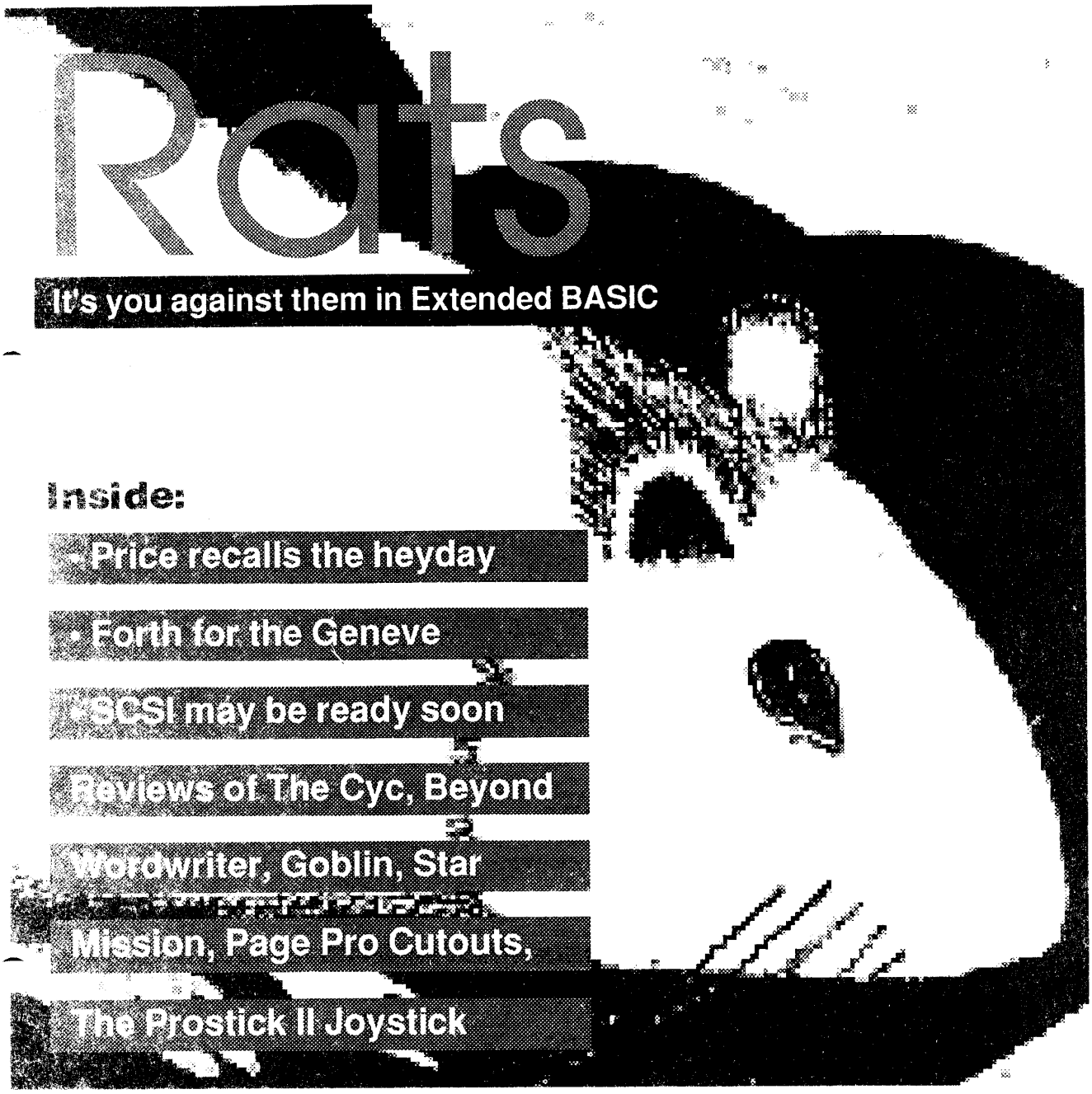
Covering the TI99/4A and the Myarc 9640

MICROpendium

Volume 10 Number 11 December 1993 $3.50



Rats

It's you against them in Extended BASIC

Inside:

- Price recalls the heyday
- Forth for the Geneve
- SCSI may be ready soon
- Reviews of The Cyc, Beyond Wordwriter, Goblin, Star Mission, Page Pro Cutouts, The Prostick II Joystick

Star Micronics Printer Bonus!

Buy A Star Printer And Get . . .

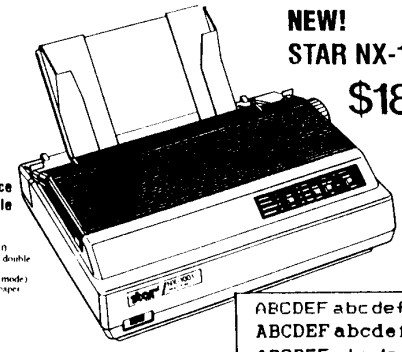
- The Most Advanced Printhead
- Letter Quality Type
- High Speed Operation
- Sophisticated Paper Handling
- Full TI® Compatibility

MULTI-FONT DOT-MATRIX PRINTER

A feature-rich 9-pin printer offering versatile typeface selection and superior ease of use at an unbelievable price!

- 9-pin parallel printer
- Printing styles: Draft, Courier, Sanserif, Orator and Orator-2 in a choice of 10 character sizes with enhancements such as: italics, emphasized, double strike, double high, quadruple high and super/subscripts modes
- 180 characters/second (draft mode); 75 characters/second (near letter quality mode)
- Both single and continuous paper handling capability with auto loading and paper parking functions
- IBM and Epson software modes
- Simplified, user-friendly control panel
- 2 year Limited Warranty

(Requires RS-232 interface and parallel cable or TurboPrint interface.)



NEW!
STAR NX-1001
$189.95 +S&H

ABCDEF abcdef 0123
ABCDEF abcdef 0123
ABCDEF abcdef 0123
ABCDEF ABCDEF 0123
ABCDEF abcdef 0123

Plus $100 In FREE

Accessories, Software & Discounts.

TEX-COMP has a great deal in store for you when you buy a STAR printer from us. With your purchase you will receive a free package of printer accessories and discounts on printer related software for the TI-99/4A with a total value of over $100. For starters, you will receive free a $19.95 copy holder, a $9.95 copy of 99 Writer II, the disk version of TI Writer, a $19.95 copy of Better Banners and the $9.95 Better Banners Companion, a $4.95 copy of the TEX-Comp STAR Demo disk, a $7.95 spare ribbon and a $9.95 Head brand printer care kit. In addition, you also receive a $5 discount on a printer stand (reg $14.95), printer cable (reg.$24.95), and a RS232 card or CorComp PIO printer interface. All discounts must be taken at the time of the printer is ordered.

copy holder

Here's the printer with the features you've been waiting for.

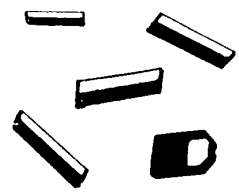
PRINTER CABLES

PARALLEL PRINTER CABLE..$24.95
(FROM PIO PORT ON RS232

Printer Ribbons

Be sure to specify your printer make & model with your order

STAR NX1000	EACH..	$7.95
STAR NX10	EACH..	$7.95
STAR GEMINI, 10, 10X & SG10	EACH..	$3.95
TI IMPACT PRINTER (EPSON MX-80)	EACH..	$7.95
AXIOM GP550 NEW LOWER PRICES	2-PAK	$19.95



HEAD BRAND

PRINTER CLEANING KIT

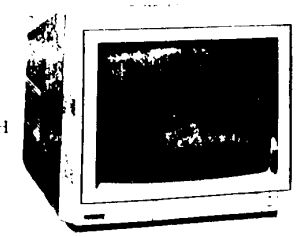
A dot matrix and daisy wheel printer head cleaner from HEAD Computer Products quickly cleans your dot matrix printer heads and daisy wheels to restore clean, crisp printing. Just insert the special cleaning sheet provided in the kit and apply a few drops of the cleaning solution. You actually see the accumulated ink and dirt disappear before your eyes. One kit lasts for dozens of cleanings and will actually extend the life of your printer heads and daisy wheels. Works with all computer printers using dot matrix or daisywheel printer heads. Not intended for thermal or ink jet printers.

NEW LOW PRICE $4.95

MAGNAVOX

RGB/Composite Color Monitor 80

THE NEW MAGNAVOX 14" COMPOSITE/RGB/ANALOG COLOR MONITOR IS THE ONLY MONITOR THAT WORKS WITH EVERY TI SYSTEM INCLUDING THE GENEVE. FEATURES INCLUDE DUAL SPEAKERS AND STEREO AMP SO YOU CAN EVEN HOOK IT UP TO A STEREO VCR OR A LASER DISK PLAYER FOR FANTASTIC PICTURE & SOUND. WE INCLUDE A TI AND A IBM CABLE. WITH A TUNER OR VCR YOU ALSO HAVE A SECOND TV. COMES WITH A 2 YEAR WARRANTY FROM PHILIPS.



INCLUDES TI-99/4A MONITOR CABLE!! **$279.95**+$10S&H

Two Year Parts and Labor Warranty • solid evidence that this is a reliable product, nationally supported by Magnavox

THE ORIGINAL TI EXTENDED BASIC IS AN ABSOLUTE MUST FOR ANY 99\4A SYSTEM. COMPLETE WITH THE ORIGINAL MANUAL. WE WERE ONLY ABLE TO GET A LIMITED QUANTITY OF THESE NEWLY PRODUCED MODULES. WITH EACH MODULE WE INCLUDE YOUR CHOICE OF 5 FREE DISKS FROM OUR FREEWARE COLLECTION, A $25 VALUE.

INCLUDES 5 FREE FREEWARE DISKS .....ONLY $64.95 +s&h

$25 Free Disk Bonus

THE SPEECH SYNTHESIZER IS BY FAR, THE TI ACCESSORY THAT IS THE MOST DIFFICULT TO LOCATE. WE WERE ABLE TO GET HOLD OF A A VERY LIMITED QUANTITY. WITH EACH SYNTHESIZER WE INCLUDE 5 FREE DISK FROM OUR FREEWARE COLLECTION. CHOOSE FROM ONE OF THE MANY DISKS WITH SPEECH CAPABILITY.

COMES WITH 5 FREE DISKS .....ONLY $59.95 +s&h

$25 Free Disk Bonus

COMES WITH 5 FREE DISKS A $25 VALUE!

TEX-COMP

P.O. Box 33084, Granada Hills, CA 91344

VISA and MASTERCARD HOLDERS CALL DIRECT

ORDER BY PHONE
24 HOURS A DAY
7 Days a Week!

TERMS: All prices F.O.B. Los Angeles. For fastest service use cashiers check or money order. Personal checks take ten days to clear. Add 3% shipping & handling (with a $3 minimum). Add 4.5% east of Mississippi. All prices shown include a 3% cash discount. Add 3% if paying by Visa or Mastercharge. Prices and availability subject to change. Include street address for UPS delivery. California orders add 6.25% sales tax. Canada, Hawaii, Alaska and overseas shipping extra.

AUTHORIZED DEALER

(818) 366-6631

CHARGE IT
WE ACCEPT VISA
or MASTERCARD

TEX-COMP

$$ DOLLAR DAYS $

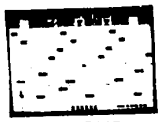
WATCH YOUR SAVINGS GROW DURING DOLLAR DAYS

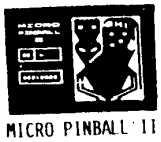
OUR LOWEST PRICES EVER ON THIS ENTIRE PAGE!

There may never be a better time than now to buy

ARCADE SPECIAL


BURGER BUILDER


TI TOAD


MICRO PINBALL II


MIDNIGHT MASON

FOUR GREAT ARCADE GAMES ON DISK AT ONE GREAT PRICE!!!

NOW YOU CAN GET FOUR OF THE HOTTEST ARCADE GAMES FOR THE 99/4A AT THE LOWEST PRICE EVER. THEY ALL AUTO LOAD WITH EX-BASIC AND HAVE GREAT GRAPHICS AND ARCADE ACTION.

MIDNIGHT MASON IS THE 99/4A ANSWER TO GHOST BUSTERS. YOU HAVE TO RETURN AT MIDNIGHT (OF COURSE) TO A HAUNTED HOUSE WHERE YOU LEFT YOUR TOOLS DURING THE DAY SHIFT. THE OBJECT IS TO GATHER UP YOUR TOOLS BEFORE THE GHOSTS GATHER YOU UP! GREAT ARCADE ACTION WHERE YOU CLIMB, RUN JUMP AND EVEN BREAK THROUGH WALLS WHILE THE GHOSTS KEEP COMING!

MICRO PINBALL II TURNS YOUR 99/4A INTO A REALISTIC PINBALL MACHINE WITH KEYBOARD OPERATED FLIPPERS AND FANTASTIC ACTION. THIS IS BY FAR ONE OF THE BEST ACTION GAMES AVAILABLE FOR THE 99/4A OR ANY OTHER COMPUTER.

TI TOAD IS EXACTLY WHAT THE NAME SAYS. IF YOU LIKE "FROGGER" YOU WILL LOVE TI TOAD AT A FRACTION OF THE PRICE BUT WITH ALL THE THRILLS IN GETTING "KERMIT" ACROSS VARIOUS ROADWAYS WITHOUT BECOMING PART OF THE PAVEMENT.

BURGER BUILDER IS KNOWN BY ALL THE COIN OP PLAYERS AS THE GAME WHERE YOU BUILD A BETTER BURGER WHILE BEING CHASED BY "FRANK" INSTEIN. ABSOLUTELY REALISTIC ARCADE ACTION AND THRILLS!

ALL FOUR ONLY **$4.95** + S & H
ON DISK

****

THREE GREAT GAMES

THORN EMI

Computer War Based on the hit movie "WarGames"®

SUBMARINE COMMANDER

RIVER RESCUE


COMPUTER WAR: Based on the hit movie "War Games." The alert bells are ringing at NORAD. Can you clear the code and destroy the simulated enemy missiles before the world is engulfed in a thermonuclear holocaust? Fast action. Uses joysticks or keyboard.


RIVER RESCUE: A group of helpless and starving refugees are trapped between a fierce battle to the north and the raging river. With your powerful boat, you must try to rescue as many as possible and deliver them to a friendly neighboring country. It won't be easy though - you must contend with hostile troops, electric eels and other hazards. High speed action for one or two players from THORN.

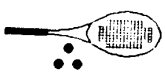
HERE ARE THREE ARCADE STYLE GAMES OF THE TYPE YOU NOW SEE FOR NINTENDO® OR SEGA® FOR OVER THIRTY DOLLARS EACH. THESE THREE GAMES WERE INTENDED FOR RELEASE ON MODULE BY ENTERTAINMENT GIANT THRONE EMI BUT NEVER MADE IT TO THE TI MARKET. NOW TEX-COMP BRINGS YOU ALL THREE GREAT GAMES AT ONE SUPER LOW PRICE

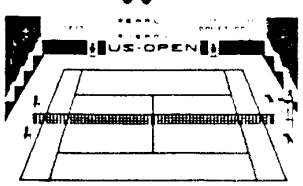
SUB COMMANDER: Simulates a sub patrol mission to hunt down and destroy enemy shipping. Use your sonar when submerged, rise to periscope level and attack. Hazards include enemy destroyers, depth charges, water pressure and an unknown sea floor. Full instrumentation display. From THORN EMI.

ALL THREE GAMES ONE ONE DISK AT ONE LOW PRICE! **$4.95** + S & H

****

PRO TENNIS+





PRO TENNIS+ LETS YOU ENJOY THE EXCITMENT, ACTION AND THRILLS OF PROFESSIONAL TENNIS WITH COMPLETE CONTROL.

THE GRAPHIC THREE DIMENSIONAL DISPLAYS ADD TO INCREASED REALISM AND OPTIONAL SPEECH AND REALISTIC SOUNDS OF THE BALL BEING STRUCK MAKES YOU FEEL THAT YOU ARE RIGHT ON THE COURT.

WE HAVE NOT SEEN A BETTER TENNIS GAME ON ANY BRAND COMPUTER OR GAME SYSTEM AND WE ARE SURE YOUR FRIENDS WITH OTHER BRAND COMPUTERS AND GAME SYSTEMS WILL BE DULY IMPRESSED.

NOW YOU CAN ENJOY THE ULTIMATE ACTION ARCADE GAME ON YOUR 99/4A! EXTENDED BASIC, 32K MEMORY EXPANSION AND DISK DRIVE REQUIRED.

COMPLETE ON DISK! ONLY **$4.95** + S & H

FREE BONUS: ON EACH PRO TENNIS+ DISK WE ALSO INCLUDE A COLLECTION OF NEVER RELEASED TI ARCADE STYLE GAMES. THESE GAMES ARE TRUE COLLECTOR'S ITEMS AND ARE WELL WORTH THE PRICE OF THIS DISK BY THEMSELVES!

****

TEX-COMP
P.O. Box 33084, Granada Hills, CA 91344

TERMS: All prices F.O.B. Los Angeles. For fastest service use cashiers check or money order. Personal checks take ten days to clear. Add 3% shipping & handling (with a $3 minimum). Add 4.5% east of Mississippi. All prices shown include a 3% cash discount. Add 3% if paying by Visa or Mastercharge. Prices and availability subject to change. Include street address for UPS delivery. California orders add 8.25% sales tax. Canada, Hawaii, Alaska and overseas shipping extra.

VISA and MASTERCARD HOLDERS CALL DIRECT:
(818) 366-6631

ORDER BY PHONE 24 HOURS A DAY 7 Days a Week! (818) 366-6631

AUTHORIZED DEALER

Send $2 for our new 1993 catalog

CONTENTS

MICROpendium

MICROpendium (ISSN 10432299) is published monthly for $25 per year by Burns-Koloen Communications Inc., 502 Windsor Rd., Round Rock, TX 78664-7639. Second-class postage paid at Round Rock, Texas. POSTMASTER: Send address changes to MICROpendium, P.O. Box 1343, Round Rock, TX 78680-1343.

All correspondence should be mailed to MICROpendium at P.O. Box 1343, Round Rock, TX 78680. We cannot take responsibility for unsolicited manuscripts but will give consideration to anything sent to the above address. Manuscripts will be returned only if a self-addressed stamped envelope is included.

Foreign subscriptions are $30.25 (Mexico); $32.50 (Canada); $30.00, surface mail to other countries; $42 airmail to other countries.

All editions of MICROpendium are mailed from the Round Rock (Texas) Post Office.

Mailing address: P.O. Box 1343, Round Rock, TX 78680.
Telephone: (512) 255-1512
CompuServe: 75156,3270
Delphi TI NET: MICROpendium
GEnie: J.Koloen
Internet E-mail: jkoloen@holonet.net
John Koloen.....Publisher
Laura Burns.....Editor

*READ THIS

Here are some tips to help you when entering programs from MICROpendium:

1. Most BASIC and Extended BASIC programs are run through Checksum, which places the numbers that follow exclamation points at the end of each program line. Do not enter these numbers or exclamation points. Checksum is available on disk from MICROpendium for $4.

2. Long Extended BASIC lines are entered by inputting until the screen stops accepting characters, pressing Enter, pressing FCTN REDO, cursoring to the end of the line and continuing input.

COMMENTS

Code change benefits Geneve users

The holiday season comes with good news for TI, and especially, Geneve users. Jim Schroeder has written a modification to MDOS that allows Geneve users with Hard and Floppy Disk Controllers to use high density, 3.5-inch disk drives. These disks handle up to 1.4 megabytes of data, four times the data that can be placed on a 5.25-inch double-sided, double-density disk.

But the good news doesn't end there. A new program called Form allows Horizon RAMdisk users to increase the size of their RAMdisks from the current 800K on a Geneve to over three megabytes! That's a lot of RAMdisk. This program, which is fairware, is expected to be available on bulletin boards. Use of Form does not require an HFDC.

So, what's the good news for TI users? Simply that Tim Tesch has supplied Schroeder with the source code of the HFDC EPROM. The hope is that there is enough space on the EPROM for Schroeder to modify it in such a way that TI99/4A users with HFDCs will also be able to use high density disk drives as well.

Keep your fingers crossed.

A COMBINED MIDWEST TI FAIR

I like Don Walden's idea of combining the Chicago and Milwaukee TI fairs and creating a Midwest fair. The fair would be held in Gurnee at the same hotel that the 1993 Chicago fair was held. Some vendors favor it because they would be spared the extra work of breaking down their displays and reassembling them at a second site. My understanding is that members of the Milwaukee user group also favor it. I wonder what the fair visitors think about it?

SCSI BREAKTHROUGH?

Bud Mills reports that maybe, just maybe, the long-awaited SCSI card will be available in 1994. Bud is hoping to have it ready for sale at the 1994 Fest West in Tucson in February. This really would be a remarkable achievement.

—JK

BUGS AND BYTES

Programmer's editor

This comes from Tony McGovern of Funnelweb fame. He writes:

Hello from Australia. Just thought I would give your internet address a tryout. In case you are interested, recent TI developments here at Funnelweb Farm have been on Delphi recently, but as a quick update I have done the first cut at a version of the F'Web programmer's editor that uses the 64K VDP RAM extension found in most 9938/9958 systems (standard unmodified Geneve excepted — but that can use My-Word) as text buffer. Typical file sizes usable are now up to about 380 sectors and/or 3,600 lines of my typical middling heavily commented assembly source code.

My TI is not connected to the world, and all things have to go by diskette from the TI at home to my office PC. This means that only D/V80 text files can be transferred, so I am trying a new experiment for transferring program files in a hex-ascii D/V80 format by e-mail by providing, also by e-mail, the source code for a hex-ascii to program file converter which can be assembled, etc. and then used to recover the intended program files. The recipient has to be able to accept e-mail files of up to 25Kb. If you are interested to experiment I can e-mail you the current set of files which I am leaving in my campus VAX account for convenience.

OPA uninvited

For probably the first time, a TI vendor has been specifically "uninvited" to a TI fair. Charles Good of the Lima, Ohio, Users Group has stated in the group's newsletter that Oasis Pensive Abacutors of Toronto, Canada, will not be allowed to have a booth at the next Lima Multi Users Group Conference unless he receives evidence that several overseas complaints have been resolved. The Lima newsletter printed complaints regarding non-receipt of items ordered from Australia, Belgium and England.

In a recent message on Delphi's TI-Forum, Gary Bowser of OPA noted that there are about 40 outstanding TIM/SOB orders and that the company hopes to resume production on them soon. Cost prohibits producing them in small lots of 10 or fewer, he said.

A Midwest-Fest

Don Walden of the Chicago Users Group says that group and the Milwaukee Users Group may hold a joint fair for two days instead of each holding one-day fairs on separate days at separate locations. Several vendors have suggested this, he notes, saying that perhaps it would become a "Midwest-Fest."

He notes that fairs provide attendees a chance to see new products and that they also come in hopes of getting special sale items.

FEEDBACK

Forward and backward

"A major new tool,"

"A new day for the TI99/4A."

These words do not come from the AEMS Project Team. Not that we haven't been saying them out loud! No, these are reprints from the MICROpendium September 1993 and October 1993 reviews of the AEMS Macro Assembler and Linker and the AMS.

It is one thing to say one's effort has produced a good product. It is far better to let others judge your work and efforts. My friends, after three long years, the above two comments made the time, blood, sweat and tears worth it!

Never in our wildest dreams did the AEMS Project Team expect such a review. With all the battles that have raged, and it was war, the unbiased reviews from Bruce Harrison were a sweet success! For those who know Bruce, he is against "new gadgets" for the 99/4A. But he put aside those biases and examined a product *for what it was and what it did!* And, most certainly for what the product offers for the 99/4A.

When I started the AEMS Project three years ago, I never imagined that a product the team and I made would be put into any TI magazine. But with perseverance, luck and by the grace of God, our efforts have succeeded! Not easily — at times the problems seemed insurmountable. But we kept on! The team grew from one, to two, to three and then to five. As the years went by, the project took on a life of its own. And with that, we jelled together and created a new memory system for the TI99/4A home computer.

The project had a goal, that this system, *unlike any before*, would institute "EASE OF USE" expanded memory. It would isolate the programmer from the paging process, which in so many other cards was detrimental to the programming process.

That we have done so is a great testimony to the team effort and the team itself. We learned from our mistakes and we compromised on certain features. Had we tried to include all of them when we first came up with the AEMS Project, the AMS would have never been done, even now. We decided on a purely memory card. And, by using off-the-shelf components, we made the system easy to make and relatively inexpensive to produce.

The actual paging process is time honored and true. In this ever changing computer world, many things tend to last only months; our system's paging process has remained a standard. Do not let the fact that "the other world" uses a similar system discourage AMS use. The 4K page is the standard for TI's minis, Unix, Windows, EMS and others.

But more importantly, with the Linker provided, someone who prefers to make 8K modules can assemble the code into blocks that size, and then link them together!The 4K page is the minimum size module for the Linker; up to 24K can be a module's size. And my challenge lies untested—does anyone out in the community, so insistent on the 8K page, wish to make a 24K subroutine, in assembly?

So this answers those who want"only" 8K programming. For those against the AMS because they have an axe to grind against Asgard, that is your loss. If Bruce Harrison could put aside his bias and give a true review, then some others should follow his example. Ask those who use and program the AMS. Get their opinion and words of experience, rather than letting someone else tell you the AMS is not worth it.

So much has gone into the AEMS Project. The large phone bills, the endless updates, the never ending conversations with team members. With the 99/4A community, a return for the effort may be doubtful, but in my view, some of the most successful projects may generate little revenue. Indeed, if one learns from the project process and applies the lessons in a professional way, then a "hobby" project has blossomed into real world engineering, albeit on a small scale. Just because a community is a "hobby" market doesn't mean that a project should be done in a "hobby" style.

To look forward, the AEMS Project continues. Refining our programming techniques. Creating new and unique, but useful, programs, and designing the next generation hardware. One result is the SUPERams, known as AMS-2 in the MICROpendium October 1993 review.

I am truly thankful. It has been hard but the rewards have been fantastic! I thank Chris Bobbitt for giving a total unknown a chance to run a project and to create the "most valuable product ever to hit the TI market," considering that only several months earlier, his biggest software project, Press, had been "put on hold."

To Tony Lewis of the Hardware Team, we have spent many a conversation on the phone and on the Internet, discussing the hardware designs. You are a great source of advice. When rumors were flying, you held your ground and taught us to stick by our guns and not believe the rumors.

To Joe Delekto and Art Green of the Software Team, guys, the software was the hardest battle and you have won! Both of you to be congratulated for a job well done! In Coast Guard lingo, "Bravo Zulu," excellent job.

An d to someone who gave me the best advice when this project was just an ide a let alone a team effort, Mr. Jim Peterso Your advice became the linchpin to the AEMS Project. And thank you for introducing me to Chris Bobbitt! Take care, sir, and God be with you in all your travels.

The journey has been rough, and the seas often hostile, But we, the AEMS Project Team, have accomplished what so few others ever have in the 99/4A community. To those who have the AMS, thank you. To those interested, ask us, write or call. The AMS is here, the software real.

Thank you all. Take care and God Bless!

Jim Walter Krych
Asgard Peripherals
Corpus Christi, Texas

Backing up Bruce

I would like to reinforce a comment made by Bruce Harrison in his October 1993 review of the AMS card. I know this card only from study of its disk documentation but it looks to be a fine addition to the TI-99/4a scene. It is a great pity t earlier makers of memory expansions had not implemented this excellent 99/8 inspired idea years ago.

(See Page 7)

FEEDBACK

(Continued from Page 6)

The point I want to back up though is Bruce's concern at the use of a non-standard DSRlink routine at the system level. A device and system software operating at such a fundamental level should emulate the existing system as closely as possible. The point in question is breaking of the normal use of PAD locations >83D0-2 for CRU/ROM search pointers. There are all sorts of levels of argument about system interfaces in the PC world (look at recent discussions of Microsoft's practices with MS-DOS and Windows interfaces). DSRlinks have always been a minefield of incomplete or misleading documentation, but in this case it IS documented in the E/A manual on page 406, and has proved to be of elegant and useful application as described by Bruce.

It does not really affect Funnelweb that much, because it has been made robust in use. Boot-tracking is mainly of value for loading from floppy disks which may be in any drive, but FW may also be configured to assume fixed drive numbers (recommended with HRDs), or else a fixed path (for historical reasons the length of pathname is constrained). If however the menu programs cannot be found then <esc> from the central menu allows the boot drive number(s) to be reset.

I wish the AMS cavalry well, and hope they have not galloped on to the scene too late.

Tony McGovern
Funnelweb Farm

***Feedback is a reader forum. The editor may condense excessively lengthy submissions if necessary. We ask that writers limit themselves to one subject per submission. Our only requirement is that submissions be of interest to those using the TI99/4A, the Geneve 9640 or compatibles. Send items to* MICROpendium Feedback, P.O. Box 1343, Round Rock, TX 78680.**

READER TO READER

❑ Kate Burkard, the treasurer/librarian of the Manasota TI99/4A User Group, writes:

I am in need of some assistance. Some time recently I read an article about a Graphic Labeler for TI-Base and I can't remember where I read about it. I wrote down the address, but now I can't find the address. I hope someone in your readership will know the address.

Write her at 2814 Hermitage Blvd., Venice, FL 34292.

❑ Harold W. Evans, of Grand Forks, ND, writes:

I have a program that accesses files stored on another disk and prints the data to screen and printer. If I place the program disk in drive 1 and the disk of files in drive 2, the program crashes. Not only that, but the name of the program disk is changed to the name of the file disk, and the names of the files are added to the program disk. There are no data in the added files, and access is denied to several of the programs.

If I place a write-protect tab on the program disk, the program still crashes. However, if I load the program and remove the program disk from drive 1, it performs as expected.

The files are accessed by disk name, not drive number, so that the disk may be placed in any drive. Why and how is the name of the program disk changed?

Write him at 293 Circle Hills Dr., Grand Forks, ND 58201.

Reader to Reader is a column to put TI and Geneve users in contact with other users. Address questions to *Reader to Reader, c/o MICROpendium, P.O. Box 1343, Round Rock, TX 78680.*.

Support Our Advertisers

Rat Infest

Exterminate or be exterminated

By JEFF AUTOR

Imagine you are in charge of a deep African Safari's food supply. Little do you know that giant rats nightly invade the area. If they should eat your food supply, the members of the safari and yourself would die. Knowing this, you attack the rats every night using only a swift Karate kick. But for some reason, one night a mutant rat joins the pack. This special rat glows a bright red and can kill you in three bites; and has taught the others how to cast a spell of teleportation. The only way to escape this quick mutant is to hide in a hut near the food. Now night is falling, and you know the rats will come again.

THE GAME

After starting the game, the computer will ask you if you want to use joysticks. A simple Y or N will do fine. Then the colorful title screen will appear. On this screen, the highest score (which is initially set at 1000), and the score of the last game played is displayed. To start the game, push the button on joystick #1 or any key on the console. The screen will turn black, and the rats will march in, with the mutant commanding from the rear.

You begin each night inside your hut (probably reading National Geographic) in the upper left hand corner of the screen. Once the rats begin to move, the food supply (represented by a chunk of cheese and a glass of wine) will appear on the screen. At this time, you may begin fighting off the rat forces.

To move your man around the playing field, use joystick #1 or the arrow keys, depending on your choice at the beginning. If you move off one edge of the screen, you will reappear at the opposite side.

To attack a rat, you must sneak up one. If it sees you, the rat will use its teleportation spell; and you will be teleported somewhere on the screen. However, if you successfully sneak up on the rat, you can kill the rat by pressing the "fire" button or the "Q" key, depending on your choice at the beginning. When attacking a rat, you do not have to be facing the rat, but you must be very close. If you are close enough, you will kick the rat. It will then scream and disappear from sight. Think carefully about your attacks, since the mutant will hear the scream and will investigate.

As you move about the screen, the rats will slowly begin to devour your food supply. At the beginning of each night, this food supply is replenished with 4 drinks of wine and 4 bites of cheese. If a rat gets a good chance, it will eat the cheese or drink the wine. When the supply is gone, the game is over.

The mutant rat will appear every night along with the normal rats. Unlike the others, this mutant is red, and is no threat to your food supply, since it does not eat. However, it poses a different threat to you. With its radioactive bites, it can kill you with only three hits. The mutant will chase after you if you come too close. The mutant rat is faster than you are and will eventually catch up. The best way to lose it is to run to your hut. If you are touching part of the hut, the mutant can't pounce on you.

If the mutant rat touches you, play will stop. On the screen your current score and number of bites you can still take without dying will be displayed. To continue playing, press the "fire" button or any key on the console.

When you are bitten for the final time, the computer will check your score against the Highest Score, (By the way, the Highest Score to date is 79,800). You will then hop into your grave and listen to "Taps" playing in the background. To return to the title screen and see your final score, press any key on the console.

SPECIAL FEATURES

When you start the game, each rat is worth 50 points. Each piece of the food supply (8 total) is worth 100 points. The points for kicking a rat are given immediately, while the points for the remaining food supply are given when the night is over. Each night the number of points for kicking a rat goes up 50 points, and the amount of points for each food piece increases 100 points.

After every three nights of attack, you gain enough experience and resistance to take another bite from the mutant without dying.

If you manage to kill all the rats that attack, play will stop, and your score and the number of bites you can take without dying is displayed on the screen. At this time the bonus for the remaining pieces of food are given. When you are ready to continue, press any key on the keyboard.

RAT INFEST

```
100 ! ************** !239
110 ! * RAT INFEST * !007
120 ! ************** !239
130 ! COPYRIGHT 1986 !084
140 !!131
150 ! BY JEFF AUTOR !036
160 !!131
170 ! VERSION 0.0.0 !213
180 ! TI EXTENDED BASIC !051
190 !!131
200 CALL CLEAR :: CALL SCREE
N(2):: CALL MAGNIFY(3):: CAL
L COLOR(3,15,1,4,15,1,5,15,1
,6,15,1,7,15,1):: DIM M(15,3
)!087
210 HS=1000 :: L(1)=5 :: L(2
)=7 :: L(3)=11 :: L(4)=14 !0
17
220 CALL CHAR(33,"0000000003
070F1F030F3FFFFFFFFFFC0F0FC
FFFFFFFFFF00000000C0E0F0F8")
!173
230 CALL CHAR(37,"1F3F7F7"&R
PT$("F",26)&"8FCFEFEFFFFFFFF
")!219
240 CALL CHAR(40,"7F5E5A5B6B
6F7D5DFFFCB0A0E0A0A0A0FF3F0D
0507070407FEEE6E6A7ADED656")
!204
250 CALL CHAR(44,"5D77777F6F
6D6F7FE060602060E0E0E0070705
05050707077EFA9ADE767EDEFE"
!062
260 CALL CHAR(60,"0000000002
010001437F3C0000000001000018
```

(See Page 9)

RAT INFEST—

(Continued from Page 8)

```
3C3C78E4C0C0A080C0C0C0C08")!
220
270 CALL CHAR(88,"E0D0D8389C
DEC6F0E0C09030B0B898D8D8C8E8
E0C0000000000000001C4F616C")
!003
280 CALL CHAR(92,"0000000000
80E06000C0E0E0C",81,"0044442
81010101")!060
290 CALL CHAR(96,"0F1F306040
00000001030E1F337BDF3FF0F81C
0E070303E3A3637FFFFEFEFCFC")
!067
300 CALL CHAR(100,"0F1F3870E
0C0C0C7C5C6FEFF7F7F3F3FF0F81
C060200000080C070F8CCDEFBFC"
)!194
310 CALL CHAR(104,"0000051B3
76F6F686A696A6F6F6F6F6F0000F
0F8FCFEFEA2AAA2AEFEFEFEFEFE"
)!055
320 CALL CHAR(108,"030F3F7C0
16F6F6D6F6F6E6F6F2C0000FEF08
3EFEFADEFEEEFEFEF8C")!184
330 CALL CHAR(112,"C0C0C0C0C
0E0E0F07F3F0F0703031F7F03030
3030307070FFEFCF0E0C0C0F8FE"
)!090
340 CALL CHAR(116,"000303030
10103030F0303060606060C0080C
08080808080C0A080C0C0C0C08")
!251
350 CALL CHAR(120,"000103010
10101010305010303030100C0C0
0C08080C0C0F0C0C060606060B")
!207
360 CALL CHAR(124,"3F3F3F3F3
F1F1F0F00",126,"FCFCFCFCFCF8
F8F00",128,"00003F3F3F1F1F0F
0",130,"0000FCFCFCF8F8F00")!
241
370 CALL CHAR(132,"000000003
F1F1F0F0",134,"00000000FCF8F
8F00",136,"00000000001F1F0F0
",138,"0000000000F8F8F00")!2
47
380 CALL CHAR(140,"0000183C3
C1E2703030C01030303030100000
00040800080C2FE3C000000008")
: DISPLAY AT(11,3):"DO QOU
HAVE JOQSTICKS" !126
390 CALL KEY(2,K,S):: IF K=1
8 THEN J=1 ELSE IF K<>15 THE
N 390 !176
400 U=88 :: V=112 :: NS=16 :
: MS=6 :: NT=1 :: YS=1.25 ::
 WI=124 :: RP=50 :: MN,BT=3
:: CB=4 :: RA=100 :: CALL CL
EAR !068
410 FOR X=1 TO 32 STEP 2 ::
CALL HCHAR(1,X,U,2):: CALL H
CHAR(24,X,V,2)!084
420 IF X<23 THEN CALL VCHAR(
X+1,1,V,2):: CALL VCHAR(X+1,
32,U,2)!180
430 U=U+8 :: V=V-8 :: IF U=1
20 THEN U=88 :: V=112 !066
440 NEXT X :: DISPLAY AT(6,9
):"WELCOME TO" :: DISPLAY AT
(9,9):"RAT INFEST" !006
450 RESTORE 470 :: FOR X=1 T
O 7 :: READ B,D,E :: CALL SP
RITE(#X,B,2,D,E):: NEXT X !2
13
460 CALL COLOR(#1,7,#2,6,#3,
15,#4,15,#5,15,#6,15,#7,12)!
047
470 DATA 100,130,50,116,140,
80,96,117,195,96,125,160,96,
83,205,96,144,190,108,130,10
0 !100
480 DISPLAY AT(12,4):"HIGH S
CORE:";HS :: DISPLAY AT(14,4
):"LAST SCORE:";LS :: D=1 ::
 RESTORE 540 !103
490 FOR E=1 TO 23 :: READ A,
B,C :: CALL SOUND(-A*500,B,7
,B+1,7,C,7)!106
500 FOR X=8 TO 12 :: CALL CO
LOR(X,L(D),L(D)):: D=D+1 ::
IF D>4 THEN D=1 !071
510 NEXT X !238
520 CALL KEY(1,K,S):: CALL K
EY(0,K,R):: IF S OR R THEN 5
70 !018
530 NEXT E :: RESTORE 540 ::
 GOTO 490 !148
540 DATA 2,233,117,2,220,250
00,2,175,117,1,175,25000,1,1
75,25000,1,196,156,1,220,156
,1,233,25000,1,196,25000 !13
8
550 DATA 1,175,117,1,147,117
,2,147,25000,1,131,131,1,131
,25000,1,196,131,1,196,25000
,1,175,175,1,175,25000 !040
560 DATA 1,262,175,1,294,250
00,2,231,117,2,231,156,3,231
,117,1,231,117 !104
570 CALL SOUND(-1,999,30)::
CALL CLEAR :: CALL DELSPRITE
(ALL)!109
580 DISPLAY AT(12,11):"NIGHT
";NT:"" :: IF NT>1 THEN 610
!023
590 DISPLAY AT(24,6):"CHECK
ALPHA LOCK" :: CALL COLOR(1,
2,1,2,2,1)!208
600 DISPLAY AT(3,4):"!"&CHR$
(34)&"#$":"  %&&'":"  ()*+
":"  ,-./" !218
610 FOR X=1 TO 15 :: RANDOMI
ZE :: CALL PEEK(-31808,E,D)!
147
620 IF E>125 THEN M(X,3)=100
 :: M(X,2)=2 :: GOTO 640 !19
7
630 M(X,3)=96 :: M(X,2)=-2 !
117
640 IF D>125 THEN M(X,1)=2 E
LSE M(X,1)=-2 !206
650 NEXT X :: DISPLAY AT(24,
6):"" !205
660 CALL SOUND(2000,262,7,33
0,7,392,7):: CALL COLOR(1,11
,1):: DISPLAY AT(12,6):"  O
N QOUR MARK" !071
670 CALL SOUND(2000,262,7,31
1,7,440,7):: CALL COLOR(2,13
,1):: DISPLAY AT(12,9):"  GE
T READQ" !064
680 CALL SOUND(2000,262,7,34
9,7,440,7):: CALL SPRITE(#1,
116,8,33,50):: DISPLAY AT(12
,11):" GET SET" !039
690 CALL SOUND(2000,262,7,33
0,7,523,7):: DISPLAY AT(12,1
2):"  GO" :: FOR X=1 TO 300
:: NEXT X :: DISPLAY AT(12,1
1):"" !213
700 A=80 :: FOR X=3 TO 16 ::
 CALL SPRITE(#X,100,15,A,1,0
,3):: CALL SOUND(200,-1,25)!
057
710 A=A+20 :: IF (X=7)+(X=12
)THEN A=80 !008
720 IF X=14 THEN A=A+20 !166
730 FOR B=1 TO 50 :: NEXT B
:: NEXT X !048
740 CALL SPRITE(#2,100,7,120
,1,0,3):: CALL SOUND(400,-2,
23)!021
750 CALL COINC(#3,80,175,10,
```

(See Page 10)

RAT INFEST—

(Continued from Page 9)

```
E):: IF E=0 THEN 770 !146
760 CALL SOUND(300,-2,18):: FOR X=2 TO 16 :: CALL MOTION (#X,0,0):: NEXT X :: GOTO 790 !076
770 IF Z>7 THEN CALL SOUND(200,-3,15):: Z=0 !242
780 Z=Z+1 :: GOTO 750 !238
790 FOR X=1 TO 15 :: A=M(X,1):: B=M(X,2):: C=M(X,3):: E=X+2 :: IF X=15 THEN E=2 !214
800 CALL MOTION(#E,A,B):: CALL PATTERN(#E,C):: CALL SOUND(100,-1,19):: NEXT X !142
810 CALL COLOR(11,16,1,8,12,1,10,12,1):: DISPLAY AT(12,15):"pr  ln" :: DISPLAY AT(13,15):"qs  mo" !226
820 CALL SPRITE(#17,124,9,89,129):: V=116 :: H=M(15,1):: G=M(15,2):: CH=M(15,3)!075
830 CALL MOVE(O,YS,J):: IF O=18 THEN 1200 !043
840 CALL COINC(#1,#2,70,B):: IF B THEN 990 !096
850 IF Z=1 THEN CALL MOTION(#2,H,G):: CALL PATTERN(#2,CH):: Z=2 !037
860 FOR X=3 TO NS :: IF WI=140 THEN 890 !170
870 CALL COINC(#X,#17,12,D):: IF D THEN 1630 !120
880 IF CB=0 THEN 900 !192
890 CALL COINC(#X,89,161,12,E):: IF E THEN 1660 !186
900 IF (X=6)+(X=10)+(X=14)THEN CALL MOVE(O,YS,J):: IF O=18 THEN 1200 !195
910 NEXT X !238
920 IF RM<21-NS THEN RM=RM+1 :: GOTO 830 !235
930 CALL POSITION(#MN,U,V)!248
940 IF U<93 THEN K=2 ELSE K=-2 !201
950 IF V<144 THEN S=2 :: A=100 ELSE S=-2 :: A=96 !159
960 CALL PATTERN(#MN,A):: CALL MOTION(#MN,K,S)!079
970 RM=1 :: MN=MN+1 :: IF MN>=NS THEN MN=3 !077
980 GOTO 830 !144
990 CALL POSITION(#1,K,S)!074
1000 IF (K>12)*(K<48)*(S>34)*(S<66)THEN 830 !063
1010 CALL COINC(#1,#2,10,X):: IF X THEN 1060 !205
1020 CALL POSITION(#2,D,E)!054
1030 IF K>D THEN C=MS ELSE C=-MS !008
1040 IF S>E THEN D=MS :: B=100 ELSE D=-MS :: B=96 !172
1050 CALL PATTERN(#2,B):: CALL MOTION(#2,C,D):: Z=1 :: GOTO 830 !159
1060 CALL SCREEN(16):: BT=BT-1 :: IF BT=0 THEN 1120 !033
1070 CALL SCREEN(2):: CALL MOTION(#1,0,0,#2,0,0)!021
1080 FOR X=6 TO 21 STEP 3 :: CALL SOUND(-50,-6,X):: NEXT X !239
1090 DISPLAY AT(7,11):"SCORE :";SC :: DISPLAY AT(9,11):"BITES LEFT :";BT :: DISPLAY AT(17,9):"PRESS ANQ KEQ" !080
1100 CALL KEY(1,K,S):: CALL KEY(0,K,R):: IF (S=0)*(R=0)THEN 1100 !119
1110 DISPLAY AT(7,11):"":"":"" :: DISPLAY AT(17,7):"" :: CALL LOCATE(#2,185,10,#1,33,50):: CALL MOTION(#2,H,G):: GOTO 830 !214
1120 CALL SOUND(-1,25000,30):: FOR X=6 TO 21 STEP 3 :: CALL SOUND(-50,-6,X):: NEXT X :: CALL SCREEN(2):: CALL MOTION(#1,0,0)!120
1130 CALL COLOR(#1,15):: CALL PATTERN(#1,104):: IF SC>HS THEN HS=SC !073
1140 RESTORE 1170 :: DISPLAY AT(8,9):"SCORE :";SC :: DISPLAY AT(16,11):"GAME OVER" :: DISPLAY AT(22,9):"PRESS ANQ KEQ" !212
1150 FOR X=1 TO 24 :: READ A,B :: CALL SOUND(A*150,B,10,B+1,10,B+2,10):: CALL KEY(1,K,S):: CALL KEY(0,K,R):: IF S OR R THEN 1190 !005
1160 NEXT X !238
1170 DATA 3,262,1,262,12,349,3,262,1,349,12,440,3,262,1,349,4,440,3,262,1,349,4,440,3,262 !105
1180 DATA 1,349,12,440,3,349,1,440,8,523,4,440,4,349,12,262,3,262,1,262,18,349 !025
1190 CALL DELSPRITE(ALL):: CALL COLOR(8,2,1,9,2,1,10,2,1,11,2,1):: LS=SC :: SC=0 :: GOTO 400 !208
1200 CALL MOTION(#1,0,0):: O=0 :: FOR X=3 TO NS !247
1210 CALL COINC(#1,#X,16,R):: IF R THEN 1230 !206
1220 NEXT X :: O=0 :: GOTO 840 !146
1230 RANDOMIZE :: CALL PEEK(-31880,K):: IF K>85 THEN 1330 !155
1240 CALL POSITION(#X,U,V):: CALL POSITION(#1,T,E)!124
1250 IF E<V THEN CALL PATTERN(#1,140)ELSE CALL PATTERN(#1,60)!225
1260 CALL POSITION(#NS,A,B):: CALL LOCATE(#X,A,B):: CALL COLOR(#NS,1):: CALL DELSPRITE(#NS):: NS=NS-1 !046
1270 CALL SOUND(400,-2,7):: SC=SC+RP :: IF NS=2 THEN 1400 !007
1280 CALL PATTERN(#1,116):: IF Z=1 THEN 840 !224
1290 CALL POSITION(#2,A,B)!048
1300 IF T>A THEN H=2 ELSE H=-2 !206
1310 IF E>B THEN G=2 :: CH=100 ELSE G=-2 :: CH=96 !233
1320 CALL MOTION(#2,H,G):: CALL PATTERN(#2,CH):: GOTO 840 !102
1330 X=3 :: U=110 !004
1340 CALL COLOR(#1,X):: CALL SOUND(-100,U,15):: X=X+1 :: U=U+50 !200
1350 IF X>16 THEN X=3 !143
1360 IF U<800 THEN 1340 !181
1370 RANDOMIZE :: CALL PEEK(-31808,B,R):: IF B>180 THEN B=180 !031
1380 IF R>220 THEN R=220 !020
1390 CALL LOCATE(#1,B+1,R+1):: CALL COLOR(#1,8):: GOTO 830 !140
```

(See Page 11)

RAT INFEST—

(Continued from Page 10)

```
1400 CALL POSITION(#2,A,B)::
 CALL PATTERN(#2,96,#1,116):
: IF B<16 THEN CALL MOTION(#
2,0,0):: CALL SOUND(300,-3,1
0):: GOTO 1420 !036
1410 CALL MOTION(#2,0,(16-B)
*.49):: CALL SOUND(300,-3,10
)!095
1420 CALL SOUND(1,-3,10):: C
ALL DELSPRITE(#1,#2)!084
1430 DISPLAY AT(6,9):"SCORE
:";SC !086
1440 DISPLAY AT(8,9):"BITES
LEFT :";BT !163
1450 FOR X=1 TO 300 :: NEXT
X :: IF CB=0 THEN 1520 !203
1460 CB=CB-1 :: ON CB+1 GOSU
B 1480,1490,1500,1510 !021
1470 CALL SOUND(380,-7,13)::
 SC=SC+RA :: DISPLAY AT(6,16
):SC :: IF CB=0 THEN 1520 EL
SE 1460 !062
1480 CALL HCHAR(12,21,32,2):
 CALL HCHAR(13,21,32,2):: R
ETURN !065
1490 CALL HCHAR(12,21,91)::
CALL HCHAR(12,22,92):: CALL
HCHAR(13,22,93):: RETURN !16
7
1500 CALL HCHAR(12,22,89)::
CALL HCHAR(13,22,90):: RETUR
N !247
1510 CALL HCHAR(12,22,88)::
RETURN !066
1520 IF WI=140 THEN 1550 ELS
E WI=WI+4 !012
1530 SC=SC+RA :: DISPLAY AT(
6,16):SC :: FOR X=500 TO 150
 STEP -50 :: CALL SOUND(-100
,X,13):: NEXT X !130
1540 IF WI=140 THEN CALL DEL
SPRITE(#17)ELSE CALL PATTERN
(#17,WI):: GOTO 1520 !104
1550 IF NT/3<>INT(NT/3)THEN
1590 !184
1560 FOR X=1 TO 300 :: NEXT
X !027
1570 DISPLAY AT(20,9):"BONUS
 BITE" :: CALL SOUND(600,131
,7,132,7,133,7):: CALL SOUND
(150,262,7,263,7,264,7)!112
1580 CALL SOUND(2000,523,7,5
24,7,525,7):: BT=BT+1 :: DIS
PLAY AT(8,21):BT !038
1590 DISPLAY AT(17,9):"PRESS
 ANQ KEQ" :: CALL KEY(0,K,S)
:: CALL KEY(1,K,X)!125
1595 DISPLAY AT(17,9):"" ::
IF (S=0)*(X=0)THEN 1590 !211
1600 CALL DELSPRITE(ALL):: D
ISPLAY AT(6,9):"": :"":"" ::
 DISPLAY AT(16,9):"" :: NT=N
T+1 :: RA=RA+100 :: RP=RP+50
 !234
1610 CALL COLOR(1,2,1,2,2,1)
:: IF NT<5 THEN YS=YS+.25 ::
 MS=MS+1 !059
1620 CB=4 :: WI=124 :: NS=16
 :: GOTO 580 !171
1630 WI=WI+4 :: FOR X=500 TO
 150 STEP -50 :: CALL SOUND(
-100,X,13):: NEXT X !0811640
 IF WI<140 THEN CALL PATTERN
(#17,WI):: GOTO 830 !126
1650 CALL DELSPRITE(#17):: I
F CB=0 THEN 1120 ELSE 830 !0
96
1660 CALL SOUND(-380,-7,13):
: CB=CB-1 :: IF CB>0 THEN 16
80 !055
1670 CALL HCHAR(12,21,32,2):
: CALL HCHAR(13,21,32,2):: I
F WI=140 THEN 1120 ELSE 830
!099
1680 ON CB GOTO 1690,1700,17
10 !101
1690 CALL HCHAR(12,21,91)::
CALL HCHAR(12,22,92):: CALL
HCHAR(13,22,93):: GOTO 830 !
175
1700 CALL HCHAR(12,22,89)::
CALL HCHAR(13,22,90):: GOTO
830 !255
1710 CALL HCHAR(12,22,88)::
GOTO 830 !074
1720 SUB MOVE(O,YS,J)!188
1730 CALL JOYST(1,A,B):: CAL
L KEY(1,O,S):: IF O=18 THEN
CALL SOUND(-200,-5,9):: SUBE
XIT !091
1740 IF J THEN 1810 ELSE IF
S=0 THEN 1840 !031
1750 A,B=0 :: ON O+1 GOTO 17
70,1840,1780,1790,1840,1800
!058
1760 GOTO 1840 !134
1770 B=-4 :: GOTO 1820 !179
1780 A=-4 :: GOTO 1820 !178
1790 A=4 :: GOTO 1820 !240
1800 B=4 :: GOTO 1820 !241
1810 IF (A=0)*(B=0)THEN 1840
 !194
1820 IF A>0 THEN CALL PATTER
N(#1,120)ELSE IF A<0 THEN CA
LL PATTERN(#1,116)!223
1830 CALL MOTION(#1,-B*YS,A*
YS):: CALL SOUND(-3000,-4,5)
:: SUBEXIT !173
1840 CALL SOUND(-1,-4,30)::
CALL MOTION(#1,0,0):: SUBEND
 !023
```

1994 TI FAIRS

FEBRUARY

Fest-West, Feb. 19-20, Santa Rita Park Inn, Tucson, Arizona. Contact Tom Wills, Fest-West '94 Committee, Southwest 99ers Users Group, P.O. Box 17831, Tucson, AZ 85731 or (602) 886-2460; BJ Mathis, (602) 747-5046; or the Cactus Patch BBS, (602) 290-6277.

MAY

Lima Multi User Group Conference, May 14-15, Ohio State University Lima Campus, Lima, Ohio. Contact Lima Ohio Users Group, P.O. Box 647, Venedocia, OH 45894.

This TI event listing is a permanent feature of MICROpendium. User groups and others planning events for TI/Geneve users may send information for inclusion in this standing column. Send information to MICROpendium Fairs, P.O. Box 1343, Round Rock, TX 78680.

THE ART OF ASSEMBLY — PART 30

Using what's there

By BRUCE HARRISON

Today's topic is using what's there, which means using the routines that come with your TI instead of inventing your own. The routines provided include some that are "documented" and some that are not. Today we"ll concentrate most of our attention on routines that work with floating point numbers.

FLOATING POINT NUMBERS

We've said before in this column that we use integer math whenever possible to speed up the execution time in our programs. There are cases, however, where nothing but floating point numbers can do the job. In a recent case, we were doing some work on a "tool" for users of Mike Maksimik's MIDI-Master. The job involved taking numbers from a file in the form of ASCII strings, multiplying or dividing by a user-input number such as 2.5, then writing the numbers back as strings into an output file record. (Portions of the source code are shown in the sidebar.)

The numbers themselves are all integers to start with, and if the divisor or multiplier were always 2.5, we could have made the calculations in integer math. However, where the user could input numbers of any kind, such as 2.3742, the problem of handling things as integers would get complex indeed. Thus, we had an opportunity to take advantage of the routines provided by TI right inside the computer.

In the TI, floating point numbers are represented by eight byte blocks in Radix 100 notation. Radix what, you ask? There is an explanation in the Editor/Assembler manual, but it's about as clear as mud, so we'll try to explain in more simple terms.

The Radix 100 notation uses an offset, or multiplier, in powers of 100. This multiplier occupies one byte. That leaves seven bytes for the number itself. Each byte of these seven can take on values from 0 through 99, thus each byte represents two digits of our common decimal number notation. That allows 14 significant digits of accuracy in all the TI's floating point number operations.

The "multiplier" is biased by >40 (64 decimal), so that the multiplier can provide both positive and negative powers of 100. In other words, 100 raised to the zero power makes the multiplier byte equal >40. Perhaps an example or two would help. Here are a couple:

Decimal number	bytes in FP format
1.0000	>40,>01,>00,>00,>00,>00,>00,>00
0.5000	>3F,>32,>00,>00,>00,>00,>00,>00
105.2	>41,>01,>05,>14,>00,>00,>00,>00
89,999.999	>42,>08,>63,>63,>63,>5A,>00,>00

We might find this easier to see if the bytes themselves were expressed in decimal notation instead of hex. These four would look like this:

1.0000	64,1,0,0,0,0,0,0
0.5000	63,50,0,0,0,0,0,0

(See Page 13)

```
*  SIDEBAR 30
*
*  SOME EXAMPLES OF USING EXISTING
*  CONSOLE ROUTINES
*  PUBLIC DOMAIN
*
*  CODE BY B. HARRISON EXCEPT WHERE NOTED
*
*
*  FIRST, SOME EQUATES USED
*
*
CNS    EQU  >0014   CONVERT NUMBER TO STRING (W/GPLLNK)
CFI    EQU  >1200   CONVERT FLOATING POINT TO INTEGER
FAC    EQU  >834A   FLOATING POINT ACCUMULATOR
ARG    EQU  >835C   FLOATING POINT ARGUMENT
CSN    EQU  >1000   CONVERT STRING TO NUMBER (W/XMLLNK)
INT    EQU  >0022    FIND LARGEST INTEGER IN F.P. NUMBER
(W/GPLLNK)
FADD   EQU  >0600   FLOATING POINT ADDITION (XMLLNK)
FMUL   EQU  >0800   FLOATING POINT MULTIPLY (XMLLNK)
FDIV   EQU  >0900   FLOATING POINT DIVIDE   (XMLLNK)
FAC11  EQU  >8355   LSB ADDRESS FOR CNS
FAC12  EQU  >8356   STRING LEN FOR CNS
*
*  FOLLOWING DISPLAYS A NUMBER USING AN
*  UNDOCUMENTED FEATURE WITH GPLLNK
*  BY CONVERTING THE INTEGER PLACED AT
*  >835E INTO A STRING, WHICH IS THEN
*  DISPLAYED AT ROW 12, COLUMN 7
*  THE NUMBER IS DISPLAYED AS AN UNSIGNED
*  INTEGER FROM 0 THROUGH 65535
*  THANKS TO MERLE VOGT FOR THIS ONE!
*
*
       LI   R0,11*32+6   SET R0, ROW 12, COL 7
       MOV  @NUMBER,@>835E  PLACE NUMBER AT >835E
       CLR  @>837C       CLEAR GPL STATUS BYTE
       BLWP @GPLLNK      USE GPLLNK
       DATA >2F7C        TO CONVERT INTEGER TO STRING (UNDOC-
UMENTED)
       MOVB @>8361,R2    GET STRING LENGTH
       SRL  R2,8         RIGHT-JUSTIFY
       MOVB @>8367,R1    GET LOW BYTE OF ADDRESS
       SRL  R1,8         RIGHT-JUSTIFY
       AI   R1,>8300     ADD >8300 HIGH BYTE
       BLWP @VMBW        WRITE STRING TO SCREEN
*
*
*  FOLLOWING GETS A NUMBER FROM A DISPLAYED STRING,
*  THEN CONVERTS THAT TO AN INTEGER IN R5
*  (ON ENTRY, R0 POINTS TO NUMBER'S SCREEN LOCATION)
*  (ON EXIT, INTEGER VALUE IS IN R5 AND AT FAC)
*
GETNUM
       MOV  R0,@FAC12    PLACE SCREEN ADDRESS AT >8356
       BLWP @XMLLNK      USE XML LINKAGE
       DATA CSN          TO CONVERT STRING TO NUMBER
*  AT THIS POINT, FAC CONTAINS NUMBER IN FLOATING POINT
*  FORMAT, AS EIGHT BYTES IN RADIX 100 NOTATION
       BLWP @XMLLNK      USE XML AGAIN
       DATA CFI          TO CONVERT FLOATING POINT TO INTEGER
       MOV  @FAC,R5      MOVE INTEGER TO R5
       RT                RETURN
*
*
```

THE ART OF ASSEMBLY—

(Continued from Page 12)

105.2 65,1,5,20,0,0,0,0
89,999.999 66,8,99,99,99,90,0,0

Still clear as mud? Then think of it like scientific notation. In that notation, 8.9999999E+4 would represent the last number shown above, where the number after the E is the power of 10 that multiplies the number before the E, in this case 10 raised to the fourth power, or 10,000. In the Radix 100 case, the number used as an exponent is a power of 100, so that 10,000 is represented by 2, not four. (10000 is 100 raised to the second power.)

We could go on like this all day and all night, but if you've got the idea by now, you can see the rest yourself. If you haven't got it by now, 30 more pages might not make it clear. In almost every case that you'll encounter, the problem of creating numbers in Radix 100 notation can be performed for you by the TI's internal routines anyway, so you won't need to understand it. (Aren't you glad we didn't discuss negative numbers in Radix 100?)

Perhaps we should start with a routine that allows the TI to make the floating point number for you, so you need not wrestle with the notation yourself. Suppose you have an integer in the range of 32767 through -32768. This could be a word in memory, for example. Converting to floating point format is simple. Place the word at FAC, which is at >834A. (FAC is an abbreviation for Floating Point ACcumulator.)

Once the number is at FAC, XMLLNK can be used to convert the integer to a floating point number by:

```
BLWP  @XMLLNK
DATA  >2300
```

This will convert the number at FAC (one word) into a floating point number, with the sign as appropriate. That is, numbers from 0 through >7FFF (0 through +32767) will become positive floating point numbers, while numbers from >FFFF through >8000 (-1 through -32768) will become negative floating point numbers. As we'll see later, we can in some circumstances either ignore or not ignore the sign of integer numbers. In a test we just ran, for example, we tried two different ways to convert an integer into a string on the screen, and one of those paid heed to the sign of the number, while another did not. Whoops, we're getting ahead of our story. We must stop doing that!

When we are going to deal in floating point numbers, we often put some equates into the beginning of our source code to give us mnemonic labels, so that it will be easier to understand what we're doing. A typical set of equates would look like this:

```
FAC     EQU   >834A   Floating point accumulator
ARG     EQU   >835C   Floating point argument
CIF     EQU   >2300   Convert Integer to floating point
CFI     EQU   >1200   Convert Floating point to Integer
CNS     EQU   >0014   Convert number (FP) to string
CSN     EQU   >1000   Convert string to number (FP)
FADD    EQU   >0600   Floating point addition
FSUB    EQU   >0700   Floating point subtraction
FMUL    EQU   >0800   Floating point multiplication
FDIV    EQU   >0900   Floating point division
FCOM    EQU   >0A00   Floating point compare
```

(See Page 14)

```
*  ANOTHER WAY TO DISPLAY A NUMBER
*  THIS ONE DISPLAYS A FLOATING POINT NUMBER
*  BY CONVERTING IT TO A STRING
*
*
       LI   R0,11*32+6   POINT AT ROW 12, COL 7
       MOVB @ZERO,@FAC11 PLACE A ZERO AT >8355
       CLR  @STATUS      CLEAR THE GPL STATUS BYTE
       BLWP @GPLLNK      USE GPL LINKAGE
       DATA CNS          TO CONVERT NUMBER AT FAC TO STRING
       MOVB @FAC12,R2    GET STRING LENGTH
       SRL  R2,8         RIGHT JUSTIFY
       MOVB @FAC11,R1    GET LOW BYTE OF STRING ADDRESS
       SRL  R1,8         RIGHT JUSTIFY
       AI   R1,>8300     ADD >8300 AS HIGH BYTE
*  FOLLOWING TWO LINES ARE USED IF
*  WE WANT TO IGNORE THE SIGN OF THE NUMBER
*  WITHOUT THESE TWO LINES, POSITIVE NUMBERS
*  ARE DISPLAYED WITH A LEADING SPACE,
*  NEGATIVE NUMBERS WITH A MINUS SIGN
       INC  R1           POINT TO 2ND CHARACTER IN STRING
       DEC  R2           DECREMENT LENGTH BY ONE
       BLWP @VMBW        DISPLAY THE STRING
*
*
* THE FOLLOWING ARE EXCERPTS FROM "TOOL4"
* AN ASSEMBLY PROGRAM THAT PROVIDED "DELAY CONVERSION"
* FOR MIDI-MASTER MUSIC SOURCE FILES
*
*
*  FIRST PART PLACES A DEFAULT CONVERSION FACTOR (2.5)
*  ON THE SCREEN, THEN ACCEPTS A USER INPUT
*
*
       LI   R0,8*32+20   POINT AT ROW 9, COL 21
CHO3   LI   R1,T5STR     POINT AT DEFAULT NUMBER (2.5)
       BL   @DISSTR      DISPLAY THAT
       LI   R4,6         SIX CHARACTERS MAX
       BL   @CRSIN       ACCEPT INPUT
       MOV  R0,@FAC12    MOVE START ADDRESS TO >8356
       BLWP @XMLLNK      USE XML LINK
       DATA CSN          TO CONVERT STRING TO FLOATING POINT
NUMBER
       MOV  @FAC,R4      MOVE THE WORD AT FAC
       JEQ  CHO3         IF THAT'S ZERO, INVALID ENTRY
       LI   R9,FAC       ELSE POINT TO FAC
       LI   R10,TWOPT5   AND TO STORAGE PLACE
       LI   R4,8         EIGHT BYTES
       BL   @MOVBTS      MOVE THAT FP NUMBER TO STORAGE
*
* THE PROGRAM READS THE MUSIC SOURCE FILE INTO MEMORY
*
* THE PROGRAM IS WRITING THE FILE FROM MEMORY BACK TO DISK
* WHERE WE RE-JOIN THE ACTION
*
* AT THIS POINT, WE'VE FOUND A FILE RECORD THAT
* CONSISTS OF "#DELAY=" PLUS A NUMBER
* AND PLACED THAT RECORD AT LABEL TEMSTR
* THE PROGRAM WILL EITHER MULTIPLY OR DIVIDE
* THE NUMBER AFTER "#DELAY=" BY THE USER INPUT
* OBTAINED IN THE PREVIOUS "SNIPPET"
*
       LI   R0,21*32+8   POINT AT ROW 22, COL 9
       LI   R4,24        24 CHARACTER FIELD
       BL   @CLRFLD      CLEAR THAT FIELD
       LI   R1,TEMSTR    POINT AT TEMPORARY STRING
       BL   @DISSTR      DISPLAY THE STRING
       AI   R0,7         ADD 7 TO POINT AT NUMERIC PART
       MOV  R0,@FAC12    PLACE THAT ADDRESS AT >8356
       BLWP @XMLLNK      USE XML LINKAGE
       DATA CSN          TO CONVERT STRING TO F.P. NUMBER
       LI   R9,FAC       POINT AT NUMBER IN FAC
```

THE ART OF ASSEMBLY—

(Continued from Page 13)

INT EQU >002 Compute greatest integer

Of these, all except CNS and INT are accessed through XMLLNK CNS and INT are accessed with GPLLNK. When using the floating point math routines, such as FADD and FMUL, the two floating point numbers to be added or multiplied are first placed at FAC and ARG. Each must be a valid eight byte floating point number. After any of the math operations, the result is a number at FAC. For adding and multiplying, it's of no importance which of the numbers is at FAC and which at ARG before the operation, except of course if one must keep one of those numbers handy, in which case ARG would be the proper place for that number, since FAC gets modified by the operation. For division or subtraction, it's important to know which number goes at which location. For subtraction, FAC is the number to be subtracted from ARG, with the result placed at FAC. For division, FAC is the divisor of the number at ARG, and again FAC gets the result. Put another way, for subtraction, we could write that FAC=ARG-FAC, while for division FAC=ARG/FAC.

In various of our projects over the years, we have used most of these utility routines. (The only one we can't remember using is FSUB.) They all work, and quite well. In one case that's memorable, we did something that really proved to us the incredible accuracy of the TI's floating point routines.

AMAZING ACCURACY

Once, we were creating a little utility for XB users, designed to place musical note values into an array variable. The XB program would give us a starting note value and the number of octaves to fill in the array, then we would take that starting number and multiply it repeatedly by the 12th root of two to make all the half-tone steps. The 12th root of two is the number 1.059463094. We expressed that number in Radix 100 notation in our source file with ten significant digits. What happens with repeated multiplication by this number is that every 12th time, the number doubles. (e.g. 110 multiplied by the 12th root of two 12 times equals 220, or 24 times equals 440, and so on.) We found that, using the TI's FMUL routine, making a seven-octave run, or multiplying 84 times in succession, then rounding the result, the result was exactly correct. Of course without the rounding, there is some cumulative error in the last couple of decimal places, but you can use the TI's own utilities to get results more accurate than most PC basic programs would yield. The rounding was done by adding .5 to the result number, then using our own "int" function, which worked faster than the function provided by the "greatest integer" function with GPLLNK. Just looking at raw numbers, without rounding, the PC basic on our trusty laptop gave a result of 14,080.05 after a seven-octave run, while the TI gave the result as 14,079.9996, which is a much more accurate number, off by only .0004 from the exact expected result.

HOW TO USE THEM

In today's sidebar are a couple of small examples taken from some of our recent work on tools for MIDI-Master. Note that in these cases we didn't have to create any floating point numbers ourselves, except for the value 0.5 used in rounding, but just used TI's routines. Taking numbers from the screen or from anywhere

```
       LI   R10,ARG       AND AT ARG LOCATION
       LI   R4,8          EIGHT BYTES TO MOVE
       BL   @MOVBTS       MOVE F.P. NUMBER FROM FAC TO ARG
      LI   R9,TWOPT5     POINT AT USER'S INPUT CORRECTION NUM-
BER
       LI   R10,FAC       AND AT FAC
       LI   R4,8          EIGHT BYTES
       BL   @MOVBTS       PLACE USER'S NUMBER AT FAC
       MOV  @BFLG,R4      CHECK FOR MULT OR DIV
       JEQ  DIV           IF ZERO, DIVIDE
*
* BFLG SIMPLY INDICATES WHETHER A MULTIPLY OR DIVIDE IS
NEEDED
* DEPENDING ON AN EARLIER USER INPUT (NOT SHOWN)
*
       BLWP @XMLLNK       ELSE USE XML
       DATA FMUL          TO MULTIPLY
       JMP  RNDOFF        THEN JUMP AHEAD
DIV    BLWP @XMLLNK       USE XML
       DATA FDIV          TO DIVIDE ARG BY FAC
*
* CODE STARTING AT RNDOFF CORRECTLY ROUNDS THE RESULTING
NUMBER
* BY FIRST ADDING 0.5, THEN TAKING THE INT OF THAT NUMBER
*
RNDOFF
       LI   R9,FPHALF     POINT AT FLOATING POINT NUMBER 0.5
       LI   R10,ARG       AND AT ARG
       LI   R4,8          EIGHT BYTES
       BL   @MOVBTS       MOVE 0.5 F.P. INTO ARG
       BLWP @XMLLNK       USE XML
       DATA FADD          TO ADD .5 TO RESULT OF ABOVE
       CLR  @STATUS       CLEAR GPL STATUS BYTE
       BLWP @GPLLNK       USE GPLLNK
       DATA INT           TO FIND LARGEST INTEGER
*
*  CODE FROM HERE ON CONVERTS THE NEW NUMBER TO A STRING,
*  PLACES THE NEW NUMBER STRING INTO THE FILE RECORD FOR
*  OUTPUT, AND WRITES THAT RECORD TO THE OUTPUT FILE
*
*
* PORTION OF DATA PART OF TOOL4 SOURCE CODE
*
FPHALF BYTE 63,50,0,0,0,0,0,0 THE NUMBER 0.5 IN FLOATING
POINT FORMAT
TWOPT5 BSS  8           STORAGE FOR USER'S CONVERSION FACTOR
AS F.P. NUMBER
TEMSTR BSS  81          TEMPORARY STRING STORAGE LOCATION
DELSTR BYTE 7           STRING LENGTH
       TEXT '#DELAY='       COMPARISON SUB-STRING TO FIND
RECORDS OF INTEREST
T5STR  BYTE 3           STRING LENGTH
      TEXT '2.5'        DEFAULT CONVERSION FACTOR STRING
*
```

in VDP memory was easy using the CSN routine. We provide a pointer to the location of the string in VDP RAM, and let the routine do the work of turning that into a floating point number. This routine reads the number from VDP, and stops it finds a space or some other character that's not a valid numeric character. If the string presented is not a valid number, the routine returns 0 at location FAC, and this can be used as an indication that no valid number was present, unless of course 0 is a valid entry anyway. In the example shown for a user input number, 0 was not a valid entry, so finding that FAC = 0 would cause our program to replace the default string in the input field and make the user re-enter either the default or some other valid number.

(See Page 15)

THE ART OF ASSEMBLY —

(Continued from Page 14)

This conversion routine will also work correctly if the string is a "scientific notation" number, such as 9.00E3, which will yield 9,000 as a floating point number. It works equally well for negative numbers, and so on.

The companion service, convert number to string, also comes in awfully handy. This gives us a really quick way to put floating point numbers on-screen as strings, without having to develop any such routines for ourselves. One small caution must be observed about the strings created by CNS. This service reserves a leading space for the minus sign, and thus for positive numbers the string will have a space in its first character. In the example shown, we were working with numbers that were always positive, and MIDI-Master would balk if there were a leading space before the number, so we simply incremented our pointer before displaying the string, so as to eliminate the space. Doing this can also perform a kind of "absolute value" function, by simply ignoring the minus sign if it's there as the first string character.

Integer numbers can also be converted into strings, and in two ways. The integer can first be placed at FAC and then converted to FP, and then to a string. This will produce a numeric string that recognizes the sign of the integer you started with, and reports strings as 0 through 32767 or -1 through -32768. You can get the computer to ignore the sign and produce a string ranging from 0 through 65535 by using the undocumented feature provided us by Merle Vogt. For this, the integer is placed at >835E, then GPLLNK is called to make the conversion as shown in the sidebar. In this case, the string is always 0 or positive, and there's no leading space to contend with.

For those interested, the details for use of these routines is on pages 254 through 261 in the E/A manual. There is a warning provided on page 261 concerning the use of the CIF routine when working from an Extended BASIC environment. This warning can safely be ignored under some circumstances. We have been able to use these routines safely even when an Option-5 Program File was loaded and run from Extended BASIC, provided only that we dumped the E/A utilities starting at >2000 and running through >23BA into low memory. (We covered how to do this in an earlier column in this series.) The fact that this all worked was one of those pleasant little surprises that come along every now and then. Another example for the "we learn something every day" category.

We've gone on at some length about this topic, but still have the feeling of having just started. Perhaps we'll continue this subject in a future article.

Roll your own

Making that assembly program do what you want it to

By BOB CARMANY

How many times have you found an assembly language program that would almost do what you wanted it to do? Or, if it was just a few sectors smaller, it might fit on that last bit of space on your RAMdisk? That is exactly the problem that I faced several months ago.

I found that the USER dictionary for SPELLIT! worked much better if it was alphabetized so I used a rather ancient text sort program written by Dave Romer and John Clulow in the early '80s to sort the USER file each time I added a few words to it. It loaded as a program image file from Funnelweb and the sort was fairly rapid. In fact, it worked quite well and everything was fine until my USER dictionary contents approached 300 words which was the limit of the program. I was faced with the prospect of splitting the USER dictionary, sorting the two parts, and recombining the lot every time that I added some new words which was not my idea of an efficient method of operation!

A few other constraints cropped up along the way as well. The increased size of the F'WEB V.5.00 editors reduced the available space on my RAMdisk to 10 sectors (after moving the original text sort program elsewhere). So, a replacement had to accommodate more than 300 80-column records *and* fit into a 10-sector space. I also wanted it to operate out of the F'WEB environment and preserve the mailbox with the stored text file name. This was indeed going to be interesting!

I had just about given up on the project when I got my June 1993 copy of MICROpendium. Bruce Harrison's sort routine looked like it might fit the circumstances, but it would take a bit of "engineering" to get the job done. I thought of rewriting the source code entirely to take advantage of some of the internal F'WEB routines but there were some register contentions and the idea of going through 500-plus lines of source code didn't attract me. The best way to attack the problem, I thought, was a rewrite of selected sections of code and the addition of the suggestions that Bruce mentioned in the article. Over the space of the next couple of days, I laboriously typed in and checked the original source code from Sidebar 24.

The original program loaded into low memory and used the entire 24K of high memory to store the text string arrays. As written, the input prompts were off-center on the screen and there was no program attribution. It was a real "diamond in the rough." Bruce made the suggestion of adding appropriate "beeps" and "honks" in his article. Some other additions could be made, but that was basically the program I started out with in the beginning.

Within these boundaries, I decided to try to customize his source code to something I could use within my system. I decided I would come up with a program image program as a final product to conserve space on my RAMdisk. Besides, program image files load much faster than object code.

(See Page 16)

ROLL YOUR OWN—

(Continued from Page 15)

Since we have gone through the preliminary constraints, let's go ahead and look at how the original program was altered. In cases where several lines of code were altered, the complete section of code will be reproduced. For the odd line or two, the tag, the line prior to the altered line and the altered line (or lines) will be reproduced. With this in mind, it is time to begin.

```
REF     VMBW,VMBR,VSBW,VSBR
REF     DSRLNK,KSCAN,GPLLNK
```

GPLLNK was added to the REF block to accommodate the use of "beep" and "honk" sound effects as suggested in the original text by Bruce Harrison.

```
        DEF     START,SFIRST,SLAST
        AORG    >2678
SFIRST  B       @START
```

SFIRST and SLAST were added to the DEF block so that the object code could be later converted to a program image file to reduce the size of the finished product. SFIRST branches to the start of the program

```
START   LWPI    WS              Load user workspace
        LI      R15,RTNSTK      Set stack for high level subroutine
        LI      R0,3            Row 1, column 4
        LI      R4,23*SCRWID    23 rows to clear - ALTERED LINE
        BL      @CLRFLD         Clear them
        LI      R1.TITLNM       Load title name - NEW LINE
        BL      @DISSTR         Display it - NEW LINE
        AI      R0,3*SCRWID+4   Come down two rows and center - ALTERED LINE
        LI      R1,INFSTR       Set for input file prompt
        BL      @DISSTR         Display prompt
        AI      R0, SCRWID      Down to row 2
        LI      R4,15           15 bytes
        CLR     R1              Clear R1 - NEW LINE
        MOVB    R1,@>837C       Move to GPL workspace - NEW LINE
        BLWP    @GPLLNK         Access GPLLNK - NEW LINE
        DATA    >0034           Generate BEEP - NEW LINE
        BL      @CRSIN          Use CRSIN subroutine
        CI      R8,15           Has F-9 been struck
        JNE     PLCFN           If not, go on
        B       @EXIT           Else quit
```

The first altered line in this section of code effectively clears the screen down past any error messages that appear. Since the program restarts after an error, this was a necessity. The next line moves the prompt down and centers it. The first two new lines display "HARRISON SOFTWARE TEXT SORT" as a title and attribution. The data required for the message is at the end of the program with the other message. The next set of four new lines generates a "beep" at the "INPUT FILE" prompt. The relevant information is from page 251 in the Editor/Assembler manual.

```
PLCFN   LI      R9,TEMSTR       Point at temporary string
        :       :       :
        :       :       :
        BL      @MOVSTR         Move data from TEMSTR to PAB data
        LI      R4,>A050        Set R4 to >A050 - ALTERED LINE
        :       :       :
```

Since I use SPELLIT! in conjunction with the word processor in F'WEB, one of the self-imposed provisions was to avoid trashing the F'WEB mailbox and retain the default file name. Thus, the beginning address of the text string array storage was moved up to >A050 to avoid any conflict and keep everything intact. The slight reduction in storage space is scarcely missed.

```
OFNF1   LI      R0,12*SCRWID+4  Set row 12 column 5 - ALTERED LINE
```

This line moves the "SORTING FILE" message down to the middle of the screen and centers the text for easier reading

```
READON  LI      R0,PAB1+5       Point at PAB+5 in VDP RAM
        :       :       :
        :       :       :
        LI      R10,>A050       Point at start of array - ALTERED LINE
```

This line points at the start address of the text array storage which was changed earlier in the program to >A050.

```
SMALL   MOV     @ENDSTR,R10     Point at the end of array
        :       :       :
        :       :       :
        JLT     DEC9            If not proceed
        LI      R0,20*SCRWID+2  13 rows to clear - ALTERED LINE
        :       :       :
        CLR     R1              Clear R1 - NEW LINE
        MOVB    R1,@>837C       Move to GPL workspace - NEW LINE
        BLWP    @GPLLNK         Access GPLLNK - NEW LINE
        DATA    >0036           Generate HONK - NEW LINE
```

The first altered line moves the error message up a couple of lines from the original location and centers it on the screen for easier reading. The four new lines generate a "honk" sound immediately after the error message is displayed. (See page 251 in the E/A manual.)

```
GETOFN  LI      R0,3            Set to row 1 column 3
        LI      R4,13*SCRWID    13 rows to clear - ALTERED LINE
        :       :       :
        AI      R0,SCRWID+4     Come down two rows - ALTERED LINE
        :       :       :
        :       :       :
        CLR     R1              Clear R1 - NEW LINE
        MOVB    R1,@>837C       Move to GPL workspace - NEW LINE
        BLWP    @GPLLNK         Access GPLLNK - NEW LINE
        DATA    >0034           Generate BEEP - NEW LINE
        :       :       :
```

The first altered line clears the screen down past the "SORT FILE . . " message since that chore has been finished at this point. The second line prepares for the display of the "OUTPUT FILE" prompt and centers it on the screen. As before, the four new lines generate a "beep" at the prompt prior to file name entry.

```
WRTF2   LI      R9,>A050        Point to the start of array - ALTERED LINE
```

Once again, we are avoiding trashing the F'WEB mailbox by writing to a point above it.

```
EXIT    BLWP    @>0             Return to XBASIC - ALTERED LINE
```

The original program had the exit to the E/A cartridge which worked just fine if you were operating the sort routine from there. I wanted it to exit to XBASIC where the AUTO program on my Quest RAMdisk would intercede and display the menu screen. It made things a lot more convenient!

(See Page 17)

ROLL YOUR OWN—

(Continued from Page 16)

```
FILER1   SLA  R1,1          Double number in R1
         ¦    ¦     ¦
         ¦    ¦     ¦
         CLR  R1           Clear R1 - NEW LINE
         MOVB R1,@837C     Move to GPL workspace - NEW LINE
         BLWP @GPLLNK      Access GPLLNK - NEW LINE
         DATA >0036        Generate HONK - NEW LINE
         BL   @KEYLOO      Stop at keyloop
         LI   R0,20*SCRWID Load R0 for 21 rows - ALTERED LINE
         ¦    ¦     ¦
```

Once again we use the GPLLNK to generate a "honk" when an error is encountered, wait for a keypress, and then clear the error message before restarting the program. The altered line clears the first 21 rows.

```
CRIX     B    @SUBRET      Return to SUBRET
```

This bit of text was omitted in the original but gleaned from Sidebar 5 where it appeared. This is a dummy to keep all of the JUMP instructions within range.

```
EDGE     BYTE >2B          Byte for input/output entry - ALTERED LINE
```

I did not care for the use of a non-displayable character as the input/output prompt delimiter so it was changed to >2B — the "+" sign for my personal preference.

```
LEFTV    BYTE 8            Left arrow
TITLNM   BYTE 27           Length byte - NEW LINE
         TEXT'HARRISON SOFTWARE TEXT SORT'
```

The TITLNM message displays the appropriate attribution at the top of the screen when the program starts.

```
SLAST    END
```

SLAST was added to accommodate the production of a program image file.

With all of the altered lines and new lines stuffed in their appropriate places, we are ready to assemble the source code. If you haven't made any errors you should get a "<0000>" ERRORS message — and a 29-sector piece of object code. At this point, the program will run from the E/A cartridge under the name "START". If this is as far as you want to go with it, change the last line in the program to:

```
SLAST    END  START
```

and the program will auto-run from E/A.

We were going for something a bit smaller than what we have created so far. The 29-sector object code file was too big to fit in the 10-sector space I had left on my RAMdisk but it should scrunch down nicely into a program image file.

One major obstacle to overcome was the fact that the program was AORGed into low memory. TI's SAVE utility will not handle a program that has been AORGed into low memory — but FSAVE that comes with F'WEB will!

To get the final product, enter F'WEB, load the object code file from LOADERS and load DSKx.FSAVE from the appropriate disk drive. Press FCTN-3 at the prompt and then ENter. Using the "S" and "D" keys, move the cursor to SAVE and press FCTN-6. Follow the on-screen prompts and the file will be saved in program image format. The result is a nine-sector file that will run *only* from F'WEB.

All of this resulted in a lively correspondence with Bruce Harrison, who suggested another way to create an E/A 5 general purpose program image file. First, remove the GPLLNK from the REF block and add the following source code lines immediately after MOVBTS.

```
*  GENERAL PURPOSE GPL LINK
*  BY DOUG WARREN/CRAIG MILLER
*
*
GR4      EQU  GPLWS+8
GR6      EQU  GPLWS+12
STKPNT   EQU  >8373
LDGADD   EQU  >60
XTAB27   EQU  >200E
GETSTK   EQU  >166C

GPLLNK   DATA GLNKWS
         DATA GLINK1
RTNAD    DATA XMLRTN
GXMLAD   DATA >176C
         DATA >50
GLNKWS   EQU  $->18
         BSS  >08
GLINK1   MOV  *R11,@GR4
         MOV  *R14+@GR6
         MOV  @XTAB27,R12
         MOV  R9,@XTAB27
         LWPI GPLWS
         BL   *R4
         MOV  @GXMLAD,@>8302(R4)
         INCT @STKPNT
         B    @LDGADD
XMLRTN   MOV  @GETSTK,R4
         BL   *R4
         LWPI GLNKWS
         MOV  R12,@XTAB27
         RTWP
```

If you have a shorter general purpose GPLLNK routine (there are several around) you can substitute it for the previous source code.

Save the assembled object code as before with FSAVE. Bruce used the file name SORTEM for his program image file but you can pick another name if you wish (e.g. SORT1). Then type in the following source code as a second file. This is a tip from Merle Vogt via Bruce Harrison for stashing the E/A utilities in a second, auxiliary file.

```
         DEF  SFIRST,SLAST,SLOAD
SFIRST   EQU  >2000
SLOAD    EQU  >2000
SLAST    EQU  >23BA
         END
```

Assemble this bit of source code into object code and use TI's SAVE utility to produce a program image file. Use the file name SORTEN for this one if you used the same name that Bruce chose

(See Page 18)

ROLL YOUR OWN —

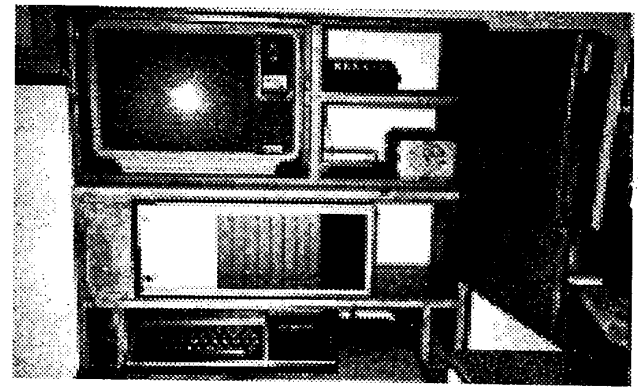
(Continued from Page 17)

for his first file name. Otherwise, increment the last character of the first file by one for the second file name (i.e., SORT1 and SORT2, etc.)

Use an appropriate disk sector editor to change the first byte of the SORTEM file from >0000 (no more file s to load flag) to >FFFF (additional file load flag) and you have two files SORTEM and SORTEN (or SORT1 and SORT2) that will load from *any* E/A option 5 loader. The only cost is an additional five sectors of disk space required for the second file. Essentially, you have loaded the required E/A utilities into low memory from >2000 to >23BA.

Incidentally, you can use this second procedure for any file that loads into low memory that you want to convert to a general purpose E/A-5 program image file.

The final word is that you can change existing source code to fit into your own system and work the way that you want it to work. It just takes a little time and a little effort to get the job done!



SYSTEM OF THE MONTH

Plenty of power at little cost

Our system of the month comes from John Collins of Fort Myers, Florida, who writes:

Here is a picture of my computer cabinet with TI99/4A computer and PE Box. It has disk controller, Memory Expansion Card, Disk Memory Drive, RS232 card, BMC green monitor and Epson-Action Printer T-1000. I also use my other computer with stand-alone disk controller, stand-alone memory expansion, stand-alone disk drive and PIO Plus from CorComp Inc., and I might get another printer later.

I do make business cards for myself and other people, too.

I bought my computer for $10 and the PE Box for $13 from Goodwill Store and other items from other companies.

I would like to thank Jim Lesher of Dallas, Texas, and Rodger Merritt of Fullerton, California, for helping out. I'm still looking for more small cliparts for business cards.

Would you like to share your TI/Geneve system with other readers? Send photo and description to MICROpendium Systems, P.O. Box 1343, Round Rock, TX 78680.

Jerry Price of Tex-Comp reminisces

Dealer for more than a decade has had customers from around the world

By LAURA BURNS

After more than a decade as a TI retailer, Jerry Price of Tex-Comp has a lot of memories.

MICROpendium asked him to share some of them.

"My educational background is as an engineer and a patent attorney," he explains. "The only computers we had when I was in college was the kind taking up a full room. I think I built an analog computer from a kit when I was in college."

Price was first introduced to the TI99/4 by a client in 1980.

"He was a former IBM engineer who had a hobby of collecting and restoring pinball machines and jukeboxes. Soon he was selling them and opened a warehouse in the San Fernando Valley."

This client called and showed Price the literature on the 99/4 Texas Instruments had sent him.

"The way the computer business was then, the only place you could walk in and buy a computer was Radio Shack," Price explains. "Computer stores sold kits and maybe a rudimentary Commodore called Pets, which had a membrane keyboard."

To dealers, TI offered as an introductory special the console, monitor, stand-alone disk controller, stand-alone memory, stand-alone RS232 and Speech Synthesizer for "around $1,600," he recalls. His client ordered three, one for himself, one for Price and one for another friend.

Price decided to enter the business because, after he got his computer, "people would come see my three-year-old son doing Early Reading and they wanted one."

At that time TI considered its home computer a department store item, selling with other appliances, but "it didn't do well. They didn't know their market or how to sell it." It was also sold in office supply stores.

In late 1980 or early 1981, he recalls, TI had an introductory fair in a hotel banquet room in the Los Angeles area. TI gave out coupons for a bonus Speech Synthesizer with the 99/4A.

Price began his mail order business when he found that owners of the computer could not get much software.

"The department stores only carried the most popular titles," he says. "I felt this would be a natural for mail order. Then in October 1991 the *99er Magazine*, which later became *Home Computer Magazine*, came out."

Tex-Comp used Lionel Trains' marketing strategy as a guide to get started in mail order of TI products

Price realized that a magazinespecifically for the TI99/4A would reach his target audience better than a generic computer magazine.

"We placed a full page ad and the rest was history," he notes.

He says it was sometimes hard to keep items in stock because of delays from TI, a company he describes as "not good with retailers."

TI would not sell its mailing list to any dealer, he recalls, but an Oklahoma butcher, Charles La Fara, got the mailing list for his International Users Group. At that time, most programs were on tape.

IUG members would trade tapes, Price recalls. "It was a sort of freeware software exchange."

But then La Fara "shifted over" and became a dealer.

This was unfair, Price notes, because "he had access to the list and no other dealers did."

The home computer marketplace began to get crowded with Atari entering and with Commodore introducing the Vic-20 and the Commodore 64. Also, Price says, internal dissension affected TI. The data products division did not like seeing software for business on the home computer. Price says TI for this reason "sat on" the TI-Count business package and finally the author sold it directly.

This attitude, Price says, "held back TI as a more powerful and capable machine for a long time."

Price says at the time he got into the mail order business "a couple of distributors and stores had mail order divisions," but he was one of the first to make it his major orientation.

"My passion is collecting Lionel trains," he says. "I took my guidelines from how they are marketed. They don't get good distribution in hobby stores because of their pricing, and the only was is mail order. I used their ads as a guide to my original mail order ads for TI stuff, because the two were analogous in my opinion."

In Los Angeles, he was able to buy from a couple of distributors.

"Later TI let us buy direct as we were moving a lot," he says, noting that Tex-Comp's business was in the millions-a-year category at one point.

Around 1981 Gary Kaplan of *99er Magazine* announced a TI fair in San Francisco. TI stepped in to underwrite it as a sponsor. This was the first chance for many people to display programs, Price notes.

Also, multi-level marketing, "selling the TI like Shaklee or Herbalife" emerged. "In my opinion, it was doomed from the beginning," Price says.

Tenex came into the market as a mail-order rival at about this time, he recalls. "I had never heard of Roger Dooley. I had a Dun and Bradstreet check done on him and found out that he was working out of his bedroom at the start. He was a good marketing guy. He got a magazine distributor to put his catalog on magazine stands where you had to pay for it, and got a lot of people to pick it up that way. Now I be-

(See Page 20)

JERRY PRICE—

(Continued from Page 19)

lieve he's in IBM stuff almost exclusively."

Interesting Tex-Comp customers through the years have included "a fellow from Saudi Arabia who came into the store in all his robes. And once during the Nixon administration we had an order from the White House. I don't know who it was," Price says. Orders have come from all over the world including from what at the time was "behind the Iron Curtain." The company had to get State Department clearance to ship articles there. Once, also, Tex-Comp shipped a computer to a Navy ship while it was in port.

The cost of building computers began to fall. TI, Atari and Commodore were "underpowered for handling major work, for which they weren't intended in the first place," and consumers began buying PCs and Apples. IBM didn't realize what was happening either, he notes, and brought out the PC Junior which "bombed" when customers realized they could get a more powerful computer for little more money.

Price says he had a 99/2 which he recently sold. These didn't come out commercially since the price had dropped on the 99/4A. The 99/2, he says, had a black and white screen but a more sophisticated tape drive which allowed for searching and a hex-bus system for peripherals.

The later 99/8 was "more powerful but too little, too late," Price says. "TI could have done more, but it was too concerned about taking away from its bread and butter," the business computers.

In 1983, losing money, the company turned to a "Gillette approach," Price says, considering the computer analogous to a razor and selling it almost at cost, while planning to make money on the "razor blades," the cartridges. But Atari, "which had all the great games at that time," started making cartridges for the TI.

TI's solution was to modify the computer so it would not run cartridges by anyone else. However, Price says, this plan to block the Atari chip backfired. "People would buy it and take the whole thing back when the cartridges wouldn't run." Stores who got the returns complained and TI stopped making the modification, he says.

TI, in big financial trouble, finally threw in the towel. When it made the announcement, its stock jumped.

"They locked their key people in a room when they made the announcement to them," Price says. This was to prevent insider trading.

"The day they announced was not anything as significant as the death of President John Kennedy," Price says, "but like that day, a lot of people can tell you where they were and what they were doing when TI made its announcement."

Tex-Comp's business is "scaled way down" now, Price says, but will continue as a service to TI users. Users now tend to be hobbyists, he notes.

"It's an inexpensive hobby to get into," he notes, citing low prices of computers and peripherals that users have found on the second-had market.

As for the TI99/4A's longevity as a computer, Price says quality is the main reason.

"It's gone way beyond what TI's engineers thought it would."

Tex-Comp's Jerry Price is pictured with former TI booster Bill Cosby during the early days of the TI99/4A.

Tex-Comp helps provide computers to charities

Sometimes people come to Tex-Comp with used TI equipment they want to sell.

Jerry Price tries to persuade them to donate it to a worthy cause instead. In this way he has provided various groups with equipment in addition to equipment they have received from Tex-Comp.

"There are still school districts that use the TI," he notes, citing the case of a Los Angeles-area special education teacher who uses them.

Another frequent recipient is James Elliott at the St. Anthony Mission where the computers are used by Zuni Indians.

"I just got a nice letter from Sister Pat (Taylor)," Price says. "Some of the older nuns (at Marian Hall in Des Moines, Iowa) had a hard time using the joysticks with the TI because of arthritis. We rigged up a sort of a trackball for them to use."

NEWSBYTES

Geneve software mod supports high density disks, 3.2Meg. RAMdisk

Programmer Jim Schroeder has written a portion of code for MDOS that allows Geneve users with Hard and Floppy Disk Controllers to use high density disk drives, according to Don Walden of Cecure Electronics. These drives write up to 1.44 megabytes of data on a single 3.5-inch disk.

Walden also says that with the use of a new fairware program called Form, "if you have the Horizon 4000, 3000 or 2000 RAMdisk you can expand the RAMdisk up to 3.2 meg." An HFDC is not required for the Horizon RAMdisk.

Users who want to format their RAMdisks to the maximum, must use the new Form program. The RAMdisk version is scheduled to be available on bulletin boards. This is a new program, not an upgrade, Walden says, so users need to make new fairware contributions to Schroeder if they use this program. Schroeder's address is 4153 South Regal Manor Court, New Berlin, WI 53151.

For the HFDC upgrade, two things are required, Walden says. A user needs a 32K buffer chip with a minimum speed of 100 nanoseconds (150s and 120s will not work reliably, he says), and needs to have the 9216-B floppy data separator with 500 kilobauds.

"Emphasize that it has to be the B version," Walden says, "not just the 9216." If it does not have a B after the 9216, it is the slow version, which operates at 250 kilobauds.

He says the MDOS code has been supplied to Beery Miller for the latest release of MDOS. For information, contact 9640 News, P.O. Box 752465, Memphis, TN 38175, (901)368-1169 (voice) or (901) 368-0112 (BBS).

Walden says Tim Tesch has received the source code to the EPROM on the HFDC and has provided it to Schroeder to create the 1.44 meg option for TI99/4A users if sufficient space exists to do so.

AMS game, TI-Nopoly, available from Delphi

TI-Nopoly, by Joe Delekto and Jon Dyer, is available for downloading from the TI-Forum on Delphi, according to Jim Krych of Asgard Peripherals.

The TI Game is 72K in length and is designed for users of Asgard's Advanced Memory System.

The TI Game is 72K in length and is designed for users of Asgard's Advanced Memory System.

Horizon hopes SCSI will be ready for sale at Fest-West

Horizon Computer hopes to have its SCSI card ready in time for Fest-West, according to Bud Mills of Horizon.

Fest-West is scheduled for Feb. 19-20 in Tucson, Arizona.

"The progress in the last couple of weeks is astounding," Mills said Dec. 6.

Hardware modifications are being made to make the card work with the TI and the Geneve, he says.

"When we started, we didn't realize how much we had to learn, but we have learned a great deal," he notes. "The DSRs (device service routines) are not done yet, but all the pieces are starting to fall into place."

Persons involved in the project include Don O'Neil, Mike Maksimik, Brad Snyder and Jeff White, he notes.

Mills says that about 12 SCSIs were sold up front "that shouldn't have been." These will need to be modified, he says, and the owners should contact him.

For SCSI information, contact Budd Mills Services, 166 Dartmouth Dr., Toledo, OH 43614-2911; (419) 385-5946 (voice) or (419)385-7484 (BBS).

For information on Fest-West, contact Tom Wills, Fest-West '94 Committee, Southwest 99ers Users Group, P.O. Box 17831, Tucson, AZ 85731 or (602) 886-2460; BJ Mathis, (602) 747-5046; or the Cactus Patch BBS, (602) 290-6277.

Sorry, wrong number

An erroneous BBS number for Tim Tesch's S&T Software BBS appeared in the listing by Tesch which appeared in last month's MICROpendium.

The correct number is that which appears at the top of the article, (414) 464-1978. Also, a typo by MICROpendium gave Tesch the wrong area code for his voice number. The correct number is (414) 464-4946.

Tesch says he believes that the erroneous BBS listing was the result of a prank by a colleague.

Howe writes book about Internet

Random House has recently released *Internet Basics, Your Online Access to the Global Electronic Superhighway* by Steve Lambert and Walt Howe.

Howe, manager of the Delphi Internet Sig, is a former manager of the TI Forum on CompuServe and has written for MICROpendium.

The book is available at bookstores or from the publisher. Also, the book is available on Delphi for $27 plus $3 shipping and handling. To order, type GUIDES at the Internet Sig prompt.

Chicago members can renew on plastic

Members of the Chicago TI Users Group can renew their memberships with Visa or MasterCard by calling Cecure Electronics in Milwaukee, according to Don Walden, president of the group. The number to call is 414-679-4343.

Walden notes that out-of-town members of the group wishing to make nominations for officers for 1994 need to do so by Jan. 31, with nominations marked to the nominating committee's attention.

Send information about your products and services for the TI/Geneve community to MICROpendium Newsbytes, P.O. Box 1343, Round Rock, TX 78680.

The Cyc

Everything you ever wanted to know about the TI

By JOHN KOLOEN

Every once in a while you run into a product that's so good that when you talk about it you sound like you're endorsing it. There've been a number of TI/Geneve products that I've felt that way about, among them the GRAM Kracker, the Myarc HFDC, the Horizon RAMdisk and Telco.

What seems to link these products is that they represented a breakthrough in how things are done. After I installed a GRAM Kracker, I knew I finally had control over my software. With the HFDC, I knew that I could have more and faster disk storage than I could probably use. With the HRD I had tremendous speed. With Telco, I had a powerful, yet easy-to-use terminal program that worked like no other. And with Cyc, I've got more information in one place about the TI than I ever thought possible.

The irony is that Cyc doesn't run on a TI, nor a Geneve, for that matter. It's native habitat is a PC equipped with Word Perfect 5.1 or higher. Cyc comes as four self-extracting, compressed files. To install it, create a directory called Cyc on your hard drive and copy the files to it. Then execute the files. When the process is completed, which will take several minutes depending on how fast your PC is, you'll be left with nearly three megabytes of data, all of it about the TI and Geneve.

The heart of the Cyc database is, perhaps, its indices of various publications that covered the TI, including *99'er Magazine*, *Texas Instruments Home Computer News*, *Computer Shopper*, MICROpendium, *The Smart Programmer* and *Enthusiast 99*. These are not mere listings of headlines, page numbers, author names and dates, though this information is included. Many of the references also include abstracts of the articles, going well beyond any of the magazine indices that I'm aware of.

REVIEW

Report Card

Performance	**A**
Ease of Use	**A**
Documentation	**B**
Value	**A**
Final Grade	**A**

Cost: $20
Manufacturer: CaDD Electronics, 81 Prescott Rd., Raymond, NH 03077, 603-895-0119
Requirements: PC with Word Perfect 5.1 or higher, 5.25- or 3.5-inch drive (specify format wanted, 360K, 720K, 1.2M or 1.44M). See text for offer by Charles Good to supply The Cyc converted for use on a TI or Geneve.

Running out of Word Perfect instead of its own database engine, each index is loaded into memory separately. Using the MICROpendium index as an example, this 400+ kilobyte file can be searched using the WP search function. The index is by volume and issue number and in page number order. It appears that virtually every article, no matter how small, is included in the various magazine indices. In the case of MICROpendium, this includes Feedback, User Notes and Newsbytes. (See Fig. 1)

In addition to the magazine indices, there are also 26 alphabetical lists and one that lists items that are referenced by numbers. These lists include information from a variety of sources, including publications and catalogs. The alphabetical/numeric lists, which are cross-referenced, are great for anyone searching for a particular product or person having to do with the TI or Geneve. Suppose you know the name of a particular product — say the Panda Expansion Box. — and want to learn more about it. Well, load the "P" list, scroll or use search to find "Panda." The entry gives you the name of Bill Nelson, who created the Panda Expansion Box. Now, go to the "N" list and search on or scroll down to "Nelson." Not only do you

(See Page 23)

Fig. 1

```
Volume 3 Number 4. May 86. 48pp.

3:4:6. Comments. Still working on the mail. Micropendium changes
mailing label system. Inside is review of Gram Kracker. New
products include The Brain from Datax; and Computer War, Submarine
Commander, and River Rescue.

3:4:8. Feedback. Triple-Tech operational quirks (see 2:11:44);
CorComp disk controller quirks (see 3:5:8) [Glenn D. Knight]. Disk
Manager initialization time includes verify (see 3:3:8) [Carlo
Angelico]. Problems of being a freeware author [Monty Schmidt].
What is a cowputer? (see 3:2:cover) [Jeff Speeth]. Verification
comes after initialization (see 3:3:8) [Gary Cox]. Speak & Spell
program can be run from Extended Basic [Ken Gladyszewski]. Using
TI-Writer editor in 40-column mode [Bernhard F. Muller]. Needs
TI-Writer update to handle lines with more than 80 columns [R.T.
Duggan].

3:4:12. Building a super keyboard by Tony Johnson. Schematics and
pictures to build an external keyboard that includes: separate
numeric keypad; separate cursor controls; a 10-key function bar;
```

THE CYC—

(Continued from Page 22)

find a precise description of the Panda Expansion Box, including its dimensions and requirements, but you also get Nelson's address and phone number. (See Fig. 2)

Two files (PHX and PRICE) list virtually every home computer product manufactured by TI and its list price. Products of other manufacturers are also included in one of these files.

The Cyc grew out of Mike Wright's need to catalog is TI software and hardware collection. It is an enormous undertaking, particularly given the fact that he and Mark Van Coppenolle have also been vorking on the PC99 emulator, a mammoth project by itself. Wright says that he called his encyclopedia "Cyc" as a result of his passion for trains and model railroading. An annual publication called Cyclopedia was the standard reference for steam locomotive users and was referred to in the trade as "The Cyc."

Fig. 2

```
Rabbit Hunt. See Cumberland Technology.

Rabbit Trail. See Funware.

Racal Vadic modem
Tip on how to redial . . . . . .
. CS 87:03:281

Racing. 1476. See Appendix, 99/4 IUG Programs.

Racing. See Kemp Software.

Racing Letters. See Microcomputers Corporation.

Radio ham. See Ham, radio.

Radio Log. 6096. See Appendix, 99/4 IUG Programs.

Radio Shack
Selling TI-99/4A modulators. . . . . .
. . .99er 5:3:60

Radio teleprinter sound
Basic listing for. . . . . .
```

What are the drawbacks toThe Cyc? Running out of a PC is one. Secondly, the indices of currently published periodicals is not up-to-date. Cyc includes only the first eight years of MICROpendium, for example. However,the indices of *99er Magazine* and other extinct publications are complete. As you can see, there aren't many drawbacks.

So what do you do if you don't have Word Perfect 5.1 or access to a PC? Well, earlier versions of Word Perfect don't recognize WP 5.1 files. If you have a word processor that accepts WP 5.1 files, you shouldn't have a problem. Or, you can find shareware conversion program that will convert the WP 5.1 files into straight ASCII text.

If you don't have access to a PC, you could ask a friend to convert the files into ASCII and then convert them into TI format using a program such as PC-Transfer. Or you can transfer the files over a modem or using a null modem. The problem with using these files on the TI is that some of them are very large and can't possibly fit into the TI's memory. MY-Word on the Geneve doesn't have the capacity to handle the larger files either. But the ambitious user could break the files down into acceptable sizes and resave them using modified file names. It is possible, you just have to want to do it.

A TI VERSION IS AVAILABLE

If you don't want to go through all of that, the easiest thing to do is to buy a copy of Cyc and then write to Charles Good. Good has converted all of the Word Perfect 5.1 Cyc files into D/V80 files which are small enough to fit into TI-Writer. Send him a paid receipt for the Cyc from C.A.D.D. Electronics, the equivalent of eight formatted DSDD disks (16 formatted DSSD disks) and a paid return mailer. He will put all the D/V80 files for the September 1993 update of the Cyc onto the disks and return the disks and receipt. He is asking for no fee for this service. He can be reached at P.O. Box 647, Venedocia, OH 45894; (419) 667-3131. Charles is very generous in offering this service and those who take advantage of it should be absolutely certain that the disks are formatted and that the return mailer includes adequate postage.

Ease of Use: If you know how to use Word Perfect, you'll have no problem using The Cyc. You just load the text files into memory and do what you like. You can easily modify the various entries, output them to a printer or save them as text files and incorporate them into other documents.

Documentation: The docs consist of a text file on disk called README. It tells you how to install the files. A second file, called INTRO, provides a history of The Cyc project.

Value: At $20, The Cyc is a bargain. It would be a bargain at twice the price.

Final Grade: As with any reference work, The Cyc is not complete. Wright hopes to continue building on it. And as long as the TI community continues to support the TI/Geneve there will be more and more to add to it. But in its current form, it's probably better than we as users have a right to expect. Wright put an enormous amount of time into it and. while he's not expecting to make much money from it, the effort resulted in a tool that any serious TI/Geneve user will find of enormous benefit.

Obviously, this is *the* encyclopedia of TI knowledge. Nothing else comes close, at least nothing else that I've seen. It's a labor of love that the entire TI community should be thankful for.

MICRO-REVIEWS

Beyond Wordwriter, Goblin and Star Mission, Page Pro Cut Out Sets, The Prostick II Joystick

By CHARLES GOOD

BEYOND WORDWRITER by DataBioTics

Funnelweb v5 is the most common TI word processor used by members of my local user group. In second place is Beyond Wordwriter, a cartridge that contains an enhanced TI-Writer editor plus the TI-Writer formatter.

This cartridge has been around since 1987 under the names Wordwriter, Wordwriter Xtra and Beyond Wordwriter, each an improvement over the former and all better than the original TI-Writer. Most "knowledgeable" 99/4A enthusiasts probably think Beyond Wordwriter is not currently in production, but this is incorrect. You can still purchase this cartridge from TM Direct Marketing, and DataBioTics will still manufacture new Beyond Wordwriter cartridges as required.

Why would anyone want a $40 cartridge word processor when other good word processors are available for disk systems?

1. Ease of use: I can't speak knowledgeably about First Draft word processor, but I do know why those members of my local user group with disk systems who use Beyond Wordwriter don't use the Funnelweb word processor. One has to use the Funnelweb word processor as part of the Funnelweb system, and these members find Funnelweb's numerous files and configuration confusing. With Beyond Wordwriter all you do is pop in the cartridge and begin typing your document.

2. Usable with a minimum system: With Beyond Wordwriter you can have the benefits of both an enhanced TI-Writer editor and text formatter with just a console and cassette recorder. You may remember from back in the old "no disk system yet" days that trying to save or load data files to tape from either BASIC takes a very long time and eats up huge amounts of tape as the computer saves or reads in one record at a time. Not so with Beyond Wordwriter, which saves documents to tape in memory image format. Cassette saving and loading of Beyond Wordwriter documents do not consume lots of tape and are almost as easy and fast as saving and loading BASIC programs. Documents saved to tape can later be loaded back into a fully expanded system for printing. Or, if you have a stand-alone printer interface ($20 at the 1993 Chicago faire) you can have a complete word processing system that does not utilize the bulky expansion box. You don't even need a tape recorder. Just create and edit your document, then print the document with Beyond Wordwriter using the built in formatter or directly from the edit buffer.

3. Column by column windowing: As you type or use the right arrow (FCTN-D), when your text reaches the right edge of the screen the display moves to the right one column at a time until the right margin is reached, and then jumps back to the left margin. This feature, not available in any other version of TI-Writer including 40-column Funnelweb, makes "what you see is what you get" word processing with a 40-column display much easier. One local user group member cites this feature as his main reason for using Beyond Wordwriter. Rapid left to right windowing is still available for browsing text.

Some additional comments about Beyond Wordwriter, not in any particular order: With a disk system the text buffer is larger than any other version of TI-Writer, probably because the software stays in cartridge RAM freeing up additional memory. You can save files in memory image format to disk, but with no obvious advantage in doing so, since they usually take up more disk space than the equivalent DV80 file. Saving files with SF automatically saves as memory image on a cassette and as D/V80 on a disk. Twelve screen color combinations are possible. When you edit, a handy status line at the bottom of the 40-column screen always shows at a glance right/left margin settings, word wrap and on or off setting (which you can also determine by checking for a solid or hollow cursor), cursor row/column position within the whole document, and normal or insert (FCTN-2) mode. A bell can be placed anywhere on the tab line to give you an audible warning when the cursor reaches the designated column. Beyond Wordwriter files load into other versions of TI Writer with no problems, but files created with other word processors should be saved with PrintFile if you want them to load into Beyond Wordwriter. The formatter is accessed directly from the editor command line by typing FF (FormatFile), and when formatting is finished you are returned to the editor command line. The formatter contains no mail/merge or define prompt feature, but the formatter does have one neat trick not found in the TI-Writer formatter. If you specify "O" (the letter O, not zero) at the WHAT PAGES? prompt, the formatter will print odd numbered pages. You can then turn the paper over and reinsert it into the printer for printing the even numbered pages on the other side of the paper.

I still prefer the Funnelweb v5 word processor. Funnelweb has many nice features lacking in Beyond Wordwriter ,such as Funnelweb's ability to mark a file for loading into the editor or formatter from the ShowDirectory menu. But if you don't like the Funnelweb system, or you like the idea of doing serious word processing without an expansion box, then Beyond Wordwriter deserves serious consideration.

Beyond Wordwriter is available for $39.95 from TM Direct Marketing, 1757 E. Bay Shore No. 12, Redwood City, CA 94063. Phone 800-336-9966

(See Page 25)

MICRO-REVIEWS —

GOBLIN and STAR MISSION by Betori Alessandro

These two games on one DSSD disk are probably the cheapest fairware in the whole world. The author and the TI99/4A E Geneve 9640 Computer Club Italia are asking users to send $1 in appreciation for the software. Star Mission is written entirely in Extended BASIC and resembles the John Phillips game Beyond Parsec. You avoid getting hit by the alien hordes as you protect your planet. There is good use of music and CALL SAY speech and plenty of "***WARNING***" on-screen messages concerning the approach of the bad guys.

The Goblin game includes very unusual speech and music and a stunning graphic at the end. This game talks to you WITHOUT the speech synthesizer. Although Goblin is mostly written in Extended BASIC, assembly CALL LINKs directly interact with the console's sound chip to produce this speech and music. Just before the game starts you are given oral instructions *in Italian*! As the game progresses you hear scratchy mysterious background music. This music is not like the pure tone plinkity plink sounds you usually get with CALL SOUND. Instead, the effect is sort of like listening to an old-time radio drama on an old radio. In the game your man is in the middle of a mysterious forest with ghosts floating around. You throw sickles at the ghosts to destroy them before one of the ghosts touches your man. When you die a full-screen high-resolution picture of a hooded goblin ends the game.

The really interesting thing about these games is the speech synthesizer-less speech. Source code for the speech and music is included. The only other 99/4A game I know about with speech synthesizer-less speech is Perfect Push, a game that originated in Europe several years ago. Goblin and Star Mission are not great games. They don't excite my children much. However, the speech and music technique and source code should be of interest to any assembly programmer, and the asking price of $1 each makes them a bargain. I am sending my $2 (two $1 bills neatly hidden in an envelope) to Italy to encourage the author to write more for us. United States and Canadian residents can obtain these two games on a DSSD disk by sending me $1 which pays for the disk and return postage, but you still owe the author his fairware donation. European users can obtain the games directly from the author by sending a disk with paid return mailer and $2 to Betori Alessandro, Via Erasmo Gattamelata n.112, C.A.P. 00176, Roma, Italia.

PAGE PRO CUT OUT SETS by Virginia Davis

My 8-year-old daughter Meaghan loves reading the "American Girl" books. (Some of you have seen Meaghan in the 1992 Lima videos being taught Logo by Eunice Spooner.) Among the extra stuff you can buy to accompany these books are sets of paper dolls and cutout clothes you strap onto the dolls. Now Virginia Davis of the V.A.S.T. user group has created the same sort of thing for TI users of all ages. Page Pro is required. You load the artwork into Page Pro, print the pictures on a dot matrix printer, color the pictures with crayons, paints or markers, then cut out the pictures. These cutouts are more versatile than the "American Girl" paper dolls I bought for Meaghan. Unlike the pre-colored glossy paper American Girl stuff, you can print out Virginia's artwork again and again, coloring the pieces differently each time. The V.A.S.T. group sent me a sample disk of these cutouts for this review. Meaghan was delighted, and spent a whole evening printing, coloring and folding her new toys. Included were paper dolls, clothes and buildings that you fold into three-dimensional structures. Because she likes the sample disk so much, I will purchase some of these cutout sets for her. My 13-year-old son Colin likes the paper airplanes and space ships that actually fly.

Here is what is available: Jenny, cutout paper doll with 27 outfits and 22 more pieces, $12. Rob, cutout paper doll with 11 outfits, $8. Entire town, 15 buildings and landscaping, $24. Indian village (canoes, scenery, animals, Indians on foot and on horseback), $15. Paper airplanes, eight planes to color assemble and fly, $8. Paper spaceships, 10 ships to color, assemble and fly, $10. These are available on either DSSD or SSSD disks. Also available are three DSSD disks of Virginia's miscellaneous artwork for $5 each. These include large pictures of cute animals, clowns, cute kids and other topics. All prices include media and shipping. A printed 19-page catalog is available for $2. Send your orders for Virginia Davis' artwork to V.A.S.T. User Group, Attn.: Wallace Knight, P.O. Box 25576, Tempe, AZ 85285-5576. Page Pro is available from Ramcharged Computers (phone 800-669-1214) or any other Asgard software dealer.

The PROSTICK II Joystick.

According to my three kids, Bruce Harrison's two kids and my own opinion, this is the best joystick available for the 99/4A. I have gone through several pairs of official TI joysticks. As we all know they are stiff and eventually some of the joystick directions quit responding. I also own an Epyx 500XJ joystick, a model favorably reviewed years ago in MICROpendium. The Epyx is mechanically very sound and doesn't wear out, but it is hard to find the diagonals. You have to position the joystick arm precisely to hit one of the diagonal switches.

The unique feature of the Prostick II, found on no other TI compatible joystick I know about, is the ability to lock out the diagonals and have just left/right up/down available when desired. You can mechanically set the ring at the base of the lever that for four directions or all eight directions. Using the four-direction setting enhances play of maze type games such as A-Maz-Ing, PacMan, Munchman and Munchman 2. It also enhances the many games in which you need only left/right movement with the joystick, such as TI Invaders. Setting the ring to the eight-direction position provides precise diagonals. There is a specific physical notch on the ring that you can see AND FEEL for each of the eight directions. I have tried the Prostick with Y.A.P.P., the 80-column TI drawing program by Alexander Hulpke. For creating pictures the Prostick is not as easy to use as a mouse, but it is easier to use than any other joystick I have used

(See Page 26)

MICRO-REVIEWS —

(Continued from Page 25)

with Y.A.P.P., allowing precise control of diagonal movement.

Other features of the Prostick II include two fire buttons for left and right-handed individuals, a steel short throw shaft with a little ball at the end, leaf switches internally, a "high impact" plastic housing that seems more durable than the housing of my Epyx joystick, and a six-foot cord with strain relief fittings at both ends.

The Prostick II comes with an amazing five-year warranty, a much better warranty than any other new TI compatible joystick. The complete rights to license and manufacture the Prostick II are owned by Ramcharged Computers, an all TI/Geneve dealer. According to Ramcharged owner Ron Markus, in the three or four years since they purchased the rights to and began selling the Prostick II no joysticks have ever been returned to them for warranty service! The Prostick II price has recently been reduced to $14.95 each. A TI joystick adapter is required. If you buy two Prosticks you can probably talk Ramcharged into giving you the adapter for free. Ramcharged Computers, P.O. Box 81532, Cleveland, OH 44181. Phone 800-669-1214.

Send your software for me to MicroReview and your requests for my various $1 software mailings to Charles Good, P.O. Box 647, Venedocia, OH 45894. Phone me evenings at (419) 667-3131.

USER NOTES

Using a RAMdisk in a single floppy system

The following item appeared in TIsHUG News Digest, the newsletter of the Sydney, New South Wales TI user group. As with all hardware projects, readers assume all responsibility for the outcome.

With the advent of RAMdisks there are probably numerous users who want to use one in a system which has only one floppy disk drive. While it is true that a RAMdisk makes a very nice second drive, there are times when a second floppy drive (as well as the RAMdisk) might also seem desirable.

For example, picture the situation where you wish to use TI-Writer out of the RAMdisk, so you use DM1000 to copy EDITA1 and EDITA2 onto the RAMdisk. From BASIC, you then change the RAMdisk to respond as DSK1 — CALL DN(1) — and proceed to run and use TI-Writer, saving your files to the RAMdisk.

So far so good. But what do you do if you want to access another D/V80 file that is on a floppy disk? Remember that the RAMdisk is currently responding as DSK1. The only solution might seem to be to quit TI-Writer, CALL DN(2) from BASIC, reload DM1000, copy the required file from the floppy disk to the RAMdisk, quit back to BASIC, CALL DN(1), reselect TI-Writer, and then load in the required file. Quit a tedious operation.

Fortunately, there appears to be a better solution.

By installing a DPDT (double pole, double throw) toggle switch, one can easily make the floppy respond to either DSK1 or DSK2 So, when using the RAMdisk as DSK1, the switch can be set on the floppy so that it responds as DSK2, and vice versa.

A subminiature toggle switch can be mounted on the front panel of the floppy drive and wired directly to the drive select socket. Simply wire the switch so that the floppy drive can be toggled between either DSK1 and DSK2, and label the switch accordingly.

Buy only the concrete and blocks you need

The following program was written by Gary Moore of the Ozark 99ers User Group of Springfield, Missouri.

The first part of this program will figure the number of concrete blocks needed for a wall of given height and length. The second part calculates how much concrete is needed for a pad of given width, length and depth.

```
100 CALL SCREEN(15):: GOSUB
420 :: GOSUB 410 !172
110 GOSUB 420 !245
120 DISPLAY AT(4,8):"CONCRET
E BLOCK AND YARDAGE" !052
130 DISPLAY AT(8,8):"ESTIMAT
OR" !241
140 DISPLAY AT(10,8):"COPYRI
GHT SEPTEMBER 1993" :: DISPL
AY AT(13,8):"GARY MOORE" ::
DISPLAY AT(16,8):"OZARK 99'E
RS" !190
150 DISPLAY AT(18,1):" PRESS
 ENTER TO START" !044
160 CALL KEY(3,K,S):: IF K<>
13 THEN 160 !213
170 GOSUB 420 :: DISPLAY AT(
8,2):"PICK PROGRAM WANTED" :
: DISPLAY AT(10,2):"(1)-BLOC
KS NEEDED ESTIMATOR" !241
180 DISPLAY AT(12,1):"(2)-CO
NCRETE AMOUNT NEEDED" !043
190 ACCEPT AT(16,2):P :: IF
P<1 OR P>2 THEN 190 !240
200 ON P GOTO 210,300 !181
210 CALL CLEAR :: DISPLAY AT
(4,3):"NUMBER OF BLOCKS NEED
ED" !178
220 DISPLAY AT(6,8):"ENTER H
EIGHT OF WALL(FEET)" :: ACCE
PT AT(8,2):H !072
230 DISPLAY AT(10,2):"ENTER
LENGTH OF WALL(FEET)" :: ACC
EPT AT(8,2):L !123
240 N=H*L*1.125 !234
250 DISPLAY AT(12,2):"NUMBER
 OF BRICKS NEEDED" !142
260 DISPLAY AT(14,2):N !223
270 GOSUB 430 !255
280 GOSUB 420 !245
290 GOTO 370 !194
300 GOSUB 420 :: DISPLAY AT(
2,2):"CONCRETE YARDAGE ESTIM
ATOR" !253
310 DISPLAY AT(4,2):"ENTER V
OLUME LENGTH(FEET)" :: ACCEP
T AT(6,2):L1 !110
320 DISPLAY AT(8,2):"ENTER V
OLUME WIDTH(FEET)" :: ACCEPT
```

(See Page 27)

USER NOTES

(Continued from Page 26)

```
 AT(10,2):W1 !102
330 DISPLAY AT(12,2):"ENTER
VOLUME THICKNESS(IN.)" :: AC
CEPT AT(14,2):T !050
340 Y=L1*W1*T/324 !030
345 DISPLAY AT(16,2):"CONCRE
TE YARDAGE NEEDED" !166
350 DISPLAY AT(18,2):Y !238
360 GOSUB 430 :: GOSUB 420 !
118
370 DISPLAY AT(8,2):"MORE CA
LCULATIONS? (Y/N)" :: ACCEPT
 AT(10,2):Z$ :: IF Z$="Y" OR
```

(See Page 28)

Forth+ for Geneve expands potential

A new software development environment, Forth+, was recently released for with with the Geneve 9640. The program is marketed by BRS Consulting, of Delray Beach, Florida. BRS stands for Bill R. Sullivan.

The company claims that Forth+ will enable programmers to write sophisticated programs for the Geneve, ranging from "Paradox-like databases" to a "Word Perfect-like word processor."

One of the strengths of Forth+, according to BRS Consulting, is that it is able to access up to two megabytes of CPU memory (with Memex and GenMod) for programming. Properly written Forth+ applications would only have "Mainline" and "error trapping" code in standard FORTH dictionary space. The remainder of the program, all code routines and some data would be placed in logical binary code 8K libraries (up to memory limit). Interpretive code and most data would be placed in virtual memory (Forth 1K Block Buffers, up to 896, or 917,504 bytes). Frequently used data would be placed in 8K buffers (up to memory limit).

Forth+ comes with five standard 8K libraries, Utility (loaded with Forth+, but can be exchanged by the user) includes string handling functions that are more powerful than ABASIC or Microsoft QuickBasic string handling functions, four sorting routines and two search routines, an input function similar to ACCEPT AT, and some basic database functions (SQL-like).

ED/ASM includes a Forth source code editor, a Forth style 9900/9995 assembler, some debugging tools, and many Forth style listing and copy utilities. FLOAT includes the Radix 100 floating point and transcendental routines, statistical algorithms, functions for calculating circles/ellipses with true aspect ratios, integer to floating point and back, double number to floating point and back, and many common calculator and conversion routines. GRAPH+ includes easy access to all video modes and video paging, character graphics, hi-res graphics, sprite (mode 1 and 2) characters and all feasible operations, sound, speech, joystick and mouse interface (Bruce Hellstrom's Mouse Driver 3.0); Forth83 interface module. The user can access these four additional libraries or libraries that can be created with simple DECLARE statements.

Forth+ itself contains conditional loaders for binary image and D/F128 files, random number generators, TI file access, time and date functions for display or file stamping, in addition to memory management functions. The Forth kernel (9640 Forth) also contains enhancements, according to BRS.

Forth+ is more extensible than standard Forth because of the modular binary code 8K libraries and 8K buffers, and can therefore be extended and updated on a continuous basis.

A Forth+ HELP and Demo application is included. Even though the on-line HELP is very large and will enable any knowledgeable Forth programmer to create adequate programs that make use of some of the advanced capabilities of Forth+, it will be virtually impossible to create programs that take advantage of all that Forth+ has to offer without the Forth+ manual. Forth+ is being distributed as fairware/shareware, but the 370-page manual is fully copyrighted and will be shipped only to users who register their copy of Forth+ and contribute $40 or more, plus $7 shipping and handling.

The Forth+ files are available on bulletin boards and some commercial telecommunications services. An archived Forth+ system will dearc to 2,372 sectors in 12 files. The FORTH_HLP fileblock, which is the largest of the files at 1,922 sectors, is available as a separate download.

For more information, contact BRS Consulting at 2727 Lowson Boulevard, #A; Delray Beach, FL 33445.

USER NOTES

(Continued from Page 27)

```
 Z$="y" THEN 100 !163
380 IF Z$="N" OR Z$="n" THEN
 GOSUB 420 :: END !180
390 IF Z$<>"Y" OR Z$<>"y" TH
EN 370 !134
400 END !139
410 CALL COLOR(1,5,5):: CALL
 HCHAR(24,1,40,64):: CALL VC
HAR(1,31,40,96):: RETURN !14
9
420 CALL CLEAR :: RETURN !21
9
430 DISPLAY AT(24,2):"PRESS
ENTER TO CONTINUE" !227
440 CALL KEY(3,K,S):: IF K<>
13 THEN 440 !238
450 RETURN !136
```

Format commands in TI-Writer

This comes from David V. Erickson of San Jose, California. He writes:

If you have had trouble with Format Commands in TI-Writer not working, perhaps this hint will help. I usually use the commands one to a line at the beginning of the line, followed by a carriage return. Recently, I inadvertently left some spaces between the end of the command and the carriage return. The result? The commands were inoperative. After some experimentation, it was found that putting the carriage returns immediately after the commands (without spaces) gave the desired results.

Christmas labels

This comes from Jim Lesher, of the Dallas TI User Group. It is another in a series of items that use The Missing Link.

This program, just in time for Christmas, lets you make your own borders on address labels. We design our characters by redefining a character set used by our printer. However, we are limited to a 7x9 matrix, which is the maximum size of each character. In this case, the character is a tiny Christmas tree. Actually, the border consists of many tiny trees, encircling a 3.5- x 15/16-inch label.

You will need to adjust the labels in your printer to make them look right. So, put the labels in with the back side toward you. You can see the labels through the backing. This way you can set up the spacing without ruining a lot of labels. Then, when you get it right, turn the labels back over and output as many as you want.

The labels look best in red, blue or green. This program is set to produce 40 labels, but you can change the number by modifying line 120. Change 40 to whatever number you want.

Here are some numbers to make other characters, just type them in to replace the numbers in line 30.

BELL

```
30 DATA 0,56,68,132,255,132,
68,56,0,0
```

CANE1

```
30 DATA 4,2,1,0,1,0,1,2,124,
0
```

CANE2

```
30 DATA 0,12,2,0,1,0,1,2,124
,0
```

CROSS

```
30 DATA 0,4,0,4,127,4,0,4,0,
0
```

TREE2

```
30 DATA 16,8,4,2,113,2,4,8,1
6,0
```

TREE3

```
30 DATA 16,8,20,2,253,2,20,8
,16,0
```

TREE4

```
10 REM TREE4 !030
20 CALL CLEAR !209
30 DATA 16,8,20,2,113,2,20,8
,16,0 !233
40 OPEN #1:"PIO" !253
50 PRINT #1:CHR$(27);"@";!04
6
60 PRINT #1:CHR$(27);"E";!05
1
70 PRINT #1:CHR$(27);"*";CHR
$(1);CHR$(126);!100
80 FOR I=1 TO 09 !113
90 READ MI !045
100 PRINT #1:CHR$(MI);!213
110 NEXT I !223
120 FOR N=1 TO 40 !113
130 PRINT N !234
140 PRINT #1:CHR$(27);"$";CH
R$(1);!003
150 PRINT #1:CHR$(27);CHR$(5
1);CHR$(12);!189
160 C$=RPT$(CHR$(126),35)!25
5
170 PRINT #1:C$ !175
180 B$=RPT$(CHR$(126),01)!24
7
190 FOR X=1 TO 09 !128
200 PRINT #1:B$;!098
210 PRINT #1:TAB(35);B$ !253
220 NEXT X !238
230 PRINT #1:C$ !175
240 PRINT #1:CHR$(27);CHR$(5
1);CHR$(12)!009
250 NEXT N !228
```

Numerology on the TI

The following item was written by Harold C. Hoyt Jr. It appeared in the St. Louis Computer Bridge, the newsletter of the St. Louis TI user group.

First, after the usual disclaimer that numerology has no scientific basis, and programs like this are for entertainment only, consider that some people, including Pythagoras, believed that our destiny was entwined with numbers.

Here is a chart from a local newspaper showing how numerology assigns a single digit number encoded from the letters in your name. Apparently, the digits 1-9 generated by your name give information about your destiny, which gets us into the old philosophical debate about whether you have free will. I prefer to believe in free will.

```
1 2 3 4 5 6 7 8 9
A B C D E F G H I
J K L M N O P Q R
S T U V W X Y Z
```

In the table, each letter is assigned a number value from one through nine. You add the numbers corresponding to the letters in your name to get a total.

For example: my name, Harold Coley Hoyt, would result in 8+1+9+6+3+4 plus 3+6+3+5+7 plus 8+6+7+2, for a total of 78. We need a single digit, so we add the seven and the eight to get 15. Still not a single digit, so add the digits one and five to get six, my personal number. Each personal number has a corresponding characteristic in a look-up table. If you are unhappy with your number, spell your name

(See Page 29)

USER NOTES

(Continued from Page 28)

differently and change your destiny!

This is just too much fun to pass up, so I have written and Extended BASIC program to do this for you. With a little more effort, I could have pulled a Regena and written it in TI BASIC, but the extra routines available in Extended BASIC are just too helpful to pass up.

Line 1 is the usual built-in file name reminder and handy file saver when typing the file in. Periodically, type 1, FCTN-X, Enter, FCTN-1, FCTN-1, Enter to have the program saved as "DSK1.NUMER."

Line 110 sets up the screen for name input. Lines 115, 117 and 119 are built-in program tests. Edit out the exclamation mark in one of these lines, and the text on that line will be placed on the screen. The ACCEPT AT statement in line 120 has a negative size, so whatever is on the screen is input as N$. The newspaper article had two examples calculating personal numbers: William Jefferson Clinton has a personal number of three while Bill Clinton has a five. The result is a completely different prediction just by using an abbreviated name.

Line 120 ACCEPTS N$ and 125 sets the DISPLAY row, R=10, clears the SUM and nulls N1$ when the program loops.

Line 130 takes each character in the name stored in N$ and fetches a corresponding number for the letter into N from the look-up table embedded in the POSition statement. The look-up table contains the letters A through Z in both upper and lowercase. The POS function returns 0 to 26 to N by using the INT with round up by adding .five to INT()/2 to group letters two at a time.

Line 150 does Modulo 10 arithmetic on N, effectively partitioning the look-up table into three rows. Line 150 replaces the number zero in the string N1$ with a space (any character typed in that isn't a letter), and +"N" for any letter.

Each loop adds "N" to "SUM" in line 160. SUM$=STR$(SUM) allows the digits in SUM$ to be fetched and added together in line 170. Line 170 also clears a scratchpad area on the screen of 180 characters and then DISPLAYs a string of all of the numbers that are added together for the SUM. This is repeated as necessary, incrementing one row each time, until the SUM is a single digit.

In line 190, SUM>9 repeats the loop.

Finally, line 200 DISPLAYs the personal number and line 500 asks if you want to do another number.

Isn't it nice that numerology shows that there are only nine kinds of people in the world?

```
1 !SAVE DSK1.NUMER !154
100 !Prog By H. Hoyt Jr. 10-
14-93 !006
110 CALL CLEAR :: DISPLAY AT
(2,9):"-NUMEROLOGY-":" FIND
YOUR PERSONAL NUMBER":"
      by":"    Harold C. Ho
yt Jr.":"         10-14-93"
 !178
115 !DISPLAY AT(8,1):"Harold
 Coley Hoyt" !236
117 !DISPLAY AT(8,1):"Willia
m Jefferson Clinton" !058
119 !DISPLAY AT(8,1):"Bill
          Clinton" !204
120 DISPLAY AT(7,2):"TYPE IN
 YOUR NAME & <ENTER>" :: ACC
EPT AT(8,1)SIZE(-30):N$ !234
125 R=10 :: SUM=0 :: N1$=""
!019
130 FOR I=1 TO LEN(N$):: N=I
NT(POS("AaBbCcDdEeFfGgHhIiJj
KkLlMmNnOoPpQqRrSsTtUuVvWwXx
YyZz",SEG$(N$,I,1),1)/2+.5)!
104
```

(See Page 30)

USER NOTES

(Continued from Page 29)

```
140 IF N>9 THEN N=N-9 :: GOT
O 140 !191
150 IF N=0 THEN N1$=N1$&" "
ELSE N1$=N1$&"+"&STR$(N)!091
160 SUM=SUM+N :: SUM$=STR$(S
UM)!077
170 NEXT I :: DISPLAY AT(R,1
):RPT$(" ",180):: DISPLAY AT
(R,2):N1$&"="&STR$(SUM):: R=
R+5 !143
180 SUM$=STR$(SUM):: FOR I=1
 TO LEN(SUM$):: N2=N2+VAL(SE
G$(SUM$,I,1)):: NEXT I :: R=
R+1 :: DISPLAY AT(R,2):"Numb
er=";SUM$;" Digit Sum=";STR$
(N2)!154
190 SUM=N2 :: N2=0 :: IF SUM
>9 THEN 180 !214
200 DISPLAY AT(20,2):"Your P
ersonal Number=";STR$(SUM)!0
52
500 DISPLAY AT(22,2):"Anothe
r? Y/N?" :: CALL KEY(0,K,S):
: IF S=0 THEN 500 :: IF K=AS
C("Y")OR K=ASC("y")THEN 100
!007
```

Tips on faster file handling in XBASIC

The following item was written by John Wineke and has appeared in several user group newsletters.

It is often convenient to write a BASIC or Extended BASIC program to handle specific filing tasks. For instance, I have a mail list program to handle our church membership and newsletter names and addresses. When you have to read in a large number of files from BASIC, it gets to be a rather long process. By experimentation I have found several ways to speed things up a bit, and I offer them for your perusal.

The program I wrote has 92 names, with six fields per name for such things as address, phone number, and so forth. I could have used an existing database program but preferred the freedom of having up to 80 characters per field and not having to set any predefined limits on myself. This way, if an address was particularly long, I would have no problem. And, if a field is blank, no memory is wasted — other than that required to DIMension for a string.

Tip No. 1: Use Internal, Variable 80 files rather than Display/Variable 80. This allows you to input numerical data as well as string. And it is faster. I cut file handling time 23 percent by doing this. I know that the manual leads you to believe otherwise, but seeing is believing.

Tip No. 2: Put all the input and output program loops in one line. By cutting three program lines to one using multistatement lines my files loaded five percent faster. (See listing below.)

Tip No. 3: Cut out extraneous program lines, especially bells and whistles. By removing a DISPLAY AT statement that showed each name as it was brought off the disk I saved nine seconds (15 percent) when inputting my 552 records.

Tip No. 4: Use a RAMdisk or hard disk. Actually, they don't save as much when using BASIC as you might suppose, because much of the time is BASIC overhead, but it did save me 12 seconds (20 percent).

Tip No. 5: Use a BASIC compiler. Now this idea is really impractical, but I'm throwing it out to show the effect it would have if it were practical. It is impractical because when you DIMension your program (mine uses 100,7) the compiler assumes you are going to use the whole DIMension up with whatever you set your string limit at. My Ryte Data (Peter Kull) compiler required you to set a maximum string limit and allows no more than 64 characters. When you do that, you have no room left for your program.

Anyway, I cut down the size of my DIMension and compiled the program and used proportioning to come up with the time saved. I figure that the whole file would have input in 30 seconds, on a RAMdisk. By not using the above techniques, the file took 78 seconds.

Tip No. 6: This is not a speed hint, but rather a good piece of advice. Use error trapping! If you don't, you are likely to lose your entire file somewhere along the way. Now that would really be a waste of time when you consider all the work it took to type it in.

I used the error trap shown in lines 110 and 1720 to 1760 below. You need a complex trap like this with I/O because you usually get two errors instead of one.

```
100 DIM A$(100,7)!107
110 ON ERROR 1720 :: GOSUB 1
110 !243
115 INPUT "DATE?":DATE$ !178
120 INPUT "1=NEW FILE 2=OLD
FILE ":OP1 :: IF OP1>2 THEN
120 ELSE IF OP1=1 THEN 200 !
170
140 INPUT "INPUT FILE NAME?
":Y$ :: OPEN #3:Y$,INPUT ,IN
TERNAL,VARIABLE 80 !029
150 CALL CLEAR :: INPUT #3:F
YL,A,B$ !139
160 FOR I=1 TO A :: INPUT #3
:C$(I),T(I):: NEXT I !124
170 FOR I=1 TO FYL :: FOR J=
1 TO A !043
180 INPUT #3:A$(I,J)!088
190 NEXT J :: DISPLAY AT(24,
1):A$(I,1):: NEXT I :: CLOSE
 #3 :: GOTO 460 !213
195 REM (AND SO ON . . . .)
!149
1720 ON ERROR 1730 !209
1730 CALL ERR(D,S,J,K):: PRI
NT :"ERROR";D;"IN LINE";K;::
 FOR I=1 TO 3 :: CALL SOUND(
900,220,0):: NEXT I !089
1740 IF D=130 AND CL=1 THEN
CL=0 :: RETURN 470 ELSE IF D
=130 THEN CL=1 :: CLOSE #3 :
: RETURN 470 !169
1750 IF D=109 AND CL=1 THEN
OPEN #3:"PIO" :: CLOSE #3 ::
 CL=0 !127
1760 RETURN 470 !040
```

The preceding routines are for demonstration purposes only. They do not constitute a finished program.—Ed.

Console 'surgery' ends lock-ups

This following item has appeared in several user group newsletters. Because the information involves opening the console case, all liability lies with the reader.

Here's a tip on curing console lock-up for good, if not longer.

Dirty contacts are, of course, the culprit. Just about every cleans the external con-

(See Page 31)

USER NOTES CLASSIFIEDS

(Continued from Page 30)

tacts, but the problem persists you'll need to do more "surgery."

The real problem lies inside the console with the cartridge L-connector. What you need to do is:

- Open up the console case
- Undo the two screws from the power supply
- Undo the three screws that hold the motherboard in place
- Disconnect the power supply and the keyboard connections and remove the motherboard.

Now, remove the cartridge connector. Looking at the male part of the connector, you may notice some indentations and black corrosion on the soldered area of the contacts. Take a piece of nylon scrub pad and buff the contacts on both sides until the indentations are gone and the contacts are smooth. Also, do this to the board edge connectors for the I/O port. It is a good idea the female part of the cartridge connector with contact cleaner while it is out.

Now, reassemble everything in reverse order. Be patient and work carefully. You should find that hardware lock-ups and erratic console behavior will be a thing of the past.

MICROpendium pays $10 for items submitted by readers that appear in this column. Send your routines, ideas and tips to MICROpendium User Notes, P.O. Box 1343, Round Rock, TX 78680.

Policy

The cost of classified advertising is 25 cents per word. Classified display (i.e., special formatting or graphics) is $9 per column inch. Classified advertisements must be paid in advance. Classified advertisers may request a category under which they would like their advertisements to appear, but the final placement decision is the responsibility of the publisher.

Classified deadlines will be kept open for as long as practical. For the purpose of classified advertising deadlines, any classified ad received later than the first day of any month cannot be assured of placement in the next edition. We will do our best to include every advertisement that is submitted in the earliest possible edition.

The publisher offers no guarantee that any advertisement will be published in any particular issue. Any damages that result either from errors in copy or for failure to be included in any particular edition will be limited to the amount of the cost of the advertisement itself. The publisher reserves the right to reject any advertisement.

The advertiser may elect to publish the advertisement in subsequent editions at the same charge, payable prior to publication. The deadline for carryover classifieds is the same as for new advertising.

In submitting an ad, please indicate whether you would like a refund if it is not published in the requested edition or whether you would like us to hold it for the next edition. Cancellations and refunds cannot be made after the second day of the month.

Send classified advertising to: MICROpendium, P.O. Box 1343, Round Rock, TX 78680.

FOR SALE

TI99/4A

PHP1100	TI, 99/4A DUAL JOYSTICKS	14.95
PHP1260	32K MEMORY CARD	99.95
PHP1500	SPEECH SYNTHESIZER	29.95
PHM3109	TI LOGO II (32K Mem. Req.)	1.95
PHA4100	TI VIDEO MONITOR (Used)	99.95
PHM3058	MINI-MEM (W/Assembler)	39.95
UM1381-1	RF MODULATOR	19.95
4530	JOYSTICK "Y" ADAPTER	7.95
AC9500	TI99/4A POWER SUPPLY	9.95

JOY ELECTRONICS INC.—P.O. BOX 542526 DALLAS, TX 75354-2526
(800) 527-7438, OUTSIDE DALLAS AREA
(214) 243-5371, DALLAS AREA
ADD $5.00 S&H—TEX RES ADD 8 1/4% TAX
10/12 **CALL OR WRITE FOR FREE CATALOG**

SEVERAL TI SYSTEMS & parts for sale. For price list, write Jim Ferguson, 22 Welland Crescent, St. Albert, Alberta, Canada T8N 3W4. 1/2

FOR SALE

ENORMOUS TI99/4A INVENTORY. CATALOGS $2.00. **BRAATZS COMPUTER SERVICES**, 719 E. BYRD ST., APPLETON, WI 54911. **1-414-731-3478.** 10/12

WANTED

AMIGA MONITOR WANTED
Looking for an Amiga monitor for use with the Geneve. Call 800-777-6632.

Use MICROpendium classifieds to buy and sell your unused equipment

Buy/Sell Used Hardware and Software

National Used Software/Hardware Club has buyers looking for TI products, as well as sellers. More than 170 TI items are listed for sale. The membership fee is only $15/year and entitles you to buy or sell computer items via the NUS/HC database. You will also receive FREE our newsletter (mailed every 8-12 weeks). If you don't buy or sell anything during the term of your membership we will refund your annual fee. For a free, no-obligation infor-mation packet, write to NUS/HC, P.O. Box 1343, Round Rock, TX 78680; or call 512-255-1512. **Here are some of the items offered for sale:**

Horizon RAMdisk (192K)	$125.00
Advanced Diagnostics	$25.00
Seagate 251 HD (40 meg)	$75.00
MG Explorer	$25.00

Call our BBS for periodic price and product updates—512-255-1557. Hours are from 5:30 p.m. to 8 a.m. weekdays and from noon Saturday through 8 a.m. Monday weekends. Supports 300-2400 baud.

The ONLY monthly devoted to the TI99/4A

Subscription Fees

❑ 12 issues, USA, **$35** ❑ 12 issues, Mexico, **$40.25**

❑ 12 issues, Canada **$42.50** ❑ 12 issues, other countries surface mail, **$40.00**

❑ 12 issues, other countries, air mail, **$52.00**

Outside U.S., pay via postal or international money order or credit card; personal checks from non-U.S. banks will be returned.

Address Changes

Subscribers who move may have the delivery of their most recent issue(s) delayed unless MICROpendium is notified six weeks in advance of address changes. Please include your old address as it appears on your mailing label when making an address change.

Check each item ordered (or list on separate page) and enter total amount here:_____

Check/MO ❑  ❑  ❑ (check one)

Card No. _____

Expiration Date _____

(Minimum credit card order is $9)

Signature _____

(Required on credit card orders.)

No sales tax on magazine subscriptions. Texas residents add 7.75% sales tax on other items, including back issues and disk subscriptions.

Mail to: MICROpendium, P.O. Box 1343, Round Rock, TX 78680

Name _____

Address _____

City _____

State _____ **ZIP** _____

The set of numbers at the top of your mailing label indicates the cover date of the last issue of your subscription.

Disks, Etc.

❑ **Back Issues**, $3.50 each. List issues: _____

_____

_____

No price breaks on sets of back issues. Free shipping USA. Add 30 cents, single issues to Canada/Mexico. Other foreign shipping 50 cents single issue surface, $1.50 airmail. Write for foreign shipping on multiple copies.

OUT OF STOCK: Vols. 1, No. 1-2; Vol. 2, No. 1; Vol. 4, No. 9

❑ **MICROpendium Index** (2 SSSD disks, 1984-1992), Extended BASIC required $6.00

❑ **MICROpendium Index II** (9 SSSD disks — 1 for each year — 1984-1992), XB required $30.00

❑ **MICROpendium Index II with MICROdex 99** (11 SSSD disks), XB required.....$35.00

❑ **MICROdex 99** (for use with MP Index II, 2 SSSD disks), XB required $10.00

MICROdex 99, by Bill Gaskill, is a collection of programs that allow users of MP Index II to modify their index entries, as well as add entries. MICROdex 99 supports many other functions, including file merging, deletion of purged records, record counting and file browsing.

GENEVE DISKS (SSSD unless specified)

❑ **MDOS 1.23F** (req. SSDD or larger, for MBASIC).....$4.00

❑ **MDOS 1.50H** (req. for MBASIC).....$4.00

❑ **GPL 1.5** $4.00

❑ **Myarc Disk Manager 1.50**.....$4.00

❑ **Myarc BASIC 3.0**.....$4.00

❑ **MY-Word V1.21** $4.00

❑ **Menu 80** (specify floppy or hard disk versions(s); includes SETCOLR, SHOWCOLOR, FIND, XUTILS, REMIND $4.00

GENEVE PUBLIC DOMAIN DISKS

These disks consists of public domain programs available from bulletin boards. If ordering DSDD, specify whether Myarc or CorComp.

	SSSD	DSSD	DSDD
❑ **Series 1**	$9.00	$7.00	$5.00
❑ **Series 2**	$9.00	$7.00	$5.00
❑ **Series 3**	$9.00	$7.00	$5.00
❑ **Series 4**	$9.00	$7.00	$5.00
❑ **Series 5**	$9.00	$7.00	$5.00

SECOND CLASS