

MICROpendium

Volume 10 Number 6

July 1993

\$2.50

Programming

Regena on
BASIC —
Telling Time

The Art of
Assembly —
*Comparing TI
assembly and
PC assembly*

INSTAROT

*An XBASIC program that rotates TI-Artist
Instances*

Buying floppy drives

*Selecting the correct drive can save money
and time*

A compiler for XBASIC?

Bruce Harrison starts work on a compiler

User Notes

*Cat(a)log & Run, a program to keep track of
your mileage and more routines for TML*

Reviews

RICHGXKB

Boot ROM Bios

Funnelweb V5 40-Column Editor

Collecting Cartridges

TEX+COMP DOLLAR DAYS

WATCH YOUR SAVINGS GROW DURING DOLLAR DAYS

SOFTWARE SUPER SALE!

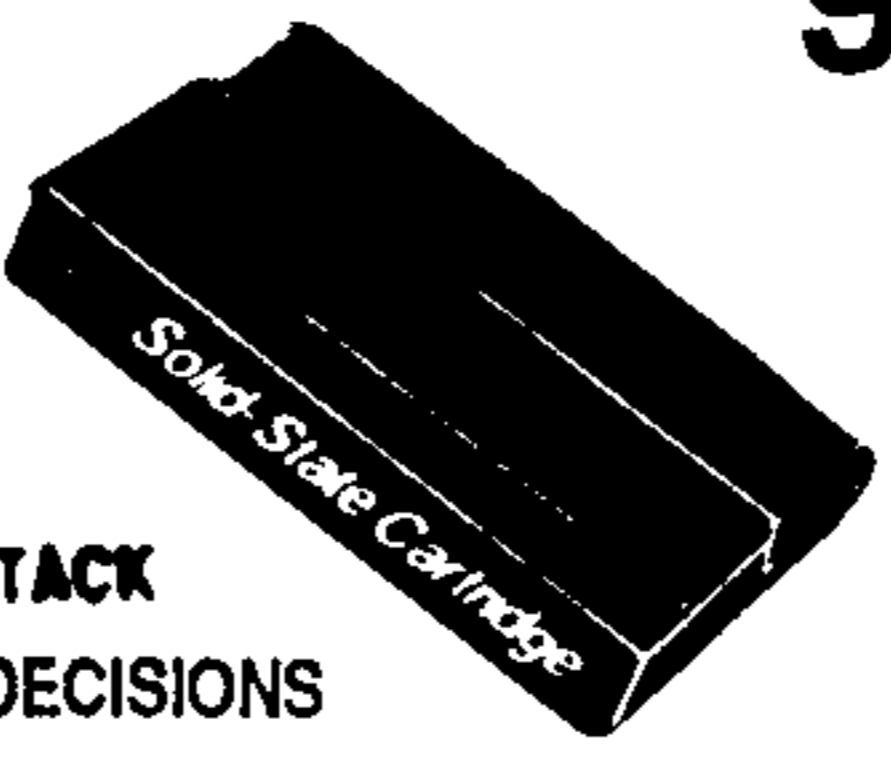
AS LOW AS \$1.95 EACH!

WHEN BUYING TEN OR MORE OR BUY ANY AMOUNT AT ONLY \$2.95 EACH.

99¢ BOOK BLOWOUT!!!

PUBLISHER'S CLEARANCE ON BOOKS FOR THE 99/4A BUY FIVE OR MORE BOOKS FOR ONLY 99 CENTS EACH + s&h. BUY ANY AMOUNT LESS THAN 5 FOR ONLY \$1.49 EACH + s&h.

- | | |
|--------------------------------------|---|
| PHM 3027 ADDITION & SUBTRACTION I | PHM 3018 TAX INVESTMENT RECORD KEEPING*** |
| PHM 3028 ADDITION & SUBTRACTION II | PHM 3022 PERSONAL REAL ESTATE |
| PHM 3093 MILLIKEN DIVISION | PHM 3029 HUNT THE WUMPUS |
| PHM 3091 NUMERATION II | PHM 3031 THE ATTACK |
| PHM 3094 MILLIKEN INTEGERS | PHM 3220 MICROSURGEON |
| PHM 3110 MINUS MISSION | PHM 3087 MUNCHMAN |
| PHM 3029 MULTIPLICATION I | PHM 3110 CHISHOLM TRAIL |
| PHM 3002 EARLY LEARNING FUN | PHM 3218 SUPER DEMON ATTACK |
| PHM 3003 BEGINNING GRAMMAR | PHM 3006 HOME FINANCIAL DECISIONS |
| PHM 3112 PARSEC | PHM 3035 TERMINAL EMULATOR II |
| PHM 3007 HOUSEHOLD BUDGET MANAGEMENT | PHM 3052 TOMBSTONE CITY |
| PHM 3044 PERS. REPORT GENERATOR** | PHM 3109 LOGO II* → → Plus \$4 additional shipping & handling |
| PHM 3089 TI INVADERS | PHM 3088 EDITOR ASSEMBLER |
| PHM 3194 JAWBREAKERS II | PHM 3010 PHYSICAL FITNESS |
| | PHM 3148 MUNCHMOBILE |



Special purchase

DONKEY KONG featuring Mario \$4.95 (regular \$9.95)

ALSO INCLUDED ARE THE FOLLOWING ARANISOFT MODULES: DEFENDER, PROTECTOR II, PICNIC PARANOIA

*ADD AN ADDITIONAL \$4.00 EACH ON THESE THREE TITLES DUE TO WEIGHT ** FOR USE WITH PERSONAL RECORD KEEPING, *** DISK DRIVE REQUIRED PERSONAL RECORD KEEPING AVAILABLE AT \$14.00

GAMES TIS PLAY THE ELEMENTARY TI COMPUTER PLAYGROUND (TI) INTRO TO ASSEMBLY LANGUAGE PROGRAMS FOR THE TI COMPUTER TI-99/4A GAME PROGRAMS STARTER PACK I STARTER PACK II GAME WRITERS PACK I GAME WRITERS PACK II

SPECIAL PACKAGE AND PRICING ON THE ADVENTURE SERIES...

ON DISK OR CASSETTE (specify) THE COMPLETE SCOTT ADAMS SERIES I (ADVENTURELAND, PIRATE ISLAND, THE COUNT MISSION IMPOSSIBLE, VOODOO CASTLE, STRANGE ODYSSEY, MYSTERY FUN HOUSE, PYRAMID OF DOOM, GHOST TOWN, SAVAGE ISLAND I, SAVAGE ISLAND II, AND GOLDEN VOYAGE PLUS KNIGHT HEART BONUS ADVENTURE

ONLY \$12.95 FOR ALL 13.

THE SCOTT ADAMS ADVENTURE SERIES II (THE HULK, SPIDERMAN, BUCKAROO BANZAI, AND THE SORCERER OF CLAYMORQUE CASTLE PLUS 2 BONUS ADVENTURES

ONLY \$9.95 FOR ALL 6

COMMAND MODULE FOR ABOVE ADVENTURES (reg \$6.95) ONLY 95 CENTS WITH EITHER OF THE ABOVE SERIES

HINT BOOK FOR ALL OF THE ABOVE SCOTT ADAMS ADVENTURES ONLY 95 CENTS WITH EITHER OF THE ABOVE SERIES

NEW First time on disk — Return to Pirate's Island with Graphics, XBASIC or Editor/Assembler Load \$4.95

ADVENTURE EDITOR NOW WRITE AND EDIT YOUR OWN GAMES FOR THE TI ADVENTURE MODULE!



\$4.95

TEX COMP is proud to announce the 99/4A Adventure Editor... an all new program which allows you to write your own adventure programs... 1. Any existing adventure for the Adventure command module... 2. New adventure games can be created freely using a template game as a start up step... TEX COMP is also organizing a new TI Adventuring Users Group... Available NOW From TEX-COMP Specify Mini Memory (Cassette) or Editor Assembler (Disk) Versions



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

HOME ENTERTAINMENT

- DISKETTE PROGRAMS
- | | |
|---|------|
| PND 3002 TI-TREE (TE-11 req. for speech) | 4.95 |
| PND 3010 Mystery Melody | 4.95 |
| PND 3015 Oldies But Goodies I | 4.95 |
| PND 3017 Oldies But Goodies II | 4.95 |
| ***SPECIAL Oldies But Goodies I & II | 7.95 |
| PND 3025 Sat. Night Bingo (Ex-Basic & Speech) | 4.95 |
| PND 3037 Draw Poker (Ex-Basic) | 4.95 |

- CASSETTE PROGRAMS
- | | |
|---|------|
| PNT 6002 TI-TREE (TE-11 req. for speech) | 4.95 |
| PNT 6010 Mystery Melody | 4.95 |
| PNT 6015 Oldies But Goodies I | 4.95 |
| PNT 6017 Oldies But Goodies II | 4.95 |
| ***SPECIAL Oldies But Goodies I & II | 7.95 |
| PNT 6025 Sat. Night Bingo (Ex-Basic & Speech) | 4.95 |
| PNT 6037 Draw Poker (Ex-Basic) | 4.95 |

COMPUTER PROGRAMMING AIDS

- DISKETTE PROGRAMS
- | | |
|--|-------|
| PND 5007 Teach Yourself 99/4A Basic | 4.95 |
| PND 5019 Teach Yourself Extended Basic | 4.95 |
| PND 5004 Programming Aids I | 4.95 |
| PND 5005 Programming Aids II | 4.95 |
| PND 5077 Programming Aids I, II, III | 9.95 |
| PND 5076 Beginning Basic Tutor | 4.95 |
| PND 5076 Text to Speech (Ex-Basic Speech) | 4.95 |
| PND 5098 TI Forth & Manual (Ed/Assem req.) | 19.95 |
| PND 5078 TI Forth Demo Disk (Ed/Assem) | 4.95 |
| PND 5079 TI Forth Source Code (2 disks) | 4.95 |

- CASSETTE PROGRAMS
- | | |
|--|------|
| PNT 6006 Programming Aids I | 4.95 |
| PNT 6007 Teach Yourself 99/4A Basic | 4.95 |
| PNT 6019 Teach Yourself Extended Basic | 4.95 |
| PNT 6067 Beginning Basic Tutor | 4.95 |

MATH AND ENGINEERING

- specify disk or tape with order
- | | |
|--------------------------------|------|
| Math Routine Library | 4.95 |
| Electrical Engineering Library | 4.95 |
| Graphing Package | 4.95 |
| Structural Engineering Library | 4.95 |
| AC Circuit Analysis | 4.95 |
- ***SPECIAL--ALL 5 OF THE ABOVE ON DISK OR TAPE--17.95

\$4.95 Per Disk **NEW BONUS SAVE AS MUCH AS 33%**
BUY FIVE DISKS GET ONE FREE
BUY TEN DISKS GET THREE FREE
BUY FIFTEEN DISKS GET FIVE FREE

- DISKETTE PROGRAMS
- | | |
|---|------|
| PND 5009 Music Skills Trainer | 4.95 |
| PND 5011 Computer Music Box | 4.95 |
| PND 5018 Market Simulation | 4.95 |
| PND 5030 Speak & Spell II (Ex-Basic req.) | 4.95 |
| PND 5031 Speak & Math (TE-11 req.) | 4.95 |
| PND 5042 Spell Writer (TE-11 req.) | 4.95 |
| PND 5026 Bridge Bidding I | 4.95 |
| PND 5039 Bridge Bidding II | 4.95 |
| PND 5041 Bridge Bidding III | 4.95 |
| SPECIAL!! BRIDGE BIDDING I, II & III | 9.95 |
| PND 5070 Music Maker Demo (use with module) | 4.95 |
| PND 6010 Mystery Melody (stop the music quiz) | 4.95 |

- CASSETTE PROGRAMS*
*see disk versions for requirements like TE-11
- | | |
|---|------|
| PNT 6009 Music Skills Trainer | 4.95 |
| PNT 6010 Mystery Melody | 4.95 |
| PNT 6011 Computer Music Box | 4.95 |
| PNT 6018 Market Simulation | 4.95 |
| PNT 6031 Speak & Math | 4.95 |
| PNT 6042 Spell Writer | 4.95 |
| PNT 6026 Bridge Bidding I | 4.95 |
| PNT 6039 Bridge Bidding II | 4.95 |
| PNT 6041 Bridge Bidding III | 4.95 |
| SPECIAL!! BRIDGE BIDDING I, II & III | 9.95 |
| PNT 6070 Music Maker Demo (use with module) | 4.95 |

MANAGEMENT AND BUSINESS

- DISKETTE PROGRAMS
- | | |
|--|------|
| PND 5001 Mailing List (upgraded version) | 4.95 |
| PND 5003 Personal Financial Aids | 4.95 |
| PND 5021 Checkbook Manager | 4.95 |
| PND 5022 Finance Manager | 4.95 |
| PND 5024 Inventory Management | 4.95 |
| PND 5027 Invoice Management | 4.95 |
| PND 5029 Cash Management | 4.95 |
| PND 5038 Lease/Purchase Decisions | 4.95 |
| PND 5075 TI/Multiplan upgrade disk | 4.95 |

- CASSETTE PROGRAMS
- | | |
|-----------------------------------|------|
| PNT 6003 Personal Financial Aids | 4.95 |
| PNT 6038 Lease/Purchase Decisions | 4.95 |

TEX+COMP

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



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



Send \$2 for our new 1993 catalog



TEX+COMP DOLLAR DAYS

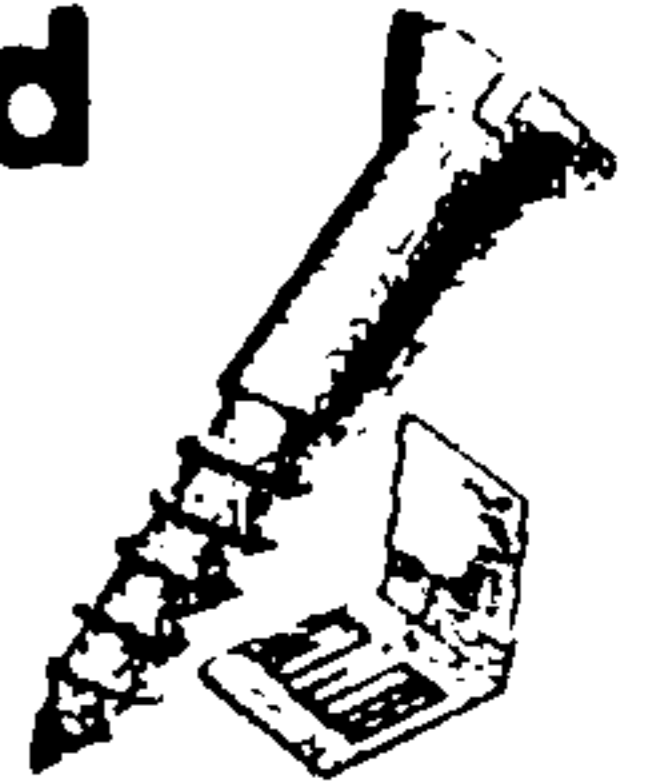
WATCH YOUR SAVINGS GROW DURING DOLLAR DAYS

OUR LOWEST PRICES EVER ON THIS ENTIRE PAGE!

Big Price Reduction for Our Full-Featured COMPLETE PRINT SHOP PACKAGE

FOR THE TI-99/4A special price \$4.95*

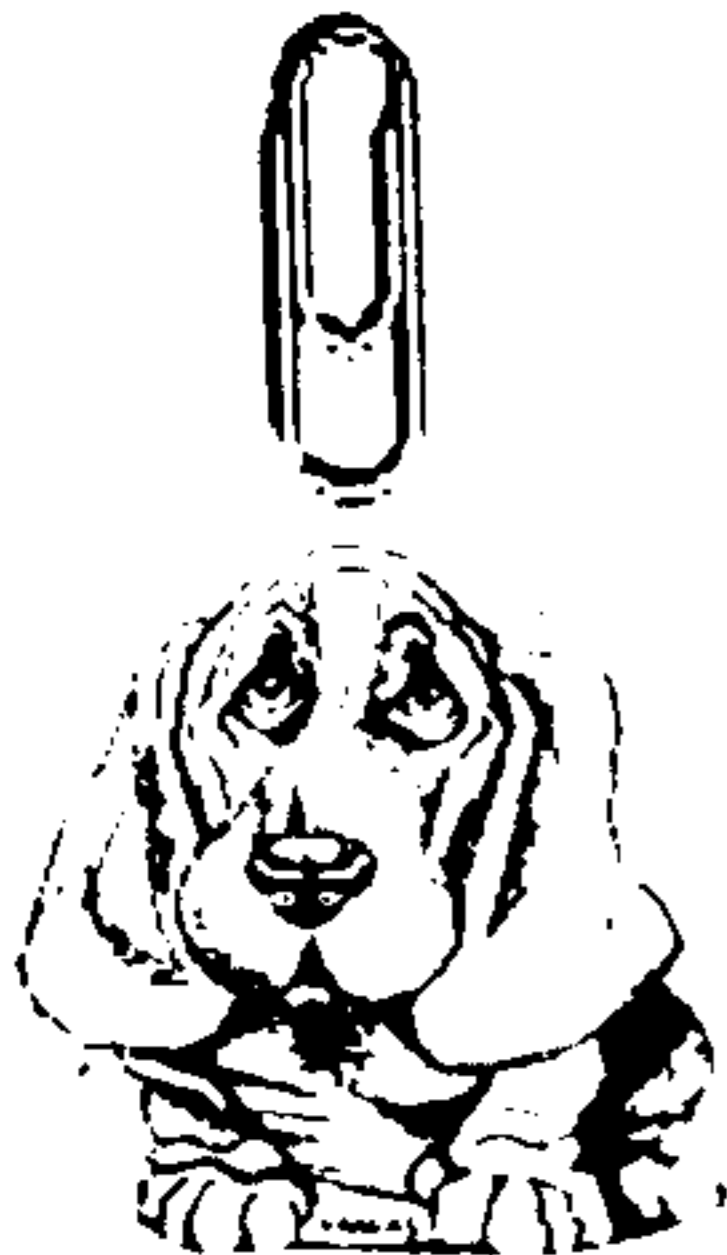
featuring GRAPHX+ with Flip & Rotate & Quick Load (REG \$19.95)



NEW FONTS INCREDIBLE

THE ALL TIME BEST GRAPHICS PROGRAM FOR THE 99/4A HAS JUST GOT EVEN BETTER WITH AN ALL NEW PACKAGE AT A GREAT LOW PRICE. ONLY \$19.95 COMPLETE WITH GRAPHX+ AND TWO DISK SIDES FILLED WITH NEW CLIPART AND FONTS. GRAPHX+ OFFERS EVERY FEATURE YOU NEED TO CREATE THE BEST 4A GRAPHICS YOU HAVE EVER SEEN. GRAPHX+ OFFERS FEATURES SUCH AS FREE HAND DRAWING AND ERASING, ZOOM FOR DETAILED WORK, AUTO FILL, COLORS, TEXT AND TITLES, AUTO CIRCLE DRAWING, CLIP STORAGE AND ANIMATION. 32K, EXBASIC, AND A STAR/EPSON COMPATIBLE DOT MATRIX PRINTER IS REQUIRED.

*BUY ANY 4 ACCESSORY PACKAGES AT THE NEW LOWER PRICE OF \$4.95 AND BUY THE COMPLETE GRAPHX+ PACKAGE FOR ONLY \$1.00.



GRAPHX DINOSAURS!

Asgard Software



\$4.95

DISK OF DINOSAURS is a fantastic collection of real and whimsical dinosaurs from that favorite 99/4A artist, Jim Gilliland. This latest utility for Graphx also contains a Bonus Disk "Land of the Dinosaurs" for creating great scenes along with all the other great clip art.

Graphx with precision resolution for superior TI Artist along with a nearly instantaneous "Auto" built-in movie-copy function, multi-speed cursor, and always available clipboard is, in the words of Chris Roberts of Asgard Software, "the best drawing tool available for the 99/4A".

Now you can "draw your own" anything from calendars to greeting cards to projects. Create dinosaur stickers, logos, and pictures faster than you can say "Brontosaurus". Can also be used with TI Artist with the conversion routine supplied. GRAPHX and a 32K disk drive system required.



MAC-FLICK

HIGH RESOLUTION CLIP ART FOR GRAPHX AND TI ARTIST

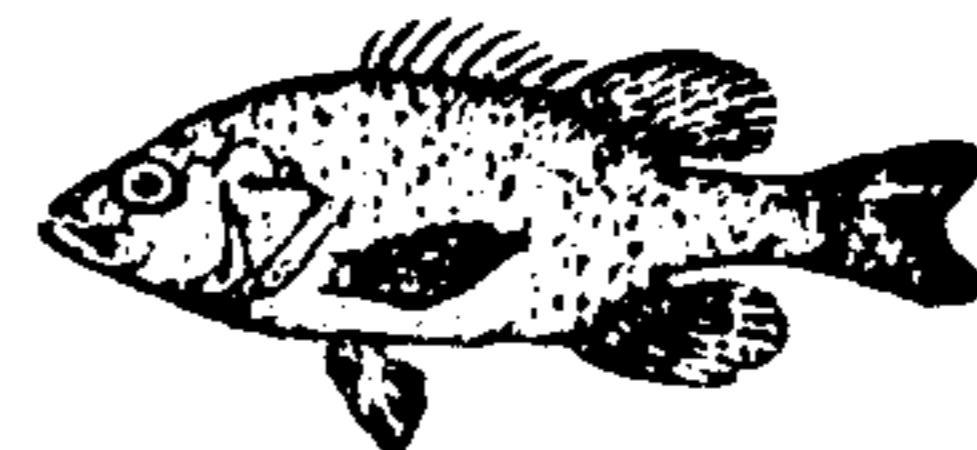


SIX DISK SIDES

TEX-COMP HAS TAKEN THE BEST MAC CLIPART AVAILABLE AND CONVERTED IT TO GRAPHX CLIPART FILES. NOW YOU CAN PRODUCE GRAPHICS WITH YOUR 4A THAT RIVALS THE BEST ANYWHERE. WE FILLED SIX DISK SIDES WITH OVER 140 PIECES OF CLIPART THAT ARE READY TO BE PRINTED OUT WITH GRAPHX OR CAN EVEN BE CONVERTED TO OTHER GRAPHIC SYSTEMS. THE SUPERIOR PRINTING CAPABILITY OF GRAPHX WILL CREATE ART THAT WILL BE OUTSTANDING!!

\$4.95

SIX DISK SIDES GRAPHX, 32K, AND STAR/EPSON DOT MATRIX PRINTER REQ.



ASGARD SOFTWARE HAS PUBLISHED FOUR SETS OF PROFESSIONAL CLIPART AND FONTS FOR USE WITH THE GRAPHX+ PUBLISHING SYSTEM SO YOU CAN CREATE FANTASTIC ARTWORK WITHOUT EVEN HAVING ARTISTIC TALENT. EACH SET IS ONLY \$4.95 (VOL I-IV)

GRAPHX SLIDESHOW BY ASGARD IS A CLIPART COLLECTION AND A SPECIAL PROGRAM FOR DISPLAYING YOUR GRAPHX CREATIONS IN AUTOMATIC SEQUENCE. GREAT FOR DISPLAYS, TRADESHOWS AND PRESENTATIONS.....ONLY \$4.95

- GRAPHX+ SYSTEM (reg. 19.95) special sale price.....\$4.95*
- ASGARD GRAPHX COMPANION I.....\$4.95
- ASGARD GRAPHX COMPANION II.....\$4.95
- ASGARD GRAPHX COMPANION III.....\$4.95
- ASGARD GRAPHX COMPANION IV.....\$4.95
- ASGARD GRAPHX SLIDESHOW.....\$4.95
- PRINTING TO GO (Clipart for GRAPHX reg 9.95).....\$4.95
- GRAPHX DINOSAURS (Dinosaur clipart reg. 12.95).....\$4.95
- MAC-FLICK (Hi-res. clipart for GRAPHX reg. 12.95).....\$4.95

*BUY ANY 4 ACCESSORY PACKAGES AT THE NEW LOWER PRICE OF \$4.95 AND BUY THE COMPLETE GRAPHX+ PACKAGE (REG \$19.95) FOR ONLY \$1.00.

TEX+COMP

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

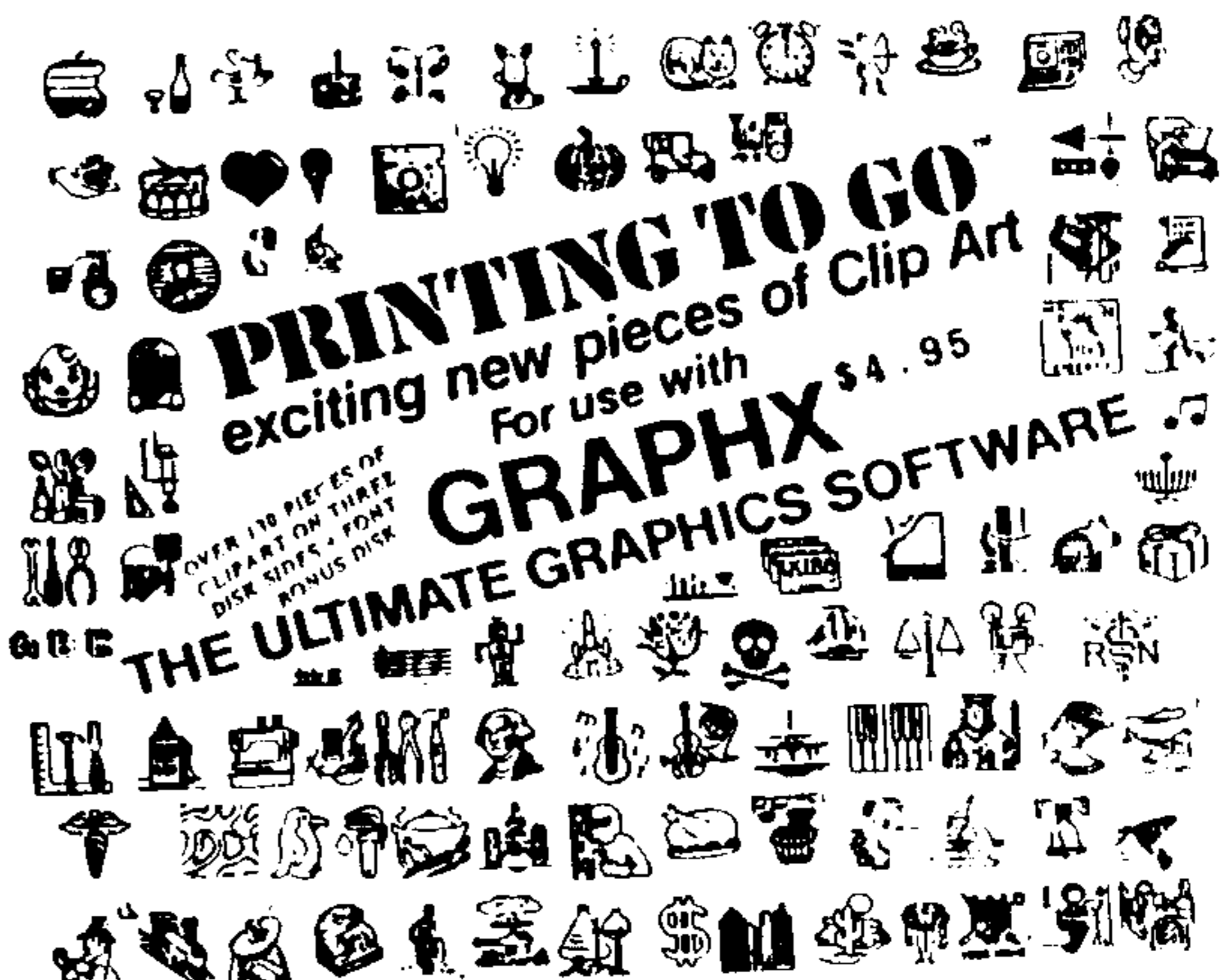


VISA and MASTERCARD HOLDERS CALL DIRECT:

(818) 366-6631



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



PRINTING TO GO
exciting new pieces of Clip Art
For use with GRAPHX \$4.95

THE ULTIMATE GRAPHICS SOFTWARE

PRINTING TO GO IS A GREAT COLLECTION OF CLIP ART AND FONTS TO ADD TO YOUR GRAPHX PUBLISHING SYSTEM FOR THE TI-99/4A. THIS GREAT PACKAGE OF FOUR DISK SIDES AT A FANTASTIC PRICE IS ADDED TO YOUR GRAPHX+ SYSTEM. GRAPHX IS THE BEST BUY IN 4A GRAPHIC PACKAGES AND BY FAR, THE BEST SUPPORTED. THIS COLLECTION CAN BE USED FOR CREATING NEWSLETTERS, ANNOUNCEMENTS OR ANY OTHER APPLICATION WHERE YOU NEED PROFESSIONAL LOOKING ART. THESE CLIPS CAN ALSO BE USED FOR CREATING PERSONALIZED STATIONARY OR EVEN FOR MAKING T-SHIRTS WITH THE TEX-COMP T-SHIRT KIT. OVER 130 PIECES OF QUALITY CLIPART & FONTS.

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 Mastercard. 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.

Send \$2 for our new 1993 catalog

CONTENTS

MICROpendium

MICROpendium (ISSN 10432299) is published monthly for \$25 per year by Burns-Koloen Communications Inc., 16506 Terrace Dr., Austin, TX 78728-1156. Second-class postage paid at Austin, Texas, and additional mailing offices. POSTMASTER: Send address changes to MICROpendium, P.O. Box 1343, Round Rock, TX 78680-1343.

No information published in the pages of MICROpendium may be used without permission of the publisher, Burns-Koloen Communications Inc. Only computer user groups that have exchange agreements with MICROpendium may excerpt articles appearing in MICROpendium without prior approval.

While all efforts are directed at providing factual and true information in published articles, the publisher cannot accept responsibility for errors that appear in advertising or text appearing in MICROpendium. The inclusion of brand names in text does not constitute an endorsement of any product by the publisher. Statements published by MICROpendium which reflect erroneously on individuals, products or companies will be corrected upon contacting the publisher.

Unless the author specifies, letters will be treated as unconditionally assigned for publication, copyright purposes and use in any other publication or brochure and are subject to MICROpendium's unrestricted right to edit and comment.

Display advertising deadlines and rates are available upon request.

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 TINET: MICROpendium

GENIE: J.Koloen

Internet E-mail: jkoloen@holonet.net

John Koloen.....Publisher

Laura Burns.....Editor

Regena on BASIC

Telling time.....Page 8

The Art of Assembly

Comparing TI to PC assembly.....Page 12

INSTAROT

An Extended BASIC program that rotates TI-Artist instances onto the left or right side.....Page 16

Buying floppy disk drives

Tips on how to do it.....Page 19

Newsbytes

Fair sites, address changes, and your help is wanted in compiling a TI BBS listing.....Page 20

Reviews

RICH GKXB.....Page 21

Boot ROM Bios.....Page 23

MICRO-Reviews: Funnelweb V5 40-Column Editor, Collecting Cartridges.....Page 24

User Notes

Catalog and Run, keeping track of mileage, more circle routines in TML and a warning on jumpering.....Page 26

Classified.....Page 31

Departments

Bugs and Bytes.....Page 20 Feedback.....Page 6

Comments.....Page 5 Reader to Reader.....Page 5

Fairs.....Page 18 System of the Month.....Page 27

*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

Harrison working on Extended BASIC compiler

Bruce Harrison, our assembly language columnist, has embarked on a noble and ambitious task — a compiler for Extended BASIC.

"I've begun work on a truly monstrous project, to make a compiler for Extended BASIC, so that XB users will be able to speed up their programs' execution. This won't be finished overnight, as it's still in the 'research and planning' stage. I'm examining in some detail how the XB interpreter handles its variables, with an eye to being able to 'share' variables between the compiled and the 'faked' parts of the resulting machine code program. I've learned some truly amazing things in this pursuit so far, and will be passing along some of these in future columns, as well as passing the info along to such XB experts as Harry Wilhelm, Jim Peterson and Barry Traver."

I've always felt that the absence of a BASIC compiler has been a great shortcoming of the TI system. There have been attempts to develop a BASIC compiler, but I know of none that has worked well. For those who are unfamiliar with compilers, a program is written in a high level programming language, such as BASIC, and then run through a compiler. The compiler converts the BASIC code into machine code. The resulting machine code program runs very fast because, unlike interpreted languages such as Extended BASIC, the computer doesn't have to interpret each line of code as it is executed.

PRICE INCREASE TAKES EFFECT AUG. 1

As of Aug. 1, the annual subscription price of MICROpendium will increase by \$10/year. The price in the U.S. will be \$35. All other subscription classes will increase by \$10 over the current rates. Readers who are unable to pay the extra cost are asked to notify us when renewing so that we can do what we can to accommodate them.

CHARLES GOOD WRITES MICRO-REVIEWS

MICROpendium has a new columnist, Charles Good. Charles will be writing the Micro Reviews column, which was formerly written by the late Stan Krajewski.

Charles is familiar to many TIers as the force behind the popular Lima TI fair. He's a college professor and father of three. Welcome aboard, Charles.

LOST USER?

Don Walden of Cecure Electronics is trying to find out the whereabouts of Jim Fetzner. Fetzner sent Myarc equipment in for repair and it has been ready for some time, but Walden has received no reply from the APO address where he sent the notification. He asks anyone knowing Fetzner's current address to contact him at P.O. Box 132, Muskego, WI 53150, or (414) 679-4343.

—JK

Reader to Reader

□ Tom Weaver, 325 Johnson St., Amery, WI 54401 writes:

I recently bought a printer at an electronic flea market. It doesn't work, so I'm looking for info, manuals, schematics, etc. The identification on the front is "NLQ 180"; looking through the inside the only other info was "Made in Japan."

□ Olden Warren, 4016 Weber Way, Lexington, KY 40514, (606) 223-4599 (home), (606) 288-2231 (work), writes:

I have read several articles on installing the Extended BASIC cartridge inside the console. I have a Triton Super Extended BASIC cartridge that I am trying to install inside my console, but I am having trouble "turning it off" when I want to use the cartridge port. Has anyone done this, and, if so, could they let me know how it is done?

□ Bruce Harrison of Harrison Software sent this reply to Ian Howle, who inquired in the December, 1992 Reader to Reader concerning using bit map mode in assembly:

Sometimes the book is just plain wrong. In the source code you showed, the setting of the color table would overlap the pattern table, because the color table gets located at the value written to register 3 times >40. Thus to place the color table at >2000 as recommended, the value for RO in your case would be >0380, thus placing the color table at >40 * >80, or >2000.

Attached is my own short test source code, which doesn't actually do anything except make a pattern on the screen, then waits for a key press and returns to E/A. This includes a loop to write the characters from 0 through >FF three times, by incrementing R1, swapping bytes, and using VSBW. At least this gives you a

controlled, non-crashing starting point. Perhaps given this starting point, you can actually *do* something with Bit-Map mode. I haven't ever tried anything serious with it myself, but couldn't ignore your cry for help.

Another potential source of help might be Harry Wilhelm, 254 Luce Rd., Groton, NY 13073. Harry is the author of The Missing Link, and knows much more than I about Bit-Map operations.

```
* BIT MAP MODE TEST
* FOR IAN HOWLE
* 27 DEC 1992
* B. HARRISON
  DEF  START
  REF  VSBW, VMBW, VWTR, KSCAN, GPLLNK
CHAR  DATA >8040, >2010, >0804, >0201
COLOR DATA >6363, >6F6F, >4343, >4D4D
ANYKEY DATA >2020          value for key detection
*
```

```
START
  LWPI >20BA          load user workspace
  LI  R0, >0002       access bit-map
  BLWP @VWTR
  LI  R0, >0206       change screen table to >1800
  BLWP @VWTR
  LI  R0, >0380       different value - book wrong
  BLWP @VWTR
  LI  R0, >0400       pattern table at 0
  BLWP @VWTR
  LI  R0, >0701       different value
  BLWP @VWTR
  CLR R0             start of character table
```

(See Page 7)

FEEDBACK

TI more fun

After trying an IBM system for just over a year, I came back to TI. You know that the IBM was nice, great graphic and good programs such as Windows and GEO, *but* there was no fun! I could *not* find any programs that even came close to the one I like the most — Page Pro 99.

What surprised me was, it was harder to get information, or at least for me it was. Now my 286 IBM system gets used very little. Even the kids hardly turn it on. They would rather use Page Pro or Graphx and TI-Artist Plus than the shelf full of IBM programs. If you ask them why, they will say, "It's easier to remember!" So with that I do agree. It's easier to remember. I wrote several programs for the TI; several have become popular with the TI community. Cassette fINDEX, Graph Jacket and the LABEL Machine are three. I dusted these off and found the joy I had lost. I'm now a TI'er again.

I will be upgrading my programs. I still like them more than anything else. I have a TI99/4A, 1 meg HRD, PGRAM+, 32K, Rave speech card and CorComp controller (with MG EPROM). I use SXB. My wish list has two things on it. One, a program like TIPS, but much easier to use, and two, a disk manager that time/date stamps programs and files like the Geneve. We have all these clock cards, etc., and make little use of them. Even if you had to type in the data, it would still help. Example: I have six disks of GRAPHJACK simply because I can't remember which one is the latest.

Well, thank you for the fine magazine. One idea that I thought you might try is free advertising every three months for the real small software or hardware companies, etc. That would help us all know what's still available, etc.

**Joseph E. Bartle
Parish, New York**

Welcome back. We publish announcements of new software and hardware in Newsbytes for small as well as large companies, and often review products for these companies, but offering free advertisements every three months would be unfair to the advertisers whose money keeps this magazine going. — Ed.

Don't quit

I think you're doing a terrific job for the TI99/4A community!

I'm sure we all would like to see *only* the subjects and items we are personally interested in, but let's face it, no publication can do that.

When MICROpendium arrives, I sit right down and read it from cover to cover, including the ads and the subjects that are not in my specific field of interest. Who knows, maybe something will start a spark of interest.

As someone said, inflation is everywhere, so do what you have to do — just don't quit. Where else would we get all this good information?

**Edna Holliday
Winston, Oregon**

MICROpendium will continue as long as there are readers who want to read it.—Ed.

Help sought for drive

For the past year I have been desperately trying to get my Tandon hard drive correctly formatted and functional on my Geneve computer.

I have properly hooked up the drive to my Myarc HFDC and the drive itself seems to be fully operational (the motor runs and the LED lights up.). But with much aggravation, I have been unsuccessful in formatting the drive with the MDM5 (v1.25) I have purchased and am using the latest version of MDOS (v1.21H) for fixed-disk, as well as the latest GPL Interpreter, from which I load the MDM5. I fear the problem lies in the HFDC card itself, but am uncertain.

When I first installed the drive I removed the Floppy Disk Controller and replaced it with the HFDC and then hooked both the hard drive and my two TEAC floppy drives as instructed. But when I turned on the system, placing the DOS in floppy drive A, my hard drive lit up once and my floppy drives did not respond. After trying this several times, making sure I hooked up all drives correctly to the card, I decided to improvise, thinking that maybe the card might not work with the floppies but might still work

with the hard drive. So I reinstalled the floppy disk controller in a different slot, gave it a new address and then hooked up my two floppy drives to it. (This is the setup I currently have.) When I start up my computer system, the computer first goes to the hard card which in turn goes to the hard drive. Finding nothing, like it should, it then goes to the floppy card where it accesses the floppy drive and the system boots up the DOS. Once I get the DOS, I load up the GPL and then the MDM5 from that. But when I try to format the hard drive from the Disk Manager, selecting all the appropriate choices on the menus, something goes wrong. The system either freezes or it just goes back to the menu. I don't know why, but figure it can be one of the following:

- Special address is needed for the HFDC.

- I need to create a new batch file (autoexec) or augment the one I have, which is:

```
TIMODE  
Ramdisk 90  
Assign C=Dsk 5  
Spool 16
```

- Or, in fact, there is no solution and the HFDC cannot be salvaged for any purpose.

- Or I am naming the drive wrong when I try to format it. I call it WSDS1 but am not sure of it.

I am in dire need and not knowing what to do I was hoping you or anyone else may have the answer. I hope this did not inconvenience you or your prestigious periodical. I greatly admire your magazine for having the guts to stick by the TI and the Geneve and want you to know that I will continue to purchase it as long as it is in production. Also on one last note, I apologize if I'm not "up to date" or the problem is totally due to my ignorance as a Geneve user (I'm 15 years old) and thank you for your time.

**Harris J. Hall
Monsey, New York**

It's never inconvenient to try to answer questions from readers. Unfortunately, I don't have a solution for you, just some more questions.

First, what kind of Tandon hard drive are you using? It must be a MFM drive, (See Page 7)

FEEDBACK

(Continued from Page 6)

not an RLL or IDE or SCSI or EISA. The documentation will tell you what it is.

Assuming it is an MFM drive, double-check the cabling and make sure that the CRU address of the HFDC is the address recommended in the HFDC manual. Also double check another other cards you have to make sure none is using the same address.

Since you hare using both a floppy controller card and an HFDC card, make sure the floppy card uses the slot closest to the internal disk drive and put the HFDC in the next slot over.

Although I don't think the software is your problem, the latest versions of MDOS for hard drives is 1.50H. The most recent version of MDM is 1.5 and the most recent version of GPL is 1.5. These are available on various bulletin board services, including GENIE and Delphi. Or, you can order them from MICROpendium (see the back page for more information).

Finally, there is always the possibility that your HFDC or hard drive is defective. If the above suggestions don't work for you, you might want to contact Cecure Electronics (414-679-4343 voice, 414-529-9659 BBS) for help. Cecure is the only authorized Myarc service center.

Thanks from Danville

The Danville 99(ers) would like to thank the (so far) 29 people who wrote to us in response to our letter in MICROpendium. Although most of the letters were just to say "Thanks," or "I feel the same way," there were also those who wanted an application for this group, and those who wanted information about this group.

All letters were answered within hours of their arrival here. The first batch contained 14 letters so they were (that afternoon) split up between me, Misty and Billy to answer.

Three were followed up with packages containing a fair amount (but by no means all) of information about the Danville group and its "alliance" members. A sample disk was sent to a couple of them. The disk was "Jitterbugs," which was Amanda's very first graphics program. She was

about to turn 11 at that time.

I must mention that a couple of the letters were a little strange. They did not mention the subject of the MICROpendium letter or request information about our group.

So from all of us here, again we say thank you for writing and stay in contact. You are, each and every one, important.

**Gene Downs (and the members of)
The Danville 99(ers)**

888 Airport Rd.
Danville, KY 40422

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

(Continued from Page 5)

LI R1,CHAR	character
LI R2,8	eight bytes
BLWP @VMBW	write
LI R0,>2000	color table
BLWP @VMBW	
LI R0,>1800	screen image table
LI R2,256*3	three times 256
CLR R1	start with r1=0
PTLOOP BLWP @VSBW	write a byte to screen table
SWPB R1	start with r1=0
INC R1	add one
SWPB R1	swap back
INC R0	point one ahead
DEC R2	dec counter
JNE PTLOOP	if not zero, repeat
SCAN BLWP @KSCAN	scan keyboard
CB @ANYKEY,@>837C	key struck?
JNE SCAN	if not, scan again
CLR R0	clear so r0=0000
BLWP @VWTR	get out of bit map mode
LI R0,>01E0	set for normal mode
BLWP @VWTR	normal
LI R0,>0200	screen origin zero
BLWP @VWTR	
LI R0,>030E	color table normal
BLWP @VWTR	
LI R0,>0401	pattern table at >800
BLWP @VWTR	
LI R0,>40*>0E	point at color table
LI R1,>4E00	colors
LI R2,32	32 bytes
OLDCOL BLWP @VSBW	write one
INC R0	inc location
DEC R2	dec count
JNE OLDCOL	not zero, repeat
LI R0,32*8*>800	point at space character
MOV R0,@>834A	
CLR @>837C	
BLWP @GPLLNK	use gpl
DATA >0018	to restore character set
LWPI >83E0	
B @>6A	

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 76860. We encourage those who answer the questions to forward us a copy of the reply to share with readers.

BASIC

Telling time

By REGENA

Our modern world usually has digital clocks and watches, but there are still a few of the old-fashioned clock faces around. First graders still spend time learning to tell time with the standard clock with an hour hand and a minute hand. The program this month presents a quiz of telling time.

The first option shows a round clock face with a random placement of the hands. The student must type in the numbers for the digital clock — the hour and then the minutes. The minutes are all multiples of 5 (00, 05, 10, 15, 20 ...) up to 55.

The second option shows the time on a digital clock in the upper left corner, and the student must set the hands on the clock. First the hour hand is set. Press the right arrow key (on D) until the arrow points to the correct hour, then press Enter to set the hour hand. Next press the right arrow key until the arrow points to the correct position for the minute hand, and press Enter. The hands will be drawn.

If incorrect answers are given, the correct answers are shown. Press the Enter key after each problem. The quiz continues until five correct responses have been given, then the program returns to the main menu screen.

I originally wrote this program on an IBM-compatible computer with the DRAW command. As the minutes became 25 or more, the hour hand would move appropriately. However, with the TI, I couldn't define more graphics characters to draw the hour hand between numbers. I explained this flaw to my seven-year-old son, and he seemed to think it was okay to have the hour hand stay on the number.

Line 120 DIMensions arrays for the 12 possible positions of the hour and minute hands. Any variable starting with H pertains to hours, and any variable starting with M pertains to minutes. HR and HC are relative row and column positions, and HG is the graphic character for the arrow at the end of the hour hand. MR and MC are relative row and column positions, and MG is the graphic character for the arrow at the

end of the minute hand.

Lines 140 and 150 set the ROW and COLUMN numbers for the center of the clock. The whole clock may be moved by changing these lines. Lines 170-210 use DATA to define HR, HC, HG, MR, MC and MG for each of the 12 positions. Lines 230-380 use DATA statements to define the graphic characters. If you have any problems with this program, these lines are the most likely trouble spots. The graphic characters are used to draw the hands of the clock.

Lines 390-500 print the main menu screen, wait for a response and branch appropriately.

Lines 510-670 are a subroutine to draw the digital clock and place the numbers around the round clock.

Lines 680-710 are a subroutine to print a message on the screen without scrolling — used to print the time in the digital clock or the correct time. Lines 720-790 print the "Press ENTER" message at the end of the problem, wait for the Enter key to be pressed, then clear the screen.

Lines 800-1400 are the subroutines to print the 12 possible hour hands. The first character of each subroutine is the arrow, then the rest of the subroutine draws the hour hand. Lines 1410-2010 are the subroutines to print the 12 possible minute hands.

Lines 2020-2570 are the subroutine for the first option, where hands are drawn on the clock and the student must designate the time. The FOR-NEXT loop with the index T gives five problems. Lines 2040-2050 print the screen. Lines 2060-2110 randomly choose an hour H and a minute M, numbers from 1 to 12. Lines 2090-2110 are used so the same hour and minute are not chosen twice in a row.

Lines 2120-2130 prepare the digital clock. Line 2140 draws the minute hand, and Lines 2150-2170 calculate the minutes. Line 2180 draws the hour hand. Lines 2190-2340 receive the student's answer. If the hour is 10, 11 or 12, two digits must be pressed, otherwise only one digit is allowed

for the hour. CY is used as a column number for the blinking cursor. Lines 2580-2660 are the subroutine to receive the answer, which must be a number for each column.

Line 2350 checks the answer, and Lines 2360-2470 print the correct time if the answer is incorrect. Line 2480 branches to the beginning of a new problem, but the T index is not incremented unless the answer is correct. Lines 2490-2540 are the sequence if the answer is correct. After the five correct times have been completed, Lines 2550-2570 return the colors back to normal and return to the main menu screen.

Lines 2670-3390 are the subroutine for the second option, where a time is given and the student must set the hands. Lines 2690-2700 print the screen. Lines 2710-2760 randomly choose a time, making sure the time isn't the same as the previous time. Lines 2770-2880 print the time in the digital clock.

Lines 1890-3010 are the procedure to set the hour hand. An hour blinks, starting with position 1. Each time the right arrow key (on D) is pressed, the arrow moves to the next position. At 12, the position goes back to 1. If the Enter key is pressed, the rest of the hour hand is drawn.

Lines 3020-3140 similarly draw the minute hand, and after the minute hand is selected, the hour hand is redrawn.

Lines 3150 checks the answer. If the answer is incorrect, Lines 3160-3290 put red stars on the incorrect hands and draw the correct hands (which may be drawn over the incorrect ones). Lines 3300-3350 are the procedure for a correct answer.

After five correct times, Lines 3360-3390 reset the colors to normal and return to the main menu screen. Line 3400 ends the program.

If you wish to save typing effort, you may have a copy of this program by sending \$4 to REGENA, 918 Cedd Knolls West, Cedar City, UT 84720. Be sure to specify that you need "CLOCK" for the TI and whether you want cassette or diskette.

REGENA ON BASIC—

```

100 REM CLOCK !070
110 REM BY REGENA !071
120 DIM HR(12),HC(12),HG(12)
,MR(12),MC(12),MG(12)!028
130 CALL CLEAR !209
140 ROW=11 !226
150 COL=16 !205
160 PRINT TAB(8);"** CLOCK *
*" !224
170 FOR J=1 TO 12 !108
180 READ HR(J),HC(J),HG(J),M
R(J),MC(J),MG(J)!215
190 NEXT J !224
200 DATA -4,2,106,-6,3,143,-
2,4,114,-3,6,147,0,4,98,0,7,
139,2,4,122,3,6,151,4,2,130,
6,3,155 !004
210 DATA 4,0,102,7,0,137,4,-
2,110,6,-3,144,2,-4,118,3,-6
,148,0,-4,99,0,-7,140 !110
220 DATA -2,-4,126,-3,-6,152
,-4,-2,134,-6,-3,156,-4,0,10
1,-7,0,136 !106
230 PRINT : : : : " ... DEFI
NING CHARACTERS ..." !159
240 FOR J=96 TO 156 !229
250 READ J$ !005
260 CALL CHAR(J,J$)!095
270 NEXT J !224
280 DATA FFFFFFFFFFFFFFFF,00
0000FFFF,00303CFFFF3C3,000C3
CFFFF3C0C,1818181818181818,1
8183C3C7E7E1818 !195
290 DATA 18187E7E3C3C1818,18
1830306060C0C,0101030306060C
0C,808,000003070F1F0F0D,0303
06060C0C1818 !059
300 DATA 30306060C0C0808,000
0000000000101,B0F0F8F0E0C,00
0000030F3CF0C,0F3CF0C,000000
0000000003 !144
310 DATA FC7CF8F02,030F3CF0C
,00000000030F3CF,C,000000040
F1F3E3F,C0F03C0F03,00000000C
0F03C0F !001
320 DATA 03,00000020F0F87CFC
,000000C0F03C0F03,F03C0F03,0
0000000000000C,3F3E1F0F04,C0
C0606030301818 !248
330 DATA 0C0C060603030101,00
000000000808,0D0F1F0F0703,1
8180C0C06060303,8080C0C06060
303,0101 !239
340 REM 134 !114
350 DATA 0000C0E0F0F8F0B,080
8080808080808,082C3E08080808
08,08080808083E1C08,000000FF
,000406FF0604 !142
360 DATA 002060FF602,1010202
04040808,0101020204040808,01
03070304040808,10102020C0E0C
08,00000000030C30C !173
370 DATA 030C30C,0F0E34C,000
00000032C70F,C0300C03,000000
00C0300C03,00000000C0340E0F,
F0702C03 !114
380 DATA 808040402020101,080
8040402020101,08080404030703
01,80C0E0C02020101 !085
390 CALL CLEAR !209
400 PRINT TAB(8);"CLOCK" !24
2
410 PRINT : : : "CHOOSE:" !13
2
420 PRINT : "1 CLOCK WITH HAN
DS—YOU WILL SEE A CLOCK WI
TH THE HANDS POINTING TO TH
E TIME." !155
430 PRINT " WHAT TIME IS IT
?" !240
440 PRINT : : "2 SET HANDS—Y
OU WILL SEE A DIGITAL CLOC
K. SET THE HANDS ON THE
ROUND CLOCK." !108
450 PRINT : : "3 END PROGRAM"
: : : !091
460 CALL KEY(3,K,S)!190
470 IF (K<49)+(K>51)THEN 460
!225
480 CALL CLEAR !209
490 ON K-48 GOSUB 2030,2680,
3400 !242
500 GOTO 390 !214
510 REM DRAW CLOCKS !231
520 CALL COLOR(3,16,5)!231
530 CALL COLOR(4,16,5)!232
540 CALL SCREEN(8)!153
550 CALL CHAR(59,"")!166
560 CALL HCHAR(1,3,59,7)!132
570 CALL HCHAR(2,3,59,7)!133
580 CALL HCHAR(3,3,59,7)!134
590 RESTORE 640 !223
600 FOR J=1 TO 16 !112
610 READ X,Y,G !245
620 CALL HCHAR(ROW+X,COL+Y,G
)!243
630 NEXT J !224
640 DATA -7,4,49,-4,7,50,0,8
,51,4,7,52,7,4,53,8,0,54,7,-
4,55,4,-7,56,0,-8,57 !078
650 DATA -4,-8,49,-4,-7,48,-
7,-5,49,-7,-4,49,-8,-1,49,-8
,0,50,-8,3,32 !015
660 CALL HCHAR(ROW,COL,96)!1
78
670 RETURN !136
680 FOR J=1 TO LEN(M$)!242
690 CALL HCHAR(MX,MY+J,ASC(S
EG$(M$,J,1)))!162
700 NEXT J !224
710 RETURN !136
720 M$="PRESS ENTER" !044
730 MX=24 !147
740 MY=15 !148
750 GOSUB 680 !250
760 CALL KEY(3,K,S)!190
770 IF K<>13 THEN 760 !239
780 CALL CLEAR !209
790 RETURN !136
800 REM HOUR HAND !083
810 CALL HCHAR(ROW-4,COL+2,1
06)!086
820 CALL HCHAR(ROW-3,COL+2,1
03)!082
830 CALL HCHAR(ROW-2,COL+2,1
05)!083
840 CALL HCHAR(ROW-2,COL+1,1
04)!081
850 CALL HCHAR(ROW-1,COL+1,1
03)!079
860 RETURN !136
870 CALL VCHAR(ROW-2,COL+4,1
14)!099
880 CALL VCHAR(ROW-2,COL+3,1
11)!095
890 CALL VCHAR(ROW-2,COL+2,1
13)!096
900 CALL VCHAR(ROW-1,COL+2,1
12)!094
910 CALL VCHAR(ROW-1,COL+1,1
11)!092
920 RETURN !136
930 CALL HCHAR(ROW,COL+4,98)
!114
940 CALL HCHAR(ROW,COL+1,97,
3)!029
950 RETURN !136
960 CALL HCHAR(ROW+2,COL+4,1
22)!083
970 CALL HCHAR(ROW+2,COL+3,1
19)!088
980 CALL HCHAR(ROW+2,COL+2,1
21)!080
990 CALL HCHAR(ROW+1,COL+2,1

```

(See Page 10)

REGENA ON BASIC—

(Continued from Page 9)

```

20) !078
1000 CALL HCHAR(ROW+1, COL+1,
119) !085
1010 RETURN !136
1020 CALL VCHAR(ROW+4, COL+2,
130) !096
1030 CALL VCHAR(ROW+3, COL+2,
127) !101
1040 CALL VCHAR(ROW+2, COL+2,
129) !102
1050 CALL VCHAR(ROW+2, COL+1,
128) !100
1060 CALL VCHAR(ROW+1, COL+1,
127) !098
1070 RETURN !136
1080 CALL VCHAR(ROW+4, COL, 10
2) !163
1090 CALL VCHAR(ROW+1, COL, 10
0, 3) !077
1100 RETURN !136
1110 CALL HCHAR(ROW+4, COL-2,
110) !081
1120 CALL HCHAR(ROW+3, COL-2,
107) !086
1130 CALL HCHAR(ROW+2, COL-2,
109) !087
1140 CALL HCHAR(ROW+2, COL-1,
108) !085
1150 CALL HCHAR(ROW+1, COL-1,
107) !083
1160 RETURN !136
1170 CALL VCHAR(ROW+2, COL-4,
118) !103
1180 CALL VCHAR(ROW+2, COL-3,
115) !099
1190 CALL VCHAR(ROW+2, COL-2,
117) !100
1200 CALL VCHAR(ROW+1, COL-2,
116) !098
1210 CALL VCHAR(ROW+1, COL-1,
115) !096
1220 RETURN !136
1230 CALL HCHAR(ROW, COL-4, 99
) !116
1240 CALL HCHAR(ROW, COL-3, 97
, 3) !032
1250 RETURN !136
1260 CALL HCHAR(ROW-2, COL-4,
126) !089
1270 CALL HCHAR(ROW-2, COL-3,
123) !085
1280 CALL HCHAR(ROW-2, COL-2,
125) !086
1290 CALL HCHAR(ROW-1, COL-2,
124) !084
1300 CALL HCHAR(ROW-1, COL-1,
123) !082
1310 RETURN !136
1320 CALL VCHAR(ROW-4, COL-2,
134) !102
1330 CALL VCHAR(ROW-3, COL-2,
131) !098
1340 CALL VCHAR(ROW-2, COL-2,
133) !099
1350 CALL VCHAR(ROW-2, COL-1,
132) !097
1360 CALL VCHAR(ROW-1, COL-1,
131) !095
1370 RETURN !136
1380 CALL VCHAR(ROW-4, COL, 10
1) !163
1390 CALL VCHAR(ROW-3, COL, 10
0, 3) !080
1400 RETURN !136
1410 REM MINUTE HAND !231
1420 FOR J=1 TO 3 !059
1430 CALL HCHAR(ROW-J*2+1, CO
L+J, 141) !106
1440 CALL HCHAR(ROW-J*2, COL+
J, 142) !176
1450 NEXT J !224
1460 CALL HCHAR(ROW-6, COL+3,
143) !090
1470 RETURN !136
1480 FOR J=1 TO 3 !059
1490 CALL VCHAR(ROW-J, COL+J*
2-1, 145) !125
1500 CALL VCHAR(ROW-J, COL+J*
2, 146) !194
1510 NEXT J !224
1520 CALL VCHAR(ROW-3, COL+6,
147) !108
1530 RETURN !136
1540 CALL HCHAR(ROW, COL+1, 13
8, 6) !077
1550 CALL HCHAR(ROW, COL+7, 13
9) !162
1560 RETURN !136
1570 FOR J=1 TO 3 !059
1580 CALL HCHAR(ROW+J, COL+J*
2-1, 149) !114
1590 CALL HCHAR(ROW+J, COL+J*
2, 150) !174
1600 NEXT J !224
1610 CALL HCHAR(ROW+3, COL+6,
151) !088
1620 RETURN !136
1630 FOR J=1 TO 3 !059
1640 CALL VCHAR(ROW+J*2-1, CO
L+J, 153) !123
1650 CALL VCHAR(ROW+J*2, COL+
J, 154) !192
1660 NEXT J !224
1670 CALL VCHAR(ROW+6, COL+3,
155) !106
1680 RETURN !136
1690 CALL VCHAR(ROW+1, COL, 13
5, 6) !088
1700 CALL VCHAR(ROW+7, COL, 13
7) !174
1710 RETURN !136
1720 FOR J=1 TO 3 !059
1730 CALL HCHAR(ROW+J*2-1, CO
L-J, 142) !108
1740 CALL HCHAR(ROW+J*2, COL-
J, 141) !175
1750 NEXT J !224
1760 CALL HCHAR(ROW+6, COL-3,
144) !091
1770 RETURN !136
1780 FOR J=1 TO 3 !059
1790 CALL VCHAR(ROW+J, COL-J*
2+1, 146) !125
1800 CALL VCHAR(ROW+J, COL-J
2, 145) !193
1810 NEXT J !224
1820 CALL VCHAR(ROW+3, COL-6,
148) !109
1830 RETURN !136
1840 CALL HCHAR(ROW, COL-6, 13
8, 6) !083
1850 CALL HCHAR(ROW, COL-7, 14
0) !155
1860 RETURN !136
1870 FOR J=1 TO 3 !059
1880 CALL HCHAR(ROW-J, COL-J*
2+1, 150) !107
1890 CALL HCHAR(ROW-J, COL-J*
2, 149) !184
1900 CALL HCHAR(ROW-3, COL-6,
152) !091
1910 NEXT J !224
1920 RETURN !136
1930 FOR J=1 TO 3 !059
1940 CALL VCHAR(ROW-J*2+1, CO
L-J, 154) !125
1950 CALL VCHAR(ROW-J*2, COL-
J, 153) !193
1960 NEXT J !224
1970 CALL VCHAR(ROW-6, COL-3,
156) !109
1980 RETURN !136
1990 CALL VCHAR(ROW-6, COL, 13
8, 6) !083

```

(See Page 11)

REGENA ON BASIC—

(Continued from Page 10)

```

5,6) !094
2000 CALL VCHAR(ROW-7, COL, 13
6) !174
2010 RETURN !136
2020 REM FIRST OPTION !091
2030 FOR T=1 TO 5 !071
2040 PRINT "SET THE DIGITAL
CLOCK": :!121
2050 GOSUB 520 !090
2060 RANDOMIZE !149
2070 H=INT(RND*12+1)!196
2080 M=INT(RND*12+1)!201
2090 IF (H=PH)+(M=PM) THEN 20
70 !252
2100 PH=H !158
2110 PM=M !168
2120 CALL HCHAR(2,4,32,5)!12
3
2130 CALL HCHAR(2,6,58)!212
2140 ON M GOSUB 1420,1480,15
40,1570,1630,1690,1720,1780,
1840,1870,1930,1990 !194
2150 M=5*M !025
2160 IF M<60 THEN 2180 !197
2170 M=0 !004
2180 ON H GOSUB 810,870,930,
960,1020,1080,1110,1170,1230
,1260,1320,1380 !008
2190 ANS=0 !153
2200 IF H<10 THEN 2250 !001
2210 CY=4 !087
2220 GOSUB 2590 !120
2230 ANS=K-48 !227
2240 GOTO 2260 !043
2250 CALL HCHAR(2,4,59)!211
2260 CY=5 !088
2270 GOSUB 2590 !120
2280 HA=10*ANS+K-48 !027
2290 CY=7 !090
2300 GOSUB 2590 !120
2310 ANS=K-48 !227
2320 CY=8 !091
2330 GOSUB 2590 !120
2340 MA=ANS*10+K-48 !032
2350 IF (H=HA)+(M=MA)=-2 THE
N 2490 !254
2360 CALL SOUND(100,165,2)!1
32
2370 CALL SOUND(100,131,2)!1
2380 MM$=STR$(M)!017
2390 IF M>5 THEN 2410 !123
2400 MM$="0"&MM$ !234
2410 M$=STR$(H)&":"&MM$ !239
2420 MX=4 !096
2430 MY=3 !096
2440 IF H>9 THEN 2460 !172
2450 MY=4 !097
2460 GOSUB 680 !250
2470 GOSUB 720 !034
2480 GOTO 2040 !078
2490 CALL SOUND(100,262,1)!1
29
2500 CALL SOUND(100,330,1)!1
25
2510 CALL SOUND(100,392,1)!1
33
2520 CALL SOUND(200,523,1)!1
30
2530 GOSUB 720 !034
2540 NEXT T !234
2550 CALL COLOR(3,2,1)!173
2560 CALL COLOR(4,2,1)!174
2570 RETURN !136
2580 REM GET NUMBER !163
2590 CALL SOUND(100,900,2)!1
29
2600 CALL KEY(3,K,S)!190
2610 CALL HCHAR(2,CY,45)!109
2620 CALL HCHAR(2,CY,32)!105
2630 IF S<1 THEN 2600 !059
2640 IF (K<48)+(K>57) THEN 26
00 !075
2650 CALL HCHAR(2,CY,K)!133
2660 RETURN !136
2670 REM SECOND OPTION !143
2680 FOR T=1 TO 5 !071
2690 PRINT "USE b AND ENTER
TO SET HOUR HAND THEN MINUTE
HAND": :!226
2700 GOSUB 520 !090
2710 RANDOMIZE !149
2720 H=INT(12*RND+1)!196
2730 M=INT(12*RND+1)!201
2740 IF (H=PH)+(M=PM) THEN 27
20 !136
2750 PH=H !158
2760 PM=M !168
2770 M5=M*5 !078
2780 IF M<>12 THEN 2800 !240
2790 M5=0 !057
2800 MM$=STR$(M5)!070
2810 IF M5>5 THEN 2830 !086
2820 MM$="0"&MM$ !234
2830 M$=STR$(H)&":"&MM$ !239
2840 MY=3 !096
2850 IF H>9 THEN 2870 !072
2860 MY=4 !097
2870 MX=2 !094
2880 GOSUB 680 !250
2890 CALL SOUND(200,900,2)!1
30
2900 REM SET HOUR !036
2910 HH=1 !072
2920 CALL KEY(3,K,S)!190
2930 CALL HCHAR(ROW+HR(HH),C
OL+HC(HH),HG(HH))!166
2940 CALL HCHAR(ROW+HR(HH),C
OL+HC(HH),32)!073
2950 IF S<1 THEN 2920 !124
2960 IF K=13 THEN 3000 !247
2970 IF (K<>68)+(K<>100)+(K<
>9)=-3 THEN 2920 !169
2980 HH=HH+1 !153
2990 IF HH=13 THEN 2910 ELSE
2920 !159
3000 CALL HCHAR(ROW+HR(HH),C
OL+HC(HH),HG(HH))!166
3010 ON HH GOSUB 820,880,940
,970,1030,1090,1120,1180,124
0,1270,1330,1390 !201
3020 CALL SOUND(200,900,2)!1
30
3030 MM=1 !082
3040 CALL KEY(3,K,S)!190
3050 CALL HCHAR(ROW+MR(MM),C
OL+MC(MM),MG(MM))!211
3060 CALL HCHAR(ROW+MR(MM),C
OL+MC(MM),32)!103
3070 IF S<1 THEN 3040 !244
3080 IF K=13 THEN 3120 !112
3090 IF (K<>68)+(K<>100)+(K<
>9)=-3 THEN 3040 !033
3100 MM=MM+1 !173
3110 IF MM=13 THEN 3030 ELSE
3040 !153
3120 CALL HCHAR(ROW+MR(MM),C
OL+MC(MM),MG(MM))!211
3130 ON MM GOSUB 1420,1480,1
540,1570,1630,1690,1720,1780
,1840,1870,1930,1990 !015
3140 ON HH GOSUB 810,870,930
,960,1020,1080,1110,1170,123
0,1260,1320,1380 !080
3150 IF (H=HH)+(M=MM)=-2 THE
N 3300 !062
3160 CALL SOUND(100,165,2)!1
32
3170 CALL SOUND(100,131,2)!1
25
3180 CALL COLOR(2,7,1)!177
3190 CALL HCHAR(ROW+HR(HH),C
OL+HC(HH),42)!074

```

(See Page 12)

THE ART OF ASSEMBLY — PART 25

Comparing PC to TI assembly

By BRUCE HARRISON
©1993 Harrison Software

As we write this, it's June, and we have recently seen Barry Traver's presentation from the Lima show on video. Barry was discussing the comparisons between the languages TI Extended BASIC and PC Quick Basic. That spurred us to pursue an idea we'd had, to compare the Assembly languages between the two machines. We have touched upon some aspects of the PC language from time to time, but never in any depth. Today's the day to really dig into this topic.

NO CLEAR WINNER

In our opinion, there is no clear winner between the TI's Assembly language and the PC's. They are different, but each has its strengths and weaknesses. That will, we think, become self-evident by the end of this article. There are of course things that the TI can do which are near impossible on the PC, regardless of language used. There are things that can be done much more easily on the PC than on the TI, partly because of the ease of getting access to information about the workings of the machine and its operating system software. Our nearby public library has several shelves full of books on the PC. It is much more difficult to probe the inner workings of our beloved TI, and many of its quirks are still not fully understood.

SIMPLE OPERATIONS

We'll start with some very basic little operations, like MOV operations. In the TI, one can move either a word or a byte from memory to register, register to register, register to memory, or memory to memory. On the PC, one can't do memory to memory with the MOV operation, but the other three are okay. One can, however, move immediate values into either registers or memory on the PC, while immediate values can only be moved (LI) into registers on the TI. Suppose, for example, we had a word in memory labeled COUNT, and wanted to set that word to >FE84. On the TI, we could do it this way:

```
LI R1, >FE84
MOV R1, @COUNT
```

On the PC, we could accomplish this more simply by:

```
MOV COUNT, 0FF84h
```

Right away, we see another difference which can cause confusion when switching back and forth between these two languages. On the TI, the order of appearance for MOV operations (Not LI operations) is OPCODE SOURCE, DESTINATION, while on the PC it's OPCODE DESTINATION, SOURCE. This makes it imperative to remember on which machine you're working at all times. Note also that the syntax for the immediate value is different. On the TI, we use the symbol > to indicate that what follows it is a hex number. On the PC, the letter H is tacked on at the end of the number to designate Hex, but a zero is added at the front to tell the assembler that what follows is an immediate numeric value, not a label name. For immediate values that begin with a numeric character, or those written in decimal, that leading zero can be omitted, as in MOV COUNT, 58.

In both machines, things get a bit more complicated for byte MOV operations than for word operations. Let's continue with the supposition that COUNT is a word in memory, but suppose that we want to move a byte value of >1F into the low-order byte of that word. On the TI, we could:

```
LI R1, >1F00
MOVB R1, @COUNT+1
```

On the PC, we could do this:

```
MOV COUNT, 1Fh
```

That, though, would also affect the high order byte at COUNT, because the assembler would take the 1Fh as a word value because the label COUNT was defined as a word. To move the value >1F into the low order byte without affecting the high order byte, we'd have to write:

```
MOV BYTE PTR COUNT, 1Fh
```

(See Page 13)

REGENA ON BASIC—

(Continued from Page 11)

```
3200 CALL HCHAR(ROW+MR(MM), C
OL+MC(MM), 42) !104
3210 IF M<>HH THEN 3260 !033
3220 CALL HCHAR(ROW-2, COL-2,
32) !034
3230 CALL HCHAR(ROW-2, COL+2,
32) !033
3240 CALL HCHAR(ROW+2, COL+2,
32) !032
3250 CALL HCHAR(ROW+2, COL-2,
32) !033
```

```
3260 ON M GOSUB 1420, 1480, 15
40, 1570, 1630, 1690, 1720, 1780,
1840, 1870, 1930, 1990 !194
3270 ON H GOSUB 810, 870, 930,
960, 1020, 1080, 1110, 1170, 1230
, 1260, 1320, 1380 !008
3280 GOSUB 720 !034
3290 GOTO 2690 !219
3300 CALL SOUND(100, 262, 2) !1
30
3310 CALL SOUND(100, 330, 2) !1
26
```

```
3320 CALL SOUND(100, 392, 2) !1
34
3330 CALL SOUND(200, 523, 2) !1
31
3340 GOSUB 720 !034
3350 NEXT T !234
3360 CALL COLOR(2, 2, 1) !172
3370 CALL COLOR(3, 2, 1) !173
3380 CALL COLOR(4, 2, 1) !174
3390 RETURN !136
3400 END !139
```


THE ART OF ASSEMBLY—

(Continued from Page 13)

```
LOOP DOOPER ;loop back
```

The single instruction LOOP DOOPER, on the PC, performs the operations of DEC CX, CMP CX,0 and JNE DOOPER. That's handy, so long as CX is available at this point in your program. If that's not so, for example in the case of two nested loops, one can still use this structure with the help of the PUSH and POP operations that the PC offers. The PC maintains a stack for temporary storage of word values from any source. The instruction PUSH places what follows on the stack, and decrements the stack pointer by two. POP does the opposite.

Let's assume we needed to nest two loops in a PC program, and needed CX for counting both. We could do it like this:

```
MOV CX,10 ;set outer loop count
OUTLP: (perform steps)
PUSH CX ;Stash CX on stack
MOV CX,5 ;set inner loop count
INLP: (perform operation of inner loop)
LOOP INLP ;Five times
POP CX ;Get old value of CX back
LOOP OUTLP ;repeat 10 times
```

On the TI, we'd just use two registers for the two loop counts, and that might be much easier, but then PUSH and POP are easy too.

The TI offers no equivalent of the PC's REP operation. There are some operations on the PC where we can use a count value in the CX register to repeat without the loop instruction. One good example is the case of moving some group of bytes from one location in memory to another.

To make this more understandable, we'll have to introduce the concept of the Source and Destination registers. These two registers, called SI (source index) and DI (destination index) are used to point to memory locations. The SI register is understood to point to the DATA SEGMENT, and the DI to the EXTRA SEGMENT, but let's for the moment assume that these are one and the same. To use the REP instruction, we must first set up SI, DI, and CX. Let's say we want to move fifty bytes from a location labeled STRING1 to a location labeled STRING2. On the PC, we can do this:

```
MOV CX,50 ;set count in CX
MOV SI,OFFSET STRING1 ;address string1
MOV DI,OFFSET STRING2 ;address string2
REP MOVSB ;repeat MOVSB CX times
```

Here we've used another new word in the PC lexicon, OFFSET. This simply means that we're moving the address of STRING1 and STRING2 into SI and DI, not the value located at that address. The single instruction REP MOVSB will move a byte from the location pointed by SI to the location pointed by DI, will increment both SI and DI, decrement CX, compare CX to zero, and loop back to MOVSB if CX is not zero. That's a powerful instruction, but must be used with care. If one makes the mistake of invoking it when CX happens to be zero to start with, a whole 64K segment will get moved before the REP stops. This can be disastrous.

Just a bit ago, we mentioned the DATA SEGMENT and EXTRA SEGMENT without explanation. There is no equivalent to these on the TI, since the entire memory address space on the TI is one 64K

"segment". On the PC, one may set aside segments of any number of bytes up to 64K. The program's code can occupy one or more such segments. Subroutines can also occupy more than one code segment. Data can be loaded into two or more such segments. One can, for example, keep all the program's variables and constants in one segment, and set aside an entire 64K for the user's data. Two registers are used to "point" to these two segments. These are called DS and ES for DATA and EXTRA. In many cases both DS and ES point to the same physical segment of memory, but they can be treated separately.

Like other registers, these can be "pushed" and "popped" to and from the stack, which has its own segment. Yes, this eats memory like crazy, but that's the nature of doing things on the PC anyway. Let's assume we want to set the ES to the same as the DS for some operation, then put it back later to its original value. We could:

```
PUSH ES ;save original ES
PUSH DS ;save DS
POP ES ;set ES to DS from stack
(perform operation)
POP ES ;restore original ES
```

The problem with using this stack business is to match up all the pushes and pops (each PUSH must have a corresponding POP) so that one doesn't mess up the stack pointer or accidentally get the wrong value into a register

We've used a lot of this space on the PC's language, but of course most of this whole series has been dedicated to the TI, so we hope you'll excuse our going on somewhat about the PC this time.

INITIALIZING DATA

As we mentioned before, our common practice when writing Assembly source code is to keep all our data in a separate part of the source file. On the PC, the data has its own segment, so it must be kept apart from the code. (There are ways to "cheat" on this process, and have some data accessed from within the code segment, but we're not going to teach cheating here.) On the TI, one identifies data as belonging to one of three kinds, DATA (a word), BYTE (a byte) or TEXT (a string of alphanumeric characters). On the PC, there are only two categories for data statements, DW (a word) and DB (a byte). (There's also DD for a Double Word of DATA, but let's skip that for now.) How then does one introduce the text for prompts and such?

It's easy. One uses the DB notation, but with single quote marks, just like the TEXT in a TI source statement. For example, a string can be done like this:

```
PAKMSG DB 25, 'PRESS ANY KEY TO CONTINUE'
```

Here we've taken advantage of the situation to stick in the length byte and the string's content in a single source code line. On the TI, we accomplish the same effect with two lines:

```
PAKMSG BYTE 25
TEXT 'PRESS ANY KEY TO CONTINUE'
```

That takes a bit more work, but accomplishes the same result, and takes up exactly the same amount of memory.

BIOS AND DOS SERVICES

By now you're all familiar with the concept of Utility Vectors on the TI, like VMBW, KSCAN, DSRLNK, and such. There are sim-

(See Page 15)

THE ART OF ASSEMBLY—

(Continued from Page 14)

ilar things on the PC, but they are called Interrupts. There are two sets of them loaded into the lower part of the PC's memory when it's "booted". The first set are the BIOS (Basic Input/Output System) Interrupts. These are sort of "primitive" routines, but can be extremely useful. The second set are called DOS (Disk Operating System) Interrupts. These are more "advanced", but in some ways harder to use than the BIOS ones. For many of these interrupts, more than one service is available within one interrupt. You tell the interrupt which service is desired by placing a number in the AH register before calling the interrupt itself. One might, for example, do this:

```
MOV AL, 'A'
MOV AH, 14
XOR BH, BH
INT 10h
```

Interrupt 10h is part of the BIOS. The call to INT 10h would place the character "A" on the screen, at whatever position the cursor happened to be, and would advance the cursor by one position on the screen. We XOR'd BH with itself here so that the character would go to the normal screen. Positioning the cursor uses a different service from INT 10h, (AH = 2) with the DH and DL registers indicating the Row and Column on the screen, and BH indicating which screen "page" is being written to. INT 10h is one of the most useful of all the Interrupts available, since it can be used for many "video"

related purposes. One could think of it as a combination of TI's VMBW, VSBW, VWTR, and VSBR. It will do more than these four TI services. Other BIOS interrupts will read the keyboard, read and write disk sectors, and so on.

In its "TEXT" mode, the PC has four screens (numbered 0 thru 3) that we can write to. We could pre-write a whole screen full of text to screen 1 while the monitor is showing screen 0, then use a BIOS call to switch screens and reveal screen 1 "instantly" on the monitor. (A few months ago, we explained how this can be done on the TI in VDP Ram by using the VWTR utility.)

The Interrupts that are part of the DOS are generally more complex, and vary considerably from one version of DOS to another. When we write for the PC, we use DOS interrupts belonging to Version 2 of DOS, as these will work on all but the oldest DOS versions, and will also work with later versions.

Like the BIOS Interrupts, the DOS ones have multiple functions built into each interrupt handler. Just the single Interrupt 21h from DOS has perhaps 50 functions. (Maybe more than that by now. We don't really know, since we haven't tried Version 5.0 of DOS. Each new version adds more services to INT 21h.)

There are some good DOS services available, and of course we've used them in our PC work. Opening, writing, reading, and closing files are all easily accomplished with INT 21h, at least from version 2.0 onwards. There are some services offered in DOS

(See Page 15)

HORIZON COMPUTER

NEW HORIZON 4000 RAM DISK

Up To 8 Meg on Single Layer Built In RAMBO and PHOENIX
RAMDISK BARE BOARD, MANUAL, ROS 814 F \$65.00

ZEROk Kit, Above + Parts, NO MEMORY \$125.00

128k Kit \$125.00 256k Kit \$150.00

384k Kit \$175.00 512k Kit \$200.00

1 Meg Kit \$325.00 1.5 Meg Kit \$425.00

2 Meg Kit \$500.00 8 Meg Kit \$CALL

ADD \$30 FOR ASSEMBLY ON RAMDISK OR P-GRAM

128k Memory Chips NOW only \$25 512k Memory Chips \$125

P-GRAM 72k Kit \$150 P-GRAM+ 190k Kit \$175

P-GRAM CLOCK \$20 P-GRAM Upgrade \$30

HORIZON MOUSE \$40 MEMEX 504k \$200

M G Eprom for CorComp Disk Controller \$35

The NEW SCSI Hard Disk Controller is STILL waiting on the DSR

We hope to have the DSR ready in July, (we said that last year too)

Order From

WIND MILLS SERVICES
166 DARTMOUTH DRIVE
TOLEDO OH 43614-2911

Call 419 385 5946 voice 6am to 8pm EST
419 385 7484 BBS

Shipping is included in pricing for US & Canada.... All other Countries add \$10 to Order

Prices subject to change

Call for latest prices

Ohio Residents ADD 6.5% ST

Visa & M C

Any HORIZON CARD REPAIRS \$50



Buy an Asgard
Mouse, get a
copy of Classic
Checkers FREE!

The most popular TI-99/4A mouse is also the best value in mice for the TI-99/4A! Get our great RS232-card mouse, plus example programs and a TI-Artist driver, along with our Classic Checkers for one low price!

\$39.95 U.S. add \$3.00 S&H
Can. add \$4.00 S&H
Airmail add \$7.00 S&H

Asgard Software • 1423 Flagship Dr.
Woodbridge VA 22192 • 703-491-1267

INSTAROT

An X BASIC program to rotate TI-Artist Instances

The following article and program were written by Lucie Dorais of the Ottawa TI User Group. The article appeared in the group's newsletter.—Ed.

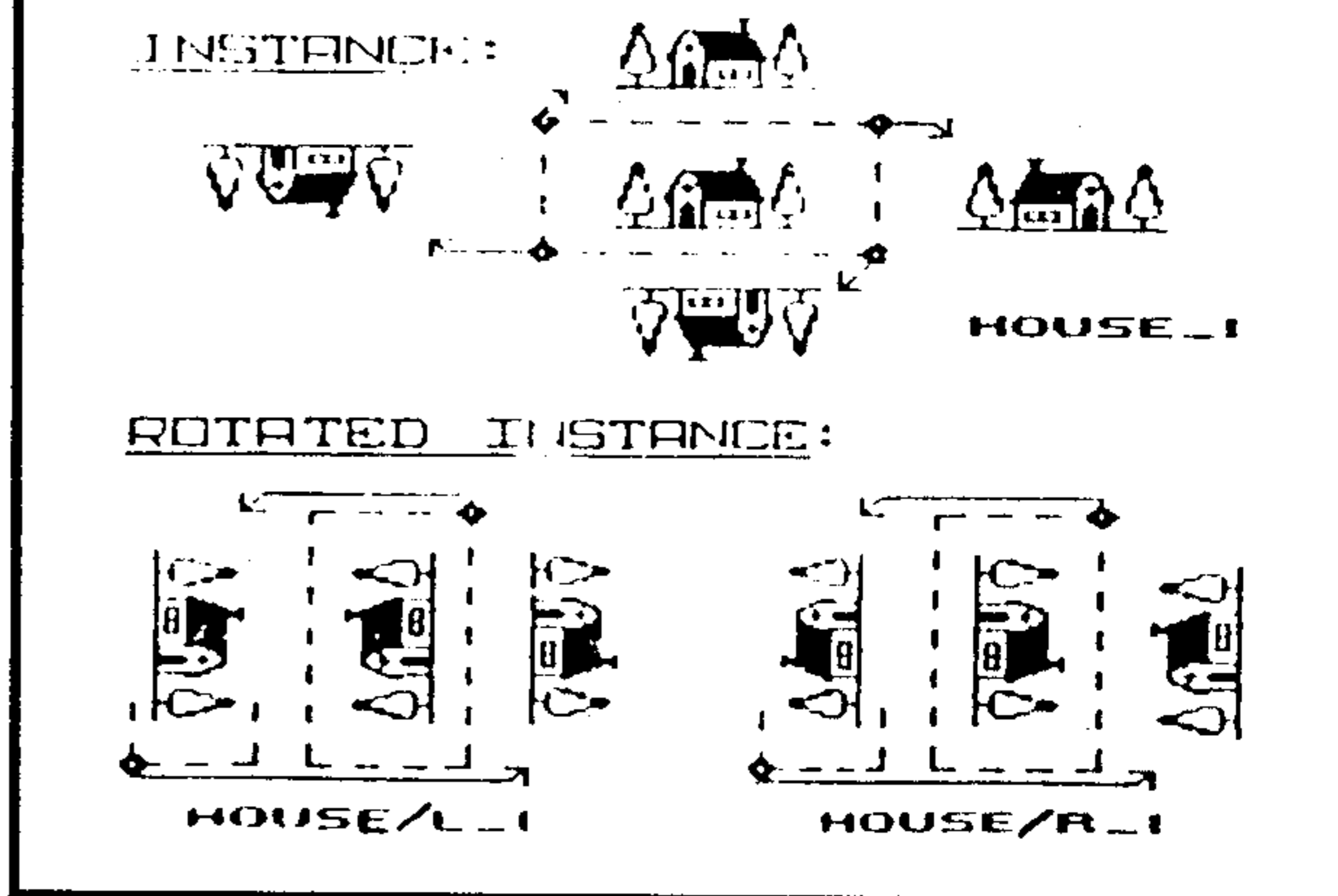
INSTAROT rotates a TI-Artist Instance on its left or right side, a function that TI-Artist Plus still lacks; its vector function does some sort of rotating, but the results are far from good. Genial Computerware's Graphics Expander does a great instance job, in any direction, but you might not have it. Joypaint's rotate function is excellent, but it is a hassle to use it with TIA instances (you must convert it to a pic, convert with JoypaintPal, load Joypaint, do the rotating, then do the reverse trip back to your original instance). Do you have Joypaint, anyway? And its Pal? My Extended BASIC utility does exactly the same thing,

although it takes a little more time.

Still unknown to many users, TIA (and TIA+) can flip a picture, i.e. reverse it horizontally (mirror) or vertically (upside down). This function is available in both the move and copy mode of the enhancement function. To understand how it works, look at Fig. 1; the original instance HOUSE_I is at the top, framed by a broken line. The big dots at the corners of the frame indicate the starting position of the cursor when you press Enter (or the file button) to "frame" the portion of the picture to be moved or copied.

If you place the cursor at the upper left

FIG. 1



corner, you copy or move the picture as is; but if you position it at the upper right corner, the picture will be mirrored (horizontal flip). If you start at the lower left corner, the picture is flipped vertically; and if you start at the lower right corner, your picture will be flipped both vertically and horizontally. The bottom part of the diagram shows you what can be done with my INSTAROT program (new instances HOUSE/L_I and HOUSE/R_I) and what you can do with the rotated pictures by placing the cursor at different positions. You will see that if you need both left and right rotations, flipping one rotated picture twice (1—upper right, 2—lower left) will give you the second rotation, so there is no need to spend a lot of time with INSTAROT.

Instances are D/V80 text files so they can be used outside the TIA environment. In Fig. 2 (left column) is the content of the (See Page 17)

FIG. 2

```

5,3          ---total COLUMNS, ROWS in instance ---
0,8,28,28,28,20,50,50    r1c1 > CALL CHAR(A,"00081C1C1C143232")
0,0,0,7,13,8,26,23      r1c2 > CALL CHAR(B,"000000070D081A17")
1,0,0,255,255,255,255,127 r1c3 > CALL CHAR(C,"010000FFFFFFFF7F")
224,192,192,224,224,240,240,248 r1c4 > CALL CHAR(D,"E0C0C0E0E0F0F0F8")
0,32,112,112,112,80,200,200 r1c5 > CALL CHAR(E,"002070707050C8C8")
34,97,97,65,65,62,8,8    r2c1 > CALL CHAR(F,"22616141413E0808")
50,48,50,23,23,23,23,23, r2c2 > CALL CHAR(G,"3230321717171717")
127,127,127,64,79,73,79,64 r2c3 > CALL CHAR(H,"7F7F7F404F494F40")
248,253,253,9,233,40,232,8 r2c4 > CALL CHAR(I,"F8FDFD09E928E808")
136,132,132,4,4,248,32,32 r2c5 > CALL CHAR(J,"8884840404F82020")
255,0,0,0,0,0,0,0,0      r3c1 > CALL CHAR(K,"FF00000000000000")
255,0,0,0,0,0,0,0,0      r3c2 > CALL CHAR(L,"FF00000000000000")
255,0,0,0,0,0,0,0,0      r3c3 > CALL CHAR(M,"FF00000000000000")
255,0,0,0,0,0,0,0,0      r3c4 > CALL CHAR(N,"FF00000000000000")
255,0,0,0,0,0,0,0,0      r3c5 > CALL CHAR(O,"FF00000000000000")
    
```

THE ART OF ASSEMBLY—

(Continued from Page 14)

which Peter Norton correctly describes as "screwball" services. Some of these were carryovers from pre-DOS operating systems like CP/M. Our practice is to avoid the DOS services except for disk file operations, and use BIOS calls for operations involving the screen and keyboard. In addition to these BIOS calls being simpler and more flexible, they also execute faster. Many of the DOS services make repeated calls to the BIOS services to perform their

tasks, and this makes them execute much more slowly than direct calls to the BIOS services.

This article could go on for many more pages, but we'll stop here having just provided a glance inside the PC's Assembly language. It's definitely not "superior" to the TI's, but it has some really powerful instructions, and in experienced hands it can make truly wonderful things happen. See you next month with an "all TI" column.

INSTAROT—

(Continued from Page 16)

HOUSE_I file, the instance used in Fig. 1. RCL means character in row one/column one, etc. The first line of the data gives the number of columns and rows, and each line after it contains one character definition. HOUSE_I, having 5 columns and 3 rows (15 characters altogether), the file has 15 lines of definition. These are expressed in eight decimal numbers, equivalent to the hex numbers we would use in a CALL CHAR statement. The file is read row by row, from upper left corner (rlcl) to lower right corner (r3c5, in the present case). To see the house, write a program with the CALL CHAR values in the right column.

To keep it as fast as possible, INSTAROT does not convert the decimal values to hexadecimal, so it cannot show the instances on the screen. But it works! I used it to draw the illustration in Fig. 1. It takes about 4.5 seconds to rotate one character; so, before proceeding, Tex will compute the estimated time needed to rotate the instance. Beware that, due to Tex's memory space, your instance has to have 250 characters or less (see the DIMension statement in line 130. If you work with a bigger instance, just save it as two smaller ones, and rotate them separately.

Half of the program is devoted to house-keeping: Initializing lines 120-160, screen interface in 180-250, get row/column, calculate estimated time and ask you to proceed in lines 270-320. The sprites defined at the beginning are an example of rotations of a frog face (what else!). Since they can be magnified, three frogs (left-rotated, normal and right-rotated) are defined in line 140. Everywhere in this program, to save memory space, I used A\$, S and K each time I needed a temporary variable. For the pre-scan, note where I put the GOTO 180 (end of line 150). It works.

The rest of the INIT is easy, so is the screen interface. Just note that the program will add "_I" to your instance name if you do not write it (line 210). Then, after opening the input file (instance to rotate), Tex will suggest a name for the rotated instance name, adding "L/_I" or "R/_I" according to your answer to R\$ (lines 230-240). You can keep the new name or change it before the output file is opened.

Line 270 reads the first line of the in-

stance file. Since it is a string with a comma, we must LINPUT it, not INPUT, and then read the segments left and right of the comma to get the C and R numerical VALues (line 270). "T" is the total of characters to be rotated. If there are too many, Tex will tell you and GOTO 510 to ask you for another. Otherwise, it will compute the time needed to rotate the instance and ask you to proceed or not (lines 300-320). An interesting point: Even if the S value can have decimals, being a multiple of 4.5 the image will display it as a round number.

To rotate an instance, two things must be rotated: The place of each character into the new row/column grid, since a row will become a column and vice-versa, and then the content of each character definition, i.e. each line from the input instance file. My solution was to use a memory array (Z\$) to hold the new lines of numbers resulting from the rotation at the correct rotated position, then saving the whole array to disk in one operation (line 500). Because of the rotation, the first line of the new file will now contain the values of "R,C", since the starting column total is now the row total, and vice-versa. A D/V80 file is always a sequential file, because each record is variable. It would be nice to use a relative file to save each new line into its new position right after it is processed, but TIA could not read the file. Therefore, the need for a memory array, which explains the 250-character limit of INSTAROT.

In line 350, AP (new Array Position) and SAP (Start-of-AP for each row) are initialized according to the R\$ rotation wanted and to the total number of characters and rows in the instance to be rotated. AP is used to save the content of each character into Z\$ after it is rotated (line 460), then the next AP/SAP values will be computed, again according to the rotation wanted (lines 470-480).

Each line of the instance (one chardef) is LINPUTed (commas!), processed and saved to memory by the FOR-NEXT Z portion of the program (lines 360-490). Lines 370-390 is an X loop to convert the string (eight decimal numbers separated by commas) into numerical VALues and then put them into the V() array to be processed. It also doubles as a loop to zero the N() array. After the rotation routine proper, N() will

be filled with eight new decimal values, one for each row of the rotated CHAR, and another X loop will transform N() into a string with commas for the new instance file (lines 450-460).

Lines 410-440, that lie between the two X loops, are the heart of the character rotation routine. I have written a very detailed explanation of how the routine works, with nice graphics and table, but it takes two full pages. So I will just summarize as best as I can what the routine does.

To rotate one chardef, Tex takes each V() decimal value (FOR Y=1 TO 8) and dissects it into the "powers of two" values it contains, from $2^7=128$ to $2^0=1$ (FOR X=7 TO 1 STEP -1). If the power-of-two value is found, it means the pixel is on, and the corresponding pixel in the rotated character has to be set on too, but in a new position in the chardef grid. Tex places the power-of-two value corresponding to that pixel position in the corresponding variable in the N() array. The total of each N() variable will, in the end, give us the decimal value of the new row. Simple, is it not?

No matter if it is still all strange to you, just type the program and watch the results.

INSTAROT

```

100 ! INSTAROT: Rotate a TIA
      Instance/ L. Dorais, OTTAWA
      UG, April, 1991 !059
110 !!131
120 CALL CLEAR :: CALL COLOR
      (13,14,1) :: CALL MAGNIFY(2) !
199
130 DIM Z$(250) :: F$="DSK1."
      :: L$=RPT$(CHR$(128),28) !02
      8
140 CALL CHAR(123,"7C9EB77B7
      BB79E7C6699BDFFD77E3C3E79E
      DDEDED793E",128,"000000FFFF
      ") ! FROGS, LINE !024
150 CALL SPRITE(#1,123,16,1,
      17,#2,124,14,1,121,#3,125,16
      ,1,225) :: GOTO 180 !128
160 A$,AP,C,K,N(),OP,P,R,R$,
      S,SAP,T,V(),X,Y,Z,ZZ :: CALL
      KEY :: !@P- !200
170 ! —SCREEN DISPLAY/INP
      UT — !224
180 DISPLAY AT(2,5): "<LEFT" ;
      TAB(19); "RIGHT >": :L$ !028

```

(See Page 18)

INSTAROT—

(Continued from Page 17)

```

190 DISPLAY AT(6,1):"INSTANC
E TO ROTATE:" : " "&SEG$(F$,
1,5) : : "ROTATION WANTED (L
'R):" !081
200 DISPLAY AT(14,1):"ROTATE
D INSTANCE:" : " DSK" : :L$ :
: A$="_I" !081
210 ACCEPT AT(8,5)SIZE(-12)B
EEP:F$ : : IF POS(F$,A$,1)=0
THEN F$=F$&A$ !152
220 F$="DSK"&F$ : : OPEN #1:F
$,INPUT : : F$=SEG$(F$,1,LEN(
F$)-2)!148
230 ACCEPT AT(11,24)SIZE(1)V
ALIDATE("LR")BEEP:R$ !077
240 DISPLAY AT(16,2):SEG$(F$
,1,11)&"/"&R$&A$ !200
250 ACCEPT AT(16,5)SIZE(-12)
:F$ : : F$="DSK"&F$ : : OPEN #
2:F$,OUTPUT !163
260 ! -GET ROW/COLUMN, TELL
TIME- !024
270 LINPUT #1:A$ : : P=POS(A$
,",",1) : : C=VAL(SEG$(A$,1,P-
1)) : : R=VAL(SEG$(A$,P+1,2)) :
: T=C*R !005
280 DISPLAY AT(20,1):R;"ROW
X";C;"COL =" ;T;"CHAR" : :L$ !

```

```

000
290 IF T>250 THEN DISPLAY AT
(23,6)BEEP:"INSTANCE TOO BIG
!!" : : GOTO 510 !113
300 S=4.5*T : : K=INT(S/60) : :
S=S-60*K ! TIME IN K MIN S
SEC !228
310 DISPLAY AT(24,1):USING "
EST TIME: ##M##S PROCEED? Y"
:K,S !001
320 ACCEPT AT(24,28)SIZE(-1)
BEEP:A$ : : IF A$="N" THEN 51
0 !235
330 ! -ROTATE ONE LINE OF 8
CHARS- !014
340 DISPLAY AT(24,1):" NOW R
OTATING CHAR:" !127
350 IF R$="L" THEN SAP,AP=T-
R+1 ELSE SAP,AP=R !062
360 FOR Z=1 TO T : : DISPLAY
AT(24,21):Z !067
370 LINPUT #1:A$ : : S=1 : : O
P=0 : : ZZ=INT(Z/C)!177
380 FOR X=1 TO 8 : : N(X)=0 :
: P=POS(A$,",",S) : : IF P=0 T
HEN P=LEN(A$)+1 !175
390 V(X)=VAL(SEG$(A$,S,P-OP-
1)) : : S=P+1 : : OP=P : : NEXT
X ! GET DEC VALUES !140

```

```

400 ! -ROTATE ONE CHARACTER
- !213
410 FOR X=7 TO 0 STEP -1 : :
FOR Y=1 TO 8 : : IF V(Y)<2 TH
EN 440 !153
420 IF R$="L" THEN N(X+1)=N(
X+1)+2 ELSE N(8-X)=N(8-X)+2
!057
430 V(Y)=V(Y)-2 !179
440 NEXT Y : : NEXT X : : A$="
" !203
450 FOR X=1 TO 8 : : A$=A$&ST
R$(N(X)) : : IF X<8 THEN A$=A$
&". " !111
460 NEXT X : : Z$(AP)=A$ ! BU
ILD DECIMAL STRING, PUT IN N
EW ARRAY !205
470 IF R$="L" THEN AP=AP-R :
: IF Z/C=ZZ THEN AP=SAP+ZZ !
161
480 IF R$="R" THEN AP=AP+R :
: IF Z/C=ZZ THEN AP,SAP=SAP-
1 !132
490 NEXT Z : : DISPLAY AT(24,
1):"SAVING NEW INSTANCE TO D
ISK" !058
500 PRINT #2:STR$(R)&". "&STR
$(C) : : FOR X=1 TO T : : PRINT
(See Page 19)

```

1993 TI FAIRS

APRIL

Northeast TI Fair, April 17, Waltham High School, Waltham, Massachusetts. Contact Ron Williams, 14 East St., Avon, MA 02322.

Canadian TI Fest, April 24, Merivale High School, Nepean, Ontario, Canada. Contact Bill Gard, 3489 Paul Anka Dr., Ottawa, Ontario, Canada K1V 9K6 or (613) 523-9396 or Fax (819) 997-2194 Attn: DMES 2.

MAY

Lima Multi User Group Conference, May 14-15, Ohio State University Lima Campus, Lima, Ohio. Contact Dave Szimpl, 4191 Patterson Haplin, Sidney, OH 45365; phone (513) 498-9713 (evenings).

Fourth Annual TI Orphans Reunion, May 15, Zurich Insurance Claims Centre, 9715 Ottewell Rd., Edmonton, Alberta, Canada. Contact Ron Hohman, (403) 456-0862.

SEPTEMBER

Western Washington TI Fair, Sept 18, Tacoma Waterworks, 3506 S. 35th, Tacoma, Washington. Jim Tompkins, (206) 756-0934.

OCTOBER

Annual International TI-Faire, Oct. 8-10, Evangelisches Fe-

rienwaldheim Weidachtal, 7000 Stuttgart 80 (Mörhingen), Weidach Gewann 8, Germany. Contact Hans Huben, Berberitzenweg 6, 7033 Herrenberg, Germany; Wolfgang Bertsch, Helenenburgweg 61, 7120 Bietigheim-Biss, German; or Dierk Warburg, Lilienweg 12, 7141 Benningen, Germany.

Chicago International World Faire, Oct. 30, Holiday Inn, Gurnee, Illinois. Contact Cecure Electronics, P.O. Box 132, Muskego, WI 53150, or Don Walden, (414) 679-4343.

Milwaukee TI Fair, Oct. 31, Quality Inn, 5311 Howell Ave, Milwaukee, Wisconsin. Contact Gene Hitz, 4121 North Glenway, Wauwatosa, WI 53222.

1994 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.

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.

Don't overlook the details

Tips on buying new floppy disk drives

By **RICHARD ROSEEN**

The following article has appeared in several user group newsletters.—Ed.

Check for quality the main mechanical parts of the drive. They should be located on a solid die-cast metal. In other words, it should have a solid metal structure throughout, as the base of the drive is what holds the motors, solenoids and other movable parts.

New drives should be sold in anti-static plastic wrap, should always be half-height (not full-height), at least two-sided, capable of 360K using double-sided, double-density disks. The 720K 80-track drives are becoming scarce because of the newer 1.2 megabyte drives. These 1.2 meg. drives can be usable at 720K. More on that later.

New 3.5-inch drives are 720K or 1.44M. They should also have die-cast bodies. Only the Myarc HFDC can handle high density drives and then only with the installation of a special chip.

Newer drives have a direct drive motor, so you won't see any drive belts as with older models.

Manufacturer warranties should be at least 12 months from date of purchase. Also investigate the seller's warranty. Typically, a seller will provide a 30-90-day warranty, in addition to the manufacturer's warranty.

Make sure you keep your receipt, along with the phone number and address of the seller. This information may be required if you ever have to make a claim based on the manufacturer's warranty.

Look for heads that are mounted on solid assemblies. Two mounting rails are preferable to one. Look for solid movement of the head assembly by the stepper motor. Since the heads take the most abuse and are often involved when alignment problems occur. Check the stepper motor that drives a screw shaft, or two strips that wind on or off the stepper motor shaft and on and off the head assembly as the heads move in either direction. An example of the stepper motor that drives a screw shaft are the 3.5-inch, 720K Chinnon and Fujitsu drives. An example of the strap winding drive is the

5.25-inch Mitsumi 360K drive.

A 3.5-inch drive can be hooked up bare without the 5.25-inch bracket with 34-pin socket IDC (Insertion Displacement Connector) connected to the square pins on the 3.5-inch drive. If this is done, the odd-ball but findable 4-pin 3.5-inch drive power connector must be used. These are not the same as the power connectors used on a 5.25-inch drive. These connectors do not have polarity tabs and can make difficult getting the proper polarity or orientation to plug in. Go for the works, get the 5.25-inch bracket and the card edge adapter board that includes a standard power connector as used on 5.25-inch drives. These adapters may have a jumper for use on a PC XT or AT. Be sure to select XT.

Unless you have help from a guru, get information from the seller or manufacturer on the drive selects, other jumper options or features, and resistor packs. On some drives, the resistor pack is soldered to a high-density logic board with a jumper to disable or enable the resistor pack. Older drives generally have removable resistor packs.

If you need a power supply, the least expensive will be found at electronic supply shops, not at a computer dealer. The same goes for cables, connectors and other parts. The electronic shop will likely have the supplies you'll need to build a linear power supply, if you need one. Instructions on building your own power supply, however, are beyond the scope of this article.

Some 3.5-inch drives require less than 1

amp for 5 and 12 volts. Some are very low power and some require only a 5-volt supply. Newer 5.25-inch half-height drives never require more than 1 amp on 5 and 12 volt lines, and may be as low as 0.5 amp. Add the amperage required for each drive (both 5 and 12 volt lines) to determine your power supply needs for your drives. Drives can be powered separately because the 34-pin cable will carry the common logic signal ground between all drives on the bus and the computer. If building a linear supply, be sure the transformer, bridge rectifier diodes and linear regulators exceed your amperage needs. The transformer should be at least 12.6 VAC RMS and 6.3 VAC RMS. Transformers are commonly rated with RMS voltage at their secondary windings.

PREFERENCES

My preferences are for Mitsumi drives, both 3.5-inch and 5.25-inch, of any density. These drives are quiet, and they have jumper blocks to enable/disable the resistor pack. I also like the NEC-1036 3.5-inch 720K drive. This is a quiet, small and durable low power drive. I also recommend the Chinnon 3.5-inch 720K drives. These are similar to the NEC drives except that the screw shaft stepper motor and extremely lower power and 5 volt only operation make it better.

INSTAROT—

(Continued from Page 18)

```
#2:Z$(X)::NEXT X !117
510 CLOSE #1 :: CLOSE #2 !END !011
520 DISPLAY AT(24,1)BEEP:"ROTATE ANOTHER INSTANCE?(Y/N)" !062
530 CALL KEY(0,K,S)::IF S=0 THEN 530 ELSE CALL CLEAR ::IF K<>78 THEN 180 !224
```

COMPETITION **SOLUTIONS**

2219 S. Hudson Milwaukee WI 53215 414-672-1600 Fx 672-8977

800-471-1600

(Nationwide & Canada orders only)

**BUY - SELL - TRADE
HARDWARE - SOFTWARE
Dual 1/2 HT drive kit \$65**

**P.E. Box Complete \$80
Star NX-1001 Printer \$160
DS/DD 1/2 Ht Drive \$30
2400 Baud Modem \$79
TI-99/4A Console \$35**

Catalog \$2 Open Daily 9-5 Sat 10-3
Huge Genuine TI Inventory Since 1982
Bankcards, Discover, Checks & UPS/COD
100 Plato Titles in Stock!

BUGS AND BYTES

Update on the SCSI

When will the SCSI and 4A Memex cards be ready? This question was asked on Usenet, and here's the response from Don O'Neil, of Western Horizon Technologies.

The SCSI is coming as soon as we can (finish it). We have been working hard on the DSR. I do not know when SCSI will be totally complete, but we are not giving up on it. As for 4A Memex, I cannot afford to put any money on the line until I know when SCSI is approaching completeness, then and only then will 4A Memex go into production for development work, after that, 1-2 months later 4A Memex will be available as a complete product. (many of the same members of the SCSI team will be developing the 4A Memex DSR). We

(WHT) have not stopped working on other projects that do not require the talents of the SCSI/4A Memex team, and hope to have them ready this summer. I cannot and will not divulge any information relating to any other projects WHT is working on until it is ready for sale. SCSI is a difficult task (4A Memex too, but not nearly as difficult) and has required hundreds of hours investigating the how's, when's and why's for SCSI communications. We do have code written to do nearly every SCSI function, but the hard disk format and high density formats are still being hashed out (Myarc HD format will allow only 212MB, not large enough for many SCSI drives) and "work arounds" to Myarc philosophies of disk struc-

(See Page 25)

NEWSBYTES

Site selected for Washington fair

The Western Washington TI Fair is scheduled for Sept. 18 at the Tacoma Water Works, 3506 S. 35th, Tacoma, Washington.

According to organizer Jim Tompkins, the fair features free entrance, free coffee and free vendor tables if one month advance reservations are made. For further information, contact Tompkins at (206) 756-0934.

Chicago fair planned for Gurnee

The Chicago International World Faire is scheduled for Oct. 30 at the Holiday Inn in Gurnee, Illinois.

Don Walden, president of the Chicago TI Users Group, says vendor tables will be "greatly reduced in price" this year. He says the club is "looking in the \$30 to \$40 range" for vendor tables.

Walden notes that, for non-computing family members of faire-goers, the Gurnee Holiday Inn is "right across the street" from the Gurnee Mills Shopping Mall. "It's a huge shopping mall. People from Wisconsin bus down to go to it," he says.

He says the fair site is halfway between Mitchell and O'Hare airports and is visible, and easily accessible, from the interstate.

For further information, contact Cecure Electronics, P.O. Box 132, Muskego, WI

53150, or call Walden at (414) 679-4343.

VAST group offers artwork catalog

The Valley of the Sun User Group has compiled a catalog of artwork for Page Pro99 by Virginia Davis.

The catalog shows samples from her disks of paper dolls, both male and female, and disks with an Indian village, a town, paper airplanes and paper spaceships.

The catalog may be had for the mailing cost of \$1 from the VAST User Group, P.O. Box 25576, Tempe, AZ 85285-5576.

Addresses change

New mailing address for the Suncoast Users Group is c/o Roy Moglia, Newsletter Editor, 5137 Burlington Ave. N., St. Petersburg, FL 33710.

New address for the Southern Nevada Users Group is c/o Rudy Johnson, 6317 O'Bannon, Las Vegas, NV 89102.

Milwaukee fair set for Halloween

The Milwaukee TI Fair has been scheduled to be held the day after the Chicago International World's Faire, as in years past. The scheduled site is again the Quality Inn at 5311 Howell Ave. in Milwaukee, Wisconsin. For further information, contact Gene Hitz, 4121 North Glenway,

Wauwatosa, WI 53222.

Tesch compiling BBS listing

Tim Tesch of Milwaukee, Wisconsin, is attempting to compile a current list of computer bulletin boards supporting the TI99/4A and Geneve computers. He asks users to send him numbers or listings they know about or have listed on their own dialers.

Write him at 3804 North 75th St., Milwaukee, WI 53216, or leave a mail message on his 24-hour BBS at (414) 464-1978.

Easy Data updated

Harrison Software has updated its Easy Data database program (reviewed May 1992). The updated version allows users to place program lines among DATA lines and allows the addition of subprograms after the DATA section. Both the menu driver and sort routines have been modified to ignore anything that is not a DATA line. The program is priced at \$6. For more information, contact Harrison Software, 5705 40th Place, Hyattsville, MD 20781; 301-227-3467.

Reach thousands of TI/Geneve users at no cost. Send your product and event announcements to MICROpendium Newsbytes, P.O. Box 1343, Round Rock, TX 78680.

RICHGKXB

Supercharging Extended BASIC

By JOHN KOLOEN

Rich GRAM Kracker Extended BASIC is more than a programming language. It's a two-year-old project that keeps getting better and better.

Rich Gilbertson started reworking GRAM Kracker Extended BASIC two years ago, at first to rid the bugs from the Extended BASIC portion of program. In the intervening years, he's added scores of functions and commands that weren't even dreamed of by the TI programmers who originally wrote XBASIC. Now, Rich is up to version 5.54, with no end in sight.

At this point, RGKXB is available only on disk. It requires a GRAM device to run. This means TI99/4A users with a GRAM Kracker, P-GRAM card, GRAMulator or similar device can use it. And Geneve owners can use it. Others will have to wait for the cartridge version to make its appearance (which, I'm told, is going to happen soon). However, one advantage to the disk version is that updates are cheap and, given Gilbertson's commitment to perfecting RGKXB, frequent. Gilbertson recommends that users not upgrade unless the version number increases by a full digit. In other words, if you have V5.0, don't upgrade until the version number hits 6.0. The current version is V5.54.

RGKXB starts with the Extended BASIC version that was produced by Miller Graphics for its GRAM Kracker. Added to it is an array of features that can only be summarized here.

PERFORMANCE: You'll notice the difference between RGKXB and TIXB as soon as you load it. The first RGKXB screen gives you the following menu selections:

Any Key = Autoload
Enter = DSK1.UTIL1
Space = Bypass Autoload
= Editor/Assembler

If you press any key, the program looks for a file called LOAD, just like XB. However, press Enter and you are flipped to a second screen, with the following options:

1. Editor
2. Assembler

REVIEW

REPORT CARD

PerformanceA
Ease of UseA
DocumentationA
ValueA
Final GradeA

Cost: \$24.95 + \$2 S&H; updates \$5 plus original disk

Publisher: C.A.D.D. Electronics, 81 Prescott Rd., Raymond, NH 03077-2624; 603-895-0119

Requirements: TI99/4A console, expansion memory, disk system, GRAM device; or Geneve 9640 with GPL

-
3. Load and Run
 4. Run
 5. Run Program File
 6. Rich GKXB
 7. Directory
 8. Memory
 9. Unused
 0. Unused

Gilbertson has incorporated both GKXB and the TI Editor/Assembler cartridge into RGKXB. The Directory option lets you catalog any drive. You can select a program from the listing and, after returning to the menu screen, select GKXB or any of the Editor/Assembler options and automatically have the program loaded.

The Memory option offers options for ROM/RAM, GROM/GRAM and VDP. These are active only if operating from a TI console. They are inactive on a Geneve.

So what's new about RGKXB? Plenty. For one thing, it supports a new device, called PDISK or Phony DISK. This uses the lower 8K of RAM as a disk drive. You can save variable or program files there and not have to remember their names since anything stored in PDISK is called PDISK. Using it with devices that support bank memory switching lets you create a PDISK in each bank, if you are so inclined.

Other new features are the ability to ac-

cess names that have been established as devices. This means that you can access the E/A main menu from simply by typing "EA" within BASIC or Rich Extended BASIC. Or, you may access RXB from the Editor/Assembler or BASIC or another cartridge. For example, to load RXB from Editor/Assembler, simply type "XB" at any E/A device prompt.

The addition of the EXECUTE command provides a faster way than a CALL LINK to execute assembly routines from RXB. Another feature, of use to assembly programmers, is CALL INIT2. This installs an Extended BASIC compatible DSR into the lower 8K of RAM that can be accessed by using a BLWP. It also loads a GPLLNK that can access every GPL support routine that the E/A cartridge can access.

But I suspect it's the Extended BASIC programmers who will get the most out of RGKXB. In many ways, the modifications that Gilbertson has done to many of the commands found in Extended BASIC are the same ones any user would want to do if he had Gilbertson's programming skills. For example, he's made it easier to create multiple sprites by incorporating an autorepeat capability into the SPRITE command. Other commands that have autorepeat capability in RGKXB include CALL KEY, CALL JOYST, CALL COINC, CALL DISTANCE and CALL MAGNIFY.

The Size command reports stack and memory free the same as TIXB but if a CALL INIT or CALL LOAD has been executed or a hidden loader installed, Size will also tell you how much A/L memory is free.

RGKXB also has a new syntax, called combined commands. With this feature, the first two characters of two XBASIC commands are combined to form a single word that represents both commands. This helps to reduce the amount of memory required for commands. An example is CALL JOKE, which combined CALL JOYST with CALL KEY. The manual includes two pages of possible command combinations and another 12 pages detail-

(See Page 22)

RGKXB—

(Continued from Page 21)

ing the use of each.

New commands include the following CALLS:

VERSION — reports RGKXB version number.

BASIC — works with GRAM device when TI BASIC is turned on.

XBASIC — same as RUN "DSK1.LOAD".

EAASS — goes directly to Editor/Assembler load Assembler request.

EAPGM — takes a string from XB and goes to E/A "Run Program File".

EALR — takes a string from XB and goes to E/A "Load and Run".

DUPCOLOR — takes the color of a set and duplicates it into a second set.

DUPCHAR — takes the character definition of a character and duplicates it into a specified character.

SWAPCOLOR — takes the color of a set and swaps it with a second set.

SWAPCHAR — takes the definition of a character and swaps it with another character.

INVERSE — inverts the bytes in the background of a specified character.

HFILL — fills the entire screen with a single character.

VFILL — same as HFILL.

DELAY — used to slow a section of a program from one-tenth of a second to 12 days.

WINDOW — places a string on the screen and wraps and breaks it according to the window's height and width values.

GWINDOW — retrieves into a string variable what is on the screen (the opposite of WINDOW).

HPUT — similar to DISPLAY AT but uses columns 1-32 instead of 3-28.

VPUT — vertical version of HPUT.

HGET — opposite of HPUT, takes a string from the screen and puts it into a

string variable.

VGET — vertical version of HGET.

RDIAG — creates a right diagonal downward.

LDIAG — left version of RDIAG.

GMOTION — a new sprite command. It retrieves the row and column velocities of a sprite and places them in variables. Opposite of CALL MOTION in XB.

RMOTION — a new sprite command that reverses the direction of a sprite.

RGKXB also includes the following new subprograms:

CALL INIT2 — does what CALL INIT does but also adds two subroutines, an XBASIC compatible DSR and a GPLLNK.

CALL INITPDISK — initializes the phony disk.

CALL PDISK — phony disk routine.

CALL MOVE — copies bytes from one memory address to another.

CALL MOVES — copies from one string to another the specified bytes, using VDP, RAM or GRAM memory.

CALL XBPGM — like in RUN in XB but uses quote marks and strings and can be run from a program.

EA DSR — allows access to the E/A section of RGKXB.

CALL EXECUTE — goes directly to a CPU address.

CALL GRAMIT — runs from within a GRAMulator and allows up to 40K assembler routines from the lower 8K used for assembler routine links.

CALL IO — allows access and control of any chip in the console or on peripheral cards.

CALL PROTECT — protects programs and files.

CALL RAMIT — similar to GRAMIT.

CALL RENAME — renames programs or files.

XB DSR — accesses RXB auto selector or auto bypass function.

Also included with the RGKXB is GRAM Kracker Utility 1 by Danny Michael. This set of enhancements is of use to those who have an 80K GRAM Kracker.

DOCUMENTATION: Documentation is on 8 1/2 x 11 sheets and consists of approximately 100 pages. Documentation is also available on disk as an ASCII file. The hardcopy docs I received are three-hole punched, which makes it easy to bind in a folder.

Gilbertson gives programming examples to illustrate every new or modified command available in RGKXB. It's written in plain English, so technophobes needn't worry about doing battle with jargon. Even the appendices, which are generally where computer manuals bog down, is cleanly written and easy to follow. I doubt the printed docs can be improved, they are that good.

EASE OF USE: Learning to use RGKXB isn't going to be a problem for Extended BASIC programmers. In fact, it should make programming easier, perhaps faster. It is 100 percent compatible with TI Extended BASIC. Of course, programs that utilize the features of RGKXB won't run in TI XBASIC.

FINAL GRADE: For the price, this package represents an exceptional bargain. Although it is not currently available in cartridge format, a cartridge is expected in the near future. For those with GRAM devices, however, RGKXB is available today. The current version is V5.50. The two years of effort that Rich Gilbertson has put into perfecting Extended BASIC is evident in all facets of the program itself and the documentation. In fact, he's almost finished with this stage of development. Next he's tackling a complete rewrite of the ROM and the the GROM in XBASIC. I wish him good luck on that project and congratulations on what he's done so far.

'Rag' subscribers pay on delivery

The second issue of The Enterprise Rag has been mailed to a reduced mailing list, according to Bruce Harrison of Harrison Software.

The Enterprise Rag is an aperiodic newsletter for owners of MIDI-Master.

Harrison says that the company asks that recipients pay \$2 per issue after receiving each issue to insure their names being kept on the mailing list for the next issue. Checks should be made out

to Dolores P. Werths. Recipients who have paid in advance for a number of issues may ignore this plea, Harrison says.

Harrison's "Toolbox" of programs to assist MIDI-Master users is available for \$1, shipped on DS/DD disk unless a floppy is requested. The "Toolbox" is public domain software and may be available through user group or BBS sources as well.

For information, contact Harrison Software, 5705 40th Place, Hyattsville, MD 20781.

Boot ROM Bios

EPROM gives quick access to MDOS

By CAL ZANELLA

Following a phone conversation with Don Walden of Cecure Electronics, I packed my Geneve 9640 for shipment via UPS to Muskego, Wisconsin. The Geneve clock battery had finally expired after 4+ years of heavy use and the LED had long ago ceased to function. Being heavily dependent on identifying dated correspondence and reports, and having little patience with resetting the time and date during boot-up, I decided it was time to replace the clock battery and LED.

During the conversation with Don we also discussed another project that Cecure Electronics, in collaboration with James Schroeder, is offering to the Geneve owner — the Boot ROM Bios. Don thoroughly explained the function of the recently released Boot ROM Bios. This is a modification to the Geneve boot EPROM and circuit board that allows the Geneve to carry SYSTEM/SYS onboard. SYSTEM/SYS, being EPROM based, is fast booting and guarantees access to an uncorrupted operating system at any time.

An additional and significant feature of the Boot Rom Bios is the inclusion of a menu that is available during either a cold boot or a warm boot of the operating system. During an initial startup (cold boot) or during a warm boot (CTRL-SHIFT-SHIFT) the user has 2 options. The first option, requiring no intervention on the user's part, is to simply allow the EPROM to boot SYSTEM/SYS which, in turn, will run an AUTOEXEC file to configure the system (more on this later). The second option is simply to hold the keyboard space bar down during the boot process. This action will bring up a menu that lists four additional and very useful options. Here are the menu options:

Select Direct Boot Device

B — Boot Bios

R — RAMdisk

H — Harddrive

F — Diskdrive

As you can see, the menu prompts for a boot path from a list of four devices. The default boot path for harddrive or floppy is set as device No. 1. The default boot path

REVIEW

for a standard Horizon RAMdisk is reportedly configurable with the use of Jim Schroeder's Form utility. The default boot path for a Horizon RAMdisk with the Phoenix modification is set as device No. 6 or No. 7, depending on the CRU location of the card. Simply press the keyboard letter representing the desired choice and SYSTEM/SYS will boot from that device, assuming that the device is present on your system and SYSTEM/SYS is located on the selected device. Should you select a device that is not present on your system, then all devices will be searched sequentially in the above order until a copy of SYSTEM/SYS is located. In the event that SYSTEM/SYS is not located on any other device, the Geneve will boot, because SYSTEM/SYS is permanently burned into the EPROM (B — Boot Bios). The menu program is also burned into the EPROM and will always be available. Yet another method of booting, which bypasses the menu, is to hold down the keyboard letter (B, R, H, F) of the desired boot path while performing a reboot.

For me, the most obvious advantage to the menu is the ability to quickly and effortlessly reboot any other version of MDOS from any device. I have three versions of MDOS available for various applications that are version dependent. Considering that my Geneve system consists of a Myarc HFDC card, Myarc FDC card and a Horizon Phoenix card, I have much flexibility as to how I might configure the system during bootup.

Another interesting note is a time delay programmed into the EPROM. During a cold boot a 15-second time delay occurs prior to boot to allow a harddrive to reach operating speed. Another interesting thing that is noticeable is the absence of the famous Myarc Swan! Tight coding on the 128K boot EPROM evidently necessitated the removal of the proud bird.

Should you decide to have Cecure perform this modification on your Geneve,

there are a few things that need to be considered. A few well planned choices need to be made prior to the modification. First is the decision as to which version of MDOS you want to be burned into the EPROM. Once burned into the EPROM, the MDOS version is somewhat semi-permanent and can only be changed by returning the EPROM to Cecure for installation of another version. And, YES, this modification is copyrighted by the authors.

The second decision should be based on a choice of path for the loading of the AUTOEXEC file. The path is coded in SYSTEM/SYS and will be burned into the EPROM. For my own personal system, I had Don edit the SYSTEM/SYS file to load the AUTOEXEC file at DSK6, my Horizon Phoenix.

Finally, if you have a RAMdisk in your system you should specify which type and size it is, i.e. 8-bit, 16-bit or Phoenix as Don will need this information should you desire to have your AUTOEXEC file located and loaded from your particular RAMdisk.

I have been using my modified Geneve now for only one week and am thoroughly pleased with the performance of this mod. It appears to be a foolproof and trouble free modification. The workmanship of Cecure Electronics is clean and very professional appearing. I have no reservations about recommending this modification to all Geneve owners.

Cecure is offering this modification for \$45, plus shipping and handling charges.

Another interesting hardware project in the works at Cecure is a similar hardware modification to the Boot Rom Bios setup that will allow the user to install any version of SYSTEM/SYS into a Flash EPROM located onboard the Geneve from any device. Stay tuned for further information. (*This modification became available in June. See the June Newsbytes for more information.—Ed.*)

For product information contact: Cecure Electronics, Inc.; P.O. Box 132; Muskego, Wisconsin 53150-0132; (414) 679-4343.

MICRO-REVIEWS

Funnelweb V5 40-Column Editor and Collecting Cartridges

By CHARLES GOOD

On rather short notice I have been asked by MICROpendium to take over the Micro Reviews column and have agreed to give it a try, so let me introduce myself.

Many of you already know of me from the articles I have written for the Lima newsletter and my efforts in organizing the Lima TI Conferences. I am 47 and have 3 children still living at home (who will help me with game and education software reviews). I teach botany and biology at Ohio State University and am an expert on certain types of fossil plants.

Does any of this qualify me as a 99/4A computer expert? Certainly not! Like most of MICROpendium's readers I am a mere mortal 99/4A user, not a programmer or hardware developer. Like many of the rest of you I purchased my first /4A in 1983 and learned about computers by studying the books that came with the console and typing in program listings published in 99er Magazine. I now use my 99/4A for almost all my home and professional computing jobs including manuscript preparation, letter writing, creating exams for my students from a multiple-choice question data base, personal finance, mail lists, and games. I have not written any fairware or commercial software and have no affiliation (other than "customer") with any of the companies or dealers that serve the TI community. I am thus an appropriately neutral reviewer.

Please loan me your 99/4A software, books, videotapes, and hardware directly to me for review. I am not interested in acquiring a free collection of commercial products so it will be my policy to return commercial products after reviewing them. If I keep anything I will pay for it. My system includes an 80-column device.

Please note that I do not own a Geneve and cannot evaluate Geneve specific products. (*Send Geneve products to MICROpendium for review.—Ed.*) Send your products for review to me at P.O. Box 647, Venedocia, Ohio 45894. My evening phone is 419-667-3131.

FUNNELWEB V5 40-COLUMN EDITOR

The Funnelweb system is frequently updated as new features are added and problems are fixed. Unlike commercial software where users have to pay in advance for updates, Funnelweb users can try out the shareware updates for free. What I am describing is the second release of the v5 40-column editor released in mid June 1993. This supersedes the initial partial release of the v5 40-column editor at the May Lima MUG Conference. There is also a June update of the 80-column v5 editor. These v5 40- and 80-column editors are designed to run from the Funnelweb v4.4 environment. Put the v5 ED and EE files and any desired character set files on your Funnelweb v4.4 system disk. The 80-column v5 editor has already been reviewed in Micro Reviews, with the reviewer stating that it is the only TI word processor he considered superior to MY-Word on a Geneve. Now 40-column users have access to the same neat new features.

The Funnelweb v5 editors have 3 features that are, to the best of my knowledge, not found in any other TI word processor (RAG Writer, TI Writer, First Draft, Writerease, Harrison).

1 — Complete multilingual capability. By holding down the space bar as the editor is loading into the computer's memory you are given your choice of the following languages; English, French, German, Dutch, Swedish, and Spanish. Selecting a language gives a command line written in the selected language with appropriate commands. Print File in French is "Imprimer Fiches", and the equivalent French is "IF". An appropriate foreign character set is loaded which allows you to display on screen and print (using your printer's foreign language character sets) accented vowels and other symbols specific to the specified language. There are several alternate ways of displaying and printing these foreign characters.

2 — ALL CHARS graphics. If you look in your printer manual you will discover a wide variety of graphic shapes coded with ASCII numbers >127. These include foreign language characters, mathematical symbols, and line graphics. The lines are useful for drawing pictures, designing business forms, and creating borders. They resemble the "lines" created by the commercial software Page Pro or Form Shop. Normally these >127 graphics built into almost all printers are not accessible directly from a 99/4A or Geneve keyboard, but with the v5 editor they are. CTRL/comma lets you switch back and forth between the normal ASCII <127 letters and ASCII >127 graphics. Everything is properly displayed on screen (WYSIWYG) and can be printed properly directly from the editor with PF (Print File). You don't need the formatter to print these graphics! As the car commercial says, "Now this changes everything." A whole new world of TI graphics has been opened up.

3 — Both the June 40- and 80-column v5 editors have enhanced hard drive support. From a ShowDirectory display, if you move the cursor next to the name of a sub directory you can display the file names of that sub directory. Also, you can bring up a display of the root directory of the currently displayed directory from within ShowDirectory. This ability to display the entire hard drive tree structure is, I believe, not found in any other disk manager available to the TI community. (I don't have a TI hard drive, so I can't try this feature out for myself. The description above is based on the v5 editor's documentation.)

Among the other nice features not found in previous versions of the Funnelweb editors are help screens displayed from the command line, wild card capability with Find/Replace String, the ability to scroll while in the command line, and the ability to goto a particular line number by typing that number and Enter in command mode. I give the Funnelweb v5 editors my very highest recommendation.

(See Page 25)

MICRO-REVIEWS—

(Continued from Page 24)

To obtain either or both of the v5 editors put \$1 (\$2 for both) in an envelope carefully wrapped in a piece of paper with your return address written on it and send it to me at the address above. If you decide to use this software, you owe the authors a fair-ware donation over and above what you have already paid for other parts of the Funnelweb system.

COLLECTING CARTRIDGES

If you are interested in computer history and perhaps enjoyed reading *The Orphan Chronicles* when it came out a few years ago then you will like this book length manuscript by Bill Gaskill. Bill has one of the largest personal collections of TI cartridge software. What began as a newsletter article has now been expanded to over 100 printed pages, with additional material planned for future revisions.

This scholarly work is based on Bill's personal cartridge collection plus a thorough study of computer magazines (Info World, Byte, Compute, 99er, Enthusiast 99, etc), third party and TI catalogs, and internal TI documents.

Bill has found references to over 360 99/4A cartridges, of which he has actually seen 275 in cartridge format. The rest are known only from gram disk files or are products announced but never released. List price on all the 99/4A cartridges that were released exceeds \$11,000! For each cartridge Bill includes information, where known, on software author, manufacturer's product number, list price, release date, along with a physical description of the cartridge and its original packaging and documentation.

The table of contents includes: Cartridge History, Collecting Guidelines, Cartridge Listings and Descriptions, Cartridge

Names Sorted by Manufacturer, Cartridge Program References (from published literature), Cartridge Trivia, Command Module Simulator, Cartridge Newsbytes, Milton Bradley MBX System, Romox ECPCs and Software Centers, and *The Other Cartridge Using Computers*. Under this last heading Bill will in future revisions include descriptions of cartridges for other 1980s computers and game machines by Atari, Coleco, Commodore, Gemini, and Mattel.

I recommend this manuscript to anyone wanting a single reference describing the cartridge software available for the 99/4A (much of which is still available from dealers) and anyone interested in computer history. *Collecting Cartridges* is packed with neat historical trivia. A check for \$15 sent to Bill Gaskill at 2310 Cypress Court, Grand Junction Colorado 81506 will get you a printed copy of the latest revision.

BUGS AND BYTES

(Continued from Page 20)

ture are being worked on. Believe me, we are taking things slow in this area to ensure fully functional, stable operations for hard disks on the TI and Geneve.

Don't give up hope yet. There are thousands waiting for SCSI to be done, and we could all use the peripheral today if it was ready. SCSI was only conceived of 15 months ago (April 1992) and to be this far along on a product is very good considering other industries' turnaround times of 24-30 months, and they have many more people working for them.

Fracas in Dallas

A livelier-than-usual March meeting of the executive committee of the Dallas TI Users Group was reported in the July issue of the Dallas 99 Interface. The executive committee had been meeting at the Walnut Hill franchise of Mr. Gatti's, a restaurant specializing in pizza. The manager told Mattie Bush, president of the group, that the committee could not continue to meet there unless at least 10 members ate from the buffet afterwards, the newsletter says.

Then Louis and Jackie Guion, BBS sysops for the club, came in with their refill mugs for Mr. Gatti's. The manager told them, says the report (written by Jackie Guion) that refills were 99 cents. Louis complained to the manager (and later to the manager's husband, also wearing a Mr. Gatti's uniform) that the mugs had been

sold with the guarantee of perpetual refills.

Getting no satisfaction, Louis headed for the meeting room, where the stunned group had just been told by the manager that they were no longer welcome at Mr. Gatti's because "that man at the counter had an attitude."

However, after a letter to the Kerrville, Texas, corporate headquarters, the Guions got a call and were informed that all refills, indeed, were free, and that this had been discussed extensively with the manager of the Walnut Hill franchise.

Jackie Guion writes, "What we do want you to know is that at least one company still believes in keeping a promise it made to customers at least four years ago when we first purchased our mugs."

After all these years ...

Looking at yet another newsletter, we find Frank Aylstock of the Bra Users Group reporting that he has been notified that he will receive a refund from Electronic Systems Design for the hard and floppy disc controller he ordered. At one time the production date for the HFDC was given as November 1990.

Aylstock received the notice from Loris Hyman Mosby, consumer specialist for the Office of the Attorney General, Consumer Protection Division, in Baltimore, Maryland, after he contacted that office.

Attend a TI fair this year

USER NOTES

Catalog and Run

This comes from Dean Mah, of Red Deer, Alberta. He writes:

To keep somewhat organized, I keep all my XB programs separate from other files. To save time I wanted to list all the files and be able to run one that was picked. What was needed was a menu program.

However, often the contents of these disks changed as programs were added, deleted or moved somewhere else. So a static menu was a hassle to keep updating every time the disk contents changed. What was really needed was a program that would read the disk catalog and allow me to select a file to be run. That's what the following program does.

The program reads the disk catalog for any program or Int/Var 254 files and creates a menu out of them. You then choose the letter that corresponds to the program that you wish to run, and that's it.

A word of warning, this program fines any program or I/V254 file and an error will occur if these are not XB programs. Trying to run an assembly program or data file will result in an error.

```
100 DIM A$(26):: B$="DSK1."
:: CALL INIT !096
110 OPEN #1:"DSK1.", INPUT ,R
```

```
ELATIVE, INTERNAL :: INPUT #1
:A$(0),B,B,B !086
120 CALL CLEAR :: A,D=0 !066
130 INPUT #1:A$(A),B,C,C !050
140 IF LEN(A$(A))=0 THEN A=A-1 :: IF A<1 THEN STOP ELSE
B,D=1 :: GOTO 170 !113
150 IF ABS(B)=5 OR (ABS(B)=4 AND C=254) THEN A=A+1 !227
160 IF A<26 THEN 130 ELSE B=1 !044
170 DISPLAY AT(B+5,1):CHR$(B+64);"->";A$(B-1):: B=B+1 !199
180 IF B<14 AND B<A THEN 170 ELSE IF A>13 THEN B=1 ELSE
200 !196
190 DISPLAY AT(B+5,15):CHR$(B+77);"->";A$(B+12):: B=B+1
:: IF B+12<A THEN 190 !160
200 PRINT " PRESS SPACE BAR FOR ";:: IF A=26 THEN PRINT "MORE" ELSE PRINT "XB" !130
210 CALL KEY(3,K,S):: IF S<1 THEN 210 !027
220 IF K=32 THEN IF D=1 THEN CLOSE #1 :: STOP ELSE 120 !015
230 IF K<65 OR K>A+64 THEN 210 ELSE CLOSE #1 !033
```

```
240 B$=B$&A$(K-65):: CALL CLEAR :: DISPLAY AT(8,6):"TRYING TO LOAD:" :TAB(6);B$ !185
250 B$=CHR$(169)&CHR$(199)&CHR$(LEN(B$))&B$&CHR$(0)!240
260 CALL PEEK(-31952,A,B):: CALL PEEK(A*256+B-65534,A,B):: C=A*256+B-65534 :: CALL LOAD(C,LEN(B$))!128
270 FOR A=1 TO LEN(B$):: CALL LOAD(C+A-3,ASC(SEG$(B$,A,1))):: NEXT A !250
280 ! * RUN "DSK1." * !209
```

Keeping track of mileage

This program was written by Bob August and appeared in the BUG News, the newsletter of the Brea (California) 99ers. The program is written in Extended BASIC and requires a disk drive, though it can be modified for use with a cassette.

The program records the date, gallons used and miles driven and calculates the mileage every time you buy gas. The data is written to disk and is recalled every time you use the program.

```
100 ! MILEAGE !151
110 ! IN TI EXTENDED BASIC !
234
120 ! BY R.W. AUGUST !092
130 CALL CLEAR :: N=0 !088
140 DIM D$(100),D(100),G(100),M(100)!007
150 DISPLAY AT(2,5)ERASE ALL : "<< M I L E A G E >>" !253
160 DISPLAY AT(6,2):"COMMAND LIST": " _____"
170 DISPLAY AT(9,2):"1) ENTER DATA FROM KEYBOARD": :TAB(2);"2) DISPLAY MILEAGE DATA": :TAB(2);"3) PRINT MILEAGE DATA" !049
180 DISPLAY AT(15,2):"4) READ DATA FROM DISK": :TAB(2);"5) WRITE DATA TO DISK": :TAB(2);"6) END PROGRAM" !117
190 DISPLAY AT(22,5):"YOUR CHOICE [1 - 6]" !051
200 CALL KEY(0,K,S):: IF S=0 THEN 200 :: IF K<49 OR K>54 THEN 200 !140
```

Cecure sees CForm hard disk formatter as predecessor of tape back software

Cecure Electronics has a new product, CForm, a hard disk formatter which runs out of MDOS, by Mike Maksimik.

According to Maksimik, CForm is written in C and assembly language and allows the user to format a hard disk through MDOS rather than using MDM5. He says it boosts the disk capacity on bigger drives by allowing the user to put more sectors on disks.

CForm is the predecessor to the company's planned tape backup software, Maksimik notes. He says he has been working to develop Quick 40 and Quick 80 tape backup software. CForm uses a lot of the same code, he explains.

CForm sells for \$15 and each copy is licensed to the purchaser.

He says the tape software will "remove a lot of hurdles to the management of hard drives," and notes that it could also be used for large collections of floppies, as in, for instance, a users group library.

Maksimik says he has recently completed testing the tape software with different tape drives and it is now undergoing beta testing. He says the first release of the software will be for MDOS, with a TI version later. The first release will be a fairware release through 9640 News, P.O. Box 752465, Memphis, TN 38175-2465, (901) 368-1169. Subsequent enhancements will be made through Cecure Electronics, P.O. Box 132, Muskego, WI 53150, (414) 679-4343.

(See Page 27)

USER NOTES

(Continued from Page 26)

```
210 CALL CLEAR :: ON K-48 GO
SUB 1000,2000,3000,4000,5000
,6000 !067
220 GOTO 150 !229
```

```
1000 ! ENTER FROM KEYBOARD !
230
1010 DISPLAY AT(2,3):"<< DAT
A ENTRY SECTION >>" !114
1020 N=N+1 :: DISPLAY AT(5,1
```

```
):"ENTER END FOR MAIN MENU"
!234
1030 DISPLAY AT(8,1):"DATA:"
:::"ODOMETER READING:"::
:"NO. GALLONS BOUGHT:" !067
1040 ACCEPT AT(9,1)SIZE(8):D
$(N):: IF D$(N)="END" OR D$(
N)="end" THEN N=N-1 :: RETUR
N !213
1050 ACCEPT AT(12,1)SIZE(7):
D(N):: ACCEPT AT(15,1)SIZE(4
):G(N)!083
1060 DISPLAY AT(12,1):" " ::
DISPLAY AT(15,1):" " :: GOT
O 1020 !065
2000 !DISPLAY DATA TO SCREEN
!118
2010 DISPLAY AT(2,2):"<< DIS
PLAY DATA SECTION >>": "DAT
E";TAB(10);"ODOMETER";TAB(19
);"GALS";TAB(25);"MPG" !027
2020 DISPLAY AT(5,1):"——
—— ———" :: M
(1)=0 !244
2030 FOR J=2 TO N :: IF G(J)
>0 THEN M(J)=(D(J)-D(J-1))/G
(J)ELSE M(J)=0 !156
2040 IF M(J)<0 THEN M(J)=0 !
171
2050 NEXT J !224
2060 K=-16 :: L=0 !129
2070 K=K+17 :: L=L+17 :: IF
K>N THEN L=N !121
2080 FOR J=K TO L :: DISPLAY
AT(5+J,1):D$(J):: DISPLAY A
T(5+J,11):USING "#####.#":D(
J)!112
2090 DISPLAY AT(5+J,19):USIN
G "##.#":G(J):: DISPLAY AT(5
+J,24):USING "##.##":M(J)::
NEXT J !086
2100 IF L<N THEN 2120 !166
2110 DISPLAY AT(24,1):"PRESS
ANY KEY FOR MAIN MENU" :: G
OSUB 2130 :: RETURN !2322120
DISPLAY AT(24,1):"PRESS ANY
KEY TO CONTINUE" :: GOSUB 2
130 :: GOTO 2070 !114
2130 CALL KEY(0,K,S):: IF S=
0 THEN 2130 :: RETURN !168
3000 ! SEND DATA TO PRINTER
!014
3010 DISPLAY AT(2,3):"<< PRI
NT DATA SECTION >>" :: M(1)=
(See Page 28)
```

SYSTEM OF THE MONTH

8 floppies and 3 hard drives make this system unique

Ralph E. Rees, of Phoenix, Arizona, has collected a large amount of hardware, which he has spread out over several systems, including a home-built portable. His main, desk-top system, consists of the following:

TI black/silver console with 14Mhz crystal and load interrupt, hold and reset switches; CorComp disk controller, Myarc Hard & Floppy Disk Controller, Horizon 512K RAMdisk with Rambo, Horizon 384K RAMdisk, Myarc 512K RAMdisk, CorComp Triple Tech, TIRS232 card, 4 Teac 360K floppy drives connected to the CorComp controller, 4 Teac 720K drives connected to the Myarc controller, 1 Seagate 44 meg. hard drive, 2 Seagate 20 meg. hard drives, 80K GRAM Kracker, Microcom 2400 baud modem, Panasonic 10911 printer, Micro-Stuffer 64K print buffer, Radio Shack 4.5" 4-color plotter, Panasonic Pen-writer (4-color plotter, typewriter, printer), Axiom PIO interfaced connected to plotters, Rave keyboard interface, Zeneth RGB/composite color monitor, Craig 7 band equalizer to tailor audio output, Milton Bradley MBX system with all modules, Marantz three-head, variable speed cassette recorder, Super Sketch drawing pad, WICO joystick adapter, Accuball trackball (modified), Micromos joystick (modified) and 3" cooling fan for console.



Rees' second system is used by his son and consists of a beige TI99/4A console, CorComp disk controller, 3 Rave 544K memory cards, Horizon 384K RAMdisk, Mecha-tronics 128K GRAM-Karte, CorComp Triple Tech card, 2 Teach 360K drives, 1 Tandon 360K drive, Amdek 300 13-inch composite color monitor, WICO joystick adapter and a WICO "red-ball" joystick.

His third system is a portable and consists of a black and silver TI99/4A console (modified as the console of his main system), CorComp Micro-Expansion System, Tandon 36K disk drive, home-made remote keyboard. All of the above is housed in a piece of Airway luggage. The system is connected to an NEC 13-inch composite color monitor.

Show in the photo is a portable system he is building. This system includes a TI console motherboard, CorComp Micro-Expansion System, 2 Teac 360K disk drives, a 9-inch monochrome monitor and a home-made keyboard.

Other equipment in his collection include a Star NX-1000 printer, TI Impact Printer, TI color monitor, Horizon 90K RAMdisk, Rave speech card, and 5 additional complete Peripheral Expansion Boxes. He also has 105 different cartridges and 100+ duplicates.

Share your system with other TIers. Send your system descriptions and photos to MICROpendium System of the Month, P.O. Box 1343, Round Rock, TX 78680.

USER NOTES

(Continued from Page 27)

```

0 !090
3020 FOR J=2 TO N :: IF G(J)
>0 THEN M(J)=(D(J)-D(J-1))/G
(J) ELSE M(J)=0 !156
3030 IF M(J)<0 THEN M(J)=0 !
171
3040 NEXT J !224
3050 DISPLAY AT(12,8):"PRINT
ING DATA" :: OPEN #2:"PIO",O
UTPUT !056
3060 PRINT #2:"DATE";TAB(12)
;"ODOMETER";TAB(23);"GALLONS
";TAB(33);"MPG" !235
3070 PRINT #2:"-----
-----" !189
3080 FOR J=1 TO N !141
3090 PRINT #2:D$(J);TAB(13);
!103
3100 PRINT #2,USING "#####.#
":D(J);!102
3110 PRINT #2:TAB(24);!150
3120 PRINT #2,USING "##.#":G
(J)!073
3130 PRINT #2:TAB(32);!149
3140 PRINT #2,USING "##.##":
M(J)!115
3150 NEXT J !224
3160 DISPLAY AT(16,9):"DATA
PRINTED" :: FOR I=1 TO 200 :
: NEXT I !059
3170 CLOSE #2 :: RETURN !162
4000 ! LOAD DATA FROM DISK !
156
4010 DISPLAY AT(2,3):"<< REA
D DATA SECTION >>" :: N=0 !1
30
4020 DISPLAY AT(12,4):"READI
NG DATA FROM DISK" !065
4030 OPEN #1:"DSK1.MDATA", IN
PUT :: INPUT #1:N !023
4040 FOR I=1 TO N :: INPUT #
1:D$(I),D(I),G(I):: NEXT I !
040
4050 CLOSE #1 :: DISPLAY AT(
16,9):"DATA LOADED" :: FOR I
=1 TO 200 :: NEXT I !2304060
RETURN !136
5000 ! WRITE DATA TO DISK !1
18
5010 DISPLAY AT(2,3):"<< SAV
E DATA SECTION >>" !014
5020 DISPLAY AT(12,9):"SAVIN
G DATA" !106
5030 OPEN #1:"DSK1.MDATA",OU

```

```

TPUT :: PRINT #1:N !134
5040 FOR I=1 TO N :: PRINT #
1:D$(I):D(I):G(I):: NEXT I !
054
5050 CLOSE #1 :: DISPLAY AT(
16,10):"DATA SAVED" :: FOR I
=1 TO 200 :: NEXT I !2165060
RETURN !136
6000 ! END PROGRAM !178
6010 DISPLAY AT(12,3):"HAVE
YOU SAVED YOUR DATA" : :TAB(1
0):"Yes or No" !126
6020 CALL KEY(0,K,S):: IF S=
0 THEN 6020 :: IF K=78 OR K=
110 THEN 150 !086
6030 IF K<>89 AND K<>121 THE
N 6020 ELSE CALL CLEAR :: EN
D !224

```

Simple circuit for program controlled switch

This comes from Alfred Slovak, of Vienna, Austria. He writes:

Referring to the May 1993 issue ('White elephant' console turns into home security system), I want to present the TI/Geneve community with a simple circuit so that the computer can be used as a program-controlled switch. I found this years ago in a German computer monthly, and its origin is from an article about the Sinclair ZX-81 computer. But it should work on any computer that can create black and write blocks on the screen.

The biggest advantage is that there are no operations inside the computer. This is helpful for people like me who are afraid to destroy something with a soldering iron. The principle is as follows:

Two light depending resistors are fixed on the screen by means of adhesive tape. Create two blocks of characters, one block

and one white, and place them just under the LDRs. The two LDRs form a voltage divider, which makes a transistor "open" or "close." The current through the transistor can be used to switch a relay or any other device. Depending on the type of the transistor and relay, every desired load can be switched by changing the colors of the blocks under the LDRs from black to white, and vice-versa (see diagram).

I used several of these circuits to switch the lights off and on in my father's house, to make the house look occupied while he was on vacation. It worked without any problem for three weeks, until my father returned.

Accessing modules

The following item is by Rich Gilbertson, of Portland, Oregon. He writes:

RXB(Rich Extended BASIC) and REA (Rich Editor/Assembler) have a access method you can place into any modules you have. Look in you GRAM device manual for Headers.

Okay, got it in front of you? Well use

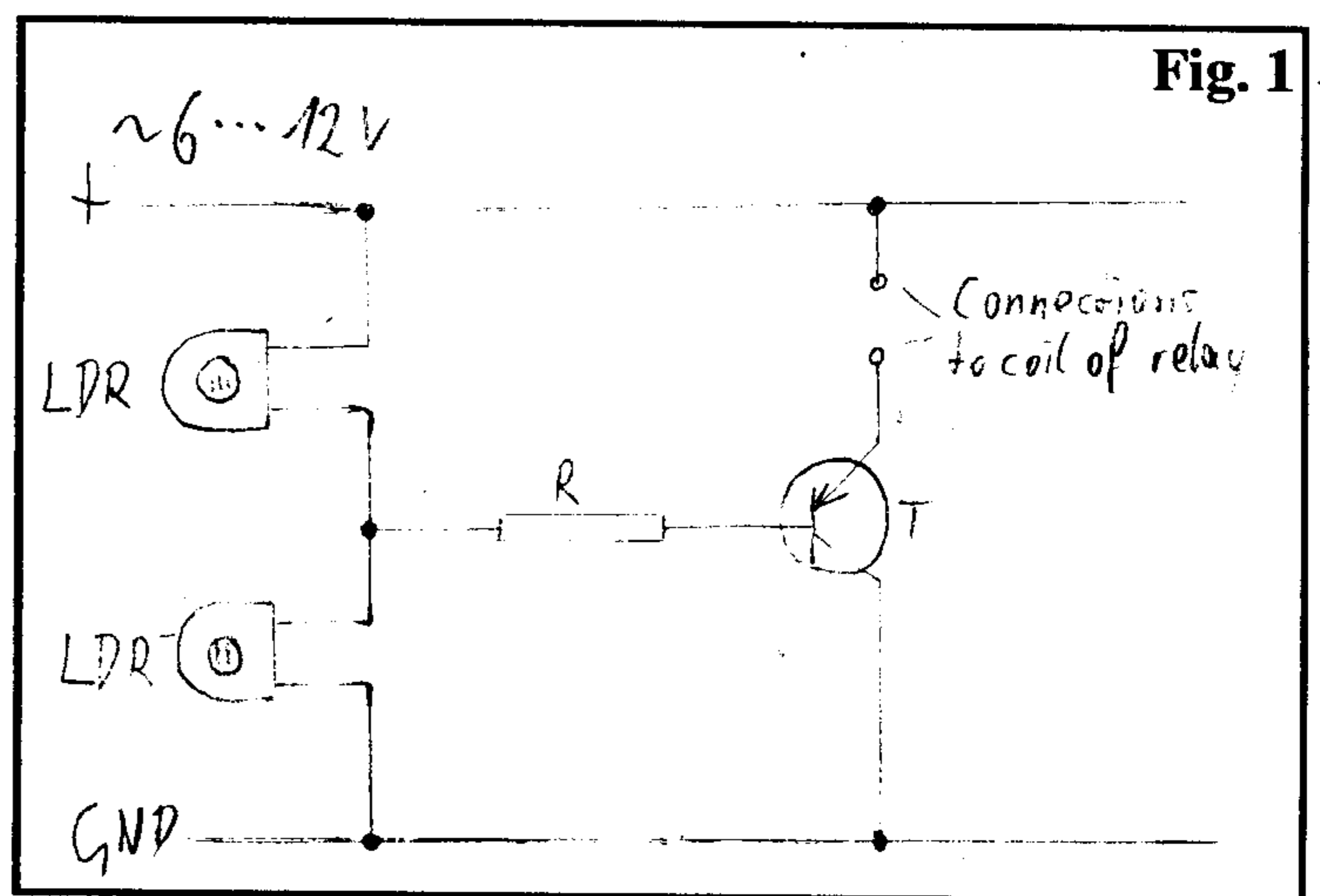


Fig. 1

your MEMEDITOR of your GRAM device to look at the Header of RXB (>6000) or REA (>E000) and you will see at >6008 or >E008 a address. Now go to that address and you will find a name of a device. In RXB it is XB and in REA it is EA. These are DSR headers. They work like this:

At >6008 is the pointer to a DSR header. At that address is the pointer to the next DSR header or if >0000 it means no more pointers. (Or no more DSR headers).

The two bytes after the pointer of the (See Page 29)

USER NOTES

(Continued from Page 28)

DSR header is the real address of where it is supposed to go when called. See the chart for an example.

Address	bytes there	what it does
>600	>6098	This points to a menu header.
>6008	>6372	This points to a DSR header.
>6098	>0000	This points to no more headers.
>609A	>6476	This is where the program starts.
>609C	"MODULE NAME"	This is the modules name you see.
>6372	>6380	This points to next DSR header.
>6374	>6687	This is where the DSR starts.
>6376	"DSR NAME"	This is the DSR access name.

If you follow me so far then just make sure you change any autostart modules to normal ones first. Follow the manual on autostart modules and how to fix them.

A simple example is to use the Car Wars game that loads into GRAM 3 (>6000). Load it up into the alternate bank of a GRAM bank while RXB/REA is in the other GRAM bank. Now go to the Memory Editor for your GRAM device and look at GRAM address >6006 that has a >6010 there. Now just put a >6010 at GRAM address >6008. Now both the Menu and the DSR headers are the same and same address.

If you leave the module and go to BASIC you can now go to Car Wars with DELETE "CARWARS" or OLD CARWARS, or use a program line like 100 OPEN #1:"CARWARS",OUTPUT. All of these will transfer you to Car Wars.

The reason I chose this module is that the name is 7 characters long, and TI allows only 7 characters to a DSR name. Also, as the Menu and DSR headers are the same, you can get a better idea of why it works. But only if you read the bloody manual of the GRAM device. So now you can do it to all modules!

You can also switch modules at will (remember the names). Blow away your friends with this one.

Another solution for Peg Solitaire

This comes from Oliver D. Hebert, of Brewton, Alabama. He writes:

The seven-move solution to the Chefred Pyramid given by Barry Traver is the same solution given in other puzzle books. While using Barry's program (see October, 1992

and February, 1993 MICROpendium), I happened across this six-move solution:

53-55-35, 44-42, 23-43, 35-33-53, 63-43, 42-44.

Encouraged, I kept

trying, and came up with this five-move solution:

44-42, 23-43, 53-55-35-33-53, 63-43, 42-44.

I haven't been able to improve on any of the other puzzle solutions.

Be careful with jumpering of regulators

This comes from Mike Doane, of Tucson, Arizona. He writes:

It has become alarming to hear of the number of Texas Instruments users whom are creating problems with one specific hardware update.

I am speaking of the conversion of PEB power supplies. I am a strong advocate of changing the internal power supplies to an IBM-style of transformer. It is an improvement which is long overdue and I am glad to see it advanced.

However, there is one major fact which has been overlooked in every article I have read. I have also noticed the fact that almost everyone who has converted a PEB claims no appreciable reduction of heat in their cards. The problem is created by the modification they have done.

The regulators on the cards themselves are still being activated! They are being "back-fed" through the output leg by the cards themselves. This, in effect, tells the regulators they are still active.

The purpose of a regulator is to restrict

voltage to a specific level and dissipate the excess voltage in the form of heat transfer. In the normal state of things, the unregulated current enters the input leg, is regulated and exits the output leg to supply the card with constant, regulated voltage.

What is done, in essence, with the "jumpering" (connecting a small piece of wire between the input and output legs) is to tell the regulator it is being activated (through the input leg) and then telling it. However, it is not working through the output leg.

Regulators are not very smart. They are designed to work by discerning the difference between power-in and power-out. We, in effect, lobotomize their "thinking" process with the jumpering process. They are not designed to be activated without a specific amount of voltage supplied to the input leg. This is usually a figure of approximately one and one-half volts and above its voltage rating. In the case of a +5 volt regulator, this figure would be 6 to 6½ volts. The card then sheds the excess voltage and supplies the needed voltage.

We are creating a "mental" breakdown by introducing power to both legs. The regulator is still active, even though it is jumpered. It is still attempting to regulate the power and is creating heat.

I speak from experience on this. Tom Wills, Jack Mathis and I managed to actually explode a 7805 regulator and blow a quarter-inch hole completely through a Myarc HFDC. The cause was from jumpering the regulator. It was an expensive and frustrating lesson.

The fix for this is simple. Remove the regulators and *then* jumper across the input/output legs. Why would you want them on there anyway? If the power supply you have installed surges, your card is toast anyway. Since you jumpered the regulator, the power is going right past it. Don't tell me it is so you can simply remove the jumper and install it in a non-altered box. You are better off never swapping your system cards between different systems (between a standard TI PEB and an "updated" one). I don't care how many warning labels you put on the card. You are asking for trouble.

I recommend that all your PEBs be con-
(See Page 30)

USER NOTES

(Continued from Page 29)

verted to the IBM-style power supplies. The cost is so low and the price of repairs to your cards is so high you really can't afford not to. A power supply costs \$30 while a typical charge to repair a card is \$50.

As usual, any and all modifications/repairs to your system and its components are done with the full assumption of the owner's risk.

Color changes in Wordcount

This comes from Bill Gaskill, of Grand Junction, Colorado. He writes:

Here is a tip on how to permanently change the color scheme of Jim Jagielski's Wordcount program (November 1986), which by default is white on black. First, copy the program to a newly initialized disk so that only Wordcount is on the disk. To change the program to white on blue, load Advanced Diagnostics and edit sector >34 byte 242. Change F1 to F\$. With DiskU,

edit sector >22 byte 242 and make the same change, F1 to F4.

Other color schemes may also be achieved by changing F1. For example:

F1 to 13 will display black on light green.

F1 to 17 will display black on cyan.

F1 to F6 will display white on dark red.

The various combinations available can be determined by referring to page 43 of the June 1985 MICROpendium. On that page Tim MacEachern shows how to modify TI-Writer colors. The same rules apply to Wordcount, once you know where the byte to be change is located.

More circle routines for The Missing Link

This comes from Jim Leshner, of Dallas, Texas. He writes:

These programs can be used only with The Missing Link program. This time, at the third screen prompt, select 1 to utilize the 16-color option. In the previous article we used 2 colors, this time we will use 3. In

addition, we will be discovering more interesting things that can be done with circles and parts of circles.

In the first program, we have a series of circles similar to the previous installment, only this time the circles are much larger than the screen. Of course, only a part of the each circle is visible. If you want to experiment with the colors, change the numbers 16,13 in line 140 for color option 1 on the third screen and 16,07 for color option 1. And for the size of the circle change, 240 in line 140 to a larger number and the 360 in line 130.

```
100 ! CIR 4
```

```
110 CALL LINK("CLEAR")
120 CALL LINK("COLOR",16,07)
130 FOR A =1 TO 360 STEP 6
140 CALL LINK("CIRCLE",96,240,A,00,16,13)
150 NEXT A
160 GOTO 110
```

The following program is a bit different. On page 18 of the TML manual you will see a circle marked off in 8 sections, with some numbers in each section. These are suppression codes. Which means that if you plug in any one of the numbers where the 15 is in line 140, you will suppress or delete that section from the circle. If you want more than one section to be suppressed, add those numbers together and replace the 15 with them.

For example, the 15 represents the sum of 1+2+4+8. When the program is run those sections will be missing from the circle, which makes it look like a bow tie. As recommended last month, use graph paper to plot your pictures and geometrics. The graph paper will become more useful when we get into squares, rectangles, cubes and cones.

```
100 ! CIR3
110 CALL LINK("CLEAR")
120 CALL LINK("COLOR",16,07)
130 FOR A=1 TO 160 STEP 6
140 CALL LINK("CIRCLE",96,120,A,15,16,14)
150 NEXT A
160 GOTO 110
```

If you would like a copy of pre-numbered graph paper or have any questions about TML, contact Jim Leshner at 722 Huntley, Dallas, TX 75214; 214-821-9274.

Ready-to-use character definitions

This item, by Rick Kellog, appeared in the MICRO Newsletter of Bloomington, Illinois.

Here are some ready-made character definitions that can be used in any BASIC or Extended BASIC program. The CAI CHAR statements use zero (48) as the designated character, but any character may be substituted.

Slashed zero

(See Page 31)

GREAT SOFTWARE!

TI ARTIST PLUS! The undisputed champion in graphics design! Features 6 dynamic design modules that allow you to create, edit, transform, scale, print and present dazzling graphics. Animated sequences can even be created using a unique movie editor! Point and shoot menu system makes this must-have 3 disk package a breeze to use.

TI BASE The only choice for TI database management! Supports 5 active databases of 16129 records each, with 17 fields per record, and 255 characters per field. Features a procedural command language consisting of over 50 dBASE-like commands, full database manipulation and on-the-fly restructuring, formatted display and printing, plus more!

TI Artist Plus! and TI Base require 32K, disk system, and either an XB, MM or EA cartridge. Geneve compatible in GPL mode only.

GREAT DEALS!

For a limited time you can purchase TI Artist Plus! or TI Base for only \$19.95 each (regularly \$24.95 each), or get both #1 best sellers for just \$32.95 (a \$49.90 value). Add \$3.50 and we'll throw in 2 disks packed with fonts and graphics designed exclusively for TI Artist Plus!.

TEXAMENTS

Serving the TI Community Since 1984

53 Center Street, Patchogue, New York 11772

Don't forget to include \$3.25 for domestic first class and Canadian delivery, \$8.50 for foreign air mail delivery. Send \$2.00 for a copy of our full-line catalog #E11. Orders usually shipped with 48 hours.

USER NOTES

CLASSIFIEDS

(Continued from Page 30)

CALL CHAR(48, "0038444C54644438")

Right arrow

CALL CHAR(48, "000804027F020408")

Left arrow

CALL CHAR(48, "00102040FE402010")

Copyright symbol

CALL CHAR(48, "003E415D515D413E")

Cent sign

CALL CHAR(48, "00083C4848483C08")

PI symbol

CALL CHAR(48, "0000FE2828282828")

Check mark

CALL CHAR(48, "0002020404482810")

Solid line

CALL CHAR(48, "00FF")

Up arrow

CALL CHAR(48, "081C2A490808080808")

Down arrow

CALL CHAR(48, "00080808492ALC08")

MICROpendium pays \$10 for items published in User Notes. Send them 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

FOR SALE

TI99/4A

PHP1100	TI, 99/4A JOYSTICKS (Pair)	14.95
PHP1260	32K MEMORY CARD	99.95
PHP1500	SPEECH SYNTHESIZER	29.95
PHM3109	TI LOGO II (32K Mem. Req.)	1.95
PHP2500	TI IMPACT PRINTER	249.95
PHM3012	SECURITIES ANALYSIS	12.95
PHM3058	MINI-MEM (W/Assembler)	39.95
PHM3122	PLATO INTERPRETER	49.95

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

10/8 CALL OR WRITE FOR FREE CATALOG

ENORMOUS TI99/4A INVENTORY. CATALOGS \$2.00. BRAATZS COMPUTER SERVICES, 719 E. BYRD ST., APPLETON, WI 54911.

1-414-731-3478.

10/12

TAKE A BREAK
THIS YEAR
ATTEND A TI FAIR

MICROpendium disks, etc.

- Series 1993-1994 mailed monthly (April 1993-March 1994)..... \$40.00
- Series 1992-1993 (Apr 1992-Mar 1993, 6 disks).. \$25.00
- Series 1991-1992 (Apr 1991-Mar 1992, 6 disks).. \$25.00
- Series 1990-1991 (Apr 1990-Mar 1991, 6 disks) ..\$25.00
- Series 1989-1990 (Apr 1989-Mar 1991, 6 disks) ..\$25.00
- Series 1988-1989 (Apr 1988-Mar 1989, 6 disks)...\$25.00
- 110 Subprograms (Jerry Stern's collection of 110 XB subprograms, 1 disk)\$6.00
- TI-Forth (2 disks, req. 32K, E/A, no docs).....\$6.00
- TI-Forth Docs (2 disks, D/V80 files)\$6.00
- 1988 updates of TI-Writer, Multiplan & SBUG (2 disks)\$6.00
- Disk of programs from any one issue of MICROpendium between April 1988 and present\$4.00
- CHECKSUM and CHECK programs from October 1987 issue (includes docs as D/V 80 file).....\$4.00

Check box for each item ordered and enter total amount here:

Name _____

Address _____

City _____

State _____ ZIP _____

Check/MO Visa M/C
(Circle method of payment)

Credit Card # _____

Exp. Date _____

Signature _____

The ONLY monthly devoted to the T199/4A

Subscription Fees

- 12 issues, USA, \$25 12 issues, Mexico, \$30.25
- 12 issues, Canada \$32.50 12 issues, other countries surface mail, \$30.00
- 12 issues, other countries, air mail, \$42.00

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 on the left of your mailing label indicates the cover date of the last issue of your subscription.

Disks, Etc.

- Back Issues, \$2.50 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, \$1.60 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 version(s); includes SET-COLR, 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
<input type="checkbox"/> Series 1	\$9.00	\$7.00	\$5.00
<input type="checkbox"/> Series 2	\$9.00	\$7.00	\$5.00
<input type="checkbox"/> Series 3	\$9.00	\$7.00	\$5.00
<input type="checkbox"/> Series 4	\$9.00	\$7.00	\$5.00
<input type="checkbox"/> Series 5	\$9.00	\$7.00	\$5.00
<input type="checkbox"/> Series 6	\$9.00	\$7.00	\$5.00

SECOND CLASS

AT EXPIRES 1/94
CHARLES BOOD
P.O. BOX 647
VENEDICIA OH 45874