THE HUGgers
HOOSIER USERS GROUP
People Helping People

September/October 1995 The HUGgers Newsletter Volume 14 Number 5

Thanks to the following who have renewed their HUG membership for the year ending in May 1996 (as of 9-1-95):

Ricky Bottoms
Walter Burns
Fred Edstrom Jr.
Greg Larson
George Lett
William M. Lucid
Bob Mikels
Stephen Moon
Bryant Pedigo
Roger Price
Bob Stahlhut
Dave Stultz

And thanks to our new member:

Ralph Metz

If your name isn't listed here, you may have forgotten to renew. Please renew your HUG membership today.

NEW S&T BBS

Hoosier Users Group
300/1200/2400/4800/9600
Baud, 24 Hours Daily

(317) 782-9942 8N1

Now with a Hard Drive
60 MEG ON LINE

MONTHLY MEETING LOCATION
WHITE HOUSE NEXT TO THE
ST. ANN'S SCHOOL
2839 S. McCLURE
INDIANAPOLIS, IN
AT 2:00 P.M.
SEPTEMBER 17, 1995

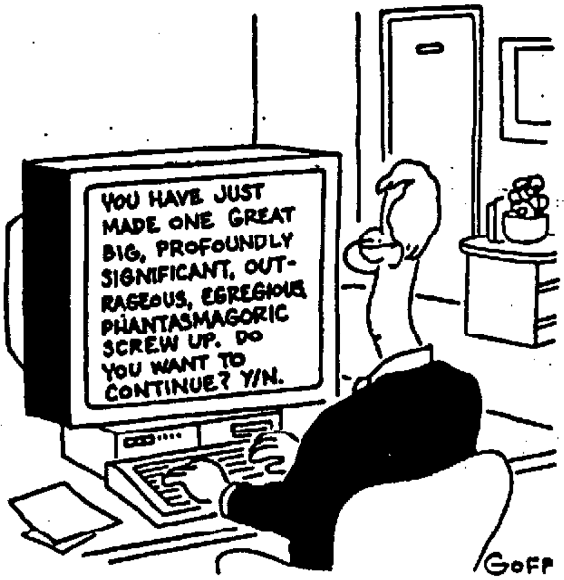
Spellbound

I have a spelling checker,
It came with my PC;
It plainly marks four my revue
Mistakes I cannot sea.
I've run this poem threw it,
I'm sure your please too no,
Its letter perfect in it's weigh,
My checker tolled me sew.

—Quoted by Pennye Harpe

HUG OFFICERS

President	Ricky Bottoms	392-2312
V.President	Bryant Pedigo	255-7381
Secretary	Greg Larson	783-4575
Treasurer	Greg Larson	783-4575
Librarian	Bryant Pedigo	255-7381


YOU HAVE JUST MADE ONE GREAT BIG, PROFOUNDLY SIGNIFICANT, OUT-RAGEOUS, EGREGIOUS PHANTASMAGORIC SCREW UP. DO YOU WANT TO CONTINUE? Y/N.
GOFF

What's old? by Dan H. Eicher

It has been awhile since I have written an article for the newsletter, so I was starting to feel guilty. Here's what I have been up to with my TI!

Several times this summer me and my girlfriend have taken trips to the park. During this time we both spend a lot of time writting and working on different papers; most of the time this would result in us coming back home and entering what we had written in our notebooks on the computer.

Everytime I have done this, I have thought, gee... I wish I had a portable with me that was small, lightweight and had batteries that would last for more than five minutes! Five minutes being how long the batteries on the portables at work last. Then I remembered the Tandy/Radio Shack Model-100.

In 86 I bought a Model-100. It has the following features.

- o A 40 column by 8 line LCD display. By redefining the character set you can easily get 60 columns!
- o A built-in 32k Ramdisk.
- o Built-in Basic, Telcom Program and Text Editor.
- o 300 BPS modem, RS232 and PIO port.
- o Real-Time Clock.
- o Battery life of 40 hours+

Unfortunately, I sold my Model-100 about 8 years ago for assorted reasons. So, I just hopped on the Internet (comp.sys.tandy) and waited until one showed up for sale. It took about three months before I found one at a good price, but got one for sixty dollars. I built a cable to hook it up to my Geneve and now can enter the text of articles, documents or manuals while sitting under tree, and then, when I get home, bring it into Funnel Writer for cleanup and printing!

The Tandy and Geneve work very well together, and if interest warrants I will demo the connection at the next Hugger metting. If you went to Lima (or watched the tapes), Charlie Good shows a very similar arrangement, only he is using the TI CC-40 and a Hex-Bus RS232!

Last week Micropendium came in and had a review of an obscure DataBiotics cartridge called TI Workshop. This cartridge has everything you could ever need except Extended Basic built-in.

These features include: Memory Manager, Disk Manager, Program Loader, Debugger, Editor/Assembler. All of these programs are enhanced over what you will usually find. All run from the cartridge no disk needed, but you will need to have 32K memory expansion of some type on your system.

The utility of this cartridge is wonderful. How many times have you been at fair or a users group meeting and someone has been shuffling through a bunch of disks, looking for a Disk Manager program to copy a couple of files or an editor to change a configuration file?

Many times when Jeff and I have been working on TI systems trying to debug piece of hardware or software we have needed either a debugger, an editor or file manager. Granted, we could have loaded up a ramdisk with Funnelweb and all its utilities or made up a floppy ect... This approach doesn't work for variety of reasons, they include, but are not limited to...

- o The Piece of hardware we are trying to debug is a bad floppy controller.
- o Adding a ramdisk and its DSR would complicate debugging.
- o We are working with a console and no Pbox or a mini-expansion unit.

This cartridge seemed like an answer to a long lost prayer. I quickly called Tex Comp 818.339.8924. The phone was answered on the second ring of my first attempt ---- by a HUMAN! I placed my order via Discover and asked that the package be sent so that I would get it by Friday or Saturday of that week.

This was on a Tuesday and like clock work, the package arrived in good shape on Friday.

This package works like advertised (except for an incompatablity with the Myarc 512K card, which I am trying to find a work around). I would HIGHLY recommend this cartridge and Tex Comp. For a full review of TI Workshop, recommend Charlie Good's review in the July '95 Micropendium.

LATE BREAKING NEWS: Version 6.0 of Ed Swartz' popular TI emulator for the PC is now out and available on our BBS! I found one error so far in a batch file.

The file LST2CNF.bat should have a %1 on the first (and only) line of the file. The setup is kind of tricky, but if you have any questions I will try to answer them on the Hugger BBS. One thing Ed managed to do in this version that thought no one would ever be able to do, is get LPC encoding ported to a sound blaster card!

If anyone wants, I will bring the TI Workshop and its 60-page manual to the August or September Huggers Meeting for your inspection/demonstration.

Dano

BITS,BYTES&PIXELS

BRUCE HARRISON'S ~~FAIRWARE~~ Freeware

Bruce Harrison has released lots of ~~fairware~~ freeware over the last several months. These disks are in the Lima software rary available to members and available to user groups who want to copy the disks at the May 13/14 Lima MUG Conference. The following is taken from Bruce's documentation of his fairware offerings:

NEXT COLUMN

DISK 937B MUSIC BACKGROUND INSTRUCTIONS

Thanks to Barry Traver, another little "gem" appears on the TI scene. This Public Domain software allows the Extended Basic programmer to have some music or sound effects running "on background" while a program is waiting for user input or even while editing a program. Barry Traver asked whether this would be possible, and we thought it probably could be done. This disk is a "twofer", in that it gives you two new ideas for the price of one. In addition to the music on background that just continues, you get another utility that allows use of music to time the ACCEPT AT operation, so that the user's input opportunity ends when the music stops. The code is written by Bruce Harrison, with help from Harry Wilhelm. The disk includes five demo programs to show how this works, plus the source and object files for the Assembly language - sub-programs that make the music, these instructions, and other stuff to allow the XB programmer complete access to this program.

DISK 935A INSTRUCTIONS for PASSWORD PROGRAMS

This Public Domain product by Bruce Harrison is intended for people who have Horizon Ramdisks on their TI systems. So far as we know, it's of no use whatsoever to those without Horizon Ramdisks. The product provides all you need to put in a "password" protection so that only those who know the password can gain access to your computer. It's not absolute, because the Menu program can be tricked on startup, but simply turning on the P-Box and Console will produce a prompt for the password, and the computer will not respond to anything until the correct password has been entered. While the password is being typed, it will not be readable on the screen, so over-the-shoulder screen readers will not see your password. Once the password has been entered on startup, everything will work normally without needing re-entry of the password.

DISK 929 FONT CONVERTER INSTRUCTIONS

This package has a utility designed to convert the screen fonts designed by Jim Peterson into CHARA1 type files. The fonts may also be edited with this same utility to create variations on Jim's original fonts, or to create whole new fonts of your own design. This package is dedicated to the memory of our dear friend Jim Peterson, whose passing has touched everyone in the TI "Community". The package is being released as Public Domain software, which may be copied, shared, uploaded to BBS systems, and such, without compensation to the authors. The materials in this package are all based upon original work done by Jim Peterson, as modified by Bruce Harrison.

The Concept

Jim Peterson's original concept for his "screen fonts"

NEXT PAGE

was to permit use of various different fonts with Extended Basic programs. He designed 127 fonts for such use. Our extension of this concept is to allow the fonts to be converted to CHARA1 type files, so that they may be used along with Assembly programs that use this type file for loading character sets. Files converted by our program may still be used with Extended Basic just like the original font object files, but these object files may also be loaded under Editor/Assembler Option 3 and used to create CHARA1 type font files.

DISK 928 DRAWING EXPERIMENT INSTRUCTIONS

The program is a set of E/A Option 5 files called DRAW1, DRAW2, and DRAW3. It can be run from E/A option 5, (DRAW1) or from XB via LOADDRAW. When it runs, a menu will appear on-screen.

Here are the options:

1. MAKE DRAWING - for starting a new drawing from scratch.
2. SAVE DRAWING - to save what you've created.
3. LOAD DRAWING - to re-load a saved drawing.
4. RECALL DRAWING - to bring back the drawing currently in process.

5. LOAD FONT - to load a CHARA1 type file as a character set. (try Funelweb's C1 or C2 files, for example.)
6. PRINT DRAWING - will work with almost any Epson compatible printer. (Tested with Star NX-1000, also works on Gemini 10X & SG10)

DISK 926A TIMEOUT INSTRUCTIONS

This disk contains a new utility called TIMER, which is intended for use with Extended Basic programs. The utility, which operates as an Interrupt while Extended Basic programs are running, is used to provide an automatic time limit on INPUT, ACCEPT AT, or CALL KEY statements in the Extended Basic program. The Assembly source code, provided on the disk, was written jointly by Bruce Harrison and Harry Wilhelm. The disk is being released as Public Domain, and may be copied, shared, uploaded to BBS systems, and so on, provided only that all contents of the disk are copied.

DISK 922B NEW LOADER INSTRUCTIONS

On this disk is a new loader for use with Extended Basic. It's designed to permit loading and running Assembly Program Files (a.k.a. Option-5 files) from Extended Basic. The disk is supplied as Public Domain software, and may be copied, shared, uploaded, etc. without compensation to its author. Except for the included GPL/DSR link routines, it's all written by Bruce Harrison.

DISK 863B TIME CALCULATOR INSTRUCTIONS

The Harrison Time Calculator is an Extended Basic program with built-in Assembly enhancement. Its purpose is to handle calculating numbers in Hours, Minutes, and Seconds. The ti inputs may be made in either the "normal" 12 hour clock format or in the "military" 24 hour format.

The program is called TIMECAL. At startup, the program simply puts a menu on the screen. Six items are on the menu, so selection requires only one keypress on the keys 1 through 6. The six selections are:

1. ELAPSED TIME
2. CUMULATIVE SUM
3. TIME MULTIPLY
4. TIME DIVIDE
5. SET 12 OR 24
6. EXIT PROGRAM

DISK 869A METRONOME Operating Instructions

Metronome does exactly what its name implies, providing a stable "tick" at a selected number of beats per minute for musicians practicing their instruments. Two versions are provided, both in the form of Option-5 Editor/Assembler Program Files. The one called METRONOME is designed for use on U.S. systems, with 60 Hz NTSC Video systems. The second version, called METROEUR, is designed for operation on European systems, with 50 Hz PAL Video systems. Both versions will run on any TI-99/4A computer, but the timi will be accurate only if run on the appropriate system. Fo example, if one runs the EUR version on a U.S. system, the number of beats selected will take 50 seconds to complete. (e.g. 60 beats per minute will produce 60 ticks in 50 seconds.)

DISK 869B OPTION-3 TO OPTION-5 CONVERSION - STEP BY STEP

Let's say you have an E/A Option-3 file which you want to convert to Option-5, that you know the entry point, and you know that it's relocatable. (We'll tell you how to find out in just a bit.) Follow these steps in order:

DONE

PASCAL WITH THE TI 99
by Anders Persson

Charles Good wrote an article in the January issue of the Lima newsletter this year, concerning the p-code card for the TI 99/4A. When reviewing software, it's usually impossible to use it enough to really find out how to take the best advantage of the program. That's especially true when the review concerns something as complex as the UCSD p-syste Since I was mentioned in that article, I couldn't restrain from adding some of my own comments.

NEXT PAGE

Bits, Bytes & Pixels

The idea of the p-system

Charles asks what's portable with UCSD Pascal? As things has turned out, the answer to that question is, unfortunately, nothing. Due to the problem with different disk formats, there has never really been any portability between different implementations of UCSD Pascal. It's of course possible to transmit the source code via serial ports, with or without modems, between different computers, but that's about it.

However, there is something that really is portable with UCSD Pascal. The original Pascal definition, by Nicklaus Wirth, is practically useless for real world interactive programming. That led to different, more or less clever extensions to the original. Some extensions were targeted to improvement of the language's behavior in limited memories, while others improved the ability to program operating systems in Pascal.

Since features introduced with version IV.0 (the one implemented on the TI 99) has been an inspiration for various designers of Pascal compilers, a lot of Pascal code can be typed in and executed on the TI. This is true also for ,code neither originally intended for the UCSD system in general, nor for the TI in particular. I have myself transferred a substantial program (compiled EXE file on the PC is 55 kbytes) from the TI to Turbo Pascal version 4, with only minor source code changes. These were mainly related to the character wide PC screen and somewhat different file handling procedures.

Since Turbo Pascal is continually being developed, in order to meet both new hardware (faster, bigger) and new operating environments (like Windows), the difference does of course grow with time. As Charles pointed out in his article, there is no current development of the p-system.

The TI implementation

The unique feature of the TI implementation is the p-code card itself. One reason for this card was probably the always apparent fear that, unless it was made impossible, people might copy software and execute it on their 4A's. By making the hardware p-code card an essential part of the Pascal system, that was much more difficult. But an advantage with this idea is that you get a ROM-disk, containing the operating system and the p-code interpreter, without wasting valuable RAM memory. We must remember that this was more than ten years ago, when 64 kbytes of RAM in a microcomputer was about as much as anyone had ever heard about. This was also a time, when running Pascal on a home computer, resulted in an impressed "Wow" from all computer friends that looked at it.

Since UCSD Pascal IV.0 wants to do a lot of things, it depends heavily on its memory management. There simply isn't room for everything at the same time. This results in a lot of disk activity when using the system. In the beginning, that was quite a nuisance, but since I made a RAM disk for my computer, that problem isn't so severe any longer. Adding four 360 kbyte disk drives (CorComp controller) also makes the system run better, since all system software can be on line at the same time, together with the application under development.

With this system, I've done the major part of all software development on my computer. Built in memory management, the capabilities for structuring your programs that's inherent in Pascal, easy assembly interface and the adaptability of the system are some of the reasons. I've modified the p-system to include true pre-emptive multitasking, full screen turtle graphics and to take advantage of my redesigned 80 kbyte RAM console.

A disadvantage of the TI implementation is that it's not complete. The unit KERNEL, for example, isn't included on the disks, at least not with the interface section intact. That unit is essential for easy system programming. Another program that's missing is the Native Code Generator, which converts p-code programs to the assembly instructions of the processor used with the target machine, the 9900 in this case. This utility, when available, is used to speed execution of programs at the cost of code size.

To use the p-system

To make the system run faster, the SYSTEM.STARTUP program should copy all essential system files to a RAM disk, to begin with. That makes loading the Filer, a disk. manager with features still difficult to find in later programs, fast enough.

I haven't had the chance to run any of the never released programs that Charles obviously has access to, but I can at least sort one thing out. A "fixed pitch printer" is one that prints like a typewriter, while a "variable pitch printer" is one capable of proportional spacing in its printout. Not an obvious feature for all low cost printers more than ten years ago.

It's not necessary to press "I" to initialize each time disks are replaced. The p-system allows you to refer to disks by name, not by number, and automatically tracks where a particular disk is located. If you move a disk to another drive, the system will look for it only once, and then remember where it's inserted. Myself, I rarely use the drive numbers, but refer to my disks with their names. Together with the ability to refer to the system disk with an asterisk and a prefixed disk with a dollar sign, you actually end up typing less than you do with the DSK# system.

It's true that you first have to format a disk and then Zero its directory before you can use it with the p-system, but you don't need an ordinary disk manager to accomplish that. The DFORMAT utility takes care of formatting a disk. Unfortunately, the original DFORMAT program, supplied on the Utilities disk, can't handle double sided disks, although it claims it can. I've developed an alternative, which handles every disk format supported by TI and CorComp controllers, including utilizing the CorComp variable interlacing feature. That means creating disks that are speed optimized for Pascal.

NEXT PAGE

The V(olume command displays all units currently available to the p-system. Unit 1, CONSOLE, is the keyboard and computer screen. Number 2 is SYSTERM (SYStem TERMinal), which is the same thing as CONSOLE, except that there is no implied echo on the screen of characters typed on the keyboard. Good if you want to read a key without displaying the character, since there is no equivalence to the CALL KEY statement in Pascal. Better still is to use an assembly routine that looks into the type ahead queue, to see if there is a new character stored there.

Units 7, REMIN (REMote INput), and 8, REMOUT (REMote OUTput), refer to the same physical port. Usually this is a serial port. On the TI, that's RS232/1 or RS232/2, dependent on where you have your printer connected. Setting the name of the physical port used for these units, as well as for unit 6, PRINTER, is done by the MODRS232 program on the Utilities disk. The source code of this program is included, together with the suggestion (in the manual) to use this procedure when you want to change the configuration within your own program. The REMOTE unit is mainly intended for communication purposes other than printing.

Unit 14, OS, contains the procedures that actually are the operating system. Most of these reside in a file called SYSTEM.PASCAL. From a hardware point of view, this is the GROM chips on the p-code card. The ROM chips contains the PME (P Machine Emulator, that interprets p-code) and low level I/O routines.

Pascal programs

Charles mentioned in his article that the p-system editor is an 80 column editor, using windowing left/right. That's true, but it's also true for the entire p-system. All programs executing under the p-system, has access to an 80 column screen.

It's a common misunderstanding that the eventual structure of a Pascal program should depend upon the structure of the language itself. That's absolutely untrue. It's perfectly possible to misuse Pascal to such an extent, that the resulting code couldn't be understood even by the original creator. The structure of a program is always a result of a structured programmer, not any particular language. However, Pascal delivers the tools a structured programmer may need to accomplish his task, of writing understandable code. The main benefit of Pascal is not the indentations that are allowed (but not compulsory), but the data structure concept. Unlike traditional BASIC, which knows nothing more complex than the array, Pascal allows a programmer to declare his own data structures, containing any mixture of data types required to handle a specific problem.

In conjunction with the separately compiled unit concept, designed mainly to facilitate memory management, it's also possible to declare a package of procedures and data structures. The benefit of this construction is, that these procedures work only with their intended data structures. That reduces the risk of hard to find errors, resulting from the correct procedure applied to wrong data. Apart from the fact that the unit is static (new copies can't be created during execution), this is similair to the class concept, found in many more modern and object oriented languages.

Charles states that he has found the p-code card as useful as the Thermal Printer, which I understand wasn't used too much. I fully agree with him in that there isn't much ready to run software available. Still I think it's the p-code card that has motivated the existence of my 99/4A. But that's only because of its overall performance as a development system for complex software. The p-system gives quite a lot of useful things, like a library system for commonly used procedures, memory management with automatic roll in and roll out of code segments, easy assembly language interface, floating point capability when needed and integers elsewhere and a lot of technical information about how it works. It also gives compatibility with , if not portability between, Pascal compilers designed for other computer systems.

Getting technical

The p-code card is located at CRU address 1F00. The reason for this is simply the fact that the p-system never releases control, once it's got it. In order to allow all other cards to execute their power up routines, the p-code card was best placed last in the search chain. What about the CorComp controller, then?

Well, since the disk controller is allocated CRU address 1100, another solution was used with the CorComp controller. That card takes command, but also takes responsibility for executing the power up routines of all other cards. The CorComp card assumes these cards to behave nicely, i.e. release control to the caller upon completition of their tasks. Usually, the only drawback with this scheme is that power up routines in GROMs are never found. These routines are rare, but do exist, for example in the Terminal Emulator II module.

When confronted with the p-code card, the usual takeover by the CorComp disk controller is prevented. The p-code card never returns from its power up routine. This explains how the p-code card appears to take control before the CorComp disk controller.

My solution to the somewhat egocentric behaviour of the CorComp controller is new EPROMs. These were once available on the market. Together with some other minor modifications, they put an end to the idea of taking control of the 99/4A before anything else. Definitely recommended.

If anyone are interested in commenting this article, you are welcomed. Either in Bits, Bytes & Pixels, or directly to me:

Anders Persson
Drottninggatan 35
S-341 36 LJUNGBY
Sweden

DONE

DOS utilities for TI-Artist files

PC users have new way to use TI artwork

By JOHN KOLOEN

TI users who also use PCs can display and print graphics created with TI-Artist using a set of programs developed by Jeffrey A. Kuhlmann.

The programs, which run on an IBM-compatible PC in DOS mode, allow users to view TI-Artist instances and pictures in monochrome. See the sample below. I outputted the graphic to a laser printer, which I don't normally have connected to my Geneve. This saved time and was a relatively painless procedure.

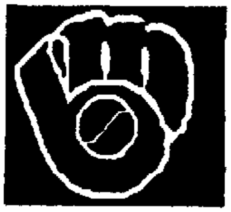
Kuhlmann notes that his software isn't user-friendly. What this means is that not only does it operate only through DOS, but you activate the programs from the command line, rather than a menu. However, this is not difficult to anyone familiar with DOS. Here are some of the commands and how they are implemented:

REVIEW

- showtia — allows IBM to view TI-Artist Instances — usage: showtia file_name graphic_mode (graphic mode is optional)
- showall — allows IBM to view all TI-Artist Instances on drive — usage: showall wildcard graphic_mode (graphic mode is optional)
- showtip — allows IBM to view TI-Artist _P file — usage: showtip file_name header_length (usually 128)
- sidetiai — a program to view TI-Artist Instances sideways,. so <shift><printscreen> will work

Also included are routines to view TI CSGD graphics. These work similarly to the TI-Artist programs so I won't list them here.

How you get the TI-Artist files into your PC is up to you. There are numerous ways, from using programs like PC-Transfer to uploading them to an electronic service from your TI and downloading them into your PC.



Sample of TIA file printed from a PC.

As would be expected from a barebones set of programs, the documentation for these programs consist of two small text files on disk. Since it's a PC program, you'll need a PC to read them. But, brief as the docs are, they're enough to get you going.

For more information about the programs, write Kuhlmann at CMR 4/6, Box D, APO, AE 09140.

MICRO-REVIEWS

Mailing List Manager like a dream come true

By CHARLES GOOD

Mailing List Manager by Bill Gaskill

Ever since I got my first 99/4A back in 1982 I have been looking for the perfect mailing list (name, address, phone number) software. I started out years ago with Personal Record Keeping and have made extensive use of three other name and address programs, each a little better for me than the previous software. My quest has been similar to searching for the Holy Grail in that I often appear close to my goal, but never quite get there. Features of existing software just don't quite seem to fit my needs. Some of the problems I have encountered over the years with such software include:

- Kludgy data entry screens and difficulty in correcting incorrectly entered data.
- Inability to enter enough data of the type I desire, such as entering a "country" and very long phone number for foreign addresses.
- Limited number of names one can put in a data file. This happens if an entire data file has to be put into memory each time you use the program.
- Difficulty in deleting a single name from a large data file. You'd be surprised how difficult it is to accomplish this very basic task with some existing mail list software.
- No ability to send special codes to my Gemini 10X printer so it will print mail labels in dark "emphasized" letters rather than thin dot matrixy "draft mode" letters.
- Slow, or no sorting ability. If sorting ability exists, you sometimes are forced to sort your list in memory each time you print sorted mailing labels since the software can't create a sorted data file from an unsorted data file.
- Etc.

NEARLY PERFECT

The closest I have come to my perfect mail list software is Mailing List Manager (MLM). The problem with switching to a new mail list program is that you have to start from the beginning. You have to manually type in all your data into the new program, data you have already long ago typed into the old program. After experimenting with MLM I was so impressed with MLM's features that I decided to transfer my user group membership and newsletter exchange mailing list from the software I had been using to MLM. This means that I spent 3 hours at the keyboard typing into MLM and checking over 150 names and associated data. I consider this time well spent.

I am very happy with MLM, and that is why I am devoting my entire column this month to this one product. I only review good software for MICROpendium, but that doesn't mean that I personally have a use for the software I review. Unlike many of the other software packages I review here, I am now actually using MLM on a regular basis.

MLM can handle name and address files of unlimited length, subject only to the physical limitations of the media where the file is stored. Each name and address data group takes one disk sector. Although sorting within MLM is limited to files with no more than 1,000 names, an alternative means of sorting larger files is provided. The fully functional fairware version of MLM comes on a SSSD disk and will work very nicely on systems with only one SSSD drive. You can use a separate SSSD data disk with up to 358 names. The program prompts you to insert the program disk into a drive whenever that is necessary.

MLM is set up to have its system files run automatically out of any floppy drive or RAMdisk because it looks for a volume name, not a drive number, when it loads parts of itself into memory. It also works with hard drives and allows you to have up to 23 characters in path names.

MLM is written in Extended BASIC, with various assembly CALL LINKs to speed things up. Default printer names, printer control codes, label spacing, and data file paths can be changed on the fly from within MLM, and you can also permanently change these defaults by changing the Extended BASIC code.

The following data fields are found in each name and address record:

lname
fname
address
city
state
zipcode
nation
group code
home phone
work phone
fax bbs phone
dates
notes 1
notes 2

You can leave any of these fields blank and update them later. Group code might be used for user group affiliation. Dates might be used to indicate when a user group membership expires. The date entry is printed on mailing labels. I particularly appreciate the nation field, something rarely found in mailing list software. This makes it easier to deal with airmail mailings to international locations.

MLM can handle name and address files of unlimited length, subject only to the physical limitations of the media where the file is stored.

ROOM FOR NOTES

Actually you can put any text you want into any of these fields. Thus, you have room to put lots of notes and comments into the last seven fields listed above. You might, for example, put XMAS in the Group Code field to identify those who are on your Christmas card list. It is not necessary, for example, to put numerical digits in all three phone number fields. You can just as easily enter some text.

When you finish typing your data in all the fields for a particular new or updated name and address and press "C" to continue, the data is immediately written to the data file. There are no complicated "exit the program" procedures required to make sure all files are closed. Wherever you are in MLM just press FCTN-9 (Back) a few times for a quick exit to the title screen. All files are safely closed when you do this, so your data remains secure.

You can sort an entire data base on any one of the data fields. When a list is sorted the computer reads the list into memory, sorts the list, and creates a new sorted fil on disk. This leaves the original unsorted file intact. Doing all this to a list of 140 names stored on a Horizon RAMdisk (sorting an unsorted list by ZIP code) took me only 1 minute 35 seconds. By 99/4A standards that is fast. An assembly language sort is used.

You can search a data file using either one or two key words in either one or two data fields. If you sort by two key words you only get a hit if both words are found. Searching for only one key word requires that you specify the same key word and data field twice when asked for the first and second key word to be used in the search. This takes some getting used to.

Partial strings can be used in these searches. For example, if you can't remember if L.L. Conner is spelled "Connor" or "Conner" you can search for "CONN". Data entry is automatically in uppercase, so you don't have to worry about what is and is not in uppercase when you do a search.

THREE WAYS TO DELETE

There are three ways to delete names form a file. In each case a new file is written without the deleted names, leaving the original file intact.

1. To delete a single name or a few manually selected names from a file, first display on-screen each name to be deleted and mark it with a caret (Shift-6) in the first space of the lname field. You can mark any number of names this way. Then select "Delete Names" from the main menu and a new file will be written leaving out the marked names. This procedure is easy, safe (you still have the old file), and fast.

2. You can also do a global delete, creating a new file that has all the records from the old file except those containing a text string you specify. For example, you could create a file that omits Christmas list people.

3. Finally, you can create a new file that contains only records from the original file that have a text string you specify. This i sort of the opposite of No. 2 above. You can make a file that contains only Christmas list people. In creating subsets of files based on text strings, you can use either

MICRO REVIEWS—

one or two text strings as described above for searching by key word.

If you can't remember a data file name you can, from within the program, display a disk directory. Then you can optionally delete any file from the disk.

Reports can be printed in either of two formats. You can also print mailing labels of the entire file or a single label of only the name currently displayed. It is possible to print labels or reports of subsets from larger lists. To do this the software lets you create an index file of a larger data file with pointers to specific records in the larger file. For example, you can make an index of all your names you have marked XMAS in one of the comment fields. Later you can select "Print using an index" to print mailing labels to your Christmas card list.

You also get some unusual software extras with MLM, all of which can be run within MLM:

- There are two different free-form mailing label editors They let you compose mailing labels on the fly and print multiple copies of these onto fan-fold labels. Examples would be return address labels or "Do Not Bend" labels for packages containing floppy disks. You can also load in templates of previously composed labels and print these. One of the two label makers also lets you print disk labels and automatically advances to the next label to continue printing if all the disk's file names won't fit on one label.
- MLM also has a 40-column text editor, which is great for writing short letters or keeping records of your correspondence in your database. This text editor has many features of the TI-Writer editor and is compatible with TI-Writer's text files. You are limited to one page of text at a time. There is no word wrap and no automatic margins.

> There are two different free-form mailing label editors They let you • compose mailing labels on the fly and print multiple copies of these onto fan-fold labels.

The software that compares most closely with MLM is Asgard's Mail Room, by Larry Tippitt. Both products have their advantages and disadvantages.

I reviewed Mail Room in one of my earlier MicroReviews columns. Mail Room has an 80-column version and allows you to use a modem to dial any phone number stored in its database, features not found in MLM. Advantages of MLM compared to Mail Room include MLM's "nation" and generous comment fields and the ease of permanently creating sorted data files and of deleting names from MLM data files. All Mail Room users should have a look at MLM.

Send me $1 and I will send you MLM on a DSSD disk ($2 for two SSSD disks). The author asks $15 to register your copy of the program. Registered owners will receive an expanded hard copy of the instructions and an update with even more features than those described here for the fairware version. You might consider saving the dollar and immediately send Bill $15 with a request that he send you the most recent version of MLM.

QUICK SEARCH OF ANY DATABASE

Lets say you have a data file created with MLM, TI-Base, First Base, PR Base, PRK, or just about any other database software usable on the TI. If all you want to do is display a particular name, address, and phone number (or some other data within a data file) on-screen, there is a quick and easy way. This method is usually much faster than loading software such as MLM and using the software's internal search engine. Instead of doing that, use an assembly language "find string" to quickly display a data file sector with the desired text you are looking for. Funnelweb's Disk Review or John Birdwell's DSKU can be used to do this. These are programs you probably already have on your computer's menu system.

From a Disk Review disk directory, move the cursor next to a file you want to search and press I (for inspect). Select 2 (File search) and then 1 (ASCII string). Press Enter when the cursor appears over the first question mark and enter your search string over the second group of question marks.

From DSKU, select File Utilities from the first menu and then select Find String. Enter the file name, drive number, "A" (for ASCII), and the text you want to find. If your file is on a Horizon RAMdisk, such string searches usually take less than 10 seconds. MLM and many other databases automatically store text only in uppercase, which takes much of the guesswork out of a string search.

ACCESS

Bill Gaskill, 10 Cypress Court, Grand Junction, Colorado, 81506.

Charles Good, P.O. Box 647, Venedocia, Ohio, 45894. Evening phone: 419-667-3131. Internet email: good.6@osu.edu

APPLICATION FOR MEMBERSHIP

Below you will find an application for membership to the Hoosier Users Group. Active membership entitles you to the Newsletter, up and download on the HUGbbs, attendance and voting rights at regular club meetings, access to the HUGger Library of Programs, special club activities and special guest speakers for one year. C5

Make check or money order payable to Hoosier Users Group. Send completed application to:

HOOSIER USERS GROUP
P.O. Box 2222
Indianapolis, IN 46206-2222

please print cut on line

Date: ____________

Name ______________________________ New membership:

Address ______________________________

City/State ______________________________ Renewals:
$22.00

Phone number ______________________________

HOOSIER USERS GROUP
P.O. Box 2222
Indianapolis, IN 46206-2222

Forwarding and Address
Correction Requested

TIME DATED
September 17, 1995
MATERIAL

May 1996

Dan H. Eicher
2720 Palo Verde Ct.
Indianapolis, IN 46227

Hoosier User's Group S&T BBS
300/1200/2400/4800/9600 Baud 8N1
317-782-9942 24 Hours Daily