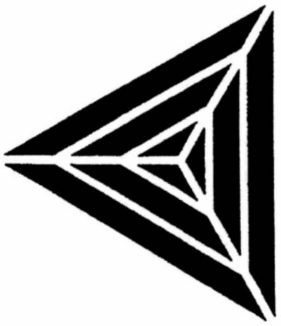
MRAS has been updated to directly handle the support of 64180/Z180 extended opcodes such as: MLT; IN0 g,(m); OUT0 (m),g; OTIM; OTIMR; OTDM; OTDMR; TSTIO m; TST g; TST m; and TST (HL); and SLP. In addition, Richard Van Houghton revised MLINK for us to support 3-byte pointers for the virtual memory file facility; thus, you should be able to link very large programs using many overlays as the new virtual file limit has been increased from 64K to 16 Megabytes. Want an update? Send back your existing MRAS disk with $10 plus $3 S&H or purchase a new MRAS disk for $15 + $3S&H. Please be honest!

Price Reductions

There have been two significant price reductions in MS-DOS products MISOSYS purchases for resale. First, Zoltrix has lowered the price of their Fax/Modem boards. Details on this plug-in board were presented in TMQ V.iii; it's a 9600 baud Group III send & receive fax plus a 2400 baud modem. With their new price structure, these boards can now be sold for $125. I still have five in stock which I bought for $175 each. My loss is your gain. **All Zoltrix ZOFAX 96/24 fax/modems will now be available for $125 + $7S&H.**

Colorado Memory Systems has also reduced the price of their tape backup drives for PCs. The DJ-10 (120M with data compression) was lowered from $399 to $250 retail; the DJ-20 (250M with data compression) was reduced from $499 to $350. These are not MISOSYS prices. Our price for the drives will be $199 for the DJ-10 and $265 for the DJ-20. These are prices for internal units - they need a 5-1/4 1/2-ht slot and a floppy connection. At these prices, we aren't including blank tapes with the price. Let me say that if you are running an MS-DOS machine with a hard drive and are still fussing with floppies for backup - or worse yet, nothing - you really ought to invest in a tape drive. Your data depends on it.

Letters to MISOSYS



Graphics Formats

Fm Robert W. Via: Since I've been messing with graphics files, I have learned there are a million formats. I know about GIF, HR, CHR, BLK but what are RLE files and how do you print or display them?

Fm Frank Slinkman [72411,650]: Robert, RLE = Run Length Encoded. The "original" RLE was a CIS invention — their first attempt at a hardware-independent graphics format. It had a resolution of 256 x 192, monochrome only. Lately, some 24-bit color graphics is being stored in a different kind of RLE.

There are a couple of CIS-RLE decoders for the TRS-80 line over in the Graphics Support Forum (GO PICS), including my own M4RLE/CMD. Look in lib:3 there, and bro trs*.

Essentially, pixels were grouped in black-white sets, and all codes were in the ASCII range, namely 20H to 7EH. If the first pixels in an image were 5 black followed by 5 white, then the first two bytes in the file would be 25H 25H, the count base 20H. If the first line of 256 pixels were all white, then the first bytes would be 20H 7EH 20H 7EH 20H 44H, meaning 0 black, 94 white, 0 black, 94 white, 0 black, 68 white.

The loose RLE format specification is in PICS lib:14 and there are a great many RLE images in QPICS lib:14 (as one in lib:0 here: ODIE/RLE, the last time I looked).

Model 1 Help

Fm William Buzbee: Hello! I'm looking for some friendly advice to make my life a little easier when transferring files between my trusty old Model 1 and my shiny new MacIIcx.

My Model 1 has a Percom doubler, 1.5X speedup mod, two DD/DS drives and is running LDOS 5.1.4. Besides a couple of hard drives, the Mac has a 3.5" drive and software capable of reading and writing MS-DOS disks. I also have Supercross/XT software for the Model 1 which allows it to read and write MS-DOS disks.

Now, my problem is that my Mac's 3.5" drive doesn't like getting 5.25" disks stuffed in its little slot. I can buy a 5.25" drive for the mac, but I kind of hate to spend the money just for file transfer.

My question is: Can I get a 3.5" drive for my Model 1? Also, does anyone know if Supercross/XT would recognize it if I did?

Fm Daniel L. Srebnick: Yes, to both questions. Roy of MISOSYS, telephone 800-MISOSYS can sell you the 3 1/2" drive for the model 1 [and a new LDOS 5.3.1 to boot]. And as far as Supercross goes, it has a multitude of selections for disk formats, one of which I believe will support the 720k 3.5" MS-DOS format. Don't have the manual here, but I know I have moved stuff from my model IV using 720k 5.25" disks. The form factor should make no difference.

Model I LDOS 5.3.1

Bill Buzbee, Santa Clara, CA: Roy, I never guessed how much money you'd cost me by bringing out LDOS 5.3.1 for the Model 1. Before you did that, my old beast had been gathering dust in the corner. Now I find myself hacking away at it again - with a shiny new 720K 3.5" system drive, MC, the SOURCE, etc. Not only that, but I've added two more Model 1's to my spare parts closet. I suspect I'll be able to keep the old thing going for at least another ten years.

Anyway, it seems my thirst for Model 1 paraphernalia hasn't died yet. The 3.5" drive has opened up a lot of space on my working system disk, so I'd like to get the /CMD file version of SuperUtility+. I assume you're following the old PowerSoft upgrade rules, so enclosed is one of my original disks along with a signed /CMD file agreement.

Thanks for hanging in there with the 8-bit world. I work as a compiler writer for Hewlett-Packard - specifically compilers for HP's line of RISC super-workstations - but I will always have a soft spot for TRS-80's. They are tremendous machines.

Incidentally, a while back you asked what people use them for. Back in '85 or so when HP was developing the first commercial RISC mini's, I was the primary engineer for HP's commercial language (COBOL, RPG, etc.) code generation support. This involved handling all of the ugly decimal arithmetic, string moves and comparisons as efficiently as possible. I worked a lot at home testing, benchmarking and comparing various code generation strategies and algorithms. How did I analyze my data? Yep, you guessed it: Visicalc on my Model 1. Eventually the complexity of my benchmark spreadsheet forced me to move on, but I got a lot of very serious, very useful work done before then. The work I did using the Model 1 was the basis of a software patent application and continues to live on in the Millicode code gen support routines on the current super-hot HP 700 boxes.

Double sided M1

Fm Pierre Noel [72321,1223]: Referencing Model I LDOS 5.3.1 FORMAT/CMD (Double sided double density SYSTEM disk):

Format :d (SYSTEM) (Model I, 5.3.1) will format side 1 of track 0 in single density instead of side 0 of a double sided disk. This problem prevents the making of a bootable double sided double density system disk for the model I. I patched FORMAT/CMD to correct that problem. With this patch, FORMAT :d (SYSTEM) will format track 0, side 0 in single density instead of side 1 (BUG?). FORMAT will still lock out all the granules of side 1 of track 0. The second granule of track 0 (side 0) will still be free, even if allocated to BOOT/SYS by SOLE. The same problem exists with single sided system disk. The third granule (track 0, side 0) will be locked out in both cases.

Other PATCHES requested! Since track 0, side 0 is formatted in single density and has only 10 sectors, this track cannot be used for files. It will be better if the three granules of track 0 (side 0) were allocated to BOOT/SYS and no granule locked out. This should be applied to both cases (1 or 2 sides) when the (SYSTEM) parameter is used. The three granules of track 0, side 1 (double sided disk) would be available in double density like on any other track. Only bad granules should be locked out. Many backup utilities will unlock good granules on the destination disk. Allocated granules will always stay allocated. If these granules were allocated to BOOT/SYS, SOLE should not increase the number of granules allocated to BOOT/SYS. BACKUP can not make a mirror backup of a double sided disk with track 0, side 0 in single density. The procedure to make a SYSTEM disk shall be used even to make a backup copy. Some other backup utilities can do a mirror backup of such disks.

Fm MISOSYS, Inc: A letter was sent to all Model I 5.3.1 recipients which included a fix for that, plus FDUBL and SOLE for Percom-type doublers, plus one small absolutely mandatory patch for SYS0/SYS - all 4 fixes strictly for the Model I version. Unless you are affected by the Canadian Postal strike, you should have received the letter by now.

Fm Pierre Noel: Roy, I did receive your letter containing the patches. The first two were already installed. I added the two others (SOLE2/FIX and SYS011/FIX). Even with these fixes, FORMAT :d (SYSTEM) still formats in single density the wrong side of a double density and **DOUBLE SIDED** diskette.

SOLE will not detect the problem and will modify the BOOT/SYS but it won't BOOT because the track is in DOUBLE DENSITY (sectors numbered 0 to 17 and formatted with 6DB6 on the first side and sectors numbered 0 to 9 and formatted with E5E5 on the second side).

I patched FORMAT so that the right side of cylinder 0 is formatted in single density. I uploaded the patches in file FOR1/FIX.

SOLE and BACKUP can be use to make a DOUBLE SIDED SYSTEM disk in DOUBLE DENSITY using the patched FORMAT/CMD and the instructions in the upgrade documentation (pages 5 & 6).

BACKUP will not make a MIRROR BACKUP of this type of disk but BACKUP "by class" can be used: in fact the whole process of creating a bootable double density disk must be used.

Since, I added other patches to FORMAT and SOLE so that the second side of track 0 (formatted in double density) is not locked out and is available for data. The first side of that track is allocated to BOOT/SYS by SOLE with no granule locked out. SOLE2/FIX still keeps one granule locked out. With this additional patch, FORMAT does not lock out any granule unless there is a bad one.

Is BIT 4 of byte CD of the GAT used to indicate a diskette with track 0 is in single density while the others cylinder are in double density ?

Fm Roy Soltoff: Pierre, Bit 4 of GAT+CD is used in LDOS 5.3.1 to indicate that cylinder 0 is single density while all other cylinders are double density. I chose not to mix densities on a 2-sided track 0 because of the complications arising from the methods used by the DOS to calculate the actual sector given the granule number.

Fm Pierre Noel: Roy, I thank you for the information about bit 4 of GAT + CD. I agree with you that the DOS can't easily deal with dual density disk. But this does apply equally to a single sided disk as well as to a double sided disk. As long as the single density track is not used by the DOS (BOOT/SYS excepted), there should be no problem with a bootable double sided system disk. The single density track granules shall be allocated to BOOT/SYS and/or locked out.

FORMAT/CMD :d (SYSTEM) has a small bug that causes the problem with a double sided disk: after formatting and verifying are completed, FORMAT re-formats track 0 in single density. In order to do that, BIT 6 of (IY+4) is reset (select single density). In the case of a double sided disk, BIT 4 of (IY+3) is set (side 1 was last accessed) and must be reset (start with side 0). Adding a <RES 4,(IY+3)> (FD CB 03 A6) somewhere in a shortened message or in memory below X'6000' shall do that. Single sided disks will not be affected with the additional code, that bit being always reset. FORMAT :d (SYSTEM) will then make a double sided disk with cylinder 0 in single density (both sides); the granules of the second side will be locked out (no patch necessary). SOLE (modified with your SOLE2/fix) will work and BACKUP (no patch required) will recognize that disk: BACKUP will make a mirror image backup of such a disk.

Without that modification, FORMAT :d (SYSTEM) re-formats only side 1 (the wrong side) of cylinder 0 of a double sided disk. SOLE installs its double density driver on side 0 of cylinder 0 which has been left in double density;: that disk can't boot. SYS0/SYS can be located about anywhere including any side on a Model I double density system disk; on a Model III system disk, it must be located on side 0, otherwise the disk won't be bootable.

I also patched FORMAT so that when (SYSTEM) is requested, BIT 7 of GAT+CD is reset (directory slots are reserved for SYSTEM files) and FORMAT terminates by automatically running SOLE. The formatted disk is ready to receive the system files, even if some files are already written to that disk before the system files are transferred to it. Automatically running SOLE immediately after formatting allocates the second granule of cylinder 0 to BOOT/SYS and prevents the DOS from trying to use that granule which is in single density and is not compatible with the other cylinders which are in double density. I think such a diskette (single or double sided) can be used safely even if the system files are not backup'd onto it.

Special deal? MISOSYS could sell a special package to Model I owners: two double sided drives (I don't think single sided drives are still available) in a case with power supply, a new drive cable and a DOS capable of using all the capacity of each drive! Some Model I users are probably still using their worn single sided <35> track drives. If someone buys new drives to be used with a Model I and afterwards the Model I should die, he could easily use its new drives with many computers, including those who use MSDOS.

Fm MISOSYS, Inc: You are absolutely correct about the bug in Model I FORMAT with the SYSTEM parameter. I didn't read you clearly before. What is happening is that after the disk is formatted in double density, a 2-sided format will leave bit-4 of DCT+3 set, indicating that the second side is selected. The subsequent routine which is supposed to re-format track 0 then starts with the second side (side 1) and does not re-format side 0. That's why a SOLE'd 2-sided disk can't boot. The problem is fixed by resetting that bit prior to the re-format.

Now the patch you worked up not only resets that bit but also resets the bit showing the disk as 2-sided [RES 5,(IY+4)], so only the first side is re-formatted to SDEN which leaves the second side in DDEN. Although you may find that more practical for your tastes, I don't. The DOS does not properly handle a diskette with a mixed configuration on the same track. The difficulty is in calculating the actual sector number of the second side. I would prefer to not create additional headaches by mixing the density on track 0. Thus, I am modifying your patch accordingly. Note that I have just NOP'd out the additional reset instruction so those who want to follow your recommendation can do so.

Now with this patch applied, Model I LDOS 5.3.1 users can boot a 2-sided DDEN disk and BACKUP can handle the disk - mirror-image - provided additional pre-formatted disks using FORMAT (SYSTEM) are used as the target. The entire cylinder 0 will be locked out from use; however, the reduction of three granules on a 360K disk is minimal.

I have finished with the Model I LDOS 5.3.1 and have to go on to other things - like something to make a buck less we drop out entirely. Next you'll ask me for an LB for the Model I!

```
. FORMAT1/FIX - Patch to
LDOS 5.3.1 Model I
. Patch corrects FORMAT
(SYSTEM) on 2-sided disks
. Apply via, PATCH
FORMAT.UTILITY FORMAT1
D01,D2=21 6F 67
F01,D2=21 63 67
. Call new code (the patch)
D01,DE=CD 63 61
F01,DE=CD 67 64
. New code located in
message at X'6763'
D07,AE=CD 67 64 FD CB 03
A6 00 00 00 00 C9
F07,AE=0A 50 72 65 70 61
72 69 6E 67 20 74
. New beginning of trun-
cated message
D07,BA=0A 20 01 EC 71 67 52
F07,BA=6F 20 01 EC 71 67 72
. Eop
```

LDOS 5.3.1 Question

Fm Bob Cunningham: Roy, A good news/bad news story: The good news-the LDOS 5.3.1 came today. The bad news-how do I default the DIR command to present a abbreviated listing? I tried the PATCH SYS6/SYS.SYSTEM (D08,92=00 00:F08,92=FF FF) which worked with 5.3 but got the message 'F08,92=FF FF FIND line mismatch'. Any suggestions?

Also, a critique item: When I performed the BACKUP :0 :1 (S,I,OLD), it blew away the part of CONFIG/SYS (?) that retained the device settings (active drives, step rates, and SYSRES files). Also, it seems to have copied ALL the files from the new disk, not just the stuff already there (seems the OLD param didn't work for me). If I had known this, I would have printed the original settings so I would have a place to start. Thanks anyway for a super product.

Fm Roy Soltoff: Bob, I have not investigated any patch to default DIR to abbreviated listing. A little work with FED on your part will most likely find the spot. As far as "blowing away CONFIG/SYS", that is not entirely correct. When you perform a BACKUP by Class, BOOT/SYS is copied when SYS0/SYS is copied. That's part of the operation of moving a set of DOS files. There is a pointer in BOOT/SYS which indicates the presence of a CONFIG/SYS file. You could - if you wanted to risk it - save the CONFIG/SYS file to another disk, SYSGEN, then COPY the old CONFIG/SYS back to the system disk (or even SYSGEN,Drive=1) to avoid overwriting the old CONFIG), but you would risk having the configuration file not match with the new DOS. Say, for instance, that you had the KI/DVR in your configuration. Unless you went through a new installation, you would not gain the advantages of the new KI driver (if on a Mod 4). Doing a BACKUP (OLD) will only copy files which exist on the destination disk - it will not copy files which aren't on the destination.

Fm Bob Cunningham: Roy, Since I don't know the ins and outs of how LDOS works, I'm not sure how I could use FED to find the spot in SYS6/SYS to make an abbreviated listing. Perhaps someone on the Forum with more skill might have already attempted it and will make it available to the rest of us.

I understand that CONFIG/SYS was copied over during the BACKUP :0 :1 (S,I,OLD) operation. What I was getting at was it might have been nice to have mentioned this in the one line of installation instructions provided with the disk. The last time I upgraded was in 1987 when 5.3 came out. I must have figured out what I wanted then and recorded it somewhere prior to the upgrade to 5.3.0 since I don't remember it being a problem then. Presently, I was able (I think) to reaccomplish customized configurations to match what I had. I was just surprised I was not informed prior to the operation.

Thanks for the help. Now I guess the software will have a chance of outlive my Model 3. At least you won't have folks complaining in eight years about why they can't input a date past 1999. Keep up the good work and thanks for the reply.

Fm Roy Soltoff: Bob, CONFIG/SYS was NOT copied over during the BACKUP because CONFIG/SYS is not on the distribution disk. What I said was that when SYS0/SYS is copied, BOOT/SYS is also moved. There is a flag in BOOT/SYS which tells the DOS whether or not a CONFIG/SYS file is present. When the new BOOT/SYS overwrote the old BOOT/SYS, the flag was reset; thus, SYS0 was not aware of a CONFIG/SYS file existing.

FED is documented as a utility of LDOS which has been provided ever since 5.1.4 - which goes back a few years - to 1984. You should learn how to use it. If you FED the SYS6/SYS.SYSTEM file of LDOS 5.3.0 looking for the record and byte location of what the old patch was changing, then look somewhere around that location - give or take a sector or so - in the new 5.3.1 SYS6/SYS file, you will probably find the spot to change. If you expect to continue to use your machine, you really ought to spend a little time and be comfortable with FED.

Fm Pierre Noel: Bob, Here is the patch to get the abbreviated listing when issuing a DIR command. It's for LDOS 5.3.1 (Model I & Model III)

```
PATCH SYS6/SYS.SYSTEM
(D08,8C=00 00:F08,8C=FF FF)
```

MARGIN/FLT

Fm Clay S. Scott, Warsaw, NC: Roy, This is in reference to the filter MARGIN/FLT from the FILTERS package no. L-32-053.

At first it seemed that this filter (MAR-

GIN/FLT) might be another handy and useful way to set the printer's left margin but when I installed it with the command FILTER *PR MARGIN/FLT (W=15) and then sent data to the printer, only the first line would have the specified margin width (w=15). All the following lines had no margin spacing. These results were obtained using a Mod 4, LDOS 5.3.1 and a Tandy DMP 430 printer.

The mention of the two character printer code sequence in the filter docs is very confusing to me. How and where is this printer control code sequence to be passed entered or input into the system and at what level?

Finally, could this filter be used as a general purpose rnethod to control the printer left margin under LDOS without using the two character control code sequence? Your comments and explanation will be greatly appreciated.

Fm MISOSYS, Inc: Clay, concerning the LDOS MARGIN filter (part of FILTERS product), MARGIN is supposed to output space characters prior to the output of a line to the printer - all lines. I just tested it out on my DMP500 printer and it works as advertised. There's a catch, though. The margin spaces will only be generated when the preceding character is either a RETURN or LINEFEED. That is not an abnormal catch. You might investigate what the output stream was. Perhaps neither a RETURN nor LINEFEED were being sent? This could be the case if the program generating the output was relying on automatic return-linefeed generation by the printer at the end-of-line.

The documentation on the control sequence is rather confusing. I didn't write that as the product was acquired from LSI; however, I can tell you what it is trying to say. In certain cases, you may want to output a two-character printer control sequence prior to emitting data output to the printer. For instance, change to condensed characters on a Line Printer V by "LPRINT CHR$(27);CHR$(14);". Some printers must have the sequence at the beginning of a line - i.e. if the margin spaces were emitted immediately following the CR-LF (not under your control), then the printer control sequence generated by your program would not be acceptable to the printer. Thus, the MARGIN has a parameter (CONT) which you use to tell the filter what the first character of the printer control sequence is to be. It will default to the ESC character, which is the first character in the example noted above. Thus, the way the filter would work, if an ESC is detected immediately following a CR or LF, then the margin spaces will be suppressed until after the character following the control code (the filter expects the printer control code to be two characters, the first of which is the character noted in the CONT parameter).

The filter certainly is a general purpose left margin controller. Note that MARGIN also does not require a WIDTH parameter to accept a MARGIN parameter.

SYS13/SYS PROBLEM

Fm Robert Hengstebeck: I have been trying to install shell20/arc on my model IV, and am getting an access denied message from the installation procedure. I have used this program before, but I admit to still being a novice with the model IV. According to the instructions "shell20/doc" the file "shellcmd/v20" will be copied into the file "sys13/sys". Therefore I guess that somehow the attributes of "sys13/sys" are set so that I cannot write to it. If I am right, then can someone show me what I should be looking for, and also how to fix whatever I find? Thanks ahead of time for your help.

Fm Frank Slinkman: If you're using LS-DOS 6.3.0, the full file name is "SYS13/SYS.LSIDOS". If you're using LS-DOS 6.3.1, the full file name is "SYS13/SYS.SYSTEM6". The program should be copied via: COPY <whatever> SYS13/SYS.password (C=N). If you're using an older DOS, such as TRSDOS 6.x.x, you need LS-DOS 6.3.1.

Fm MISOSYS, Inc: As an additional point, if SYS13/SYS was previously deleted from the target disk, COPY will not be of benefit; you have to BACKUP SYS13 from a full system disk to the intended target (don't forget the S parameter) so the file will be properly positioned in the directory. It is possible that once previously removed, another file may wind up occupying the SYS13/SYS directory entry which would inhibit you to add SYS13. If that is the case, start with a fresh system disk.

TRS-80 Emulator

Fm Yves Lempereur [74016,1741]: Well, I'd like to tell the story of a pretty unbelievable coincidence that happened to me recently. So, here goes:

A couple of days ago, I got an EMail from a friend of mine who lives in Belgium, telling me that he just ran into somebody he and I hadn't seen since 1983. This is back when I was making a living at writing computer games for the TRS-80 model I/III for a company called FUNSOFT, INC. I switched over to the Apple Macintosh back in 1984 and have been writing software for it since then (all this is actually important, so hang in there). Anyway, that somebody is none other than Vincent Van Den Berghe, author of the TRS-80 Emulator for the IBM PC, "TRS80.ZIP". He and I knew each other back in 1983 (I still lived in Belgium then, I moved to Los Angeles in early 1984). This is where things get a little weird; it turns out that about six months ago, I, myself, wrote a TRS-80 emulator for the Macintosh (out

of boredom and nostalgia, I guess). How's that for a coincidence?

Anyway, I never gave the emulator to anybody out of fear from the copyright gods! What I'd like to know is: is anybody interested in this emulator and is there a legal way to distribute it?

A couple of technical details: it is completely written in MC68020 assembly language and runs on any Macintosh with an 020, 030 or 040. It gives true TRS-80 model I performance on a Macintosh IIci and actually runs faster than the real thing on a Macintosh IIfx. It ran everything I threw at it (and I have a lot of stuff). However, it only runs from virtual disks (these are files that emulate disks) since there isn't a 5 1/4 inch disk drive on a Mac. The TRS-80 screen is reproduced pixel for pixel inside a window on the Mac screen. It is MultiFinder compatible and actually keeps running when switched to the background. Its interface is very rough right now, but I will make it better as soon as I can find the time to do so.

Fm William Buzbee: Hi, Here's one Model 1 user who migrated to Macintosh. Incidentally, I also wrote a Model 1 emulator, but used Pascal. It's not terribly fast, and only supports pseudo cassette I/O. I haven't released it because of similar copyright concerns, but I doubt there would be any trouble if it were released without the ROM image.

I'd love to get a copy of your emulator - can you upload it without the ROM image - I can reconstruct that myself if you let me know what format you use.

Fm MISOSYS, Inc: Anyone who has a legal TRSDOS 6 disk with the MODELA/III file on it should have a near-copy of the Model I ROM. True, the Model III ROM was not identical to the Model I, but BASIC was virtually identical. On the other hand, there is more utility of dealing with a Model III or 4 emulator. But that is a personal observation.

Fm Yves Lempereur: William, The emulator still needs quite a bit of work on the interface as well as the floppy disk controller. I wrote the current floppy routines by disassembling the Z-80 routines of TRSDOS and guessing at what they needed. I recently received the data sheet for the INS1771-1 Floppy Disk Formatter/Controller after dealing with half a dozen people at National Semiconductor over a period of two months. I will re-write the floppy routines completely as soon as I get a chance. Also, it seems that my screen emulation does not work under System 7.0 (incidentally, I do emulate 64 AND 32 character modes). I will have to look into that as well.

Finally, I seem to have a problem with the keyboard emulation and have been talking back and forth with MacDTS (Macintosh Developer Technical Support) and it seems to be a bug in the Macintosh OS itself. Anyway, I really would like to hear from you about how you resolved some of the more obvious problems involved in writing the emulator.

Either a technical description of the problems/answers or the source code itself would be highly appreciated (you might have come up with clever answers to problems I resolved in a pretty brain dead manner). My goal is to make the best (read fastest) possible emulator and then release it in the public domain.

Model III High Bit Chars

Fm John Shanafelt: My brother, Gary Shanafelt, asked me to post the following message: I have a Model 4, and in Model 3 mode when I try to print high bit characters, the printer driver in ROM seems to tamper with all the characters above 159. 160-191 are incremented by 64, while 224-255 are decremented by 64. In other words, if I try to print character code 160, I get 224; to get 160 on the printer, I have to send it code 192. What is going on here? I notice further that the codes print as they should if I have LDOS PR/FLT installed, and they print normally in Model 4 mode. Is this deliberate, or a bug in the ROM driver?

Fm MISOSYS, Inc: These are all adjustments performed by the ROM printer driver. When you use the LDOS forms filter (PR/FLT), it replaces the ROM printer driver with its own printer driver - essentially a driver which passes all codes to the printer except those translated by the filter itself.

SuperCross

Fm Gary Phillips: Roy, is SuperCross/XT something you now handle? Mine is version 2.0 dated January 31, 1985. If there was anything in the way of an update after that, I'd like to get it. My copy was from Breeze/QSD-PowerSoft, but I don't recall seeing SuperCross listed in your ads after you picked up their product lines.

Fm MISOSYS, Inc: Supercross/XT is not a MISOSYS product. I believe that Powersoft purchased it from Hypersoft and resold it, or were licensed to manufacture it. Check with them (him), as the Hypersoft ad in CN80 shows a hypercross product (probably a more recent version).

Kermit for Model IV

Fm Tom Bourke: Just saw in '*The Quarterly*' that someone was asking for a copy of the model 4 version of Kermit.

Well, it landed on my shoulders to develop bits and pieces for it. However, time passed by and I got in touch with Greg Wonderly (ex Oklahoma State) and got a version from him. This version is (was?) later than the one in the Kermit distribution.

I'll attempt to dig it up (bit chaotic here, I'm preparing to move!), and send it to Vic, the librarian. Also, I'll see what the smallest size is, and upload it here (to CompuServe PCS49)...

TMQ

Fm Bob Cunningham: Roy, According to page 3 of The Blurb section of TMQ Vol VI.i, "...I hope as this last year of publication proceeds...", I understand that you intend to call TMQ quits with VI.iv. If so and my address label says 92/05, how many more issues do I get? How much do I pay to finish out the volume VI series? I hate to see it go, but I can understand with a shrinking product base, you have hung on about as long as possible.

Fm MISOSYS, Inc: TMQ can not continue publication with only 300+ subscribers. That's why it goes. Unless, of course, all of a sudden, I gain another 300+ subscribers. With 92/05 expiration, that carries you to issue VI.iii; a $6.50 fee would extend that to 92/08 - good until VI.iv.

More TMQ

Fm Jim Potsch: Roy, Seeing your message to Bob Cunningham reminds me - I need to extend my subscription to TMQ to [whenever]. Also, I need to order LDOS 5.3 for use on the 4P in III mode. I only have TRSDOS for the Model-III mode. I have 5.1.4 for the Model I - don't have that set up but may order in the near future - that was a good machine. [Loaded. 64K with EI and RS232, Percom doubler, Omikron CPM - with (gasp) 48K Wordstar, DR Basic, etc.] Gad, was Wordstar sllloooowwww on a M1. Used Scripsit.

A question also: I'm trying to apply the HDBOOT patch (Adam Rubin as modified by Gary Phillips for the 4P) to get the 4P to boot off a RS 15 Meg drive. (ex Bob Haynes) Now this may seen silly, but how do I examine the DCT for Drive :0 in the Hard Disk? According to the instructions I need the sectors per cylinder and the sectors per granule, in hex, to apply to the patch. Also, on a 306 cylinder Drive :0, on what track would the directory be? I am trying to recover a couple of accidentally deleted files.

Fm MISOSYS, Inc: Well, you can examine the DCT using DEBUG with either *The Programmer's Guide to DOS6* or Tandy's *Technical Reference manual* in hand. Or you could get a direct readout of the DCT data - in English - from the DCT utility which is part of the GO:MTC utility package (recently reduced 11/1 to $39.95 plus $4 S&H. Note also that GO:MTC includes the UNREMOVE utility which effortlessly recovers removed files where the released file space has not been reused by another file.

LS-DOS 6.3.1

Fm Bruce D. Berry: Roy, I am interested in knowing what programs are available for my TRS-80, Model 4D. I am using your LS-DOS 6.3.1. I am very interested in some sort of PRINT SHOP program to make birthday cards, etc.

I also have encountered two problems with my system. I am using a Radio Shack 15 meg hard drive and a DMP-107 printer.

1. After using DeskMate 01.00.00, and returning to LS-DOS, I find that my RS232 parameters have changed from the settings I have sysgened.

2. Sometimes after exiting DeskMate, my DEVICE table prints scrambled information at the bottom of the screen.

One other question. I have level 1H LS-DOS and have seen some kind of upgrades listed in your forum's libraries to level K or so. How can I download them (if needed). I use COMM (capture protocol) to access COMPUSERVE.

Fm MISOSYS, Inc: If you see scrambled information after running DESKMATE, it means that the system stack has overflowed. DESKMATE must apparently be using the OS stack area - a bad practice when a program uses a lot of stack space. I can offer no solution to that. DESKMATE may also be changing the COM parameters since it needs, and uses, the COM / DVR.

The latest LS-DOS 6.3.1 disk is Level 1H, so your disk is current.

MISOSYS Catalog

Fm Bruce D. Berry: Roy, Thanks for your speedy response in mailing me your catalog. I do have a couple of questions for you.

1. Can PowerDraw and/or PowerDOT 2.0 run on Model 4? If not, can they be CONVerted to run on Model 4? The reason I am asking is that I am set up for Model 4 and use a Radio Shack 15 Meg. hard drive. I realize that I could boot up in III and run such programs, but would prefer to use 4.

2. Although PowerDOT 2.0 states that it can use a Radio Shack printer, (I have a DMP-107) PowerDraw does not say if it supports an R/S printer. Does it?

3. What is the price of the U-72 PAL chip for the 128K upgrade? I have 3 Model 4 computers and would like to take the memory chips out of one to expand the memory in another and remember reading something in a message here about needing the U-72. I have read the instructions in my Hardware manual, and assume all I need, besides the extra memory chips, is the U-72.

Fm MISOSYS, Inc: PowerDraw and Dot are strictly Model III programs. They cannot be easily converted. I suppose with sufficient advance orders, I could get Model 4 versions - but that's a big if. There is a version of PowerDot for Tandy DMP printers - but since not all of Tandy's DMP printers support the same codes, I can't confirm if the 107 would work properly. If you want to try PowerDot, you can return it within 30 days for a refund (less S&H). I sell the U-72 PAL for $8 + $3S&H.

Fm Bruce D. Berry: Roy, I have received PowerDOT and it works just fine! One question though... To erase dots, or to move the cursor quickly, you hold down the <CLEAR> key and press the appropriate direction key. This is a little awkward as the clear key is on the right hand side of the keyboard. Is there a possible PATCH to use the <CTRL> key instead of <CLEAR>?

Fm MISOSYS, Inc: Well I don't know of any patch. It would take me too long to evaluate the source code to see if a patch is even feasible.

Adds25 Host/Term

Fm Dan Martin: Roy, My use of LS-DOS is only done across Adds25 Host/term. Would it be possible to turn off the video driver to wring more speed out.

Fm MISOSYS, Inc: Turn off the driver from where? If you are referring to a remote Model 4, the time for the remote to operate its video is miniscule compared to the time it takes to transmit video information to your machine.

International DOS 6.3.1

Fm Claude Wyngaert, BELGIUM: Roy, I receive the letter from TMQ dept. of JULY, announcing the release of LDOS 5.3.1.; it was a great surprise for me, I thought that LDOS 5.3.0. was a finality; THANKS to continue to develop it!

Roy, we are a lot of people to use LS-DOS 6.3.0. in AZERTY version with our MODEL 4 in EUROPA (and in CANADA); and we are still waiting more than one year that you release the AZERTY version of the 6.3.1.

Roy, we NEED the upgrade to continue to follow new releases, patch's in TMQ, new versions of many program's...

I don't know if it's a long work for you to make an AZERTY version of the 6.3.1.; when I read THE SOURCE it appears to me that only the keyboard table, the printer driver and an AZERTY flag in SYSO must to be changed; it must be the same job as the 6.3.0. AZERTY so you don't need to perform an important 'brain work'! (and maybe you've yet a JCL to do it).

Can you find an afternoon to compile it; all french TRS80) MODEL4 users will appreciate a lot and I order to you immediately a disk; I think that your work will be gratified, you've the CANADIAN, FRENCH, and BELGIUM market (and maybe others?).

I yet purchase the version QWERTY of 6.3.1. (for my collection) but can't use it for intensive work. And I'm a subscriber for TMQ since 3 years now and have all the other numbers.

I tried myself to decode SYSO of the 6.3.1. and change the keyboard table but without success, I'm a very poor 'assembler programmer'.

I thank you yet for your future help MISOSYS give the best service in the MODEL-4 TRS80) world , and I know that you'll do something!

Fm MISOSYS, Inc: Claude, The problem with MISOSYS generating the international versions of LS-DOS 6.3.1 is that Logical Systems was not the company which generated the international versions. To my knowledge, those versions were generated by Tandy. Thus, I do not have the keyboard and printer drivers or other particulars of those versions. Without those drivers, I cannot perform the DOS generation.

On the other hand, I have received from Hans Bongartz of Germany, a disk con-

taining patches to a LS-DOS 6.3.1 disk to turn it into a 6.3.1G. I will send you a copy of that disk and include it when I ship the order which accompanied your letter. I suspect the German version is the one you use.

Perhaps I can make some inquiries in Tandy to locate the files associated with the other two international versions. I had expected to include Hans' work in an upcoming issue of *The MISOSYS Quarterly*, so perhaps others may be able to derive the French version from the patches.

PROWAM 'CALC'

Fm Hans G. Bongartz, Germany: Roy, The autumn issue of TMQ, VI.i is packed again with many ideas. Some of them require an answer, I think. Your idea of a new DOS manual for LDOS/LS-DOS for all the model 1/3/4 computers is just right for me! I have all those machines up and running until today and the new model 1 and 3 Upgrades of the 5.3.1 LDOS are of great interest. I knew that the DOS is originally not designed for the foreign keyboards! The success to adapt the famous LS-DOS 6.3.1 for german model 4 keyboards - it runs for over one year without any trouble - make me think to try the new releases of LDOS 5.3.1 too.

Next you mentioned a coming version 2.2 of the LB Data Manager with the new SIMILarity function and automatic generation of formats for a screen and print format. I'm very interested to get an upgrade for my current model 4 LB Data Manager V 2.1 when it becomes available!

My next point is a question concerning the PROWAM 'CALCULATOR', when used to add some numbers - in the range of hundreds with two places after the decimal point - e.g. when I added the amount of my current order, I get sometimes a one in the fifth place after the decimal point, although nothing is entered in the third and fourth places! Because there is no overflow condition, too big numbers or the like, I can't isolate the reason why the CALCULATOR does crazy things. It has been invoked from within DOS command level as well from inside the TED text editor, in both cases you will get the unexpected '1'. Did you get another report of this behavior of the PROWAM CALCULATOR?

All the other things sold from you - software and hardware - will do the job and made the old model 4 a fairly powerful machine, which I got in spring of 1986 as a 'last chance buy' before it disappeared from the market here in Germany, when R/S closes the computer shops. I began with the Model I in 1978, adding a Model III in 1984 with the 'new LDOS 5.1.4', a very big new manual to learn, but with the ending 'date range until 12/31/87' as you know. Then I got this Model 4 as a cassette only version, had to add a second 38 watt power source and disk controller board, and so on I finished with a serial interface and a Hires graphics board, two dual side TEAC drives and those 128K of memory. Looking for new software to keep the Model 4 more useful I soon got attention to your product line. I would not stress you with the story of my way to the Model 4, but it is somewhat like a thank you, to keep this old machine running on a useful level with 40 MB hard drive and real time clock now surviving for the nineties, compared to 12/20/33 MHz of modern PC's, slow with its 4 MHz clock.

Your winter vacation reminder directs the eyes to the declining 1991 and of course, Christmas time. I owe you another thank you for your support to the old Z-80 based machines and express my best wishes to you and your family for a nice Christmas and a happy NEW YEAR 1992!

Fm MISOSYS, Inc: Hans, Right now I am working on release of 2.2 of LB Data Manager; I am very excited about that. I am just about finished with the new LB. I didn't introduce automatic generation of print screens, but I did introduce an automatic generation of an edit/update screen. LB is just about at its memory limit for the Model 4. What I have been spending all of my time on for the past month is a database conversion utility to be bundled with LB. It supports direct generation of an LB database from fixed fielded records, dBASE II, dBASE III, Profile 4, and pfsFILE4. I'm in the process of adding DIF to LB. It also currently handles conversion of an LB database to dBASE II, dBASE III, DIF, tab delimited, comma delimited, and ASCII strings. It's big - about 42K so it will be a squeeze to get in the DIF to LB conversion.

The reason why you can occasionally see a 0.000001 pop up in a calculation using PRO-WAM's CALCulator, is because of the inexactness of converting decimal representation to and from binary floating point and the errors introduced during calculation. There are methods to mask these small values during conversion to display; however, because of the space considerations in a PRO-WAM popup utility and the amount of code space taken up by floating point routines, it is next to impossible to introduce the *fixup* code in CALC. You'll just have to live with it.

Your history with the Tandy computer lineup sort of parallels mine. My first Model I was acquired in 1978. I continued with the Model III and 4 as soon as they were out. Of course, my development efforts with the Model 4 DOS provided a very early exposure to the 4. Some of the first machines we were using to evaluate the DOS still were in the old gray Model III cabinets.

I am really going to have to apologize to my foreign friends over some real in-house snags. You won't believe this. During some conversation with one of my Tandy system software contacts, I asked him who did the foreign keyboard drivers for the Model 4 DOS. I was informed that LSI did, but they required Tandy to manufacture and support the foreign versions. I

was told that LSI should have had the driver(s). Well is my face red. I always knew that the version 6 system code had conditionals for the French and German versions, but I never realized that the foreign keyboard drivers were part of the files in my possession. Sure enough, after some search, I found the combined foreign keyboard driver and was able to generate a foreign version. I need at this point to verify that I installed the handful of 6.3.1 patches into the source code so that a system generation will not re-introduce bugs previously corrected. But I will be generating the Model 4 DOS foreign versions and making them available. Announcements to that effect will be in the next issue of TMQ. The Model III LDOS is a different story.

XHIK & Aerocomp HD

Fm Jane A. Layman, Waukesha, WI: Roy, I am in receipt of XHIK or the LDOS 5.3 XLR8er Interface Kit. Whenever I try to use it from my Aerocomp 20meg hard drive, however, the computer hangs when it needs to write to disk. The same problem occurs with SET2RAM from the first Interface Kit. It occurred to me that the LDOS version of the Aerocomp driver might include "writes-to-ROM" as did TRSHD3/DCT. I did look at the installed driver to see if I could spot an instance of a "write-to-ROM", but am not familiar enough with Assembly language to be successful.

I am enclosing a disassembly of the Aerocomp driver part of the CONFIG/SYS file used for booting the hard disk in hopes that you can use it to spot any offending bits of code. If this is insufficiently helpful, I would be happy to send any disk copies of files you might need. I would really like for XLR2RAM to work on my system as it occupies no high memory compared to the 196 bytes used by H.I. Tech's RAMDISK/DCT.

SASE enclosed. Thank you for your attention.

Fm MISOSYS, Inc: Jane, This concerns your system crashing using the XLR8er interface kit in Model III mode with your Aerocomp 20 Meg hard drive. Aerocomp's driver is indeed the culprit, and the problem is virtually identical to the problem with the TRSHD3/DCT driver. In Aerocomp's HD20SA3/DCT driver, the I/O buffer address is adjusted to 0 if a VERIFY function is requested. The driver then reads the sector into page 0 - which would be transparent if page 0 was ROM but not RAM. Although HD20SA3 sets the buffer address to 0 in a manner slightly different (SBC HL,HL) than TRSHD3/DCT (LD HL,0), I can come up with a patch that should work. Use the following:

```
PATCH HD20SA3/DCT
(D04,23=26 38:F04,23=ED 62)
```

This will load register H with 38H; regardless of the value of register L, the new buffer address in HL will be a usable 256-byte region in the keyboard mapping; the resulting VERIFY will be non-destructive if the Model 4 is in the RAM-Model III mode.

If you want to make the change in your CONFIG/SYS file, the corresponding location is X'FED0'. Patch that so you don't have to re-do your configuration.

Incidentally, MISOSYS has acquired all hard disk drivers from Aerocomp - including their CP/M drivers and LDOS/TRSDOS drivers for their own hard drives. MISOSYS has also acquired their hard drives, FDC cards, serial cards, and Model I DDC.

LB's Auto-Date field

Fm Karl Krelove, Levittown, PA: Roy, I'm running LB v.2.1.0. I am doing a selection (Main menu - option 5) in which I simply want to select all records in which Field 28 (L1) contains a "Y" and Field 30 (Date of last update) is today's date (1991/10/23 for October 23, 1991). I don't need the list to be sorted, only selected according to these two criteria. All goes well until I try to specify the date criterion for Field 30. LB acts as if this were a numeric field, accepting only numbers, "+", "-", and"."). It won't accept the slash (which is stored on the disk in the "date of last update" field), nor will it accept an asterisk as a wild card character. I can't find a way to enter a date for the sort module to match in this field. Is there a way I've missed to select by specific dates from this field? The protection level is "E", I guess, because the program maintains the field automatically and it would be meaningless if a user could alter it during Edit/Update mode. But does the protection category block data entry in the sort module (the manual only says Calculated fields will not work)?

I'm sending a copy of the field definitions. I can get around the problem by defining Date as a literal and entering it myself from the keyboard each time I need an accurate date for any reason, but I'd rather have the date entered automatically - it's more reliable and accurate. I need to send a letter to the parents of any child whose record I have tagged in the L1 field during the current school day. I generate a SuperScripsit merge file directly from the LB print module using this index file, printing the report to disk. I don't want the records of other children who have been tagged previously to show up in the merge file because their parents have already been sent the letter. I don't want to erase the "Y" from any of the records because I want to be able to tell who has already

received the letter so that a duplicate of the same one is not sent. Needless to say, I can also simply select all the "Y" entries in Field 28, sorted by the contents of the entire Date field, then go into the resulting report file (the merge file I create by printing a report to disk) with SAID and simply delete all the records earlier than the date I want. Since the goal is as much automation as possible, this is also a distinct second choice. My first choice is to be able to select on a specific date in the Date of Last Update field. Can I do it?

As always, thanks for your help. I really have tried, by the way, to find an answer to this in the manual. I've struck out.

Fm MISOSYS, Inc: Karl, You have found a legitimate problem in LB. There should be nothing which prevents you from selecting a "date-last-updated" field, type '\'. Unfortunately, when that field type was implemented, its use as a selection field was not explicitly programmed.

I have worked up a patch to the TRS-80 version which provides a kludged work-around to this problem. For selection, the field will be treated as a literal field. That is not the correct way to eliminate the problem as you really ought not be permitted to enter any character in the selection string. On the other hand, it does no harm. A more rigorous solution will await the next release when I can touch up a few modules to clarify the '\' type's character entry for selection as NUMERIC DIGITS ONLY plus '/'.

Here's the work-around patch. Apply it to LB/OV7.

```
. LB7A/FIX - Allows entry
of '/' when searching '\'
fields.
D19,69=2A B7 64 7D FE 5C
CA F5 3E 3E 20 BD 30
F19,69=21 20 00 ED 5B B7
64 CD 13 D2 7C B5 28
. Eop
```

LB Database Template

Fm Barry Thrippleton, AUSTRALIA: Roy, I have recently been updating my database records to the new Release 2 version. I must say the changes are significant and certainly make using the programme very enjoyable. Further to your request in TMQ Vol VI.i I have enclosed a disk containing an example of one of my database uses. It relates to keeping records of my stamp booklet collection. In regard to the LB2: Cursors article my experience is similar to Karl Krelove's. It does not detract from operation of release 2 however it does appear redundant when it appears after the highlighted command bar.

With regard to Indexes: I appreciate that to include newly added records into an existing index requires that all the data needs to be resorted. Is there a means of recreating an identical index file without having to re-enter all the select parameters? i.e. to do a resort of the data without having to re-enter the select criteria.

Is there a means of easily checking what the selection criteria are in each of the indexes. For instance, in my family tree database, I always have difficulty remembering which index does what. Thanks again for the support of our TRS80 computers. I have included this letter as an ASCII file onthe enclosed disk.

Fm MISOSYS, Inc: Barry, if you want to repeat a particular sequence of commands in LB, such as a set of sort criteria, simply turn on AUTO and save the keystrokes to a JOB file. Then all you need to do is to use that job file at some later time. For instance, I generate three indexes to sort my customer database by 1) last name\first name, 2) ZIP code, 3) Company name. That's a lot of keystrokes to have to re-enter. But I only had to do it once. From the main menu, I turned on AUTO and provided the name of a new job file. Then every keystroke used from that point on to enter the sort module, perform all three selections, then return to the main menu, was saved in the job file. Re-invoking the AUTO command from the main menu turned off the recorder (ESC would have done it also). Once that was done, I now use that job file and all of the commands are performed automatically.

Database Field Definitions

Name:	Type:	Length:	Protect:	Required:
Book retail price	R	4	N	N
Contents	L	6	N	N
Book Series:	B	15	N	N
Book No:	D	12	N	N
Book No: (ext)	B	1	N	N
Code letter:	U	1	N	N
Bar Code No:	R	6	N	N
Pane No:	U	4	N	N
Cylinders:	U	10	N	N
Printers (Stamps):	A	8	N	N
Printers (Covers):	A	8	N	N
Qty:	R	1	N	N
Value (each):	N	6	N	N
Value:	C	0	C	N
Source:	U	1	N	N
Special Features:	L	40	N	N
Location:	L	10	N	N

Under LB's architecture, no selection criteria is saved after a selection completes. Under LB 1.0 through LB 2.1, only the number of the key field was saved. Starting with LB 2.2, an index file saves the field numbers of all fields used in the comparison string. This is used by the duplicate find function, and the first two are used by the EDIT/UPDATE find function. Using LB 2.2, you could at least display the names of those fields by invoking the find duplicate function; the names of the fields used as criteria in the attached index file will be displayed.

pfs-FILE & Hard Drives

Fm Norbert A. Klemmer: I bought PFS File (Cat No. 26-1518) about six years ago and in that time have made my five copies to use. About 2 years ago I bought a Hard Drive for my TRS-80 Model 4 Computer. I have tried to install PFS File on my Hard Drive but can't do it. I have tried using "COPY", "BACKUP FIL/CMD:5:0", and just "BACKUP:5:0". This does not work.

My question is: Is there some way I can install this program on my Hard Drive?

Fm MISOSYS, Inc: If you have a ZAP utility, change byte X'C6' of track 0, sector 2, to X'05'.

Fm Norbert A. Klemmer: Thank you for answering my letter dated October 30. I'm sorry I cannot make out your reply.

I have your program (LS-DOS 06.03.01, LS- DOS63 LEVEL- 1C) installed on my hard drive (drive :0). Every time I want to use PFS FILE (Cat No. 26- 1518) I have to insert a floppy disk in drive :4. I have tried to install PFS FILE on my hard drive by using "COPY", "BACKUP FIL/CMD", AND "DISKCOPY". I can copy this program to another floppy disk but not to the hard drive by using "DISKCOPY". It states in the book that "the program diskette can only be backed up FIVE times" and I have used my 5 times before I got the hard drive. I have tried "PATCH" but it comes up that it is a protected file (must be a special password).

I also noticed that you have a MSDOS Software called LB Data Manager 2.1. How does that compare with PFS FILE? If I would get it, can I convert my PFS FILE's over to your LB. I have four years of records on PFS now. Could you send me some information on LB?

Any information on how to install PFS on the hard drive would be helpful. I am not a computer expert, so please use simple language.

Fm MISOSYS, Inc: Norbert, The password for BOOT/SYS should be in the DOS upgrade manual. In any event, using LS-DOS 6.3, the BOOT/SYS password is "SYSTEM6". Thus, you should be able to re-adjust your used-up pfsFILE master via the command,

```
PATCH BOOT/SYS.SYSTEM6:d
(O=N,D02,C6=05)
```

Note that the "O=N" uses the letter "O", whereas the "D02,C6=05" uses the number 0. The "d" should be replaced with the number of the drive containing the pfs master diskette.

Now we do have an MS-DOS version of LB (LB86) as well as a TRS-80 Model 4 version of LB which is suitable for your machine. I consider LB infinitely superior to pfsFILE. There are methods to convert FILE's data to LB. I am enclosing an excerpt from an issue of *The MISOSYS Quarterly* which explains a method. I am also in the final stages of developing LB version 2.2 which will include a database conversion utility; LBCONV will directly generate an LB data base from a pfsFILE database as well as from PROFILE, DIF, dBASE II, dBASE III, and fixed fielded record format. Note that our 50% tradein offer would enable you to obtain LB at $49.50 plus shipping. If you wish to take advantage of this prior to release 2.2, I could provide you with the LBCONV utility for your use - it currently supports everything but DIF.

LDOS & Utilities

Fm Robert L. Hales: I have been using the LSI utilities disk for years. I have recently added 360K & 3.5" drives to my most used Model I and to my Model 4.

I use MultiDOS occasionally for the utilities that come with them. The /CMD files won't run on LDOS. My favorite utilities are FMAP/CMD. CAT/CMD, ZAP/CMD, VFU/CMD and the basic utilities to remove spaces and combine lines in a basic program. Are there comparable utilities for LDOS?

I have a question about the Attrib of DIRectory command. On page 12 of the LDOS 5.3.1 Upgrade Documentation, I haven't found a description of the PaDS attribute of SIP*+ in this manual or the LDOS 5.1.3 manual. What is it?

If I have read the manual correctly the Model III LDOS 5.3.1 includes the functions of the KI4/DVR that: is a part of the Hardware Interface Kit. Are any of the other Hardware Interface Kit functions been incorporated into the newest version of LDOS?

I love the Model I LDOS 5.3.1! LDOS 5.1.3/5 works good but the new version seems to be more compatible with TRSDOS 2.3 data files and LDOS 5.1.x than LDOS 5.3.0 for the Model III.

I had no problem in making a DDSS bootable Model I 5.3. 1 system disk on a 5.25". It: was my understanding that I could make a DDDS bootable Model I

LDOS 5.3.1 system diskette. I have tried several times unsuccessfully. I have one more idea that I haven't tried. Format a DDDS disk, use FED to lock out the grans on the 2nd side, make a DDSS copy as I have done before and then use FED to unlock those grans on the 2nd side. Any thoughts?

Fm MISOSYS, Inc: Robert, You ask whether MISOSYS sells products equivalent to Multidos utilities FMAP, CAT, ZAP, and VFU, but you neglect to provide any specifications or descriptions of them. How am I to know what they do? Judging from the names, I'll assume (you know what that means?) that FMAP presents some sort of file map of a disk layout. We used to have a full scrollable file mapper as part of the MACH2 package, but that was discontinued a long ago due to extremely low sales volume. A single file mapping utility is part of the Utility Disk #1 (L-32-070). What's CAT, a disk cataloger or the equivalent of LS-DOS's CAT command? If the former, we discontinued ZCAT long ago due to extremely low sales volume. If the latter, the only equivalent is DIR (A=). You could also dig into the SYNONYM facility published in an old LDOS JOURNAL. VFU? You got me on that one - Virtual File Utility? Very Funny User? Haven't the foggiest. There were a number of BASIC compression programs published moons ago for Model I or III operation, but they have probably disappeared over the years. I do not know of any currently published.

The PaDS attribute is not new to 5.3.1; thus, there was no need to mention it in the documentation. It is not an attribute alterable by the ATTRIB command, but is indicative of DIR+0, bit 5. It was incorporated into the system when our PaDS utility was released a long time ago. The Model I/III PaDS product was discontinued some years ago due to extremely low sales volume. Incidentally, LSI also used the "PaDS" attribute for its diskDISK product, taken over by MISOSYS after we acquired LSI's retail products.

For a ZAP utility, you ought to consider SuperUtility 3.2.

Only the Hardware Interface Kit keyboard driver function is integrated into LDOS 5.3.1 (Model III version).

I can't imagine why there should be any more compatibility with TRSDOS 2.3 data files in our 5.3.1 release of LDOS. Could you be more specific? On Model I LDOS 5.3.1, double-sided single density disks can boot and single-sided double density disks can boot. Check my letter to Pierre in this issue for the correction associated with booting DDEN 2-sided.

LDOS KI/DVR & CAPS

Fm Ronald W. Wick: Dear Roy, When I installed the 5.3.1 for the Model III, I also installed the new KI/DVR, but after I was up and running, the computer would no longer respond to the SHIFT+0 for CAPS. So, I removed the new KI/DVR and installed the old KI/DVR again and everything works fine now.

I had hoped that someone would have said something and you would have brought it up in TMQ, but either no one has said anything or I am the only one having the problem.

Can you tell me if Powersoft is still in business? I wrote to them recently about their Bi-Tech multiplexor products that they took over, but I never received an answer. I wanted to buy another host adaptor.

I see that you took over the Breeze software line. Does that mean you will support the WD/DCT HD driver I have to use because of that multiplexor?

And one last question. Where on the Model I does the FORMAT program get the default density from? The Model I is attached through the multiplexor to my HD's and has one boot floppy, but sometimes I re-assign drive positions back to the floppy drives and floppy drive positions 1-3 will not format in double density no matter what I do. And putting the density in the command line won't do it either. Can you help me with this problem? By the way, those drive positions will read double density, they just won't format it.

Fm MISOSYS, Inc: Ronald, No one reported a problem with 5.3.1's KI/DVR as far as the CAPS function. It appears to work correctly here. Now if you have installed the 5.3.1 LDOS KI/DVR on a Model 4 computer, then SHIFT+0 will indeed not affect the CAPS mode as that is handled by the CAPS key on the Model 4 keyboard. I suspect that you are using a Model 4 in Model III mode. Let me know if that is not the case.

Powersoft closed down in November 1989. I have their WD drivers but not their Bi-Tech hardware. I'll try to handle problems with Powersoft software that is currently sold by MISOSYS; but I can do nothing for you on the Bi-Tech hardware.

Bit 6 of DCT+4 must be set for the DDEN question to be asked by FORMAT. Under the older LDOS 5.1.3, the DDEN bit was set in the DCT on all floppy-associated drive slots when the RDUBL or PDUBL driver was installed. But if you subsequently enabled other drive slots with the MOD1/DCT, the DDEN bit would not have been set as that was only done by the doubler driver - MOD1/DCT was not smart enough. Thus one of the undocumented functions added to 5.3.1 was a change in MOD1/DCT to set the DDEN bit if the FDUBL driver was installed - I added some "smarts" to it. With 5.3.1, you won't have that problem.

LDOS 5.3.1 & TED

Fm Clay S. Scott: Roy, While playing around with my brand new LDOS 5.3.1 disk that I just recently received from you, I discovered what appears to be a fault with TED/CMD, Ver 1.2.

When I press <CTRL -F> the message is "Load Filespec".

When I press <CTRL- L> the message is again "Load Filespec".

The best I can make out from the info sheet that came with LDOS 5.3.1 is that by pressing <CTRL-F> the message should be Filespec and not Load Filespec. Please advise me if this operation is normal or do I need a patch?

Fm MISOSYS, Inc: Clay, You have found a legitimate bug in TED. The ^L command should prompt with "Load filespec"? but the ^F command should prompt with just "filespec?". After checking out your report, I discovered that a single assembly language instruction had been omitted from the source code. It took a bit to squeeze it in with a patch, but I did it.

The TED5311 patch listed here should correct TED so that it follows its documentation.

```
. TED5311/FIX - Corrects
^F filespec prompt
. Apply via, PATCH
TED.UTILITY TED5311
D04,38=4F 7E 36 00 B7 21
B4 5B 20 0C 21 54 5B 0C
0D
F04,38=7E 36 00 B7 21 B4
5B 20 0D 21 54 5B 0C 0D
28
D04,4B=28 02 2E 4F
F04,4B=03 21 4F 5B
. Eop
```

LDOS 5.3.1 & KI/DVR

Fm Lloyd Evans: Roy, This is a very belated bug report on 5.3.1. I am having a house redone and it has been a nightmare.

TED does not display the 'Load Filespec' message as it should. When I compared it to 6.3.1 I found what I believe to be a missing instruction. I added the LD C,A instruction it seems to work fine.

I have put a 20 meg Miniscribe and a 720 k floppy in a 4P. When I tried to format the HD with the WD drivers it thought it was a 77 track 8" drive. FORMAT from 5.3.1 seemed to be the culprit because when I used FORMAT from 5.3.0 on the install disk it did ok. If this is not your problem I will understand.

The screen print option in the new KI/DVR does not work. I am not sure but I believe the type ahead buffer may not be right either. Since I must have both these options I am still using KI4.

If any of these problems have been corrected please use the enclosed SASE to send me the patches. Have a happy holidays and thank you again for your continued support.

Fm MISOSYS, Inc: Lloyd, I worked up a patch for TED about a month ago to answer another user's query [see above].

The DOS formatter does not format hard drives; at most, you can issue a FORMAT (SYSTEM) to add the directory information to a hard drive previously formatted at the low-level by a rigid formatter - if the rigid bit is set in the DCT to indicate a hard drive. Please be more specific as to what you are doing.

There is nothing wrong with the type-ahead in the new LDOS 5.3.1 KI/DVR. But you have stumbled onto a bug in screen print. When KI is installed on a Model 4, the SHIFT-DNARROW is no longer used for CTRL - the CTRL key is. This requires that the screen print module check for a ':' and not the shifted '*' character. Thus, Model III/4 LDOS 5.3.1 users need to put in the following patch.

```
. KI1/fix - Patch to MIII
LDOS 5.3.1 KI/DVR
. Apply via, PATCH KI/
DVR.DRIVER KI1
. Fixes access to screen
print on Model 4
X'5236'=CD 20 59
X'5920'=32 64 52 B7 C8 3E
3A 32 91 57 C9
. Eop
```

This changes the test if the installation is done on a Model 4.

LDOS 5.3.1 & FORMAT (SYSTEM)

Fm Lloyd Evans: Roy, Sorry than I was not clear on the problem I had with FORMAT. The problem is with FORMAT (SYSTEM) command. The WDFORMAT program calls this dos command and it is returning a bad number.

I formatted a 5 meg head with TRSHD5 and TRSFORM5 and got the free space map that I expected. One hundred and fifty three tracks with track one locked out. I then did a FORMAT (SYSTEM) command on this head. It reported that the directory would be placed on track 38. I think this number is the source of the problem.

The free space map on the 5 meg still showed the directory to be on track 76. Track one still showed to be locked out. On 5.3.0 this command would give me back track one. I should have looked at

track 38 to see if I had another directory there, but I did not. I hope this is enough information to track down the problem.

Fm MISOSYS, Inc: Lloyd, A feature that was added to FORMAT in the 5.3.1 release caused your problem. FORMAT now first tries to use the directory cylinder entry in the DCT when formatting hard drives. Unfortunately, with the addition of the "DIR=" parameter, the code does not work properly for re-using the DCT entry as the execution stream continues with the code which divides the initial directory value (which would normally be the maximum cylinder number) by two. What was happening wa the value selected for the directory wound up being half of the initial value - or one fourth of the maximum. After the format was complete, the BOOT record was updated but the DCT is not updated at that point - only a disk log operation would do that. The easiest solution is to eliminate re-use of the DCT value as the initial value; this would restore FORMAT to its previous condition yet still retain the DIR command. Here's patches for Model I and Model III versions:

```
Model I LDOS 5.3.1
PATCH FORMAT.UTILITY
(D02,82=06:F02,82=09)

Model III LDOS 5.3.1
PATCH FORMAT.UTILITY
(D02,35=06:F02,35=09)
```

Fm Ronald W. Wick: Roy, I have encountered a problem with the installation of the LDOS 5.3.1 for the Model I. The problem is when I try to format the hard drives from the Model I under 5.3.1 using WDFORMT/CMD. [remaining text describing problem in detail is deleted].

Also, how do you stop KI/DVR from starting up in lower case?

Fm MISOSYS, Inc: Ronald, Your FORMAT problem should be solved with the patches noted above.

You can't start KI/DVR in CAPS mode during installation; however, once installed, the status of CAPS is contained in the KFLAG$ which is part of the CONFIG.SYS file. If you SYSGEN with KI installed and in CAPS mode, when you re-boot your system, it will remain in CAPS mode until you change the mode.

Response to last issue

Fm Jonathan Armstrong, CANADA: Roy, I received TMQ VI.i yesterday, and I am impressed by the code examples included. I think I would have ordered DiskNotes 6.1 even if you hadn't withheld Mr. Slinkman's patches - I normally do not hesitate to key stuff in, but there's a lot of good stuff in this issue. Accordingly, please send me DiskNotes 6.1.

I appreciate your explanation of the reasons TMQ will not continue forever. I regret this, as I think the TRS -80 (in its final incarnation, the model 4D) is a sound design, and had the customer base justified it, it could continue to be upgraded, eg. with the Z280. Unfortunately the customer base isn't enough (and even if it were, modern machines would still vastly outperform the most upgraded TRS-80 imaginable). <Sigh>. An era is passing.

I suppose I wouldn't mind so much if the usurper weren't so awful. I refuse to buy 'Big Blue' on principle, and I have yet to feel the pangs of a Mac attack.

Graphics-90 coming

Fm Gary W. Shanafelt: I just got the latest issue of the *Quarterly*, and wanted to thank you for your response to my query about interrupt processing with LDOS. I tried what you suggested, and it works as you said it would. It took me a while to figure out that those two bytes which the DE register must point to need to be the address of the actual task subroutine, in MSB/LSB format. That done, the routine worked like a charm.

Of course, I didn't go to all this trouble just to make a letter blink in the corner of the screen, so you might want to know what I did expect to get out of it. As you'll see in the next issue or so of TRSTimes, I've worked over a new version of Software Affair's Graphics-90 package which should be fully compatible with LDOS (it was originally compatible only with my favorite operating system to hate, TRSDOS 1.3). Among other things, I've added a utility to print the Model III screen on a Laserjet or Deskjet printer, using a special soft font which I created for the purpose and which includes all the TRS-80 block graphics and special characters. To avoid the vagaries of the screen dump routine built into the ROM and thus unchangeable, my routine patches the keyboard driver so that CTRL-P dumps the screen. This works with any DOS. But the drawback is that it only works with programs which use the keyboard driver (many don't, as you well know); and further, it only works when those programs are actually looking for keyboard input.

When I was studying the LDOS manual, it struck me that if I could base my routine on the interrupts rather than the keyboard driver, it would be a lot more effective - for you could dump the screen anywhere in a program so long as the program didn't disable the interrupts. The routine would only work with LDOS, of course, but

that's the DOS any Model III program should be using anyway. Now that I know how to use the task processor, it should only be a question of time (the ultimate four-letter word!) until I can rewrite the routine to take advantage of it. This really is a powerful feature of LDOS, and I'm surprised that so few programs seem use it.

Oh, to answer your query about new DOS manuals: I, for one, would be interested, not just so that all the revisions would be incorporated in the same place but because my current manuals are falling apart from overuse!

Thanks again for the assistance; your expertise continues to make my computer not just useful, but fun!

Return Policy, etc.

Fm Daryl L. Biser: Roy, In the flyer you included with my last order, I noticed your "no-questions-asked" return policy for software. I am really grateful that you have such a generous policy, because I realized once I received my copy of LDOS 5.3 that it is much more than I really need. I use my Model 4 in III mode only to run some BASIC programs I wrote a number of years ago. A couple of them manipulate a lot of data, and are fairly large, so the loss of approximately 1K of memory in LDOS 5.3 BASIC is fatal. In fact, I have not converted them to Model 4 BASIC because of the memory constraint in LSDOS 6.3. No, I haven't tried chaining segments of the programs, although it would probably be the way to go. I did try chaining with Model 4 ZBASIC, but the program wouldn't pass all the variables I was using. I am sure a more clever programmer would have succeeded.

Your package looks extremely impressive, but I simply won't use it. I would, however, like to try your latest version of Little Brother. The only database I have is DeskMate, and that leaves a lot to be desired. I use AppleWorks at school, and need a database with similar power and simplicity for my Model 4. Please give me credit for the returned LDOS 5.3 and the Manual toward the purchase of LB Data Manager 2.1.

One more thing while I am writing. On page 11 of this month's Computer News 80, I noticed under HARDWARE INTERFACE KIT mention of SET2RAM/CMD which uses @EXMEM and @BANK to use extended memory with BASIC in LDOS 5.3. If I read this correctly, this works in Model 4 mode as well. If true, and it really allows BASIC to use extended memory, it would solve my conversion problems. Please send me any information you have about the interface kit, if it really does allow basic to use the additional memory. (I have the 128K upgrade, and a MegaMem board with 1 Meg installed.) I know I could use a RAMDISK, but that isn't what I have in mind - I want the extra memory to be there when I boot up BASIC. If I am confused, not to worry. It is my normal state of mind.

I will appreciate any help you can give me. And while I am thinking of it, please extend my subscription to *The MISOSYS Quarterly*. I really look forward to receiving it, and I read every word, even though, as you can probably tell from the last paragraph, I don't understand all of it (a lot of it?). I really regret that I never had the time to learn assembly programming. And thanks for your continued support of our old TRS-80 computers. I really love my old 4P and want to keep it going as long as possible.

Fm MISOSYS, Inc: Daryl, Our policy is that any software product - except those sold on closeout - may be returned within 30-days of purchase for a full refund excluding and S&H charges; products must be in "like-new" condition. I believe that policy best serves the customer by removing all doubt as to whether or not a particular product will serve their needs. I'm glad to accept your return - especially when its used to purchase LB!

Let's clarify the issue with the Hardware Interface Kit. A number of years ago, I published an article in The MISOSYS Quarterly which covered the design and installation of an extended memory manager, called @EXMEM. The memory manager provided the ability of accessing extended memory - through the @BANK service call - from any program. The manager handled the program stack and allowed single-byte I/O and page (256-byte) I/O. With the USR11 facility in LSDOS 6.3.x BASIC, a Model 4 user could easily access the extended memory; a few articles appeared in The MISOSYS Quarterly which provided concrete examples of how to accomplish this.

Model III mode does not have any inherent bank-switching capabilities. Some time ago, MISOSYS released a product called the "hardware interface kit" which provided an @BANK and @EXMEM service call when LDOS 5.3 was in use on a Model 4 computer. It achieved this by switching the machine into a RAM Model III mode, populating the RAM with an image of the ROM. It then replaced some of the cassette I/O routines in what was the ROMC area with code to handle the @BANK and @EXMEM functions. This, of course, required a 128-K Model 4.

David Goben took my idea a step further and re-wrote the Model III memory manager to support the XLR8er board. This package - noted in CN-80 - was provided to MISOSYS and I released it as the "XLR8er hardware interface kit". This product requires a Model 4 equipped with an XLR8er board. Thus, when installed, it allows programs to access extended memory via both byte I/O and page I/O. But note that the program has to be able to handle access of the new service calls. LDOS BASIC does not have any USR11 capability; therefore, any use made of the new service calls from Model III BASIC would have to be implemented via some additional USR routine written in assem-

bly language to provide an interface between BASIC and the service call. Although it may sound like a useful article for TMQ, at this point in time I believe I have sold about three copies of XHIK. Does that sound like I should spend time on that?

LBASIC4

Fm Brad Stiles: Roy, Thanks for the recent updates of LDOS and LB. In general, I feel that my Model 4's are pretty well taking care of my needs. That is in large part due to MISOSYS products and your continued support of those products. The only areas where I see the Model 4 suffer, re., the MS-DOS machines is in regards to accounting and investing software. I continue to muddle along in those areas with software written in BASIC for the Model 3 mode. It is fortunate that these products were written in BASIC, since I have had to rewrite portions of each, to get them to run on LDOS and allow the use of the normal LDOS system features (PR/FLT, KI/DVR, etc.).

I would really prefer to be able to run these programs in the Model 4 mode, but the conversions are too time consuming. Thank you for LDOS 5.3.1!

But now, you raise the possibility of an LBASIC for the Model 4 mode [Article "LDOS 5.3.1 Released" (TMQ Vol. Vi.i, P .4)]. Boy would that be convenient! If the demand is sufficient to justify your effort, please put me down for a definite PURCHASE! With LBASIC4 in hand, I would also require DoubleDuty. Just imagine? VersaLedger on one bank, MultiPlan on the other and ProWam available as well.

Fm John P. Jones: In answer to TMQ Vi.i page 4, yes, I would be interested in running Model 3 LBASIC directly in the model 4 mode. Use would be more limited if the interpreter requires too much memory - one reason I still use mostly the 3-mode. Would sacrifice speed for storage if overlays are needed.

Fm Henry H. Herrdegen , Canada: Roy, I've just read Lance's news flash in the Nov/Dec TRSTimes, that you may be developing a III/4 BASIC. He suggests to let you know what we, out there in dinosaur land think about it. For what it's worth, here it is from me.

I do a bit of Programming in BASIC, no assembly language, both in III and 4 mode, having gone from a Model III (83) over a 4 (89] to a 4D (90). If You read the Computer News 80, You may remember seeing my name. I do not run LDOS, but a patched (mostly with Andy Levinson's,) TRSDOS 1.3, NEWDOS80 and LS-DOS 6.3.1. You may even be familiar with my "PUP" patch disk, sold by CN80, and know what my TRSDOS 1.3.3 looks like. So much for an introduction.

I found the original TRSDOS BASIC a bit restricted in many ways, and got into the habit of programming in NEWDOS, mainly for the so convenient way to move lines around, duplicate them, renumber/move blocks and the references printout. Renumbering, "NAME", in TRSDOS BASIC is the pits, only all the way to the end, and the spaces added at the GOTOs and GOSUBs!

The NEWDOS DOS has, in my opinion, too many unnecessary wrinkles, bells and whistles, which the average user does not need, just creating confusion as to what SYSTEM/PDRIVE is on the disk. Hard to keep some sort of uniformity in your disk file. So I stuck with the patched TRSDOS for running the programs written in NEWDOS. I am a bit mad at people who dismiss TRSDOS out of hand as ancient, "run on occasion for some unknown reason", and so on. Granted, TRSDOS 6 is not worth the disk it's on. Thank You for LS-DOS, but there is nothing wrong with a patched 1.3, with DIR and LIST paginating, LIST in ASCII, no superfluous TIME entry, use the convenient period key instead of slash, choice of cursor, CAT, error messages, and all the other goodies from patches Andy and others have given to us.

LS-DOS BASIC has many of the good NEWDOS features, but lacks a few others and some good ones from the TRSDOS BASIC. I hope you can and will combine these, and come up with a SUPER BASIC: (S-BASIC?)! Taking LS-DOS BASIC as datum, it would be nice to add (but nothing existing taken out):

REM (hardcopy) feature from NEWDOS. BREF is a pain in the *#%&, not being accessible from BASIC during program writing.

Require a "D" before a line number to delete it! As in NEWDOS. Who hasn't punched <ENTER> after a line number, and then had to re-construct it"

CMD"F" in NEWDOS has some overkill, but also some good uses, like the F=SS, single stepping through a program could help, as may the F=POPx

CMD"I" is already there as the (too long) SYSTEM "dos cmd", better than the non-returning CMD"I", but please keep the (undocumented?) "!" substitute for the 6-letter command word!

CMD"J": does anybody use "ddd/yy"? Besides, it does not get the year anyway, and the necessary "-yy" input is a bit far out. But a conversion from the (outmoded) mm/dd/yy to the more sensible (and I.S.O. standard) yy/mm/dd could he helpful and avoid some string handling, sorting by date, etc.

CMD"O": maybe with the added NEWDOS features, multi array sorting, etc?

CMD"'P ": it would have to distinguish between III and 4 'mode. (PEEK (14312)

versus INP (248))

CMD"Z" (or CMD"*") very useful in many circumstances, rather then lots of duplication with LPRINTs, and better, even if not as universal (keyboard & RS-232 needed?), then CMD"ROUTE .....

Peek & poke the video screen enabled me to get an alphabetized CAT for the III, sorted in columns, not lines, as is the 4 CAT. Is there a way to read the screen in LS-DOS?

Could "F" (or "S" or REM) include the line number after a GOTO, GOSUB, or @? Would be nice for de-bugging, to find a specific PRINT@ or GOTO!

Spaces between keywords? Will "LINE INPUT" in an "old" 4 program run as well as a "LINEINPUT" from an "old" III?

Make THEN optional before GOSUB as it is for GOTO. As well as retaining THEN 1200 without the GOTO! And who needs or still uses LET?

Maybe an improved TRON/TROFF system is possible? Stop when waiting for an input, instead of filling the screen with the same line number?

And please consider the TRSDOS 1.3 BASIC, not just LBASIC. Judging from the number of people which have bought my 1.3 patch disk, many Model III users do still work with the 1.3, and anything new for the TRS bunch should not exclude a fair number of them!

I have worked for several months with MS-DOS's BASIC, and must say that I like our editing features better. It is nice to see a block of code on the screen and chase the cursor around, but I still prefer the simplicity of our commands and scrolling. A version against something new, and comfort with the well known? I don't know.

Fm MISOSYS, Inc: Henry, You pose lots of interesting features, but I believe that's diverse of the point. When I originally noted the concept, the BASIC in my mind was a version of the Model III LDOS BASIC running in the native Model 4 mode which would run Model III BASIC programs. I have begun to work on that effort as time permits; however, I do not envision adding features to "LBASIC". If anything, I would expect to eliminate features. One difficulty in such an implementation is to ensure that the memory space available to user programs is greater than that available from the native Model III mode. Model III BASIC takes up about 20K of memory space; that's about 11 K in the ROMs and 9K in Disk BASIC extensions. In Model 4 mode, such a BASIC would load at 2600H. Without reserving space unused by BASIC for the 3C00H-3FFFH video RAM image, that would leave just about the same amount of user memory available to BASIC programs as that available in Model 4 BASIC. I would be able to save space by reducing the double jump interface between "ROM" BASIC and disk BASIC, and would be able to optimize the total program eliminating, for instance, the use of "RAM" for division and port I/O. I would also eliminate all cassette routines (i.e. CSAVE, CLOAD, SYSTEM), perhaps 2-letter error abbreviations, etc. I have also considered two distinct BASICs: one without a programming interface (i.e. listing, trace, editing, etc) which would enable additional user memory during runtime; and one adding the programming capabilities. The most significant feature which could be added to an LBASIC4 would be the availability of additional user memory space.

Building LBASIC4 is a tremendous editing job; also, there is a quite intricate Model 4 memory mode which would be required to enable the video RAM to be active at the Model III memory address so existing BASIC programs which peek and poke the video would continue to function. There is also a question in my mind about which Model III addresses typically peeked and poked to support (i.e. Model III-mode DCB data?). If I do pull off such a feat as LBASIC4, my philosophy would be to adhere to LDOS's LBASIC application program interface - not NEWDOS's.

As far as peeking and poking the video screen, I published "DEMOSVC/BAS" in The MISOSYS Quarterly issue IV.i (Fall 1989). That demo illustrated a program for using LS-DOS 6.3's USR11 function to read the video screen using the @VDCTL DOS service call. That issue is still available for $4 +S&H (minimum order of $15).

XBE & Prosoft's Trashman

Fm John P. Jones, 17 Bel Manor Drive, Fairmont, WV 26554: Roy, I would like to do more programming in the Model 4 mode. However, I usually find myself using the Model 3 mode because of the availability of two utility programs:

1. Computer Applications Unlimited's "Full Screen Text Editor for the Model 1 and 3" (Reviewed as "XBE" in "80 Microcomputing", Nov. 1981, page 42, copy attached - the version I use is XEDIT48/CMD).

2. Prosoft's "Trashman"

The former allows switching from BASIC operating mode to editor in a couple of keystrokes. I find this a must for testing first the syntax and then the operation of [complex] lines.

I'm afraid I'm incapable of programming with a word processor because of the problem not being able to test as I go. The "find" mode is also a must, both for variables and finding what line contains the "NetPay" printout or the "GOSUB 2000". Note that XEDIT converts tokens to keywords, the same as the LIST command from BASIC.

"Trashman" is useful not only for handling string arrays (not really a problem on the Model 4), but for avoiding the long delays (1 min, 12 sec.) on data entry. I'm afraid this would be intolerable in our business. Adding frequent X= FRE(AS) (as suggested in "Model 4 by Chris") just seems to slow things down to me.

Can you advise me as to whether you know if such software is available for the Model 4. POWERSOFT was incorporated into MISOSYS, but I doubt that PROSOFT is related. I assume Computer Applications Unlimited is out of business, but I don't know if they did a Model 4 version of XEDIT before quitting. And other companies may have had similar software, or there may be something in the Public Domain.

Fm MISOSYS, Inc: Anyone with information on Model 4 equivalents may contact John directly.

CLAN genealogy system

Fm Patrick Hamel: Roy, Well, I successfully installed the RSHARD program and LDOS after a full-day learning experience, the cable works fine. EVERYTHING WORKS BUT THE PRIMARY OBJECTIVE.

I know I'll probably have to pay for the advice, but perhaps with the enclosed symptoms it won't cost too much to have you find out why the "CLAN genealogy system" will work on floppies with LDOS 5.3.0 and won't work with LDOS 5.3.0 or 5.3.1 on a hard disk.

I used the trace - and the system dies at the initialization" portion of the "menu program". The last address shown (before the screen goes blank for about 5 minutes and then reboots asking for "date") is 20010 which is where the listing on the attached page starts. Apparently the CLAN programs communicate. between themselves by using the memory area that this portion of the "menu" program reserves.

My only hope is that you will have enough demand for the solution to the "CLAN on a hard disk problem" to make my cost minimal. I am enclosing an SASE for your reply.

```
20010 POKE16561,253:POKE16562,255
'Protect memory for 'beyond start' flag
20020 CLEAR 5000: DEFINTA-Z:DIMVD(511),NA$(10,1),LA$(10,2)
```

Fm MISOSYS, Inc: Patrick, The program fragment you supplied provides the entire clue to your dilemma. Notice the first line: POKE16561,253:POKE 16562,255. You really should understand what that is doing, in light of the comment included therein. The values 16561 and 16562 translated to a hexadecimal address indicate that the BASIC program is poking a value into the pointer BASIC uses for indicating the last byte usable by BASIC - 40B1H-40B2H. The value the program is poking is 0FFFDH. As such, that program expects to have all of upper memory free from use when BASIC is started. Some other portion of that program is going to poke into the memory address locations 0FFFEH and 0FFFFH. Once you install a hard disk driver into the top of memory, that program will poke its data into your hard disk driver. The result is crash city.

The solution to the problem is (1) lower the DOS high memory pointer prior to installing the hard disk driver; and (2) change the BASIC program so if the BASIC high pointer is below that address, then don't poke anything into it.

Starting from a floppy-only system disk with MEMORY showing X'FFFF', you can install your hard disk driver after you lower HIGH$ via the command, MEMORY (HIGH=X'FFFD'). You can then SYSTEM (SYSGEN,DRIVE=d) that system disk and boot from it when you want to run CLAN.

Why must you also modify the program? Good question, and here's why. If you performed the first step, HIGH$ after loading the hard disk driver would probably be around X'FEA0'. Let's use that for this discussion. When you invoke BASIC, BASIC peeks the HIGH$ value and uses the lesser of HIGH$ or the MEM parameter entry as the last memory address to be used by BASIC, storing that value into 40B1H-40B2H. BASIC then reserves some memory from that point downward for string storage space. If you didn't alter that CLAN program, BASIC would initialize the pointer to X'FEA0'. Now the program would start, poking a new higher value into the pointer location. At some point in running the program, BASIC would probably overwrite the hard disk driver with strings. That would then cause a crash when access of the hard disk was attempted. Therefore, the program needs to be modified to PEEK the pointer then update it only if its initial value is X'FFFF' So change the program by adding something akin to: IF (PEEK(16561) <> 255) OR (PEEK(16652) <> 255) THEN 20020 ELSE POKE 166561,253 'Protect... It is not necessary to POKE into 16562 at that point because it already contains a 255. Incidentally, line 20070 with its PEEK(-2) is obtaining the value at X'FFFE'.

1000 TL/2 Hard drive

Fm Dave Busenbark: I need help with a hard drive upgrade question. I have a model 1000 TL/2 without a hard drive. A friend has a seagate 251 20mb drive and controller that he will give me but I don't know if it will work in my computer or not. Any comments will be appreciated.

Fm MISOSYS, Inc: Depends on the controller. To begin with, the 251 is a 1/2 ht 5.25" 40 Meg drive. Your 1000 TL/2 needs an XT 8-bit controller or you need

an 8-bit IDE drive.

Fm Fred Oberding : Dave, the ST-251 will work fine in the TL/2. The controller board needs to be an 8 bit unit (not 16 bit) and also less then 10.5 inches long

Tandy 1000RL-HD

Fm Frank Poirrier [71435,1363]: HELP! I have a 1000RL-HD that is soon to become a permanent part of the landscape. I have had it with this thing. Since 1978 I have been hacking with all of the toys from Tandyland; Model 1, 2, 3 , 4, 100, CoCo and never have I come across anything as UN-hackable as the 1000RL. All I want to do is connect an external 360K 5.25 floppy to it. I made a cable and was able to make it work but only by itself. It will operate only with the internal 3.5 disconnected and the internal 3.5 will only operate with the external disconnected. The cable connects the motherboard to the internal 3.5 pin to pin. It is then routed externally to a 34 pin edge card plug with pins 1, 2, 3, 4, 5, 6, 7, 9, and 11 removed to eliminate the 5VDC power supplied from the 1000RL. The external drive has its own supply.

The problem comes in with drive select and motor on signals. For some reason Drive A and B on the 1000RL are the same. There is NO WAY to separate them. TandyDOS has a feature called SETUPRL. I dug through all the supposed DOS books and found out about DRIVER.SYS and ASSIGN but still cannot separate A&B drives. I am offering a REWARD to anyone who can give me a solution other than buying a Pacific Rim Drive (that's the one that plugs into any parallel printer port). This 1000RL is only an interim machine until I can afford a real computer. I need the advantage of a 5.25" drive to utilize HYPERCROSS and TRSCROSS, in my transition to the world of MSDOS from LDOS.

TRSCROSS

Fm A. C. Mullen: I purchased your TRSCROSS software program from a Radio Shack store at Greenbriar Mall in Chesapeake, VA. last Saturday.

I have attempted to use it. Trying for several hours to follow the instructions in the manual provided. Enclosed is an example of the success I have had. In addition to this unsatisfactory output the program has reformatted one of my trs80 disks without my selecting that function on the menu.

I have a Packard-Bell legend IV computer with a 5 1/4" and a 3 1/2" floppy drive and a 40mb hard drive. I made a backup of your disk on a 3 1/2" disk and I copied your disk onto the hard drive. I copied my 5 1/4" disk onto my hard drive using item 1 from the main menu. It copied without presenting an opportunity to take any action on the "select screen" as described in last pard page 9 of your manual. I printed the enclosed by using the print command from dos.

Please check your disk which I am returning to you to see if contains the TRSCROSS as designed. If it does please advise me what I am doing incorrectly to bypass the select screen and thus prevent its use. If it does please advise me if the enclosed output is the best I can expect from this program.

Fm MISOSYS, Inc: You don't provide sufficient information for me to pinpoint your problem; however, if I may make a guess, it appears that you may be pressing [ENTER] after typing the main menu command. When you invoke TRSCROSS, the main menu, which displays the six options, appears. On page 8, the User Manual states simply press 1-6 or the corresponding function key. You don't press [ENTER] - just a single keystroke.

When you enter the "1" or F1, or simply [ENTER] to select the default of function "1", TRSCROSS will then prompt you for the *number* of the drive containing the TRS-80-type diskette. If you had pressed "1" followed by [ENTER] at the main menu, the "[ENTER]" would have responded to this question before it had been asked.

It takes an [F3] at the menu on page 13 to instruct TRSCROSS to begin copying; thus, I can't explain why for you it immediately copied. Did you by chance invoke TRSCROSS with command line parameters to have it operate without menu operation? For instance, if you had entered **TRSCROSS /1**, the program would have automatically copied all files from drive :0 (the a: drive) to the DOS current directory. That would match the operation you described.

Finally, it appears that the file you copied may have been a SuperScripsit document. If that were so, then you need to invoke TRSCROSS using the /RF option and designate all SuperScripsit files as a mode of "SS" (using F10) to generate DCA-RFT files, or without the /RF option but still designate all SuperScripsit files as a mode of "SS" (using F10) to generate ASCII files.

Please write with specific details if you still have problems.

More TRSCROSS

Fm Ernest G. Robinson: Using MS-DOS 4.01, I could not use DOS DISKCOPY command to backup the original disk. I received an error message (see enclosed Screen Print), but using XCOPY A: B: command, I got a copy (5 1/4 to 3 1/2). I used COPY command to copy the files to the Hard Disk.

I like to program in BASIC language, so I had many programs made with my Radio Shack MOD 4 computer that I wanted to transfer to the MS-DOS Packard Bell computer - with out re -typing them in. Your software TRSCROSS done the job!

The program works well. I transferred many BASIC programs that were made in both the MOD 4 mode (TRSDOS) and MOD 3 mode (DOSPLUS). BASIC programs made in MOD 3 (DOSPLUS) had to be saved on TRSDOS or LS-DOS disk, then transferred with TRSCROSS.

I also transferred many data files made with BASIC programs. The data saved while using a database software named AIDSPLUS-II (MOD 3 mode - LDOS System), caused some problems. Unless they had been saved "vertically", so they could be read by a BASIC program, I had to add delimiter fields of "*" between the existing fields. I then made related database fields with MS-DOS LOTUSWORKS database to accept data "TRSCROSS'd" from AIDSPLUS II of the R/S computer.

SuperScripsit documents MOD 4 mode (TRSDOS) transferred well. SCRIPSIT documents MOD 3 mod (LDOS) transferred almost as well, but LeScript documents didn't do as well. They transferred with much scattered "garbage", but could be salvaged.

Nice work, gentlemen/ladies!

Fm MISOSYS, Inc: Ernest, Our TRSCROSS product is distributed on a disk formatted one sided, eight sectors per track. That's the lowest common denominator disk structure. All versions of MS-DOS must be able to read that kind of disk. However, it's possible that DISKCOPY of MS-DOS 4.01 won't copy it. All you need to do is to copy the files to another disk. It was not necessary to use XCOPY since there is no subdirectory on the TRSCROSS disk. But XCOPY won't hurt anything.

The problem you had with Lescript documents was really not a *problem* per se, but the result of the files not being ASCII and TRSCROSS performing no conversion of the imbedded formatting controls; that was your *scattered* garbage. I am glad you found all other aspects of the conversion worked well.

Still More TRSCROSS

Fm Max F. Homfeld: I bought TRSCROSS when I bought my IBM clone, an Emerson 8200. I use it to transfer text and data files and basic programs from my Radio Shack model 4.

A constant problem is that it seems to read wrong sectors at times. In text, I get several sentences repeated while others are missing. In basic programs, I find 4 or 5 lines missing, sometimes in two places. Every file seems to need repair. What causes this? What can be done?

I always get the "abort or ignore" statement, but the document warned me of that.

Fm MISOSYS, Inc: Max, Typically when most files converted by TRSCROSS have some data missing and some superfluous data showing up, the reason is the inability to properly read the TRS-80 diskette. You highlight this problem when you stated you always get the "*abort or ignore*" message.

My hunch is that the diskettes you are trying to read were formatted with a version of TRSDOS 6 earlier than the 6.02.01 release. If that is the case, you are somewhat out of luck unless you still have access to your TRS-80. Your letter implies that you still use your TRS-80. If you do, the easiest cure is to format a TRS-80-type disk on your PC using TRSCROSS's menu option 3 - Format TRS-80 disk. Take that disk to your TRS-80 computer and copy files to it that you wish to move to your PC.

Another fix is to obtain the latest release of the TRS-80 DOS for your computer and recopy your current disks to ones formatted by the new LS-DOS. That's the LS-DOS 6.3.1 release available from Radio Shack Consumer Mail, part number 700-2297, or directly from us using part number M-11-043. Our price is $39.95 + $4S&H.

If you are using TRSDOS 6.2.1 or newer, then the other possible reason could be a significant alignment difference between the two disk drives (the TRS-80 drive and the PC drive). Again, it's possible that the first cure mentioned may work.

Fm Max F. Homfeld: The advice in your November 8 letter was excellent. I've had no errors when the disk is formatted with TRSCROSS. And no "abort or ignore". My TRSDOS is 6.02.00. We have two Model 4's; one has a 5 meg hard disk.

TRSCROSS & pfs-FILE

Fm Robert E. Jinks: Roy, Before I get to the purpose of this letter I want to tell you how much I appreciated your commitment to the TRS -80 Model 4 community and how much I enjoyed your *"THE MISOSYS QUARTERLY"* and products. It was with some trepidation that I upgraded to an IBM compatible computer. The company that I worked for, until retirement, used the PC-compatibles and to maintain compatibility between home and office I decided to take the leap.

Thank goodness for TRSCROSS, without it I doubt that I would ever have converted my Model 4 files to the PC. The only problem that I have encountered is converting my PFS:file data to the LotusWorks data base. I accomplished this by writing reports to disk then converting them, through TRSCROSS, to the PC as ASCII files and editing them to add delimiters and remove some garbage. Some data was lost and there seems to have been some duplication, still it was much better than reentering all data from scratch.

M.A.D. ROM and internal HD, etc.

Fm MISOSYS, Inc: The following letter exchange is between a TRS-80 user and M.A.D. Software. Ordinarily, this would not appear in TMQ; however, the information provided by M.A.D. is sufficiantly relevant to TMQ readership that I have taken the liberty of providing the exchange to my readers.

Fm John Cerul to M. A. D. Software: Back in August I purchased, from you, a BOOT-ROM S/N SC00715-23010-36 and HBUILD6 S/N ZE111 - 2(13) to install in my Model 4P. This 4P is one of the non-gate models that has been somewhat modified. i.e. Anitek Hypermem (1 mb.), Anitek speedup kit to 6.1 megahertz and a MISOSYS Hard-drive kit installed internally, with a Tandy 3 1/2" floppy as drive :0. Additionnally I have replaced the original Kalok 320 HD with a Seagate 251-1.

This makes a nice installation and I am pleased to say that with only one exception your BOOT-ROM is a very desireable addition. Not your fault though. The "problem" comes about because of expectations raised by comments made by Fred Oberding in his review in TMQ V.i.

—"7)Hard disk booting will wait for slower drives to come on-line. This is useful if you expect your system to power up unattended."—

My observation is that this wait will never take place with the ROM as presently coded. Therefore my workaround is to keep a boot disk, with a system and the following auto'd JCL in physical drive :0.

```
.
//delay 25
BOOT
//STOP
```

With no disk in drive :0 upon power up, boot from the floppy is attempted and we get the "Close the floppy door and try again" error message. If a forced HD boot is attempted (F1), we get the "The hard disk drive is not ready" error message. Using the RESET button after the HD has come up tp speed, of course will cause a normal HD boot

I don't have another 4P with an "external" HD that I can check against to see if the "problem" is installation specific or not. From the information I have to work with, I think not. I believe that if any combination is powered up from the same power switch — Harddisk and the 4 or 4p the results will be the same — no boot from harddisk. Power up "must" be Harddisk first then when the disk is up to speed power up the computer. This of course without a boot delay programmed into the ROM or using a boot floppy with a delay as above.

I would like your comments, and ask that you forward a copy of them to MISOSYS, for possible inclusion in TMQ. I am forwarding a copy of this letter to Roy also.

Fm M.A.D. Software, P. O. Box 331323, Ft. Worth, Texas 76163: We regret you are having difficulty in using the 4P-BOOT ROM with your system. Since your configuration is considerably different from a stock Model 4P system, it has taken us time to configure a system that is similar to yours to attempt to reproduce the problem you encountered.

Let me say up front that the 4P-ROM is designed to delay long enough for a hard disk drive to come up to speed from a stopped state and boot correctly, even with an XLR8er adapter installed.

For our testing, we set up configurations with 4P gate- and non-gate-array systems (same ROM), both types of MISOSYS SCSI controllers, three different hard disk drives (TM503, TM602 and ST4096), and with and without an XLR8er adapter. We do not have an Anitek system, nor do we have a Kalok 320 drive.

In a nutshell, we were unable to reproduce your problem. testing confirmed that the 4P ROM will wait 3 seconds at 4MHz for the disk controller to receive power (this is done in case you use a relay-controlled power strip that applies power to the CPU first), then the hard disk is given about 25 seconds to become ready. The timeout was checked by physically preventing the spindle on a drive from turning (or holding RPMs below 3600) and waiting for the ROM to time-out. Another test applied power to the CPU, interface and controller, but not to the drive. We also tested the devices on a standard power strip (all devices receive power at the same time). Over several dozen cycles, each allowing the drives to come to a complete stop, no failed boots were seen.

We do have a couple of suggestions. First, you describe your system as having the hard disk, SCSI interface card and SCSI/MFM controller inside the Model 4P. With the standard power supply, you are certainly exceeding, the power capacity of the power supply, and one or more voltages may be out of spec during the first few minutes of operation. As mentioned earlier, if the controller is not energized in three seconds, the ROM will assume you do not have a hard disk controller. This can be confirmed by temporarily connecting SCSI controller, drive and interface card to a second power supply. Note that both of our machines were incapable of operating the interface card, controller and drive from the 4P power supply. We used an external 65 watt power supply during our testing.

If your 4P has a Tandy power supply, you might try replacing it with an Astec, preferably a larger one. The original 4P with two floppy drives consumed nearly 60 watts, and the power supply is only rated at 65 watts with forced air. Despite using the small hard disk drive, it plus the SCSI controller are consuming 25 watts or more, putting the power supply well out of spec. You might measure the +12 and +5 at the controller when the system is first turned on to verify that +5 is not below +4.85 and +12 is not below 11.5. These are the minimum operating voltages of the main logic board; your drive and controller may require voltages even closer to +5 and +12 for proper operation.

If the "Drive not ready" message does not appear for at least 25 seconds, the drive is taking too long to become ready, and this may be caused by the drive performing lengthy self-tests during power-on. Since most drives cater to the IBM PC market, and most PCs busy themselves when first turned on with memory tests, the drive would have sufficient time to perform its own tests. If this is being done by the drive, you might check to see if these tests can be disabled.

Finally, you are running a speed-up modification. Without having one to test, it is possible that this device runs the machine at 6MHz all the time including during the boot sequence. (The XLR8er adapter does not increase its speed until the operating system is Loaded.) With the CPU running 50% faster than normal, the 3 second delay for the disk controller would be cut to about 2 seconds, and the 25 second motor delay would be reduced to about 18 seconds. These values may be insufficient for your controller and drive. If possible, try forcing the system to run at its normal 4 MHz speed (or less) during the boot period.

Your second question has to do with the replacement of the 5 1/4" floppy drive with a 3 1/2" unit. When there is no floppy in the 3 1/2" drive, you get the message "Close the floppy drive door and try again". With a 5 1/4" floppy drive, this message only appears when a floppy is in the drive and the door is open, or the floppy is inserted backwards.

What you are seeing is perfectly normal for the 3 1/2" drives since they do not return the same status as a 5 1/4" drive when the drive is empty. The original Model 4P boot ROM also displays the "Close the door" message when a 3 1/2" drive is used. The reason for the difference has to do with the drive mechanics. On a 5 1/4" drive, the index hole is detected optically, and the write- protect is detected optically or with a switch. An empty drive "sees" index at all times and does not "see" a write-protect tab. A ready drive may or may not see a write-protect tab, but it will see the index signal pulsing on each revolution of the diskette. A drive with a diskette inserted and the door open never sees index (the disk isn't rotating. The CPU may also see a write- protect tab.

On a 3 1/2" drive, things are different. Index is generated by a sensor located in the motor assembly and is only passed to the CPU when the diskette-loaded switch is pressed (at the front of the drive). When there is no diskette in the drive or if it is in the ejected position, the CPU sees no index signal and a write-protect tab. Subsequently, the 4P ROM code mistakes this condition for a partially inserted floppy.

Inside TMQ

pfsFILE Data Structure

by Roy Soltoff
MISOSYS, Inc.

During the last quarter of 1991, I occupied most of my programming time with a data base conversion program project. The culmination of this effort was the LBCONV utility currently being provided as part of my version 2.2.0 LB Data Manager. There are two classes of potential users of LB; one is those folks who need to computerize their collections of data and who do not already own a data base manager, and those who are already using an existing - but inferior - data manager program. LB version 2.2 incorporates still more features than the previous release (2.1.0) in an attempt at attracting those in the former class. However, when you are already using a particular program - be it a data base manager or a word processor - you have a general reluctance to convert to another product which uses a different data structure; you want to easily re-use your existing data.

My intent with LBCONV was to eliminate, from existing data base program users, virtually all resistance from *wanting to switch*. To accomplish that, I had to support conversion from the popular existing data bases. There are versions of LB available for both the TRS-80 Model 4 and MS-DOS.

Under the Model 4, the other two popular data base programs besides LB are Profile 4 and pfs-FILE; both of these were sold by Tandy. dBASE II was also available for CP/M. Some users may have used a spreadsheet program to store data; data porting from spreadsheets is usually handled by the Data Interchange Format (DIF) originated by Software Arts - the originator of Visicalc. Finally, other users may be utilizing a BASIC program with some form of direct-accessed file in a fixed record format (i.e. fielded data elements in a fixed record length file). LBCONV can handle all of these formats; the latter being described by the operator in a manner similar to that used in our DSM4 sort utility.

In the MS-DOS world, the pseudo-standard of data base interchange is dBASE. Virtually all data base programs either use a dBASE format directly as their normal data file structure or provide a dBASE format for data interchange. There are a few dBASE formats in use. The oldest and most restrictive is dBASE II. It is a very primitive format allowing few fields and few field types. dBASE II was followed by dBASE III, an expanded format which provided additional field types, a greater number of fields, and a larger maximum number of records. LBCONV supports conversion to and from dBASE II, dBASE III, and DIF data structures so that an operator may port data to or from LB and dBASE formats, or LB and DIF.

Some programs can only import tab or comma delimited data. LB version 2.1 can easily generate such a data file by means of a user-prepared print screen with the output report directed to a disk file. On the other hand, I included the direct generation of tab and comma delimited output in LBCONV for ease of use.

A PRO-WAM Help Displayer

by
Matthew Kent Reed
Box 368
West Lebanon, NH 03784

One very useful (and often overlooked) feature of LS-DOS is its help system. With HELP/CMD and the supplied help file for DOS, it becomes much easier to remember the syntax of DOS commands. You can also use a help file generator to write your own help files for other programs. However, since HELP/CMD only runs from the DOS command line, help files cannot be viewed while you're using other programs. It would be very useful to be able to look at the help entry for DEBUG while you are using DEBUG, but HELP/CMD does not have this capability.

I wrote HELP/APP to solve this problem. It performs similarly to HELP/CMD, and it uses the same help files, but HELP/APP is a PRO-WAM application. This means that help is now available from within almost any program. Also, because HELP/APP is a PRO-WAM application, you can import help topics and export most help text. (Due to PRO-WAM's design, help text in reverse video can't be exported.)

Assemble HELP/APP as a core image file (use the -GC and -CI switches with MRAS)

```
*GET EQUATES
*GET MACROS
;-------------------------------
; Help File Displayer, copyright (c) 1989, by Matthew Reed
; Displays help file directory and lists help topics
;-------------------------------
@WAM EQU   7CH          ; PRO-WAM SVC number
     ORG   2700H        ; Program starts at 2800H
;-------------------------------
; Information sector
;-------------------------------
     DB    'PROWAM'
     DB    'Help display',03H
     DC    .HIGH.$<8-$+256-56,0
     DB    'copyright (c) 1989, by Matthew Reed, '
     DB    'all rights reserved'
;-------------------------------
; Open window and request information
;-------------------------------
START LD   HL,BUFF          ; HL = BUFF
     LD    (HLI),HL        ; Put in HLI
     LD    B,7             ; WCREAT (opens window)
     LD    HL,0707H        ; Start of window
     LD    DE,043BH        ; Size of window
     SVC   @WAM            ; Create window
     JR    Z,ST1           ; If no error, skip to ST1
     JR    BEEP            ; Beep and return
ST1  LD    HL,HFD$         ; Opening screen
     CALL  KEY             ; Get keyboard input
     JR    C,CLOS          ; If break or export, CLOS
     LD    BC,0800H        ; WCLOSE and return
     SVC   @WAM            ; Close it
;-------------------------------
; Open file, parsing the command line pointed to by HL
;-------------------------------
     LD    HL,(HLI)        ; Get buffer address
     LD    DE,FCB          ; 32 byte FCB
     SVC   @FSPEC          ; Put it into FCB
     JR    NZ,START        ; Prompt again if error
     LD    (HLI),HL        ; Save buffer address
     LD    HL,HLP          ; Default extension
     SVC   @FEXT           ; Add extension to FCB
     SVC   @FLAGS          ; Get pointer to SFLAG$
     SET   0,(IY+18)       ; Do not set file open bit
     LD    B,1             ; LRL of 1
     LD    HL,BUF          ; 256 byte I/O buffer
     SVC   @OPEN           ; Open file
     JR    Z,HPRS          ; If no error, skip past
     CALL  BEEP            ; Beep
     JR    START           ; Prompt again if error
```

The structure of a Profile data file set was discussed briefly by David Krebs in an article which appeared in *The MISOSYS Quarterly*, issue V.iii (Spring 1991). I may expand on that data structure at some point; however, I want to concentrate on the pfs FILE data structure because it has not been discussed - to my knowledge - in print. This article will not present any programming techniques to deal with the pfs FILE data file; for that you may as well acquire LB. The data structure is quite complex which caused me to spend considerable time figuring it out. I hope I got it right. Nevertheless, TMQ readers are an inquisitive lot and I thought I would share my discovery with you.

Let me first define some terminology for the sake of understanding the discussion. Folks usually refer to the collection of data associated with one entity as a *record*. FILE calls this a *FORM*. Perhaps they use this term because you're always filling in forms these days. FILE's form can be composed of more than one page; thus, it uses the term *PAGE* to designate these divisions. Under FILE, a page will appear on one and only one screen display; the program provides a means to move from one page of a form to another - and from one form to another. Data is displayed on the video screen by using a blank form as a template through which the form data is displayed. I will call these templates, *SCREEN FORMS*.

To begin with an explanation of the data structure, FILE's files are addressed in 128-byte blocks; these are numbered from 0 to n. When you create a new file, it is initially established with 128 blocks. At two blocks per 256-byte sector, such a file is 16K. The first block (block 0) is a system block and contains a header record. Table I illustrates the data in that header - at least as far as I was able to determine.

Data blocks are used starting from the last block in the file. The first screen form is stored starting in that last block. I say *starting*, because a form occupies as many blocks as it needs to store its character string layouts. Pointers in each block are

```
;-------------------------------------
; Act upon input
; Either display help directory or display a help topic
;-------------------------------------
HPRS  LD    HL,0000H      ; Full screen
      LD    DE,1850H      ; window
      LD    B,7           ; WCREAT
      SVC   @WAM          ; Open window
      RET   NZ            ; Return if error
HA    LD    HL,(HLI)      ; Pointer to command line
      LD    A,(HL)        ; Value of delimiter
      CP    0DH           ; Is there no topic?
      JR    Z,HD          ; If so, skip to directory
      CALL  HLIST         ; Otherwise, list topic
      JR    C,CLOS        ; If break or export, CLOS
      JR    NZ,HD         ; Get directory if error
      JR    HC            ; Skip to enter keyword
HD    CALL  HDIR          ; Get help directory
      JR    C,CLOS        ; If break or export, CLOS
      JR    Z,HC          ; If no error, skip to HC
      LD    C,0           ; Close and return
      CALL  CLOS          ; Close window
      CALL  BEEP          ; Beep
      JR    START         ; Start at the beginning
HC    LD    HL,RBYTE      ; Reverse video byte
      LD    (HL),11H      ; Reset reverse video
      LD    HL,EK$        ; "Enter keyword: "
      CALL  KEY           ; Get keyboard input
      JR    C,CLOS        ; If break, skip to CLOS
      CALL  E9            ; Clear screen
      JR    HA            ; Go back to process input
;-------------------------------------
; Routines to beep and close windows
;-------------------------------------
BEEP  LD    B,2           ; One beep
      SVC   @SOUND        ; Beep
      RET                 ; Return
CLOS  LD    B,08          ; WCLOSE
      SVC   @WAM          ; Close it
      RET                 ; Return
```

Table I - block 0 - file header

word: block # of first screen form page
word: block # of last screen form page
word: # of pages in the form
word: # of active forms (records) in the file
word: block # of last data form (record) added
word: unknown - varies with # of active forms
word: unknown - varies with # of active forms
word: first free block
string: "TYPE 3" - used to designate a pfs FILE file
word: block # of screen form page 1
word: block # of screen form page 2
...
word: block # of screen form page n (32 max)

```
;-----------------------------------------
; Displays message in HL, gets keyboard input,
; and converts it to uppercase (KEY)
;-----------------------------------------
KEY   LD    B,10          ; WDSPLY (displays string)
      SVC   @WAM          ; Display it
      LD    HL,BUFF             ; Buffer for keyboard input
      LD    (HLI),HL      ; Save it in HLI
      LD    BC,0020H      ; WKEYIN for 32 characters
      SVC   @WAM          ; Get keyboard input
      RET   C             ; Return if break or export
      LD    HL,BUFF             ; Retrieve BUFF
KEY1  LD    A,(HL)        ; Get byte
      CP    0DH           ; Is it CR?
      RET   Z             ; If so, return
      CP    'a'           ; Is it above "a"?
      JR    C,KEY2        ; No, so skip to KEY2
      CP    'z'+1         ; Is it below "z"?
      JR    NC,KEY2             ; No, so skip to KEY2
      AND   0DFH          ; Uppercase it
KEY2  LD    (HL),A        ; Save it in HL
      INC   HL            ; Next byte
      JR    KEY1          ; Loop
;-----------------------------------------
; Displays the byte in A, interpreting reverse video,
; expanding tabs, and paging (BDSP)
;-----------------------------------------
BDSP  PUSH  HL            ; Save HL
      PUSH  DE            ; Save DE
      PUSH  BC            ; Save BC
      CP    7FH           ; Is it reverse video
      JR    Z,RVRSE             ; If so, RVRSE
      LD    HL,LBYTE      ; Last byte
      LD    B,(HL)        ; B = value of last byte
      PUSH  AF            ; Save A
      LD    A,B           ; Copy B to A
      OR    7FH           ; Set bits 1-6
      LD    B,A           ; Put it into B
      POP   AF            ; Restore A
      LD    (HL),A        ; Save A as last byte
      AND   B             ; Combine A and B
      BIT   7,A           ; Was bit 7, and now tab?
      JR    NZ,TAB        ; Yes, go to TAB
      LD    C,A           ; Transfer A to C
      CP    0DH           ; Is it a CR?
      JR    Z,ENT         ; If so, go to ENT
      CP    20H           ; Is it below 32?
      JR    C,DONE        ; If so, leave
BD1   LD    B,09H         ; WDSP
      SVC   @WAM          ; Display it
      LD    HL,LBYTE      ; HL = LBYTE
      BIT   7,(HL)        ; Was bit 7 set?
      JR    Z,DONE        ; If bit 7 not set, done
      LD    A,1           ; If so, one iteration
TAB   AND   127           ; Reset bit 7
      LD    B,A           ; Transfer A to B
T1    PUSH  BC            ; Save B
      LD    BC,0920H      ; WDSP and C equals space
      SVC   @WAM          ; Display it
      POP   BC            ; Retrieve B
      DJNZ  T1            ; Loop until done
DONE  SCF                 ; Set carry flag
      CCF                 ; Complement (reset)
      POP   BC            ; Restore BC
```

used to link the multiple blocks which constitute a single form. In Table I, a *word* is a 2-byte unsigned integer value. For instance, the first word contains the block number of the first page of screen form 1. If your file contains more than one page per form, the second word will contain the block number of the last screen form page. This value would be the same as the first value if the file contained only one screen form template. The third word contains the number of pages in the form template. Note that FILE does not require you to complete all pages in each active form.

The next word contains the number of active forms; this is used by the program in traversing the pages.

If you use FILE, you will note that when you search a file, the search procedure works from the last form added up through the first form added. This sequence doesn't necessarily make too much sense to me; a last-in first-out sequence makes sense more in inventory control - and definitely not for food storage and consumption which should be first-in first-out. Nevertheless, one reason for this approach is that FILE doesn't keep a pointer to the first record! The next word in the header structure is a pointer to the block number which contains the first page of the most recent form added. So a search will start form this block and traverse through the file; each form also contains pointers which link together the sequence of forms.

The following two words appeared to vary in value according to the number of records added to a file as I proceeded with my investigation of FILE's data structure. My best educated guess is that they are used in some fashion to force an expansion of the file from its initial 16K size at some point. But they would be of no consequence in porting data from a pfs file; therefore, I chose to ignore them.

In order to add a record to a file, there must be some indicator of the first unused block. This is the next word of the header. So when a new form is added - or an existing form is increased and needs an additional block, this word is used to locate it. Blocks are always used from the top down; thus, when block 124 is used, the next free block would be 123, then 122, and so forth. Blocks would be available down to block 1 (0 is used for the system header structure), at such time the file would need to be expanded by 16K so that block 256 would be the next available block.

The following six bytes of the header contains the string, "TYPE 3". My hunch is that this is used as an integrity check of a pfs FILE data file. I never bothered to explore a Model III pfs FILE file as it is on a diskette with a format not compatible to TRSDOS or LDOS (i.e. it uses a 512-byte sector size). But my hunch is that the "TYPE 3" string may stem from the pfs

```
CDN   POP   DE          ; Restore DE
      POP   HL          ; Restore HL
      LD    A,(LBYTE)   ; Get value of A displayed
      RET               ; Return
ENT   LD    B,04H       ; WGCUR
      SVC   @WAM        ; Get cursor
      LD    A,17H       ; A = 23
      CP    H           ; See if end of screen
      JR    NZ,BD1      ; Return if not end
      LD    BC,0000H    ; WKEYIN and 0 characters
      LD    HL,MSB      ; Dummy buffer
      SVC   @WAM        ; Do it
      POP   DE          ; Eliminate BC
      JR    C,CDN       ; Return if break or export
      PUSH  DE          ; Put BC back
      CALL  E9          ; Clear the screen
      JR    DONE        ; Skip to finish
RVRSELD     A,(RBYTE)   ; Get reverse video byte
      XOR   1           ; Toggle between two
      LD    (RBYTE),A   ; Save it
      LD    C,A         ; For display
      LD    B,09H       ; WDSP
      SVC   @WAM        ; Do it
      JR    DONE        ; Skip to finish
E9    LD    BC,091CH    ; WDSP to go to upper left
      SVC   @WAM        ; Do it
      LD    BC,091FH    ; WDSP to clear screen
      SVC   @WAM        ; Do it
      RET               ; Return
;-------------------------------
; Positions disk file to table pointed to at the end of
; the file (PTAB)
;-------------------------------
PTAB  LD    DE,FCB      ; DE = FCB
      SVC   @LOF        ; Calculate EOF, in BC
      RET   NZ          ; Abort if error
      DEC   BC          ; Go to pointer
      DEC   BC          ; two before
      LD    (FCNT),BC   ; Save pointer in FCNT
      LD    DE,FCB      ; DE = FCB
      SVC   @POSN       ; Position to pointer
      RET   NZ          ; Return if error
      CALL  MPNT        ; Position to pointer value
      RET   NZ          ; Return if error
      LD    HL,(FCNT)   ; Get pointer again
      XOR   A           ; Reset carry flag
      SBC   HL,BC       ; Find length of table
      LD    (FCNT),HL   ; Save in FCNT
      XOR   A           ; Set Z flag
      RET               ; Return
;-------------------------------
; Display directory of help topics (HDIR)
;-------------------------------
HDIR  CALL  PTAB        ; Go to topic table
      RET   NZ          ; Abort if error
      LD    HL,DIR$           ; Directory message
      LD    B,0AH       ; WDSPLY
      SVC   @WAM        ; Display it
      LD    C,04H       ; Four tabs in a line
D0    LD    B,13H       ; Twenty characters in tab
D1    CALL  GETT        ; Read byte from topic list
      RET   NZ          ; Return if error
      LD    A,(HL)      ; A equals byte from disk
      CALL  BDSP        ; Display it
```

```
        RET     C               ; Return if break or export
        DEC     B               ; Decrement character count
        JP      Z,ZNZ           ; Abort if overlong
        BIT     7,A             ; Was bit 7 set?
        JR      Z,D1            ; No, loop to D1
        SET     7,B             ; Make tab character
        DEC     C               ; If set, decrement count
        JR      NZ,D3           ; If four tabs not used, D3
        LD      BC,0D04H        ; B=CR, C=four more tabs
D3      LD      A,B             ; Either tab or CR
        CALL    BDSP            ; Display it
        RET     C               ; Return if break
D2      CALL    GETT            ; Skip past the pointer
        RET     NZ              ; Abort if error
        CALL    GETT            ; Finish skipping
        RET     NZ              ; Abort if error
        CALL    TDONE           ; Is table over?
        RET     Z               ; If so, return
        JR      D0              ; Loop
;-----------------------------------
; Display contents of help topic (HLIST)
; HLI = topic entry
;-----------------------------------
HLISTCALL   SSRCH               ; Position to topic text
        RET     NZ              ; Abort if error
        LD      HL,(HLI)        ; HL = topic entry
        LD      B,10            ; WDSPLY
        SVC     @WAM            ; Do it
HL1     CALL    GETT            ; Read byte of file
        RET     NZ              ; Abort if error
        LD      A,(HL)          ; A equals byte from disk
        CP      0CH             ; Is it end of help?
        RET     Z               ; If so, return
        CALL    BDSP            ; Display byte in A
        RET     C               ; Return if break or export
        JR      HL1             ; Loop
```

Table II - pfs form block

word: block # of page N+1; 0 if last page
word: block # of page N-1; 0 if first page
word: page # of form
word: form #; Note different on screen form (98BD)
word: block # of form N+1; 0 if last form
word: block # of form N-1; 0 if first form

word: length of string + 5
byte: screen column of string
byte: screen row of string (bit 7 set)
var: string (bit 7 set); optionally followed by 04H if string len was even;
string can contain embedded space compression coding
of "01 nn A0" which count only 3 in length byte: NULL
repeat previous block until byte 126

word: block number of block continuing the data; 0 if last

file Model III format. That's just a hunch, though! I use a check of the "TYPE 3" string in LBCONV.

Following the integrity check string are a variable number of words - each one contains a pointer to the first block of a screen form page. From the space remaining in the 128-byte header, there could be 53 pages to a form; however, FILE limits the number of pages to 32. This completes the discussion of the header record.

Actual data (i.e. form pages) take up as many 128-byte blocks as they need. At this point it may be useful to discuss how FILE *packs* its data into blocks. Appendix C-1 of the pfs user manual covers the number of bytes (characters) needed under various conditions. Let me rehash that material and tell you why those counts exist.

The manual says that FILE "uses the first 14 bytes of every page". That's not entirely correct; it uses the first 12 bytes as a header and the last two bytes as a link pointer. The use of these bytes is illustrated in Table II. The first word is the block number of the next higher-numbered page; a zero value indicates no higher page exists. The second word contains the block number of the next lower-numbered page, or zero if its the first page. Word three is the page number of the form; however, the screen form does not use this word for that function. The next two words of the header contain the block numbers of the next and previous forms - if they exist. The last two bytes of the 128-byte block are always contain a pointer which points to the next block of the page; a zero value indicates no further blocks are in use for that page.

The appendix further states that each item entered on a page takes up five bytes for FILE internal parameters. Here's what they are used for. A 2-byte value (word) contains the length of the character string plus five for the parameters. A byte is used to contain the screen column (0-79) where the first character of the string was entered; the next byte contains the screen

row where the first character of the string was entered. This value is always excess 128 (i.e. bit 7 is set); thus, the value ranges from 128 to 151 for screen rows 0 through 23. Those four bytes are followed by the *encoded* string. Each string character is stored with bit-7 set - with one exception. The manual states that "a string of three or more blanks inside a filled-in item takes three bytes". That's because contiguous multiple spaces are encoded into a 3-byte run length representation. An encoding is indicated by a value of "1" followed by a one-byte count of contiguous spaces, which is then followed by the hex value, X'A0', which happens to be the ASCII value for a SPACE with bit-7 set. Thus, the two encoding bytes don't set bit-7. Under this scheme, it is possible to compress any contiguous stream of identical characters - not just a SPACE - but FILE only compresses SPACE characters. This type of encoding is called run-length encoding. The last byte of the five referenced in the manual is a terminating NULL which designates the end of the field.

When you prepare a screen form, a field *name* includes all characters from the first non-blank character up to the next colon character. The number of characters available for a field are the total number of character positions up to the next non-blank character. FILE does not maintain any concept of field lengths, other than the number of blank positions available in the screen form.

The character strings which define the various fields continue up through the 126th byte position of the block. At that point, the page must be expanded to another block. The next free block available is allocated to the form and its block number is stored in the last two bytes of the current block. It is important to note that a field's string of characters can traverse a block. Note also that since there is no need to refer back to first block of the page, the blocks used in an expansion of a single page contain no header; they contain only a single word pointer in the last two bytes of the block.

```
;-----------------------------------------
; Moves to address of pointer in file (MPNT)
;-----------------------------------------
MPNT  LD    DE,FCB        ; DE = FCB
      LD    HL,LSB        ; HL = LSB of address
      SVC   @READ         ; Read byte into buffer
      RET   NZ            ; Abort if error
      INC   HL            ; HL = MSB of address
      SVC   @READ         ; Read byte into buffer
      RET   NZ            ; Abort if error
      LD    BC,(LSB)      ; BC = address
      SVC   @POSN         ; Position to address
      RET
;-----------------------------------------
; Searches disk file for topic entry (SSRCH)
; HLI = topic entry
;-----------------------------------------
SSRCHCALL   PTAB          ; Position to table start
      RET   NZ            ; Abort if error
S1    LD    DE,(HLI)      ; Get address of topic
      CALL  RD            ; Get next byte to test
      RET   NZ            ; Abort if error
      CP    (HL)          ; Do they match?
      JR    C,ZNZ         ; If too high, abort
      JR    NZ,ADVNC      ; If not, skip to ADVNC
SLOOP INC   DE            ; Advance topic byte
      CALL  RD            ; Get next bytes to test
      RET   NZ            ; Abort if error
      XOR   (HL)          ; Do they match?
      JR    Z,SLOOP             ; If so, loop again
      CP    128           ; Was bit 7 set, but match
      JR    NZ,ADVNC      ; If not, ADVNC
      INC   DE            ; Is it the
      LD    A,(DE)        ; end of the
      CP    0DH           ; topic?
      JR    NZ,ADVNC      ; If not, ADVNC
      CALL  MPNT          ; Position to proper place
      RET                 ; Return
A1    CALL  GETT          ; Read next byte
      RET   NZ            ; Abort if error
ADVNCLD     A,(HL)        ; Get byte from disk
      BIT   7,A           ; Is bit 7 set?
      JR    Z,A1          ; If bit 7 is not set, loop
      CALL  GETT          ; If end of entry add two
      CALL  GETT          ; to get past pointer
      RET   NZ            ; Return if error
      CALL  TDONE         ; Is table over?
      JR    Z,ZNZ         ; If so, error, skip to ZNZ
      LD    DE,(HLI)      ; Restore pointer to entry
      JR    S1            ; Head back to beginning
ZNZ   OR    0FFH          ; Make NZ
      RET                 ; Return NZ
RD    CALL  GETT          ; Read a byte
      LD    A,(DE)        ; Load one from the topic
      RET                 ; Return
;-----------------------------------------
; Tests to see if the table is done (TDONE)
;-----------------------------------------
TDONEPUSH   HL            ; Save HL
      LD    HL,FCNT             ; HL points to table length
      LD    A,(HL)        ; A = first byte
      INC   HL            ; HL points to next byte
      OR    (HL)          ; Are both zero?
      POP   HL            ; Restore HL
```

```
        RET                     ; Return
;-------------------------------------------------
; Gets a byte from disk while saving DE (GETT)
;-------------------------------------------------
GETT    PUSH    DE              ; Save DE
        LD      DE,FCB          ; DE = FCB
        LD      HL,LSB          ; HL = UREC
        SVC     @READ           ; Read a byte
        LD      HL,(FCNT)       ; HL = length of table
        DEC     HL              ; Reduce one
        LD      (FCNT),HL       ; Load back into table
        LD      HL,LSB          ; HL = UREC
        POP     DE              ; Restore DE
        RET                     ; Return
;-------------------------------------------------
; Data area (messages)
;-------------------------------------------------
HFD$    DB      11H,0AH
        DB      '  Help File Displayer,'
        DB      ' copyright (c) 1989 by Matthew Reed'
        DB      0AH
        DB      '    Enter category: ',03H
DIR$    DB      'Directory of help topics:',0AH,0DH
EK$     DB      11H,0AH
        DB      'Enter keyword: ',03H
;-------------------------------------------------
; Data area (storage)
;-------------------------------------------------
HLP     DB      'HLP'           ; Help extension
FCNT    DW      0000H           ; Counter for topic length
HLI     DW      BUFF            ; Points to keyboard buffer
LSB     DB      00H             ; Used as LSB and scratch
MSB     DB      00H             ; Used as MSB and scratch
LBYTE   DB      7FH             ; Last byte (used by BDSP)
RBYTE   DB      11H             ; Reverse video byte
BUFF    DC      33,0            ; Keyboard buffer
FCB     DC      32,0            ; File control block
BUF     DC      256,0           ; Disk I/O buffer
;-------------------------------------------------
LAST    EQU     $
        IF      LAST.GT.3000H
        ERR     'Program is too long!'
        ENDIF
        IFLT    LAST,3000H
        DC      .HIGH.$.SHL.8-$+256,0
        ENDIF
        END     START
```

There are some significant differences in the header of subsequent pages of a form, for those forms which have more than one page. Only the first page of a form is used to reference links to the previous and subsequent forms in the chain. Also, the number of the form is stored once in the header of the first page. Thus, the first block of pages 2-n of a form contain a shortened header limited to the first three words presented in Table II; the length byte of the first character string starts with byte 6.

As if all this wasn't sufficiently confusing, here's one last wrinkle. In building up and examining sample files, I noticed that all of the character string lengths were even numbers. Here's what's happening, but don't ask me why! If the string contains an even number of characters, the string is lengthened by one and the string is terminated with a additional byte of 04H which precedes the normal terminating NULL. A field which is totally blank contains zero characters - an arbitrarily even number. Thus, a blank item would contain the five internal parameter characters plus the 04H for a total length of six. There's the reason for the pfs manual's statement, "a blank item takes 6 bytes". But it will take a greater mind to understand the significance of a required even length!

For those who really want to tinker with this type of file, Table III contains the data structures I used in LBCONV to access the two block types: system header block and form header block. In the design of LBCONV, I wanted the program to do as much as possible in the data conversion effort. Thus, LBCONV performs two passes of the data pfs file. It first examines all screen form pages to detect the total quantity of fields. It then scans the entire file to determine the longest character string in each field. It uses these values to fix the field lengths for LB. LB uses a descriptive field name of up to 19 characters for display in on-line prompts. So LBCONV also extracts the rightmost 19 characters of the descriptive name used in the pfs screen form. Why the rightmost 19? Chapter 1 of the pfs user manual describes a procedure to design a file. The first example starts with fields described as: "SS#:", "Age:", and "Date:" , followed by a row of hyphens. The fourth field is described as "Name:". It's this fourth field which creates a slight problem. Remember I previously stated that "a field *name* includes all characters from the first non-blank character up to the next colon character". In this fourth field, the *name* includes the hyphens. To automatically use the first 19 characters of the descriptive name would chop off all of those names which were preceded by a sequence of dashes - not to mention any titles in the screen form. My choice of using the rightmost 19 is an arbitrary one; I felt that folks would tend to be the most descriptive at the location of the colon.

Another compromise was to avoid unpacking the space compression codes if one occurred within the 19 characters selected as the name. LB provides an easy

method of editing the field names, anyway.

LB provides a rich set of field types; however, all fields of the pfs data file are treated as literal data because there is no way to discern the intent of an entry. After a data file is converted to LB, a user can use the LBREDEF utility to re-define the type. Besides, at a 48K file size, LBCONV is at its limit on the TRS-80 without resorting to complex overlays as is the case with LB itself.

All in all, LBCONV affords an easy way to convert from a pfs FILE data file to LB. In fact, because LBCONV also can convert from an LB data structure to dBASE II, dBASE III, DIF, and comma delimited data structures, LBCONV can even be used to convert pfs to dBASE or as an ASCII file easily importable into just about any MS-DOS database program by using LB as an intermediate file structure. I have thought about writing a data base conversion program based on LBCONV to bundle with my TRSCROSS product. But that's another story...

Table III

```
struct {
   unsigned form1;
   unsigned formn;
   unsigned npages;
   unsigned nrecs;
   unsigned lastrec;
   int fill[2];
   unsigned fblock;
   char pfsfile[6];
   unsigned forms[32];
   } pfs0 = {0,0,0,0,0,{0},0,
{'T','Y','P','E',' ','3'},{0}};

struct {
   unsigned nextpg;
   unsigned prevpg;
   int pageno;
   int formno;
   unsigned nextblk;
   unsigned prevblk;
   } pfshdr;
```

LB Data Base Manager
a review

Ken Strickler
P.O. Box 773
Stanwood, WA 98292-0773

Source: MISOSYS, Inc.
Technical Support: MISOSYS, Inc.

Ordering Information:
MISOSYS, Inc.
P.O. Box 239
Sterling, VA 22170
Phone 1-800-MISOSYS

INTRODUCTION

As with all of the Software and Hardware Packages which MISOSYS becomes involved with, LB is another fine example of taking a rough cut diamond, and polishing its many facets to a brilliant luster! The TRS80 Model 4 computer has long needed a full featured, large capacity Data Base Manager, like LB. This package adds another dimension to the TRS80 and again demonstrates the inherent power of the 8 bit machine. Additionally, the files generated can be transferred to an MS-DOS machine, and LB86 for the MS-DOS machine can process the files!

Before continuing with my review, and in order to save space and time, some background information is necessary. You probably ought to get a cuppa' jo (coffee) and a quiet place to do this!

Step 1 - Locate a recent copy of the TMQ, preferably Volume 5, Issue IV. On the inside of the BACK COVER is a quick description of LB, and right there in the upper right hand corner is 'ole ROY checkin' that last bit of code for any errors! (Can't find any more - huh Roy!)

Step 2 - Read the BLURB on pages 2,3 and 4 of TMQ V.IV where Roy explains the features that he has added or enhanced. (Nice work ROY! Any old programmer will appreciate the time and effort that has gone into improving this package!)

Step 3 - The PACKAGE!

The LB Data Base Management System (LB-DBMS) is a flat-faced (Equal sized record) filing system, containing the necessary modules for creating and manipulating information containing any single point of commonality. In this program, extreme care has been taken to effectively utilize system resources, to allow the program to run on a wide variety of available computer systems.

As delivered, for the TRS80 model 4 computers, an Installation Manual, Users Manual, Installation Floppy, and Execution Floppy Diskette.

The PLATFORM(s)

The SYSTEM requirement is a TRS80 Model 4 with at least 128K of memory, 1 FLOPPY Disk Drive, if a HARD Disk is available, or 2 FLOPPY Disk Drives.

Evaluation was performed on several different Model 4 computers, using different configurations, as follows:

(1) 128K - 2 Disk Drives (40 trk,ss,dd) - (standard)

(2) 128K - 2 Disk Drives (1-40 trk,ss,dd; 1-80 trk,ds,dd)

(3) 384K (xlr8er) - 2 Disk Drives (1-40 trk,ss,dd; 1-80 trk,ds,dd)

(4) 384K (xlr8er) - 2 Disk Drives (1-40 trk,ss,dd; 1-80 trk,ds,dd) - 40 Meg Hard Disk

Systems Using the 384K RAM was allocated as 128K RAM and 256K RAMDISK.

The INSTALLATION

Following the 8 page Installation Manual provided, selecting the desired configuration resulted in ERROR FREE installations.

It is apparent from this point and on, that ATTENTION TO DETAIL has made LB-DBMS an easy system to use!

The USERS MANUAL

The real HEART of the success of any software program is the ability to convey to the user the intent of the program, and how the user might implement the program to perform the desired task. It is important that the USERS MANUAL guide the user to the area rapidly, to answer any question which might arise. Further if the MANUAL could remain open to a specific page, and not "snap" closed whilst the command is entered into the computer, learning to use the program is greatly enhanced! A "sample" program provided to demonstrate the use of the program is always a HELP! Another consideration would be the arranging of the manual in the order which the program is to be used, so that continual flipping of pages is avoided! It would always be of great help, if portions of the manual were available as HELP screens, while running the program, to help check methods or syntax without going to the USERS MANUAL.

Take HEART! LB Version 2.x.x does it all! From Page 1 to page 170, information is provided with descriptions of the "how to" and "why for" sprinkled generously throughout! True, the Table Of Contents, starting in the body at Menu #10 may seem a little strange at first, it is the first choice to set up a Data Base! Starting with Page 1, Installation, First Time Users, and upgrading from Version 1 are explained - Then the INTRODUCTION - and with Roy's permission, a little quote:

"What exactly is a 'database'? It is nothing but a fancy name for an organized collection of information. The box or recipes in your kitchen is a database. The telephone book is a database. That list of movies in your videotape library is a database. The public library is a database. Your little black book is a database."

It couldn't be stated more simply yet more completely! Everyone uses a database daily! From this beginning, complete information is presented concerning the capabilities and requirements are logically outlined. A sample Mailing List program called Mailfile is emulated, and the necessary files are provided for you to look at. Important information is marked by BOXES, MARGIN LINES, or different TYPE styles to attract your attention. Especially helpful if one is scanning the book for a particular item which "I was sure that I had seen!".

Since this is an extensive package, don't expect to be finished in 15 minutes! At nearly $100, if I could be completely familiar with a complete package in an hour or so, I would figure that I had been "ripped off". Not to worry, I have been testing the program for a couple of months, and I still don't think that I know it all, however, there are an endless number of variations to be tried!

I could continue from this point, and give a description of the many fine features that are provided with the program, including the up to 10 screens each for entering and printing information, the Index file operations and the screen outputting functions! Where would I stop? I will say that the ON-LINE HELP function that is provided will provide HELP for the current screen, and not just an endless help file which must be scanned as some programs do!

After looking at the sample database, I built my database using Titles, Singers and locations of songs on some CD's that I recently purchased. That gave me 195 entries to work with. I tried ADDING and DELETING, SELECTIONS and SORTING, VIEWING, FORMATTING and PRINTING, and everything that I did worked flawlessly, or it was my fault!

Changes, sure! I copied page 7 of the installation manual, and trimmed the section called "Special Keyboard Characters" and "Using the Input Editor" for the TRS80, and taped it to the page, just before page 1! (It is blank) I never can seem to find that information! Now I know just where to look. The manual is delivered in one of those ever popular plastic multi-ring types, and frankly Roy has maybe put a few too many pages in for that size binder! They seem to get caught on each other if much page flipping is done! Care must be taken, or I'm afraid some tearing might occur. [Note from MISOSYS: the binding fingers currently used are now 3/4" versus the 5/8" previously used] The other option might be one of those SPIFFY, MISOSYS 3 RING binders used for PRO-WAM and the GOLDEN OLDIES! It would also allow the addition of tabbed pages, and the ever popular "cheat notes" that are derived by each individual user! Since the manual is laid

out as used, a double Table of Contents, with the Numerical Menu Items might be useful.

In the back of the Manual, is another section called Maintenance Utilities, (cost extra in the old version) which allow the manipulation of the data base format 'after' it has been constructed. I had to use it on my very first effort because I never seem to get things right the first time! I'm sure that none of you good readers suffer from the same problem! (Actually, the price of LB Version 2 is $5 cheaper than Version 1 and the old Maintenance Utility Package!)

It won't take long before you get a little I/O bound waiting for modules to load, and if you have a FLOPPY system ONLY (low capacity drives) the time needed for DISK SWAPPING. After checking to see that the various configurations would work correctly, my testing was completed using platform 4, with the data files stored in the RAMDISK. This made the process quite quick! As a matter of fact, I have enclosed the following table which gives time comparisons for switching from MENU to MODULE and MODULE to MENU for a FLOPPY DISK system and the HARD DISK system from MISOSYS -: (written using the NEW FILE STORE section in option 4!)

I could shave another 'second' off the HARD DISK time if I put the LB modules in the RAMDISK instead of my DATA FILE, but that substantially increased my 'Loading' time.

Along the same lines of thought, a little simple math will show that if you have a large data base file, with many entries, you will easily run out of capacity to store and manipulate your files using only a FLOPPY system. (Remember, you will need a work file, possibly as large as your main file for sorts and a temporary file.) These problems are resolved by ordering a HARD DISK drive system, and configuring the partitions to 13 MEG each! (Roy has the control formats for his 40MEG hard disk system!) If you must go to the maximum that LB will handle, a nice MS-DOS system with at least 130+ MEG will be required! (AH - get the COLORADO TAPE BACKUP from ROY while you are at it!)

RAM CRAM and LB

I always like to check the 'corners' while I'm testing to see how FRIENDLY the new software is to the HARDWARE and other RESIDENT SOFTWARE. LB ranks VERY HIGH in it ability to work with other resident software, like PROWAM or ZSHELL, KISTORE or IOMON. If you CRAM too hard, however, you will get the CAN'T LOAD message, no memory, and you will have to drop some of your resident programs. If you have managed to fill memory prior to entering LB and have a HIGH MEM pointer much LOWER than x'E990, LB starts to lock up your system! Since memory is dynamically allocated by LB during RUNTIME, this boundary is not checked. Module 3 - Update and Delete locks between x'E990 and x'E980: Module 8 - Define Screen Formats locks between x'E586 and x'E580: Module 9 - Define Print Formats locks between x'E7B0 and x'E7A0. If you happen to lock up your system, check the HIMEM boundary and insure that you haven't violated LBs space. You still have around 5K for drivers and filters. In a machine with only 64K in the base unit, with 16K reserved at the bottom for the operating system, 5K at the TOP is really generous! Again it shows the care that has been taken to provide a top quality program to those who have 'stayed behind' in the run for 'wallet'!

THE NEXT STEP !

The next step is for the reader to sit down and write out a check for LB! You have been waiting for this program, You Know You Have! I heard you say just the other day, that if there was a really fine Data Base Manager for this old TRS80 of mine, that you were going to get it! Now admit it, YOU DID! Now, just think, after we all have BOUGHT our copies (DID YOU CATCH THAT - THE OPERATIVE PHRASE IS 'TO BUY' AS IN 'PURCHASE' - as opposed to 'COPY' AS IN 'PIRATE'!) then we can 'SHARE' all of those NEAT Data Base Templates, like the one for MISOSYS articles written a couple of years back! Oh - YES, and while you are at it, RENEW YOUR SUBSCRIPTION TO TMQ for another year!

IN CONCLUSION

Well, the cuppa' jo should be gone, or cold by now! Thanks for sharing this time with me. I hope that this information will be of value to you the reader, and that a LB is in your future!

LB Module Switching Times (Secs)

Direction	Floppy	Hard Disk
Menu to 1	00.00	00.00
1 to Menu	00.00	00.00
Menu to 2	11.50	02.59
2 to Menu	19.17	04.03
Menu to 3	12.57	02.93
3 to Menu	20.59	03.98
Menu to 4	11.63	02.43
4 to Menu	18.33	03.79
Menu to 5	12.87	02.80
5 to Menu	19.77	03.90
Menu to 6	00.00	00.00
6 to Menu	00.00	00.00
Menu to 7	02.82	00.11
7 to Menu	00.11	00.00
Menu to 8	12.70	02.81
8 to Menu	17.84	03.70
Menu to 9	13.72	03.24
9 to Menu	17.90	03.75
Menu to 10	07.35	02.13
10 to Menu	03.80	01.85
Menu to 11	06.70	00.83
11 to Menu	00.00	00.00
Menu to 12	03.84	00.66
12 to Menu	03.84	00.66
Menu to 13	00.00	00.00
13 to Menu	00.00	00.00
Menu to 14	00.00	00.00
14 to Menu	00.00	00.00
Menu to 15	05.97	01.44
15 to Menu	18.72	05.20

by Matthew Reed
Box 368
West Lebanon, NH 03784-0368

SYSFLEX:
The Flexible /SYS File Loader

A 128K Model 4 contains two "extra" 32K memory banks. These memory banks are generally unused by the operating system, although many programs do exist that can take advantage of the extra 64K. For example, the MEMDISK driver included with LS-DOS can use the memory as a small, super-fast disk drive. If you copy all the LS-DOS system files (SYS0-SYS13) onto this RAM disk and then establish it as drive zero, your Model 4 will experience truly spectacular performance. Library commands will execute instantly, and physical drive zero will then be available for data diskettes.

Another excellent use for the extra 64K is PRO-WAM, the window controller and applications manager from MISOSYS. Once PRO-WAM has been installed, pressing a special key combination "pops up" a menu of small applications. These applications, which can include a calculator, calendar, address book, to-do list, and more, can be used from within any program, without exiting to LS-DOS and without disturbing your program in any way. PRO-WAM applications can even transfer data to and from the interrupted program; for example, you could "import" numbers from a word processing document into the calculator and "export" the sum back to the document.

Now that I have used a RAM disk and PRO-WAM, I never want to work without them. The problem is that PRO-WAM uses 32K of the Model 4's extra memory. That leaves only 32K for the RAM disk, and that isn't enough for all the LS-DOS system files. Creating a RAM disk with only some of the system files won't work, because if LS-DOS can't find a necessary /SYS file, it displays "Error 07H" and often locks up the machine. This means I could install PRO-WAM, or I could in-

```
;-----------------------
; SYSFLEX modified /SYS file loader
; copyright (c) 1991 by Matthew Reed
; all rights reserved
;-----------------------
@DSPLY      EQU   10
@LOGOT      EQU   12
@PARAM      EQU   17
@ERROR      EQU   26
@CKDRV      EQU   33
@DCSTAT     EQU   40
@GTMOD      EQU   83
@DIRRD      EQU   87
@HIGH$      EQU   100
@CKBRKC     EQU   106
VAL   EQU   80H
SW    EQU   40H
ABR   EQU   10H
DIRRD@      EQU   18BBH  ; Address of @DIRRD
OVHK EQU    1ABAH        ; Hook into SYS loader
SYSFCB      EQU   0092H  ; Drive in system FCB
SVC   MACRO #NUM
      LD    A,#NUM
      RST   28H
      ENDM
      ORG   3000H
;-------
; Initialize
;-------
START LD    (STACK+1),SP ; Save old stack
      LD    SP,START     ; New stack
      SVC   @CKBRKC      ; Is break pressed?
      JP    NZ,ABORT     ; If so, abort
      PUSH  HL           ; Save command line
      LD    HL,OPEN$     ; Opening message
      SVC   @DSPLY       ; Display it
      POP   HL           ; Restore command line
;-----------------
; Parse parameters (if any)
;-----------------
      LD    DE,PARM$     ; Parameter table
      SVC   @PARAM       ; Parse parameters
      JP    NZ,PARMERR   ; Abort if error
;-----------------
; See if module should be removed
;-----------------
      LD    BC,0         ; Default REMOVE=OFF
RMVVAL      EQU   $-2
      LD    A,B
      OR    C
      JP    NZ,REMOVE    ; Remove module
```

```
;-------------------------
; Make sure SYSTEM and DRIVE are numeric
;-------------------------
     LD    A,(SYSRSP)    ; Are SYSTEM
     LD    C,A
     LD    A,(DRVRSP)    ; and DRIVE
     AND   C
     BIT   7,A           ; numeric?
     JP    Z,PARMERR     ; Abort if not
;-------------
; Make sure drives exist
;-------------
     LD    BC,0          ; SYSTEM value
SYSVAL    EQU   $-2
     LD    A,C           ; Put in
     LD    (SYSTEM),A    ; driver
     CALL  TSTD  ; Test for valid drive
     LD    BC,1          ; DRIVE value
DRVVAL    EQU   $-2
     LD    A,C           ; Put in
     LD    (DRIVE),A     ; driver
     CALL  TSTD  ; Test for valid drive
;-------------
; Make sure SYS1 is on drives
;-------------
     LD    A,(SYSVAL)    ; Drive to search
     CALL  TSYS1 ; Test for existence
     JP    Z,INSTALL     ; Install if there
     LD    A,(DRVVAL)    ; Drive to search
     CALL  TSYS1 ; Test for existence
     JP    Z,INSTALL     ; Install if there
;--------
; Display errors
;--------
NTINST    LD    HL,NTINST$    ; Module not
installed
     DB    0DDH
CNTRMV    LD    HL,CNTRMV$    ; Module not
removed
     DB    0DDH
IMDV LD    HL,IMDV$      ; Improper DOS
version
     DB    0DDH
NS1  LD    HL,NS1$       ; No SYS1
     DB    0DDH
DRN  LD    HL,DRN$       ; Drive doesn't
exist
     DB    0DDH
DNID LD    HL,DNID$      ; Disk not in
drive
     DB    0DDH
MEMERR    LD    HL,MEMERR$    ; Memory
error
ERR  SVC   @LOGOT        ; Log message
;--------
; Abort or exit
;--------
ABORTLD    HL,-1         ; Error
     DB    0DDH
EXIT LD    HL,0          ; Success
STACKLD    SP,$-$        ; Old stack
     RET                 ; Return
```

```
;--------
; Error routines
;--------
DRNE LD    A,30H         ; Zero
     ADD   A,C           ; Make into number
     LD    (DRV1),A      ; Put in message
     JR    DRN           ; Signal error
;
DNIDELD    A,30H         ; Zero
     ADD   A,C           ; Make into number
     LD    (DRV2),A      ; Put in message
     JR    DNID          ; Signal error
;
PARMERR   LD    A,44   ; Invalid param-
eters
;--------
; DOS errors
;--------
DOSERR    LD    L,A           ; Put error
     LD    H,0           ; in HL
     OR    0C0H          ; Abbreviate,
return
     LD    C,A           ; Display
     SVC   @ERROR        ; error message
     JR    STACK         ; Abort
;------
; Test SYS1
;------
TSYS1LD    C,A
     LD    B,3           ; SYS1
     SVC   @DIRRD        ; Read dir entry
     JR    NZ,DOSERR     ; Fatal error
     LD    A,(HL)        ; Get overlay
     AND   50H           ; Was it purged
     XOR   50H           ; or non-system?
     RET
;-------
; Test drive
;-------
TSTD CP    8             ; Is it invalid?
     JP    NC,PARMERR    ; Abort if error
     SVC   @DCSTAT       ; Does it exist?
     JP    NZ,DRNE       ; Abort if no
     SVC   @CKDRV        ; Is there a disk?
     JP    NZ,DNIDE      ; Abort if no
     RET
;---------
; Install SYSFLEX
;---------
INSTALL   LD    DE,FLEX$      ; look for
module
     SVC   @GTMOD
     JR    NZ,INHG       ; Put in high if
not
     PUSH  HL            ; Save pointer to
module
     LD    BC,SYSTEM-BEGIN     ; Point to
SYSTEM
     ADD   HL,BC
     LD    A,(SYSTEM)    ; Get SYSTEM drive
     LD    (HL),A        ; Put in module
     LD    BC,DRIVE-SYSTEM ; Point to DRIVE
     ADD   HL,BC
     LD    A,(DRIVE)     ; Get DRIVE drive
```

```
        LD      (HL),A          ; Put in module
        JR      LINK            ; Link into system
INHG    LD      A,(OVHK)        ; Test byte
        CP      0BBH            ; Is it 18BBH?
        JP      NZ,IMDV         ; Abort if error
        LD      A,(OVHK+1)
        CP      18H
        JP      NZ,IMDV         ; Abort if error
        LD      HL,0            ; Retrieve address
        LD      B,H             ; High memory
        SVC     @HIGH$
        JP      NZ,MEMERR       ; Abort if error
        LD      (OLDHI),HL      ; Save HIGH$ in
module
        LD      BC,LNTH         ; Length of module
        OR      A               ; Reset carry
        SBC     HL,BC           ; Find new HIGH$
        LD      B,0             ; Set new HIGH$
        SVC     @HIGH$
        JP      NZ,MEMERR       ; Abort if error
        PUSH    HL              ; Save for later
        EX      DE,HL  ; Point DE to destination
        INC     DE
        LD      HL,BEGIN        ; Point HL to source
        LD      BC,LNTH         ; Length to transfer
        LDIR
;---------------------
; Make linkage into system loader
;---------------------
LINK    LD      BC,17           ; Length of header
        POP     HL              ; Restore HIGH$
        ADD     HL,BC           ; Find SYSFLEX
        LD      (OVHK),HL       ; Install hook
;---------------------
; Announce success and exit
;---------------------
        LD      HL,SCC$         ; Display message
        SVC     @DSPLY
        JP      EXIT            ; Exit
;---------------
; Turn SYSFLEX off
;---------------
REMOVE  LD      HL,DIRRD@ ; System SYS
loader
        LD      (OVHK),HL       ; Reload it
        XOR     A               ; Drive 0
        LD      (0092H),A       ; Put in system FCB
;---------------------
; Remove module from memory
;---------------------
        LD      DE,FLEX$        ; Module name
        SVC     @GTMOD          ; Look in high
memory
        JP      NZ,NTINST       ; Abort if not found
        PUSH    HL              ; Save module address
        LD      HL,0            ; Get HIGH$
        LD      B,H
        SVC     @HIGH$
        JP      NZ,MEMERR       ; Abort if error
        EX      DE,HL           ; DE = HIGH$
        POP     HL              ; HL = module address
        DEC     HL              ; Back one
        OR      A               ; Compare
        SBC     HL,DE
        JP      NZ,CNTRMV       ; Abort if error
        EX      DE,HL           ; Switch to HL
        INC     HL              ; Go to OLDHI
        INC     HL
        INC     HL
        LD      E,(HL)
        INC     HL
        LD      D,(HL)          ; DE = (OLDHI)
        EX      DE,HL           ; Switch to HL
        LD      B,0             ; Set new HIGH$
        SVC     @HIGH$
        JP      NZ,MEMERR       ; Abort if error
;---------------------
; Announce success and exit
;---------------------
        LD      HL,RMVSCC$ ; Remove success message
        SVC     @DSPLY
        JP      EXIT
;--------
; Messages
;--------
OPEN$DB     'SYSFLEX, modified SYS file
loader 1.0',0AH
        DB      '  copyright (c) 1991 by Mat-
thew Reed',0AH
        DB      '  all rights reserved',0AH,0DH
SCC$    DB      'SYSFLEX correctly installed'
        DB      ' and operational',0DH
RMVSCC$     DB      'SYSFLEX is turned off'
        DB      ' and memory is reclaimed',0DH
NS1$    DB      'Neither drive has SYS1!',0DH
DRN$    DB      'Drive 0'
DRV1    EQU     $-1
        DB      ' is not enabled!',0DH
DNID$DB     'There is no disk in drive
0!',0DH
DRV2    EQU     $-3
MEMERR$     DB      'High memory address'
        DB      ' can''t be changed!',0DH
IMDV$DB     'Overlay hook already in
use!',0DH
CNTRMV$     DB      'SYSFLEX is turned off'
        DB      ' but memory can''t be
reclaimed',0DH
NTINST$     DB      'SYSFLEX is not
installed',0DH
FLEX$DB     'SYSFLEX',03H
;---------------
; Parameter table
;---------------
PARM$DB     80H
        DB      VAL.OR.ABR.OR.6
        DB      'SYSTEM'
SYSRSP      DB      VAL
        DW      SYSVAL
        DB      VAL.OR.ABR.OR.5
        DB      'DRIVE'
DRVRSP      DB      VAL
        DW      DRVVAL
        DB      SW.OR.6
        DB      'REMOVE'
        DB      SW
        DW      RMVVAL
        DB      0
```

```
;----------------
; SYSFLEX high memory module
;----------------
BEGIN JR    SYSFLEX ; Branch around linkage
OLDHI DW    $-$     ; Last byte used
      DB    7,'SYSFLEX'  ; Name of module
MODDCB DW   $-$     ; No DCB pointer
      DW    0       ; Reserved by DOS
;-------
; Start of code
;-------
SYSFLEX   LD    A,0         ; SYSTEM
drive number
SYSTEM    EQU   $-1
      LD    (SYSFCB),A  ; System FCB
      LD    C,A
      CALL  DIRRD@      ; Read dir entry
      RET   NZ          ; Return if error
      LD    A,(HL) ; Was overlay purged?
      AND   50H   ; or is it non-system?
      XOR   50H
      RET   Z     ; Return if successful
      LD    A,1   ; DRIVE drive number
DRIVE EQU   $-1
      LD    (SYSFCB),A  ; System FCB
      LD    C,A
      JP    DIRRD@      ; Read dir entry
LNTH  EQU   $-BEGIN     ; Length of module
      END   START
```

PowerMail Plus

"This is one of the BEST mail-list programs we've seen for the TRS-80, and for its price, it's a down-right bargain."
INFOWORLD

Rated FOUR STARS in the May 1984 80-Micro!

PowerMail Plus
★★★★
MISOSYS, Inc
PO Box 239
Sterling, VA 22170
Model 4
(Model I/III and II/12/16 versions also available)
$150- now only $39.95 + S&H

Easy to use?	★★★★☆
Good docs?	★★★★☆
Bug free?	★★★★★
Does the job?	★★★★★

● *INFOWORLD's Essential Guide To The TRS-80* said...
"If you need to maintain large mailing data-bases, you should definitely consider PowerMAIL Plus... It is exceptionally fast The program is intrinsically efficient... overall concept and design are excellent. PowerMail Plus offers the kind of features that businesses, in particular, need."

● *80-MICRO* said in their May '84 issue,
"If you're looking for a mailing-list system, I highly recommend PowerSoft's PowerMail Plus. (The) manual is high in quality, content, and workmanship. The commands were easy to use and remember."

● *OMNI's Complete Catalog of Computer Software* wrote...
"This is a very potent mailing-list program with many versatile features. ...It lives up to (PowerSoft's) normal high-standards... it's an excellent mailing list program."

PowerMAIL Plus is a highly acclaimed, deluxe mailing/data storage system written entirely in machine language for maximum operating speed. Besides being a mailing system, it is also capable of generating custom personalized "form letters" to whatever groupings of labels you wish using the TEXT-MERGE module, available separately. Many of its features, such as extensive data "flags" enable you to use it as a limited data-base manager, too. PowerMAIL+ is a "DISK-BASED" system, rather than "RAM-BASED", therefore it can keep track of over 520,000 individual entries by combining up to eight different mailing lists (which, of course would require a hard drive). There are no "slow" periods when PowerMAIL+ is running. Features have been added to this program that othershave always lacked. You now have the ability to keep track of mailings using 24 user defined "flags" that are incorporated into the PowerMAIL+ program. Separate any category in any manner you wish.

PowerMail+ can use multiple floppies by logging them in and out of the system. The program will run in as little as a 32K one drive environment, but dual drives or 80 track or double sided drives, or a hard drive are recommended for more serious use. The more available freespace, the more names you can store. A Model III or 4 data disk will hold about 1150 names. Double that for 80 track or 40 track double-sided drives (4D). Works great on hard drives as well as floppies! Handles drives 0-7. PowerSoft used it on their hard drive system for over four years with excellent results! Does not pre-allocate the entire drive, but allows you to define how many records you need at initialization time. If you need more room later, there is a way get your names onto a LARGER file easily. Also you may have a file on each disk or platter, if you have multiple drives, or a hard drive. This further increases overall storage, as PowerMAIL+ will treat the files individually, or as an overall total. TWELVE LEVEL SORT! The program will sort up to TWELVE levels that you wish in any order. Other features include the ability to separate your flags and put them onto another file, merge files together, and then separate whatever you want, improved "key" search, improved field lengths, improved disk I/O, and much improved print routines. The main thing here is that PowerMAIL+ was designed to be easy to use, easy to start up, and easy to train people to enter your names.

With PowerMAIL+, you have many print options based on Labels or Lists. If you choose labels, you have a choice of two modes and are asked how many labels across and the spacing required. The default settings are for the standard one up label commonly available (you may choose up to 4 across). if you choose listings, you have four choices to choose from there. if printing a LIST rather than LABELS, a user-definable "header" will be printed, along with the page number. This really makes an impressive printout, as well as adds to the usefulness.
PowerMAIL Plus' print system can be controlled from flag settings, allowing you to print only a part of the file. You may keep lists within lists. No need to keep separate files for different classifications. Use the flag or DATA fields for separating them for printout. There is a lot of flexibility here! In addition, you can tell it to SET FLAGs after printing to designate WHO got a label, etc.

This will contribute tremendously to the efficiency of your mass mailings, because for the first time you can keep track of who has been sent a particular mailing. Avoid duplicate mailings with ease! Let PowerMAIL Plus do it for you!
PowerMAIL Plus also incorporates a very fast search mode for locating a particular entry in the shortest possible time (supports wild cards too), as well as a sort-merge routine which allows you to sort files larger than can fit into memory. Best of all, PowerMAIL Plus includes routines which will convert files created by some of the other popular mailing list systems to its own internal format, so you don't waste any time retyping.

Converts data files from these mail systems:

Radio Shack MailList Expanded	POSTMAN
Radio Shack MailList Compressed	PowerMail 1.0
Special Delivery	Galactic Mailfile
Extra Special Delivery	

Text-Merge

TEXT-MERGE is a stand-alone form letter utility for PowerMAIL Plus. It will take a form letter prepared in ASCII format by any word processor or text editor (i.e. TED), and merge the contents of a PowerMAIL Plus ADDER file into the letter. The form letter may be as long as available memory (usually about 32K) or as short as a few lines. TEXT-MERGE is ideal for creating personalized mailings or specially-formatted mailing labels. You use PowerMAIL Plus' flag system to separate those records for which you want form letters printed into an ADDER file, then simply run TEXT-MERGE.

Prompts allow you to specify such parameters as page size, number of printed lines per page, line length, left margin, whether or not linefeeds follow carriage returns, and whether of not you want to pause after each page. During printing a count of letters printed is displayed on the screen.

Since TEXT-MERGE accepts all ASCII codes from 0 to 255 as valid data, you can create special print effects by embedding printer control codes in your form letter text. if you have a word processor which will permit this.

Fields from the PowerMAIL Plus record are inserted into the form letter at places marked by @n, where n is a digit from 0 to 9, which corresponds to the record fields. Field data may be inserted in any order, and each field may appear as many times as necessary.

Here is how you might start out a form letter;
Mr. @2 @1
@3
@4 @5
@6, @7 @8

Dear Mr. @1 ,

Here is what the resulting printout would look like;
Mr. Robert Jones
ABC Paper Co.
1632 North St. Suite 101
Ft. Worth, TX 76751

Dear Mr. Jones,

Model I/III/MAX-80 or Model 4/4P. Please Specify which computer!

LS-DOS 6.3.1 has something for everyone

International versions now available!

- ☆ The DATE command, "Date?" prompt on boot, and the @DATE SVC now support a date range of 32 years; from **January 1, 1980 through December 31, 2011.**
- ☆ **Enable or disable the printer time-out** and error generation with SYSTEM (PRTIME=ON|OFF)
- ☆ Customize the display of the time field in the DIR command to display **12-hr or 24-hr clock time** with SYSTEM (AMPM=ON|OFF).
- ☆ Both ASCII and hexadecimal display output from the LIST command is **paged a screen at a time.** Or run it non-stop under your control.
- ☆ MEMORY displays (or prints) the status of switchable memory banks known to the DOS, as well as a **map of modules** resident in I/O driver system memory and high memory.
- ☆ Specify SYSTEM (DRIVE=d1,SWAP=d2) to **switch drive d1 for d2.** Either may be the system drive, and a Job Control Language file may be active on either of the swapped drives.
- ☆ The TED text editor now has commands to **print the entire text buffer**, or the contents of the first block encountered. Obtain directories from TED, too!
- ☆ Have extended memory **known to the DOS**? The SPOOL command now permits the BANK parameter entry to range from 0-30 instead of 0-7.
- ☆ **Alter the logical record length** of a file with "RESET filespec (LRL=n)"
- ☆ Specify "RESET filespec (DATE=OFF)" to restore a file's directory entry to the old-style dating of pre-6.3 release. Specify "RESET filespec (DATE=ON)" to establish a file's directory date as that of the **current system date and time.**
- ☆ Felt uncomfortable with the *alleged* protection scheme of 6.3? **LS-DOS 6.3.1 has no anti-piracy protection!** MISOSYS trusts its customers to honor our copyrights.
- ☆ Best of all, **a 6.3.1 diskette is available as a replacement for your 6.3.0 diskette for $15** (plus $2 S&H in US). There's no need to return your current master.
- ☆ The 6.3.1 diskette comes with a 30-day warranty; written customer support is available for 30 days from the purchase date. Versions for the Model 4 and Model II/12 are available. **If you do not already have an LS-DOS 6.3.0, order the 6.3.1 Upgrade Kit with 90 days of customer support for $39.95** (+$4 S&H).

MISOSYS, Inc.
P. O. Box 239
Sterling, VA 22170-0239
703-450-4181 [orders to 800-MISOSYS (647-6797)]

LDOS 5.3.1: the support continues

- ☆ The DATE command, "Date?" prompt on boot, and the @DATE SVC now support a date range of 32 years; from **January 1, 1980 through December 31, 2011;** time-stamping, too.
- ☆ **Double-density BOOT support for Model I** with embedded SOLE and FORMAT (SYSTEM). Supports mirror-image backup, too. Reworked FDUBL driver eliminates PDUBL and RDUBL and takes less memory; enhanced resident driver eliminates TWOSIDE.
- ☆ Model III version auto-detects Model 4 for installation of KI4 keyboard driver; supports CAPS, CTRL, and function keys.
- ☆ SYSTEM command supports removable and reusable BLINK, ALIVE, and UPDATE memory modules.
- ☆ The TED text editor now has commands to **print the entire text buffer**, or the contents of the first block encountered. Obtain directories from TED, too!
- ☆ The SPOOL command offers Pause, Resume, and Clear parameters. (OFF) attempts to reclaim memory used.
- ☆ **Alter the logical record length** of a file with "RESET filespec (LRL=n)"
- ☆ Specify "RESET filespec (DATE=OFF)" to restore a file's directory entry to the old-style dating of pre-5.3 release. Specify "RESET filespec (DATE=ON)" to establish a file's directory date as that of the **current system date and time.**
- ☆ Both Model I and Model III support similar commands: all features of Model III 5.3.0 are in Model I 5.3.1. That includes such facilities as DOS and BASIC help files, SETCOM and FORMS library commands, TED text editor, BASIC enhancements, etc. All DOS commands have been groomed for Model 4 LS-DOS 6.3.1 syntax where possible.
- ☆ Best of all, **a 5.3.1 diskette is available as a replacement for your 5.3.0 diskette for $15** (plus $3 S&H in US and Canada, $4 elsewhere). There's no need to return your current master.
- ☆ The 5.3.1 diskette(s) come(s) with a 30-day warranty; written customer support is available for 30 days from the purchase date. Versions for the Model I and Model III are available. **If you do not already have an LDOS 5.3.0, order the 5.3.1 Upgrade Kit with 30 days of customer support for $39.95** (+$4 S&H). Some features require lower case or DDEN adaptor.

With a 20 or 40 MB MISOSYS Hard Drive connected to your TRS-80 Model III or 4, your computer will sail through data access.

Order any hard drive kit or unit from MISOSYS and we'll pre-install either LS-DOS 6.3.1 or LDOS 5.3.1 at no extra charge.

MISOSYS has been shipping complete drive kit packages since September 1989 which plug into Model 4/4P/4D and Model III computers; let us build one up for you! Our host adaptor, which interfaces the 50-pin expansion port of the TRS-80 (host) to the 50-pin SCSI port of the HDC, sports a hardware real time clock using a DS1287 clock module. With its internal battery lifetime in excess of 10 years, never enter date and time again. It even adjusts for daylight saving time! An available option is a joystick port and Kraft MAZEMASTER joystick with a port interface identical to the old Alpha Products joystick; thus, any software which operated from that joystick will operate from this one.

Software provided supporting the S1421 and 4010A controllers includes: a low level formatter; an installation utility and driver; a high level formatter; a sub-disk partitioning utility; utilities to archive/restore the hard disk files onto/from floppy diskettes; a utility to park the drive's read/write head; a utility to set or read the hardware clock; a keyboard filter which allows the optional joystick to generate five keycodes; and a utility to change the joystick filter's generated "keystroke" values after installation. Optional LDOS 5.3 software is available.

Aerocomp Hard Drives now available from MISOSYS

Twenty megabyte drive packages are currently built with a Kalok 3.5" hard drive; Forty megabyte packages use a Seagate ST251-1 28 millisecond drive. Drive packages are offered as '*pre-assembled kits*'. Your 'kit' will be assembled to order and fully tested; all you will need to do is plug it in and install the software. Drive kits include a 50-pin host interface cable and the hardware clock. Add a joystick for but $20 additional (see price schedule).
MISOSYS is also the sole source of remaining brand new Aerocomp hard drives. A limited number of NEW 5 megabyte units are available. All Aerocomp drives include status LEDs, software driver and formatter, power and host cables, and installation Job Control Language.

Prices currently in effect:

Complete MISOSYS Drive Kits:	
20 Megabyte kit:	$450
40 Megabyte kit:	$575
Joystick option	$20
LDOS software interface	$30
Aerocomp Hard Drives:	
5 Meg unit	$250
20 Meg unit	$400
40 Meg drive	$500
MISOSYS H/A with software	$75
Xebec 1421 HDC	$75
Adaptec 4010 HDC	$75
Drive power Y cable	$5
XT drive cable set	$5

Note: freight charges are additional.
Prices subject to change without notice

Pre-installed program collections at unheard of prices!

For a limited time, MISOSYS presents fantastic program sets pre-installed on your newly purchased hard drive. For just a modest additional charge for each collection, we will install all software on your hard drive and provide you with all documentation. But hurry, these offers are available for only a limited time and are offered only at the time of a new hard drive purchase.

Programmer's Powerpack - $50

MC: C compiler; MRAS: macro assembler; HartFORTH: FORTH compiler; DSMBLR: Z80 disassembler; CON80Z: converts 8080 to Z80; UNREL: converts REL to ASM; *The SOURCE*: 3-volume book set

Model 4 Business Bonanza - $50

LB Data Manager; PRO-WAM: Pop-up desk-top; Mister Ed: More wammies; LS-LED: text

Model III Games Galore - $40

Leo Cristopherson's Greatest Hits; Kim Watt Game Disk; Lance Micklus' Greatest Hits; The Gobbling Box; Bounceoids; Crazy Painter; Frogger; Scarfman; Space Castle; MISOSYS HD Joystick option

Model III Business Bonanza - $50

PowerMail: Mailing list; TextMerge: form letters; AFM: data file manager; FILTER Pack; LED: text editor

Model 4 Ultimate Utilities - $50

GO:Maintenance; GO:System Enhancements; GO:Command Utilities; Powersoft's Toolbelt; LS-Utility Disk

MISOSYS, Inc.
P. O. Box 239
Sterling, VA 22170-0239
703-450-4181
orders to 800-MISOSYS (647-6797)
Note: Our 800 ORDER LINE can accept calls from all 50 United States and CANADA.

MISOSYS, Inc.

MISOSYS sponsors a forum on CompuServe: PCS49

When you don't have to write in stone, don't let your editor weigh you down. You need SAID-86!

Editing was never so easy!

SAID-86 is a fast, flexible, full screen text editor for PC's. It is perfect for editing batch files, program listings, README files, CONFIG.SYS files, and anything you now do with EDLIN or the non-document mode of a word processor. Why struggle with huge editors; when all is said and done, SAID-86 will be your text editor of choice!

Check out this list of features

- ✓ WordStar-like editing commands are easy to use
- ✓ Pull-down menu system for commanding SAID-86
- ✓ Supports nine editing buffers with automatic swap to disk
- ✓ Supports up to 30 user-defined macros; 255 characters each
- ✓ Undelete the last nine deleted lines can save your bacon
- ✓ MOUSE support with automatic recognition
- ✓ HELP facility; shell to invoke DOS commands from SAID-86
- ✓ SAID-86 can expand or contract TABs

SAID-86 is reasonably priced at just $29.95 + $3S&H

MISOSYS, Inc.
P.O. Box 239
Sterling, VA 22170-0239
800-MISOSYS
(US&Canada)
or 703-450-4181
M/C & VISA accepted.
S&H are U.S. only.

TRSCROSS

Now you can transfer TRS-80 Model III/4 files directly to your MS-DOS disks right on your PC. Convert BASIC programs; Convert SuperScripsit document files to DCA-RFT. **Only $89.95 + $4S&H**

HartFORTH-86™

HartFORTH is a Direct Threaded Code implementation full 79-STANDARD FORTH which runs under DOS; the Virtual Memory that it accesses for storage and retrieval purposes is a file created and controlled by the operating system. HartFORTH's enhancements include functions to call the DOS file handling routines so that other files may be created and accessed if required. A library of standard screens is supplied with HartFORTH to provide often used extensions to the language, such as double length and floating point math, editing of source screens, string manipulation, arrays, etc. **Priced at $59.95 + $5S&H**

- HartFORTH programs can invoke other programs via **EXEC** and **EXEC.PROG**.
- Functions create new files from within HartFORTH, and allow the current Virtual Memory file to be changed for another and manipulated at the individual block level.
- Provides the recommended 79-STANDARD DOUBLE NUMBER STANDARD EXTENSION word set that implements 32-bit operations.
- **CASE:** and **SWITCH:** functions allow multi-way branching decisions to be taken with execution continuing in-line once the word branched to completes.
- String manipulators include: `"VARIABLE`, `"CONSTANT`, `"!`, `"LEFT`, `"RIGHT`, `"MID`, `"+`, `"COMPARE`, and `">`.
- DOS software and hardware interrupt vector access support via: `GET.VECTOR`, `PUT.VECTOR`, `THIS.SEG`, `DI`, and `EI`.
- V24 program input, output, and interrupt input support.
- *Overlay management words*: `FORGET.OVLY`, `OVLYNAME`, `PUT.DATA`, `OVLY.ENTRY`, `SAVE.OVLY`, `CORRECT?`, `LOAD.OVLY`, `NEW.OVLY`, `RUN.OVLY`, and `LEAVE.OVLY`.
- Screens provide trigonometric functions: `SIN.`, `COS.`, `TAN.`, `SIN`, `COS`, and `TAN`.

Are you still fussing with floppies for BACKUP? CMS' DJ10 or DJ20 tape drive from MISOSYS is your solution!

The Colorado Memory Systems' JUMBO tape drives fit all computers. Internal mounting in AT's, XT's, and PC's, they connect to your floppy disk controller. Tape adaptor board needed when two floppies are in use. Kit converts Jumbo to external use.

- In about 5.5 minutes, a DJ10 backs up 10MB's file-by-file - the fastest in the industry! 40MB's gets backed up in about 18 to 20 minutes. Uses industry-standard DC2000/DC2120 tape cartridges.
- DJ10/DJ20 plugs into your floppy disk controller to save cost, power, and a slot. Needs 5-1/4" (or 3.5" with faceplate) mounting slot.
- Optional adapter board mounts in your computer to provide a tape port. When used with the external DJ10/DJ20, it lets you share your drive between computers. Note: external adaptor includes "Tape Adaptor"
- DJ10 has up to 120 megabytes of capacity using compression with a DC2120 tape; DJ20 has up to 250 megabytes of capacity.

DJ10 Jumbo	$199 ($7S&H)
DJ20 Tape drive	$265 ($7S&H)
A10 Adaptor	$75 ($3S&H)
K10 External Kit	$110 ($5S&H)
DC2000 tape	$20.00
DC2120 tape	$25.00

NEW LOWER PRICES

Why buy just a FAX board, when the ZOFAX 96/24 from MISOSYS includes a 2400 baud modem for a few bucks more? Turn your PC into a FAX machine!

- ✓ Send and receive FAX from any CCITT Group III Fax Machine or PC Fax
- ✓ Auto receive and print incoming Fax messages ✓ Background receiving
- ✓ Distribut Fax messages to multiple destinations ✓ Fax mail merge
- ✓ Time schedule transmission to take advantage of low nightime rates
- ✓ 2400 bps Fully Hayes Compatible Modem
- ✓ Includes powerful but easy to use BITCOM and BITFAX software

Further price reduction: Just $125 + $7 S&H

EXPANZ!™ Disk Expander Card

With the EXPANZ! data compression card, you can boost hard disk capacity up to three times. EXPANZ! plugs into any open slot and intercepts calls to and from the disk controllers. Compresses and decompresses in real time. Requires PC/XT/AT or compatible running DOS 3.x or higher. **Now Just $150 + $7S&H.**

MISOSYS, Inc.
P.O. Box 239
Sterling, VA 22170-0239

800-MISOSYS
or 703-450-4181

Let our LB Data Manager solve your data storage problems

LB Version 2.2: A Flat File Data Manager with more powerful and easy to use features in this latest enhancement of Little Brother! Now with data conversion utility for DIF, pfs, Profile, dBASE, ...

We've added many features asked for over the past few years by LB users; yet LB is still about the easiest, most flexible data manager you can use for managing your data. Absolutely no programming is needed to create a database with numerous fields, construct input screens for adding and editing data, and create your own customized report. Quickly you define your data fields in response to LB's prompts, and then draw your data input screen using simple keystrokes - or have LB automatically create your input screen. In no time at all, you're entering data. Customize your printed reports with user-definable print screen definitions. LB is just what you need in a data manager! Now even more in version 2.2!

Data capacity per database:
LB supports up to 65,534 records per data base; 1,024 characters (64 fields) per record; and up to 254 characters per field.

Field types supported:
LB allows ten field types for flexibility: *alphabetic* {A-Z, a-z}, *calculated* {operations on "numeric" fields using +, -, *, /}, *date last modified* {YYYY/MM/DD automatically maintained}, *dollar* {±ddddddd.dd}, *floating point* {±ddddddd.ddddddd, *literal* {any ASCII character}, *numeric* {0-9, -, .}, *right-justified numeric* {flush right numeric}, *upper case alphabetic* {A-Z, automatic conversion of a-z}, and *upper case literal* { literal with automatic conversion of a-z}. All field types utilize input editing verification so invalid data cannot be added to a record. Field name strings can be up to 19 characters long.

Data entry and editing:
LB allows you to design up to ten different input/update screens to provide extreme flexibility for selectively viewing your database fields. Using a database password provides the capability of selectively protecting fields from being displayed or printed without entry of the correct database password, or they can be protected from being altered. This is quite useful in a work-group environment. Fields may selectively be established to require a data entry before a record being added or edited is saved. You can enable a special index file to keep track of records being added. This can be subsequently used, for example, for a special mailing to newly added *customers*. Flexible editing includes global search and replace with wild-card character match and source string substitution. Search and replace can be performed on all records, or on records referenced in an unsorted or sorted index file.

Record selection and sorting:
You can maintain up to ten different index files to keep your data organized per your multiple specifications. Records may be selected for reference in an index file by search criteria using six different field comparisons: EQ, NE, GT, GE, LT, and LE. You can select on up to eight different fields with AND and OR connectives. Index files can be left unsorted, or you can sort in ascending or descending order. By attaching a sorted index file, any record may be found within seconds - even in a very large database. LB even includes a special command for automatically finding duplicate records!

Report generation:
Report generation incorporates a great flexibility. Your report presentation is totally customized through print definition formats which you define on the screen as easily as you define the input/update screens. You can truncate field data, strip trailing spaces, or tab to a column. You control exactly where you want each field to appear. LB provides for a report header complete with database statistics: database name, date, time, and page numbers. A report footer provides subtotaling, totaling, and averaging for dollar, floating point, and calculated fields; print number of records printed per page and per report. Attach any of the ten index files and you control exactly what records get printed; even a subset of indexed records can be selected for printing to give you a means of recovering from that printer jam halfway through your 30-page printout. You can even force a new page when the key field of an index file changes value. Up to ten different printout definition formats can be maintained for each database. Reports may be sent easily to a printer, the console display screen, or to a disk file - useful for subsequent printing or downstream data export to other programs. Report formatting allows for multiple across mailing labels, multiple copies of the same record, or even printing one record per page for sales books. You can easily generate mail/merge files of address or other data for your word processor. Or you can use LB's built-in form letter capability.

Automatic operation:
For automating your processing needs, LB can be run in an *automatic* mode, without operator intervention. Frequently used procedures can be saved by LB's built-in macro recorder for future use. Entire job streams may be produced, so that LB operations may be intermixed with literally any DOS function that can be *batch* processed.

Maintenance utilities:
To make it easy for you to grow your database as your data needs grow, we provide two utility programs for managing your database. One allows you to construct a new database with an altered data structure and populate it with data from your existing database Another allows you to duplicate your database structure, copy or move records from one to another, or automatically purge un-needed records. **A third utility converts to LB from pfsFILE4, Profile4, DIF, dBASE II&III, and fixed record; also converts to DIF, dBASE, and delimited.**

Help is on the way:
The main menu even provides a shell to DOS so you can temporarily exit LB to perform other DOS commands. LB provides extensive on-line help available from almost every sub-command. A 200-page User Manual documents every facet of LB's operation.

Trade-up policy:
Send in the Table of Contents page from any existing database program and get LB Version 2 for half price.

Specify MS-DOS or TRS-80 version. LB is priced at $99 + $5 S&H (US; $6 Canada; $7 Europe; $9 Asia, Pacific Rim, and Australia). To trade up from any other database, send Table of Contents page and $49.50 +S&H. Remit to:

MISOSYS, Inc.
PO Box 239
Sterling, VA 22170
703-450-4181 or orders to 800-MISOSYS

MISOSYS, Inc.

Aerocomp Hardware is now available from MISOSYS

Model I DDen Controller (DDC)	$45 + $6S&H
Model III/4 FDC board	$45 + $6S&H
Model III/4 RS232 board	$45 + $6S&H
Model III/4 RS232 Kit	$50 + $6S&H
Aerocomp 5 Meg HD	$250 + S&H
Aerocomp 20 Meg HD	$400 + S&H
Aerocomp 40 Meg HD	$500 + S&H
MM CP/M 2.2 HD drivers	$29.95 + $3S&H

SUPER UTILITY PLUS

The greatest utility ever written for the TRS-80!

Reads, repairs & works with popular TRS-80 DOS'Mods I, III, 4!

SU is completely menu-driven, is configurable for all the popular TRS-80 DOSs, regardless of density, track number, number of sides, or system used. SU removes or decodes passwords, reformats a disk without erasing the data, fixes problems, backs up most protected disks, etc. SU has over 65 functions and features. A fantastic buy. Does not work on hard disks. (Specify Mod I&III or Mod4)

Super Utility+ Half-Price special: $22.98 + $5S&H

The Cornsoft Group
Model I/III Action Games Special

All five games on a single disk for one low price!

You get Bounceoids, Crazy Painter, Space Castle, Scarfman, and The Official Frogger; all five for $20 + $3S&H

PRO-WAM Sale Price Plunge

If you are not using PRO-WAM on your 128K Model 4, you're not using your 4! You'll get a pop-up desktop manager with ADDRESS, BRINGUP, CAL, CARD, CALC, PHRASE, and more. Export|Import across windows. PSORT your data files. $37.48 + $6S&H

Sale prices good through Mar 31st.

MISOSYS, Inc.
P. O. Box 239
Sterling, VA 22170-0239
703-450-4181
orders: 800-MISOSYS

DoubleDuty doubles your 128K Model 4
Now on sale at half price!

DoubleDuty divides your Mod4's memory into three complete and independent partitions. Two operate as they were each their own 64K computer. Get the best task switcher you can buy. Our 2.6 release also works with extended memory. If you thought you needed a second computer, think again. At $24.98 + $3S&H, you can't afford to not have DoubleDuty.

New LB Version 2.2.0 released. Details inside!
International LS-DOS 6.3.1 now available. Details inside!
MISOSYS acquires Aerocomp's remaining hardware!

MISOSYS, Inc.
PO Box 239
Sterling, VA 22170-0239
U.S.A.

Contents: Printed Matter

BULK RATE
U. S. POSTAGE
PAID
Sterling, VA
PERMIT NO. 74

Attention Postmaster: Address Correction Requested
Forwarding and return postage guaranteed