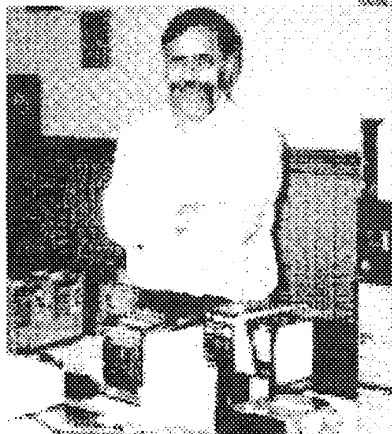


# MICROpendium

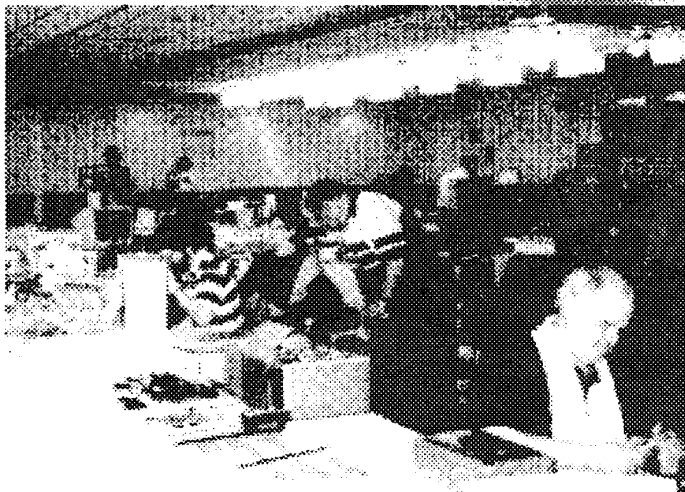
Volume 11 Number 10

November 1994

\$3.50



Photos by Gary Cox



## The 1994 TI International World's Faire

See Page 17

**XBASIC programs galore  
for work and play**

# CONTENTS

## MICROpendium

**MICROpendium (ISSN 10432299) is published monthly for \$35 per year by Burns-Koloen Communications Inc., 502 Windsor Rd., Round Rock, TX 78664-7639. Second-class postage paid at Round Rock, Texas. POSTMASTER: Send address changes to MICROpendium, P.O. Box 1343, Round Rock, TX 78680-1343.**

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 \$40.25 (Mexico); \$42.50 (Canada); \$40.00, surface mail to other countries; \$52 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 & FAX: (512) 255-1512

CompuServe: 75156,3270

Delphi TINET: MICROpendium

GENie: J.Koloen

Internet E-mail: jkoloen@io.com

**John Koloen.....Publisher**

**Laura Burns.....Editor**

## Extended BASIC

- This boomerang, from the Ottawa Users Group's Lucie Dorais, is one you will come back to.....Page 6
- Forget Me Knots and Reminders create a memo system on your computer .....Page 13
- Bill Gaskill creates a program to help you make decisions about investing for retirement.....Page 17
- Communicating with Vegans in a program by the late Jim Peterson, the Tigercub .....Page 22

## The Art of Assembly

- It's about time .....Page 8

## Geneve 9640

- Using BREAD XOP routine to review disk sectors .....Page 11

## TI International World's Faire

- Highlights of the event sponsored by the Chicago and Milwaukee users groups.....Page 17

## Reviews

- MICRO-Reviews: Newsletter Editor and Formatter and USVBA Power Volleyball .....Page 25
- Funnelweb V.5.00 40-Column Editor .....Page 26

## User Notes

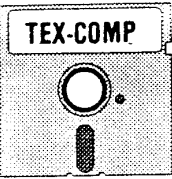
- An answer to a question, a new 80-column card from Germany, and a Maze Maker game.....Page 28

## Classified .....Page 31

### \*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.



# ANNOUNCING NEW LOW PRICES ON FREWARE DISKS

FROM  
**\$2.95 TEX-COMP \$2.95**

TEX-COMP HAS SLASHED PRICES ON THE BIGGEST & BEST COLLECTION OF FREWARE FOR THE 99/4A. NOW ONLY \$2.95 PER DISK WITH A 5 DISK MINIMUM. CHOOSE FROM HUNDREDS OF GREAT PROGRAMS. NO EXTRA CHARGE ON PROGRAMS THAT REQUIRE MORE THAN ONE DISK SIDE. ALL PROGRAMS HAVE BEEN TESTED AND ALMOST ALL HAVE BEEN PROVIDED WITH EXBASIC AUTOLOAD. CHOOSE FROM THE BEST IN GAMES, UTILITIES, GRAPHICS, HOME & BUSINESS AND DISK BACKUPS OF DOZENS OF YOUR FAVORITE MODULES THAT ARE NOW OUT OF PRODUCTION.

GAMES • BUSINESS • GRAPHICS • WORD PROCESSING • UTILITIES • DATABASE • MUSIC • COMMUNICATIONS • HOME

- |   |   |  |   |
|---|---|--|---|
| <p><b>GRAPHICS, MUSIC &amp; ANIMATION</b></p> <ul style="list-style-type: none"> <li>#1 THE SINGING II VOL. 1 (S) (2)</li> <li>#4 PRINTART (2) (P)</li> <li>#5A MUSIC/GRAPHICS</li> <li>#6 EXBASIC MUSIC (2)</li> <li>#7 SPACE SHUTTLE MUSIC/GRAPHICS</li> <li>#9 MONA LISA PRINTOUT (P)</li> <li>#11 ANIMATED XMAS (WOODSTOCK)</li> <li>#14 FIGURE STUDIOS (P) (PG)</li> <li>#12 EXBASIC XMAS MUSIC (2)</li> <li>#41 VIDEO GRAFMS (M)</li> <li>#52 ANIMATION 99* (2)</li> <li>#69 PLAYER PIANO/KEYBOARD ANALYSIS</li> <li>#93 KGBD GIRLIE CALENDAR (4) (P)</li> <li>#103 SORGAN THE II ORGAN</li> <li>#107 STAIRERK MUSIC ALBUM</li> <li>#111 POP MUSIC &amp; GRAPHICS</li> <li>#114 FAHORAMA</li> <li>#115 GRAPHICS DESIGN SYSTEM</li> <li>#120 BITMAC (2) (P)</li> <li>#230 THE SINGING II VOL. 2 (S) (2)</li> <li>#231 THE SINGING II VOL. 3 (S) (2)</li> <li>#246 THE SINGING II VOL. 4 (S) (2)</li> <li>#313 J-D WORLD (CAD for the TI)</li> <li>COMPUTER UTILITIES, PRINTER UTILITIES &amp; PROGRAMMING LANGUAGES</li> <li>#3 DUMPIT (E/A)</li> <li>#15 STAR/EPSON PRINTER DEMO (P) (2)</li> <li>#16 SIDWAYS PRINTOUT (F) (2)</li> <li>#18 TI DIAGNOSTIC (HM) (2)</li> <li>#28 LOADERS &amp; CATALOGS</li> <li>#30 HOUSEHOLD BUDGET PRINTOUT (P)</li> <li>#35 PROGRAMMING AIDS &amp; UTILITIES I</li> <li>#42 FUNNELWEB FARM (SHELL UTILITY)</li> <li>#53 HACKER CRACKER</li> <li>#55 SCREEN DUMP (P)</li> <li>#62 DISK MANAGER II (M)</li> <li>#75 DISK CATALOGER</li> <li>#76 PROGRAMMING AIDS &amp; UTILITIES II</li> <li>#78 ANTCOM GRAPHIC CONVERSION</li> <li>#79 DISK MANAGER 1000</li> <li>#80 BIRDWELL DISK UTILITY (2)</li> <li>#85 AUTOBOOT UTILITY</li> <li>#86 COLUMN TEXT III (P)</li> <li>#87 ARCHIVER III</li> <li>#89 PROCAL DECIMAL/HEX CONVERTER</li> <li>#96 STATISC &amp; SORT ROUTINES</li> <li>#97 MEMORY MANIPULATOR</li> <li>#101 ENHANCED DISPLAY PACKAGE</li> <li>#108 FUNFLUUS (FUS!)</li> <li>#110 DISK AID</li> <li>#117 UNIVERSAL DISASSEMBLER</li> <li>#119 RAG LINKER CONVERSION</li> <li>#127 FIX GRAPHICS UTILITY</li> <li>#243 OS/99 (GD)</li> <li>#251 PC TRANSFER TI/IBM (DD)</li> <li>#253 THE EXPLORER/DH1000</li> <li>#254 NIBBLE/TURBO</li> <li>#260 TI FORTH (DISK ONLY - add \$8 for manual)</li> <li>#17 TI FORTH DEMO</li> <li>#116 TI FORTH TUTORIAL</li> <li>#5078 FORTHE SOURCE CODE</li> <li>#104 C29 COMPLIER &amp; LIBRARY</li> <li>#5007 TEACH YOURSELF TI BASIC</li> <li>#5019 TEACH YOURSELF EX-BASIC</li> <li>#5067 BEGINNING BASIC TUTOR</li> <li>#307 GRAPHICS CODE GENERATOR</li> <li>#3912 SUPER RUGGER</li> <li>#3913 BETTER BAWMRS</li> <li>#3914 CERTIFICATE 99</li> <li>#3915 HOROSCOPE MAKPR</li> <li>#3916 GRAFHX+ PRINT SHOFFF</li> <li>#3917-1720</li> <li>GRAFIX COMPANIONS 1-4</li> <li>#3721 MAC FLICK (C)</li> <li>#3722 PRINTING TO GO (C)</li> <li>#3723 GRAFIX DIMOSAURS (G)</li> </ul> | <p><b>BUSINESS, ACCOUNTING, WORD PROCESSING, DATA BASE</b></p> <ul style="list-style-type: none"> <li>#10 OFFICE &amp; HOME</li> <li>#10 GOTRIC PRINTOUT (P)</li> <li>#19 TI WRITER/MULTIPLAN UPGRADE</li> <li>#20 ACCOUNTS RECEIVABLE (2)</li> <li>#21 DATA BASE DEMO</li> <li>#21 WILL WRITER</li> <li>#29 LABEL MAKER I (F)</li> <li>#36 STRICTLY BUSINESS (2)</li> <li>#56 SPREAD SHEET</li> <li>#58 PR BASE (database)</li> <li>#59 CRM MAKER</li> <li>#74 LABEL MAKER II (F)</li> <li>#77 MICROdex 99 (database)</li> <li>#81 HOME ACCOUNTING SYSTEM</li> <li>#83 HOME APPLICATION PROGRAMS (2)</li> <li>#90 JET CHECKBOOK MANAGER</li> <li>#92 HOUSEHOLD INVENTORY</li> <li>#109 TI WRITER MINI MANUAL</li> <li>#112 INVOICE PACK</li> <li>#113 LABEL MAKER III</li> <li>#129 CASH DRAWER (point of sale)</li> <li>#130 THE ORGANIZER</li> <li>#147 CALENDAR-NOTEPAD</li> <li>#177 HOUSEHOLD BUDGET MANAGEMENT (M)</li> <li>#30 HBM DATA PRINTOUT (F)</li> <li>#221 PERSONAL REAL ESTATE (M)</li> <li>#249 HAFMAKER</li> <li>#252 99 WRITER II (TI WRITER) (P)</li> <li>#253 AMA MAILING LIST</li> <li>#255 TRAF-A-CHECK</li> <li>#257 DAILY DIARY</li> <li>#257A EXER-LOG</li> <li>#23 WILL WRITER</li> <li>#310 SELF HELP TAX CUT</li> <li>TELECOMMUNICATIONS (MODERN)</li> <li>#57 IELCO</li> <li>#57D TELCO (FOR SYSTEMS WITH OS DRIVES)</li> <li>#118 FAST TERM</li> <li>EDUCATOR &amp; PERSONAL DEVELOPMENT</li> <li>#22 ASTROLOGY</li> <li>#24 ENGINEERING CALCULATIONS (2)</li> <li>#25 MEDICAL ALERT</li> <li>#27 KIDS LEARNING I (2)</li> <li>#31 HORSE CODE TRAINER</li> <li>#37 LAPD COOKBOOK (2)</li> <li>#40 ARTIFICIAL INTELLIGENCE (ELIZA)</li> <li>#54 ASTRONOMY</li> <li>#66 HEBREW TYPEWRITER</li> <li>#67 GENEALOGY (2)</li> <li>#71 KIDS LEARNING II (2)</li> <li>#95 WEATHER FORECASTER</li> <li>#138 FIREHOUSE COOKBOOK</li> <li>#142 TOUCH TYPING TUTOR (M)</li> <li>#184 FACE MAKER</li> <li>#194 ST. WCK/GHOSTLY SPELLING (M)</li> <li>#199 TINY LOG</li> <li>#199 HILLIKEN ADDITION (M)</li> <li>#200 HILLIKEN DECIMALS (M)</li> <li>#203 HILLIKEN FRACTIONS (M)</li> <li>#204 HILLIKEN INTEGERS (M)</li> <li>#205 HILLIKEN LAWS OF MATH (M)</li> <li>#211 MIND CHALLENGERS (M)</li> <li>#212 MIND MISSION (M)</li> <li>#213 HILLIKEN PERCENTS (M)</li> <li>#214 STORY MACHINE (M)</li> <li>#215 BEGINNING GRAMMAR (M)</li> <li>#216 METEOR MULTIPLICATION (M)</li> <li>#217 HANGMAN (M)</li> <li>#222 MUSICMAKER (M)</li> <li>#223 PHYSICAL FITNESS (M)</li> <li>#225 ALIEN ADDITION (M)</li> <li>#226 ALLIGATOR MIX (H)</li> <li>#227 DEPOLITION DIVISION (M)</li> <li>#228 DRAGON MIX (H)</li> <li>#270 TRIGONOMETRY</li> <li>#271 CALCULATORS &amp; CONVERSIONS (2)</li> <li>#272 HIGHER MATH (2)</li> <li>#273 ASTRONOMY 2.</li> <li>#306 SPEAK &amp; SPELL II (S) (EX)</li> </ul> | <p><b>GAMES</b></p> <ul style="list-style-type: none"> <li>#2 WH. OF FORTUNE, BLKJACK, POKER</li> <li>#2 TI TRPK</li> <li>#8 LOITTO PICKER</li> <li>#13 STRIP POKER (PG)</li> <li>#13 R-RATED NOVELTY GAME</li> <li>#13 CHECKERS &amp; BACKGAMMON</li> <li>#14 SOLITAIRE &amp; SCRABBLE</li> <li>#18 GREAT II GAMES VOL 1</li> <li>#19 GREAT II GAMES VOL 2</li> <li>#43 BEST OF BRITAIN GAMES VOL 1</li> <li>#43 BEST OF BRITAIN GAMES VOL 2</li> <li>(LEGEND OF CARFAX ABBY GRAPHIC-INTERACTIVE ADVENTURE)</li> <li>#46 SUPER TRIVIA 99</li> <li>#47 INFOCOM RAPID LOADER</li> <li>#48 GHOSTMAN (FROM U.K.)</li> <li>#49 DEMON DESTROYER (FROM FRANCE)</li> <li>#50 OH MUMMY! (HIT FROM GERMANY)</li> <li>#51 BERLIN WALL (FROM CANADA)</li> <li>#50 FREDDY (HIT FROM GERMANY)</li> <li>#51 THE NINE (FROM GERMANY)</li> <li>#52 ASTROBLITZ &amp; MAZOG</li> <li>#54 MAJOR TOM &amp; SPACE STATION PIETA</li> <li>#55 PERFECT PUSH (HIT)</li> <li>#58 CHESS (SARGON)</li> <li>#70 TI RUNNER II (HIT)</li> <li>#72 CEBERUS (HIT SPACE GAME)</li> <li>#73 CRVIO (GRAM)</li> <li>#82 CROSSWORD (PUZZLES)</li> <li>#84 GALACTIC BATTLE &amp; SPY ADVEN.</li> <li>#88 AUSSIE GAME COLLECTION VOL 1</li> <li>#91 THE MAZE OF GROG (WOODSTOCK II)</li> <li>#94 GREAT II GAMES VOL 3</li> <li>#98 DAYS/POORS OF EDEN (BIBLE ADV) (AM)</li> <li>#99 GREAT II GAMES VOL 4</li> <li>#100 ASSAULT THE CITY (TD)</li> <li>#102 COLOSSAL CAVES (ADVENTURE)</li> <li>#105 KINGS CASTLE (M)</li> <li>#106 QUEST (D&amp;D)</li> <li>#121 SUPER YANITZEE &amp; WHEEL II</li> <li>#122 ADULT ADVENT &amp; GAMES (PG)</li> <li>#123 GREAT II GAMES VOL 5 (2)</li> <li>#124 GREAT II GAMES VOL 6 (2)</li> <li>#125 BLACKJACK &amp; POKER (M)</li> <li>#126 VLDZ CHESS (M)</li> <li>#128 TETRIS (HIT FROM RUSSIA)</li> <li>#131 COMPUTER CRAFTS</li> <li>#132 AMBULANCE (M)</li> <li>#133 DRIVING DEMON (M)</li> <li>#134 ROTO-RAIDER (M)</li> <li>#135 ARTURUS (HIT-ZARGON)</li> <li>#136 ANT-EATER (M)</li> <li>#137 CROSS-FIRE (M)</li> <li>#139 MOOHMINE (M)</li> <li>#140 HASH (M)</li> <li>#141 MOONKEEPER (M)</li> <li>#143 CONGO BONGO (M)</li> <li>#144 STAR TREK (M)</li> <li>#145 BUCK ROGERS (M)</li> <li>#148 KENO &amp; SLOTS</li> <li>#149 GREAT II GAMES VOL 7 (WITH HIT BLOCKRUSTR)</li> <li>#150 ULTIMATE TRIVIA</li> <li>#151 JUNGLE HUNT (M)</li> <li>#152 GOLF POSITION (M)</li> <li>#153 DONKEY KONG (M)</li> <li>#154 PROTECTOR II (M)</li> <li>#155 PAC MAN (M)</li> <li>#156 CENTIPEDE (M)</li> <li>#157 DEFENDER (M)</li> <li>#158 SHAMUS (M)</li> <li>#159 MS. PAC MAN (M)</li> <li>#160 DIG-DOG (M)</li> <li>#161 PICNIC PARANOIA (M)</li> <li>#162 MOON PATROL (M)</li> <li>#3711 ARCADE SPECIAL (4 GAMES)</li> <li>#3712 THREE GREAT GAMES</li> <li>#3713 PRO TENNIS+</li> </ul> | <p><b>INFOCOM ADVENTURE BACKUPS</b></p> <ul style="list-style-type: none"> <li>#163 ZORK I</li> <li>#164 ZORK II</li> <li>#165 ZORK III</li> <li>#166 HITCHHIKER'S GUIDE</li> <li>#167 WITNESS</li> <li>#168 ENCHANTER</li> <li>#169 INFIDEL</li> <li>#170 PLANETFALL</li> <li>#171 SORCERER</li> <li>#172 DEADLINE</li> <li>#173 CUTTHROATS</li> <li>#174 SUSPENDSD</li> <li>#175 STARPOSS</li> <li>****</li> <li>#176 AMAZING (M)</li> <li>#178 DEMON DESTROYER (M)</li> <li>#179 POPEYE (M)</li> <li>#180 QUEBERT (M)</li> <li>#181 METEOR BELT (M)</li> <li>#182 BLASTO (M)</li> <li>#183 CAR WARS (M)</li> <li>#184 FACE MAKER (M)</li> <li>#185 SUPER FLY (M)</li> <li>#186 SPACE BANDITS (M)</li> <li>#188 KILLER CATERPILLER (M)</li> <li>#190 BLACK HOLE &amp; SPACE AGGRESS. (M)</li> <li>#191 GREAT II GAMES VOL 8</li> <li>#192 GREAT II GAMES VOL 9</li> <li>#193 SPY'S DEHISE (HIT) (M)</li> <li>#194 ST. NICK+ (M)</li> <li>****</li> <li>#196 JOITO</li> <li>#197 PRO TENNIS+ (HIT) (M)</li> <li>#198 TI INVADERS/TOHSTONE CITY (M)</li> <li>#202 CONNECT FOUR (M)</li> <li>#205 HOPPER (M)</li> <li>#206 TREASURE ISLAND (M)</li> <li>#206B SLYMOIDS (M)</li> <li>#207 OTHELLO (M)</li> <li>#208 FARSEC (M)</li> <li>#209 SOCCER (M)</li> <li>#210 SENEPRIMA (M)</li> <li>#218 HUSTLE/FOOTBALL (M)</li> <li>#219 CHISOLM TRAIL (M)</li> <li>#220 ZERO ZAP (M)</li> <li>#224 ATTACK (M)</li> <li>#229 4A FLYER (FLIGHT SIM.) (M)</li> <li>#232 TUNNELS OF DOOM (MOD BACKUP PLUS 2 NEW ADVENTURES) (M)</li> <li>#233 MS ADVENTURES (3 ADV-EXBASIC)</li> <li>#248 STRIKE THREE BASEBALL (M)</li> <li>#248 GREAT II GAMES VOL 10</li> <li>#250 BARRAGE/SPOTSHOT</li> <li>#318 THREE GREAT GAMES</li> <li>#319 ARCADE SPECIAL</li> <li>#317 BEANSTALK ADVENTURE</li> <li>****</li> <li>SCOTT ADAMS ADVENTURES (USE WITH ADVENTURE MODULE)</li> <li>#349 TI ADVENTURES 1-13+</li> <li>#350 TI ADVENTURES 14-16+</li> </ul> |
|---|---|--|---|

P = Printer required  
G = Graphx required  
S = Speech required  
M = Module Backup  
MM = Mini Memory req.  
E/A = Editor Assem.  
Exbasic and 32k mem req. for most programs.

**ORDER BY PHONE**  
(818) 366-6631 24 HOURS A DAY



**TEX-COMP**  
America's Number One TI computer retailer

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

TERMS: MINIMUM ORDER FOR \$2.95 PRICE IS 5 DISKS (REG. 4.95) ADD \$4.00 PER ORDER FOR SHIPPING (U.S.) ALL PRICES ARE FOR CASH/CHECK, ADD 3% FOR CREDIT CARD ORDERS. INCLUDE STREET ADDRESS FOR U.P.S.

# COMMENTS

## A new 80-column card

A group of German TI developers have produced an 80-column card for the TI. The announcement was made at the Göttingen TI fair. The group is also developing a multi-module card that uses flash EPROMs and plans a PEB-based motherboard. For more information, see User Notes.

### CREDIT CARD SALES

There's no way to avoid it any more. I just got my latest credit card merchant's statement. It's very depressing. We're now being charged more than 5 percent for sales done on credit cards. We can't afford it. So, starting immediately, all credit card orders will be charged a 5 percent surcharge. I don't blame you if you don't want to pay the 5 percent. But you can always send a check or money order. We've been covering these expenses for many years but we can't continue to do so. I hope this doesn't offend anyone.

### CHICAGO FAIRE

Attendance at this year's Chicago TI fair was significantly less than in years past. What I noticed most was the absence of casual visitors, people who'd pay their admission and stay for an hour or two. Virtually everybody who attended stayed the entire day.

Some camped out in the seminar room while others roamed the exhibit hall with frequent side trips to the seminar room.

A number of vendors I spoke with thought a broader publicity campaign would have increased attendance. Even so, visitors came from all over the country and Europe. There just weren't many from the Chicago area. Also, the proximity to Gurnee Mills a large mall across the street from the hotel, may have emptied the fair of the normal contingent of family traffic. Wives and children were conspicuously absent.

Unlike years past, when the seminar rooms were directly accessible from the exhibit floor, this year the two rooms were at opposite ends of the hotel. This was a particular problem for presenters who had to haul their systems from the exhibit hall and set them up before their session could start. It also discouraged casual attendance at the seminars.

Although the seminars were excellent, it's clear that TIers need to be reaching out to their brethren to boost attendance at these events. Even knowing that you're going to get more than your two dollar's worth isn't enough. Getting the word out is probably more important now than ever.

—JK

## MICROpendium disks, etc.

- |   |   |
|---|---|
| <input type="checkbox"/> Series 1994-1995 mailed monthly (April 1994-March 1995)..... \$40.00 | subprograms, 1 disk) .....\$6.00  |
| <input type="checkbox"/> Series 1993-1994 mailed monthly (April 1993-March 1994)..... \$25.00 | <input type="checkbox"/> TI-Forth (2 disks, req. 32K, E/A, no docs).....\$6.00  |
| <input type="checkbox"/> Series 1992-1993 (Apr 1992-Mar 1993, 6 disks) .. \$25.00             | <input type="checkbox"/> TI-Forth Docs (2 disks, D/V80 files) .....\$6.00   |
| <input type="checkbox"/> Series 1991-1992 (Apr 1991-Mar 1992, 6 disks) .. \$25.00             | <input type="checkbox"/> 1988 updates of TI-Writer, Multiplan & SBUG (2 disks) .....\$6.00                              |
| <input type="checkbox"/> Series 1990-1991 (Apr 1990-Mar 1991, 6 disks) ..\$25.00              | <input type="checkbox"/> Disk of programs from any one issue of MICROpendium between April 1988 and present .....\$4.00 |
| <input type="checkbox"/> Series 1989-1990 (Apr 1989-Mar 1991, 6 disks) ..\$25.00              | <input type="checkbox"/> CHECKSUM and CHECK programs from October 1987 issue (includes docs as D/V 80 file) .....\$4.00 |
| <input type="checkbox"/> Series 1988-1989 (Apr 1988-Mar 1989, 6 disks)...\$25.00              |   |
| <input type="checkbox"/> 110 Subprograms (Jerry Stern's collection of 110 XB                  |   |

Name \_\_\_\_\_

Address \_\_\_\_\_

City \_\_\_\_\_

State \_\_\_\_\_ ZIP \_\_\_\_\_

Texas residents add 7.75% sales tax.

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

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

Credit Card # \_\_\_\_\_

Exp. Date \_\_\_\_\_

Signature \_\_\_\_\_

# FEEDBACK

## Problem in TI-BASE

In the June 1994 issue of your magazine I found an interesting article about the programming of TI-BASE because I made my own tries with the TI-BASE program.

My problem is that I missed the RETURN at the end of the SETUP/C file which is created by Ron Warfield

In my understanding, the RETURN-command of TI-BASE brings the program back to the level which called the program file in which the RETURN is programmed!

So the RETURN in START/C will bring the program back in the SETUP/C file and from there the program falls back to the command line without a RETURN. So this may work but it is not programmed correctly. On my Geneve it causes trouble when I run the program some times more. (Error: System hangs up!)

Keep on compuTing.

**Martin Zeddies**  
Wolfsburg, Germany

## An open letter to the TI community c/o Charles Good

*(This is condensed from the November 1994 issue of Bits, Bytes & Pixels, the newsletter of the Lima 99/4A Users Group in Lima, Ohio)*

Looking through the latest issue of *B.B.&P.*, I thought it was time that I sat down and wrote you a letter — that could be published if you like — concerning Asgard, Harry Brashear, and the current state of my computer philosophy.

As you know, as of November 1993, Chris Bobbitt turned Asgard over to me lock, stock and barrel for the price of saying "Yes, I'll take it." He wouldn't have done that if there had been any more money to be made on the company. Sales had taken a plunge as of the Lima that year and by October were down to zip. Most of the lack of sales stemmed from my inability to get the author of First Draft to fix a couple of minor bugs that were causing a lot of problems. I had

to stop selling the program after Lima '93 on account of them and never got the program off the ground again. Believe me, I did everything I could, but it's impossible to deal with someone a thousand miles away who doesn't want to be dealt with — the TI community has had this experience more than once.

All of that occurred under the old Asgard structure so by the time I took it over, there was nothing new to sell. Besides, it appeared that the size of the community had taken a steep plunge after the '93 Lima show. To supplement my catalog, I made arrangements with Ken Gilliland to handle all of his current software, quickly made up 25 copies of each and began to dump out flyers on it. I still have 25 copies of each almost a year later. To make matters worse, a number of authors decided to pull the rug out from under me because they felt that Chris and I had no right to dump the company the way we did. No problem, most of what was pulled hadn't sold a copy in the preceding year anyway.

Because of nothing new, there has been no reason to advertise and I think that, now, some people have decided that Asgard is dead too. Not true. Yes, the company is dead because I decided not to officially resurrect it, but I am still around personally if people should happen to need something out of the old catalogs. So, bottom line; Asgard is dead, I'm not, and if I should get some new software worth the time and money, I will advertise it under a new name.

I have something to say to those people who have or are considering jumping ship.

Over the last 11 years, no one has been more of a TI advocate than I. I would like to think, even, that I was partly responsible for adding some TI life to a couple of groups and a few individuals. It was tough because eight years ago, my wife became involved with Delphi in the PC SIG and has kept it up to a point where, today, she is the data base manager and gets paid for her work. I think you can appreciate the fact that I have had a lot of high tech equipment shoved down my throat 24 hours a day, yet managed to keep pounding on my TI up to three

years ago. At that stage of the game I started having to deal with half meg files of writing and had to move over for the sake of file size, if nothing else. First Draft would have held me (That's really why I took on the First Draft project.) and even today's Funnelwriter 5.0+ would have given me a half life since I had a TIM in my system, but neither of these were available.

Before I go on, I want to tell you that I still use the TI for all my business invoices and year-end bookkeeping; I made those programs years ago, still use them monthly, yearly, and have no reason to change. They work! And if there is something new I need done — reports etc. — I can go in and change the programming.

What I want to tell you is that I have found a way of working on an IBM and keeping the TI attitude, now dealt with venomous feeling toward Bill Gates, Microsoft and the !\$%^@&#! world of Windows — I use Geoworks Pro on my 386.

Geoworks is not sold in stores. It is sold only from the company. It has user groups world wide, has three monthly newsletters and is user supported. It has drawing capabilities, word processing, over 700 fonts available, a spread sheet, and interactive data base, a communications package, spell checking, thesaurus, date book, scrapbook, full file and disk management, will auto run DOS-based programs and return, has built-in games, camera ready printer output, etc., etc. They all run at once and you can cut and paste from one to the other with ease. Plus it will run on even an 8086, though I wouldn't recommend it speed wise. Even the Australians have gotten on the Geos band wagon, so you know it has to be good — and rebellious.

No, I'm not selling Geos, though I'd like to find a way. What I'm telling you is that there is a place to go if you feel you must move on, where the TI99 spirit is still alive, where you can find that friendship, that attitude we are so famous for, and find the old challenge that we yearn to maintain. If I have insulted you

(See Page 6)

Extended BASIC

# You'll return to this boomerang

By LUCIE DORAIS  
(c) 1992 Lucie Dorais

When your imagination has run dry, you can always count on someone else's! I had wanted for a long time to design an animated "sprity" game, but could not come up with an idea. When I found, among my collection of programs to type, this little gem, I picked it up for my column. Published in the French computer magazine *MICRO 7* a few years ago, it was written by Pascal Peitrequin. I have tried to improve(!) his code, mostly by simplifying the graphic routines and translating most of the screen text into graphics, but the look of the screen and, most importantly, the routines, are totally his.

This is a shoot-'em-up game (there are six ducks to kill), but with a twist: the man shooting the boomerang is placed in the lower left corner; when he shoots his weapon, it follows a curve towards the left, immediately leaving the screen to reappear on the right (because of the sprite's wraparound feature). It is therefore very difficult to aim with precision.

Another factor is that the boomerang's speed is controlled by its trajectory; the higher you shoot, the faster it will run: the boom's height and velocity depend on the key you press to launch it, from "1" for a very low/slow boom to "7" for the highest trajectory. Finally, as a good boomerang should do, it goes up in a full curve, then comes down until it hits the ground, or hits a duck (this, of course, can also happen on the way up). It is a bit hard to master, but it is quite possible to kill all six ducks with the 10 boomerangs you start with.

The very simple instructions are displayed by lines 150-200, and some booms fly by while you read them (line

160). In line 120, the DIM refers to the ducks' velocity, which is random and slightly increased after each hit. The graphics are defined and colored by lines 220-320, while the boomerangs continue to fly around the screen, until they disappear in line 340.

The game screen is then displayed: first the tree trunk is made up into a string, then the ground is CALL CHARED, followed by the DISPLAY of two trees and tall grass (line 360). The sky is CALL CHARED in three shades of

**The boomerang's speed is controlled by its trajectory: the higher you shoot, the faster it will run**

blue (line 370), and the man appears at the lower left corner; his boomerang and his dog are sprites that will be animated later. The six ducks are finally set in motion, with their random speed V(X); the red duck-sprites (Nos. 2-4) are placed on pixel-rows 100, 120 and 140 respectively, and they travel toward the right; the yellow ones (Nos. 5-7) are placed higher in the sky, on pixel-rows 40, 60 and 80, and they travel toward the left (negative value, hence the IF in line 400).

The game itself starts at line 440 by checking for an "end of game" situation: you have shot all 10 boomerangs, or you have killed all six ducks. If not, the remaining number of weapons is displayed

at the upper right corner (you start with nine, since the man already holds the tenth). While Tex waits for your input in line 460, the flying ducks are animated by changing their PATTERNS. When you press a key from "1" to "7," the boom is launched with appropriate sound, its velocity and trajectory are determined according to the key you pressed (I did not write the routine in line 500, but it works, *don't try to change it!*), and the SHOT counter is incremented.

Lines 510-540 control the boom's trajectory and check for its COINCidence with any sprite (I had to modify the game at some point, because the program also reacted when the boom hit the dog...). The good working of this whole trajectory routine depends on the time Tex takes to run through these lines, so if you want to alter them, be prepared for some fine tuning (I did, until I decided to leave it as the author had written it, which was the only way it worked well; I just added one more CALL COINC, but even then it misses a few when the boom travels too fast). The POSITION call in line 530 checks if the boomerang is still in the sky; when it hits the ground without having hit a duck, it disappears and a new boom is put in the man's hand (line 550).

A duck has been hit when CALL COINC returns a value of "-1" into the variable C. It takes some time for Tex to react, and usually the boom is a bit ahead of the hit duck, even if Tex stops the boom's motion immediately after the hit (line 570). The boom's position on the screen needs to be determined next, to tell Tex which duck has been hit. The sprite number SP is found by comparing the position of the boom, in lines 590-640, with the general space occupied by  
(See Page 7)

## FEEDBACK

(Continued from Page 5)

with this thought, I apologize, but believe me, I still have the heart of a 99er and an awfully big stock of TI programs.

Harry Thomas Brashear

2753 Main St.

Newfane, NY 14108

Send your letters and comments

to *MICROpendium Feedback*, P.O.

Box 1343, Round Rock, TX 78680.

# BOOMERANG—

(Continued from Page 6)

each duck. Line 580 takes care of the dog problem mentioned above.

The successful boomerang (sprite No. 1) disappears, and the hit duck, now dead, changes to black before starting its fall (line 660). The duck's position is checked until it hits the ground (pixel-row 175); when it does, its pattern is changed to that of a broken duck cadaver (this is a violent game) and, after a little delay, the dog is sent to fetch it.

The dog routine is a separate sub, even if it is called only from line 680, no doubt to keep the code easier to read. In line 800, the dog is set in motion toward the right; until it reaches the dead duck, its pattern changes to indicate walking, and its POSITION column CDOG is checked regularly; if it comes near the dead duck position column CSP, it stops, sort of barks (line 820) and the dead duck sprite disappears. The dog then turns around, now holding the duck in its mouth. It walks towards the left until it reaches pixel-column 15, where it stops before sitting at its master's feet (lines 830-870).

Upon returning to the main program, the HIT variable is incremented, and the new dead duck added to the total displayed below the man with appropriate sounds and delay (lines 690-700). The velocity of all ducks is increased (the ones that have been DELSPRITED will not show it of course) and you are sent back to line 550 for a new boom.

When the game is over, your score is displayed, and you are asked to either play Another game or Quit (lines 740-780).

## BOOMERANG

```

100 ! ** BOOMERANG ** P. PEI
TREQUIN (MICRO7/mai 1985) !1
04
110 ! == adapted by L. Dorais / Ottawa UG / Feb. 1992 !0
97
120 DIM V(7):: CALL CLEAR ::
CALL SCREEN(12):: CALL MAGN
Y(2)!072
130 GOTO 150 :: C,CDOG,CSP,D
,K,HIT,R,S,SHOT,SP,T$,X,VC,V
L :: CALL KEY :: CALL SOUND
:: CALL COLOR :: CALL CHAR :
: CALL HCHAR !215
140 CALL SPRITE :: CALL DELS
PRITE :: CALL PATTERN :: CAL
L MOTION :: CALL POSITION ::
CALL LOCATE :: CALL COINC :
: !0P- !208
150 DISPLAY AT(1,10):"BOOMER
ANG" :: CALL CHAR(91,"402010
080402013F")!089
160 FOR X=1 TO 5 :: CALL SPR
ITE(#X,91,9,20*X+100,30*X,-5
,15*X):: NEXT X !096
170 DISPLAY AT(4,1):"Use the
numeric keys 1-7 to launch
the boomerang. Its heighth
and speed depend on the key
value:" !091
180 DISPLAY AT(10,6):"1 = lo
west":" 7 = highest": :
:"You are allowed 10 shots
to kill 6 ducks." !232
190 DISPLAY AT(24,3):"PRESS
ANY KEY TO START..." !012
200 CALL KEY(0,K,S):: IF S=0
THEN 200 ELSE CALL CLEAR !0
94
210 ! ==== init values ====
!047
220 CALL COLOR(12,2,3,13,7,1
,2,13,1)! ground/man/grass !
073
230 CALL CHAR(96,"",104,"",1
12,"",125,""):: CALL COLOR(9
,2,5,10,2,6,11,2,8)! sky !00
2
240 CALL CHAR(46,"9252544952
643820",47,"50525949496A1C18
")! grass !149
250 CALL CHAR(42,"21397F7FFF
FFFFFF",43,"20F8FEFEFEFEFEFFF
",44,"FFFFFFFFF7F3F1F",45,"
FFFFFFEFEFEF8")! tree fol
iage !215
260 CALL CHAR(130,"010101010
1010101",131,"80808080808080
80")! tree trunk !105
270 CALL CHAR(123,"F03036FF7
C3870E0",124,"00F036FF7CF8")
! red ducks !228
280 CALL CHAR(40,"03060C58FF
1E0C07",41,"00000F58FF1E")!
yellow ducks !183
290 CALL CHAR(122,"000102060
E5EFC")! dead duck !213
300 CALL CHAR(128,"393939117
F7F7CBC",129,"B838382C6C6C44
C6")! man !158
310 CALL CHAR(132,"0407877C7
C7C4444",133,"0407877C7C7CE3
81")! going dog !144
320 CALL CHAR(134,"20E0E1BEB
E3E2222",135,"20E0E1BEBE7EC7
81")! returning dog !178
330 ! ==== display game scre
en ==== !027
340 CALL DELSPRITE(ALL):: CA
LL CLEAR :: CALL MAGNIFY(1)!
038
350 T$=CHR$(130)&CHR$(131)&"
." :: CALL HCHAR(24,1,125,32
)! ground !167
360 DISPLAY AT(21,16):"*+ *+
":TAB(16);",-,-":TAB(13);"/
//"&T$&T$&" /././" !098
370 CALL HCHAR(2,1,96,96)::
CALL HCHAR(5,1,104,256):: CA
LL HCHAR(13,1,112,224)! sky
!017
380 CALL HCHAR(22,3,128):: C
ALL HCHAR(23,3,129)! man !20
1
390 CALL SPRITE(#1,91,2,162,
22,#15,132,2,177,9)! boomer
ang/dog !109
400 FOR X=2 TO 7 :: V(X)=INT
(15*RND)+4 :: IF X>4 THEN V(
X)=-V(X)! speed !162
410 IF X<5 THEN CALL SPRITE(
#X,123,9,20*X+60,8,0,V(X))EL
SE CALL SPRITE(#X,40,11,20*X
-60,248,0,V(X))!246
420 NEXT X ! put ducks on sc
reen !193
430 ! ==== game ==== !101
440 IF SHOT=10 OR HIT=6 THEN
740 ! game over !127
450 CALL SOUND(100,1000,0)::
DISPLAY AT(3,18):" "&RPT$("
[,9-SHOT)! avail.booms !132
460 CALL KEY(0,K,S)!187
470 CALL PATTERN(#2,124,#3,1
23,#4,124,#5,40,#6,41,#7,40)
! animate ducks !243
480 CALL PATTERN(#2,123,#3,1
24,#4,123,#5,41,#6,40,#7,41)
!055
490 IF S=0 OR K<49 OR K>55 T
HEN 460 ELSE CALL SOUND(10,1
30,0)!146

```

(See Page 8)

## THE ART OF ASSEMBLY — PART 41

# It's about time

By **BRUCE HARRISON**

©1994B. Harrison

Today's subject is time. Recently, it's been important to us in several respects. First, it's February 1994 as we write this, and we've just learned of the death of our dear friend Jim Peterson. This reminds us that we're all working against an unknown time limit, but that's not the kind of time constraint we're concerned with in today's column. It's the use of our beloved TI to do things that are related to "keeping time" in matters of seconds. We'll talk about two time-related assembly projects, and show how these use the internal Vertical Interval timer in the TI to provide an accurate source of "ticks" that we can count.

### THE METRONOME

As many of our readers are aware, we are a "musical" family, having produced lots of "computer music" both with and without MIDI capability. Our two little boys have started learning to use "real" musical instruments, so we're getting used to having the sounds of a clarinet and violin in the house. When kids are learning to play instruments, they need a stable source of timing to keep them playing at a steady tempo. Our local music store has all kinds of metronome devices, from the simple mechanical ones to the most exotic electronic ones. They all work quite well, but all are expensive, considering that all they do is produce a repetitive "tick" at a constant rate in beats per minute.

(See Page 9)

## BOOMERANG—

(Continued from Page 7)

```

500 SHOT=SHOT+1 :: VL=(3*(K-
47))+6 :: VL,VC=-VL ! boom's
velocity !125
510 CALL COINC(ALL,C):: IF C
=-1 THEN 570 !157
520 CALL MOTION(#1,VL,VC)::
VL=VL+2 :: VC=VC-2 !181
530 CALL POSITION(#1,R,D)::
IF R>175 THEN 550 ! boom hit
s ground !058
540 CALL COINC(ALL,C):: IF C
=-1 THEN 570 ELSE 510 !230
550 CALL MOTION(#1,0,0):: CA
LL LOCATE(#1,162,22):: GOTO
440 ! new boom !066
560 ! ==== hit ==== !016
570 CALL MOTION(#1,0,0):: CA
LL POSITION(#1,R,D)!193
580 IF R<30 OR R>150 THEN 51
0 !103
590 IF R>130 AND R<=150 THEN
SP=4 :: GOTO 650 !075
600 IF R>110 AND R<=130 THEN
SP=3 :: GOTO 650 !070
610 IF R>90 AND R<=110 THEN
SP=2 :: GOTO 650 !025
620 IF R>70 AND R<=90 THEN S
P=7 :: GOTO 650 !242
630 IF R>50 AND R<=70 THEN S
P=6 :: GOTO 650 !237
640 IF R<=50 THEN SP=5 ELSE
570 !023
650 CALL DELSPRITE(#1):: CAL
L SOUND(50,1300,0,-3,0):: CA
LL COLOR(#SP,2):: CALL MOTIO
N(#SP,16,0)! dead duck falls
!075
660 CALL POSITION(#SP,R,CSP)
:: IF R<175 THEN 660 !027
670 CALL MOTION(#SP,0,0):: C
ALL PATTERN(#SP,122)!dead du
ck hits ground !049
680 CALL SOUND(10,-5,0):: FO
R D=1 TO 100 :: NEXT D :: GO
SUB 800 ! send dog !185
690 HIT=HIT+1 :: CALL HCHAR(
24,2,122,HIT):: CALL SOUND(5
0,400,0)!090
700 CALL SOUND(50,800,0):: C
ALL SOUND(50,300,0):: FOR D=
1 TO 100 :: NEXT D !146
710 FOR X=2 TO 7 :: IF X<5 T
HEN V(X)=V(X)+1 ELSE V(X)=V(
X)-1 !249
720 CALL MOTION(#X,0,V(X))::
NEXT X :: GOTO 550 ! increa
se ducks' speed !101
730 ! ==== game over ==== !0
65
740 CALL SOUND(100,400,0)::
CALL SOUND(100,250,0):: CALL
SOUND(150,600,0)!124
750 S=10*(10*HIT+60-6*SHOT):
: DISPLAY AT(3,2)SIZE(-12)BE
EP:" SCORE:";S !178
760 DISPLAY AT(6,2)SIZE(-16)
:" 1 ANOTHER GAME " :: DISPL
AY AT(7,2)SIZE(-16):" 2 QUIT
" !017
770 CALL KEY(0,K,S):: IF S=0
THEN 770 ELSE CALL SOUND(10
0,600,0)!070
780 IF K=49 THEN HIT,SHOT=0
:: GOTO 340 ELSE END !051
790 ! ==== sub: dog routine
==== !207
800 CALL MOTION(#15,0,20)! g
oing right !091
810 CALL POSITION(#15,D,CDOG
):: CALL PATTERN(#15,133)::
CALL PATTERN(#15,132):: CALL
SOUND(10,-5,0)!228
820 IF CDOG<CSP-8 THEN 810 E
LSE CALL MOTION(#15,0,0):: C
ALL SOUND(50,-1,0)!143
830 CALL DELSPRITE(#SP):: CA
LL MOTION(#15,0,-8)! returni
ng !039
840 CALL POSITION(#15,D,CDOG
):: CALL PATTERN(#15,135)::
CALL PATTERN(#15,134):: CALL
SOUND(5,-6,0)!188
850 IF CDOG>15 THEN 840 ELSE
CALL MOTION(#15,0,0)!009
860 CALL SOUND(30,-4,0):: FO
R D=1 TO 100 :: NEXT D :: C
ALL SOUND(50,-6,0)!079
870 CALL SPRITE(#15,132,2,17
7,9):: RETURN !122

```



# THE ART OF ASSEMBLY—

(Continued from Page 8)

Being a typical "TI Cheapskate," it occurred to me that our beloved computer ought to be able, with the right programming, to produce those ticks for "free." We plunged right into Assembly, and in just a couple of days we had a new Public Domain "product" ready for any TI users. This product, which is called METRONOME, is available through the Lima and Chicago User Groups at their normal copying fee. It's also been authorized to be placed on BBS, etc. Thus it should be readily available to anyone who needs it. Actually the disk contains two versions of the program, one for U.S. users and another for "European" users. This is necessary because the actual timing is provided by the Vertical Interval timer, and that operates at 60 Hz for U.S. systems, and at 50 Hz for "European" systems, including those in Great Britain and Australia.

We've not included the source code for the metronome in today's column, partly because it's a lot of stuff, and partly because it's already provided on the Public Domain disks. We will, however, cover some salient points.

First on our list of "musts" was that the user interface had to be extremely simple but effective. The input for the user had to be expressed in Beats per Minute, since that's what is used in the musical world. (Sheet music often indicates tempo by metronome setting, and the number given is always understood to be in Beats per Minute.) Of course nothing in the computer itself understands Beats per Minute, so we have to provide a kind of "translation" from B/M to a number expressed in 60ths (or 50ths) of a second. We also wanted all this to be done as accurately as possible, so we used floating point numbers to do the calculations in this mathematical "translation" process. In the following, we're describing the U.S. version, then we'll briefly cover what's different in the "European" version.

The user gets a prompt "BEATS PER MINUTE" on the screen, with an allowed range indicated (15 through 500). He enters a number in that range, let's say 120, for example. As soon as he presses ENTER, we use an internal service called Convert String to Number to take the number right from the screen and convert it to a floating point number stored at FAC (>834A). What we're after is a number in 60ths of a second that will provide a limit count for the Vertical Interval timer at >8378. To get there, we take the number of 60ths of a second in one minute (3,600) expressed as a floating point number, and place that eight-byte quantity at location ARG (>835C) in memory. Now we use another built-in service called Floating Point Divide to divide 3,600 by 120. That gives us the number 30, or exactly half the number of 60ths in one second.

This is a number that can be used as the limit on our time count. We use the built-in Convert Floating Point to Integer routine (this conversion rounds the number correctly to the nearest integer), place that number into memory as an integer value and start making our "ticks." First, we clear the counter at >8378, then send a few bytes to the sound generator to make a sound. We enter a loop that checks constantly the number in >8378 against the limit number (30 in this case). The sound we've sent is silenced after

(See Page 10)

## Sidebar 41

```

BACK B *R11 ELSE RETURN

TIMEUP MOVB @>9802,R3 Reads address of grom byte sched-
uled to be read next
SWPB R3
MOVB @>9802,R3
SWPB R3
DEC R3 the true address in GROM

CZC @HX1FFF,R3 if address that was to be read was
>A000 then we can't
JNE TIMEU2 read one byte lower because >9FFF is
impossible in grom

MOVB R3,@>9C02 restore grom address.
SWPB R3
MOVB R3,@>9C02
JMP BACK and return

TIMEU2 DEC R3 want to read the preceeding byte
MOVB R3,@>9C02 GROM Write Address Register
SWPB R3
MOVB R3,@>9C02
SWPB R3
INC R3
CLR R2
MOVB @>9800,R2 Read preceeding GROM byte; with au-
toincrement grom
* pointer will be same as before we
tinkered with it
MOVB @>8373,R4 \
SRL R4,8 checking previous stack entry to
see if we've just popped from the stack
AI R4,>8302 /
C R3,*R4
JNE TIMEU1
CI R2,>0300 GPL for keyscan

TIMEU1 JNE BACK if EQ then keyscan just happened

MOVB @ENTER,@>8375
SOC @MASK,@>837C
TIMEU5 CLR @>83C4
JMP BACK

* DATA SECTION
*
ENTER BYTE >0D ASCII FOR "ENTER" KEY
HX0C BYTE >0C enter minus 1
MASK DATA >2000
HX1FFF DATA >1FFF
WS BSS 32 OUR WORKSPACE
CUMNUM DATA 0 COUNT NUMBER (60THS OF A SECOND)
LIMNUM DATA 0 LIMIT NUMBER
SIXTY BYTE 64,60,0,0,0,0,0,0 SIXTY IN F.P. NOTATION
CHKON DATA TIMER ADDRESS OF MAIN INTERRUPT
ANYKEY BYTE 32 KEY PRESSED VALUE
END

*
*
* EXTENDED BASIC TEST PROGRAMS
* FOR TESTING THE ABOVE WITH XB
* (LISTED IN 28 COLUMNS)
*
* FIRST, USING AN INPUT STATEMENT
* AND AN ACCEPT AT

10 CALL INIT
20 CALL LOAD("DSK1.TIMER/O")
30 CALL CLEAR

```

## THE ART OF ASSEMBLY—

(Continued from Page 9)

the first three counts, so it's just a brief "tick," after which the computer gives silence for the rest of the 30/60ths period. When that count finishes, the computer reclears >8378, sends another tick to the sound generator, and starts counting the next period.

This gives exactly the desired result, with two ticks per second, or 120 per minute, being produced by the monitor's speaker. By using the Vertical Interval as our source of timing, we've made this able to be accurate regardless of whether it's run on a standard TI, or a "bus modified" TI, or a Geneve at any of its clock rates.

### RANGE LIMITS

The timer we're using (at >8378) actually uses only one byte, at >8379. Thus, the number we can count has an upper limit of 255 counts before it "zeros" itself. Thus, there's a lower limit of 15 beats per minute, as that calculates to 240 counts (14 would wind up with a count of 257, and that's too big).

The other limit, 500, is arbitrary, but necessary so some "off time" will occur between ticks even at the fastest rate. With 500 entered, the number of counts for the interval would be 7.2 (rounds to 7) so that the sound would be on for three 60ths, then off for four 60ths, and so on. The program checks the value of the count before starting the ticks, and rejects the entry if the count is less than 7 or more than 240. The user will see the entry field clear, ready to accept a new input. The accepted range (15 through 500) should be enough for nearly any musical purpose, covering as it does from an extremely slow dirge through a super prestissimo.

While this ticking goes on, we flash the cursor onto the screen during each tick, so that there's a visual cue in case the musician's playing drowns out the sound of the tick. We also check for a keyboard input during the ticking, so that the user can stop the ticking by simply pressing any key. Pressing a key other than Function-9 during a stopped condition will just restart at the same rate. Pressing Function-9 while the ticks are stopped will clear the entry field so the user can enter a new rate.

### THE "EUR" VERSION

For our friends in Europe and Australia, the "EUR" version, called METROEUR, is designed to operate with their PAL video system, which has a 50 Hz vertical rate. The essential difference is that, for the 50 Hz system, there are 3000 vertical intervals per minute instead of the 3600 on the U.S. system. Thus, where, in the preceding discussion, we put 3600 as a floating point number into ARG, the EUR version places 3000 at ARG, then divides that by the user's input number. Because of this different number, the EUR version allows a lower bottom limit of 12 beats per minute, but that's the only obvious difference, except that the EUR version has a screen legend saying "EUROPEAN VERSION." This public domain product is available from the Lima Users' Group as disk 870A. (Contact the group c/o Dr. Charles Good at P.O. Box 647, Venedocia, OH 45894.)

### A TIMED INPUT FIELD

Ever since we started programming on the TI (in Extended BASIC) we've wished for some way to limit the time allowed for an INPUT or ACCEPT AT statement. As we've mentioned before,

```

40 CALL LINK("SETTIM",4.5)
50 CALL LINK("ACT")
60 INPUT "ANY STRING ":XS
70 IF XS="" THEN PRINT "TIME
'S UP" ELSE PRINT XS
80 DISPLAY AT(12,1):"ANY STR
ING"
90 CALL LINK("ACT")
100 ACCEPT AT(12,15):XS
110 IF XS="" THEN PRINT "TIM
E'S UP" ELSE PRINT XS
*
* SECOND XB TEST, USING CALL KEY
* THIS WILL RUN UNTIL STOPPED BY
* FUNCTION-4
*
10 CALL INIT
20 CALL LOAD("DSK1.TIMER/O")
30 CALL LINK("SETTIM",1)
40 CALL LINK("ACT")
50 CALL KEY(0,K,S):: IF S=0
THEN 50
60 N=N+1 :: PRINT N
70 GOTO 30

```

our friend Jim Peterson sometimes issued challenges to the assembly practitioners. Both he and Barry Traver have said that at times they'd like to be able to place a "time limit" on a user input as in some quiz games they've written in Extended BASIC. As you all know, once Extended BASIC starts executing an INPUT or ACCEPT AT statement, it just waits until the user is finished with his entry, even if an hour goes by. This can take some of the fun out of a game program, since the user could go to the library to look up the answer, and the computer would wait till he or she got back to answer the question.

As we were fooling around with this timing for the metronome application, this problem kept haunting our thoughts. Perhaps we could keep a time count through a "user interrupt," and then find some way to terminate the INPUT or ACCEPT command after a number of vertical intervals, without the user having typed anything from the keyboard.

Before we go further, let's confess that the "user interrupt" process is one that we rarely use, and don't really understand. Thus, getting the interrupt to do what we wanted was difficult. As you'll see in the sidebar, the "final product" is fairly simple. Two "entry points" are used with CALL LINK from Extended BASIC, plus the Interrupt code itself. Here's how it works.

Assuming the code shown in the sidebar has been assembled and loaded under Extended BASIC, the program that's using the code would perform a CALL LINK("SETTIM",TL), where TL is a number in seconds. Anything from 1 second through 546 seconds can be used for TL. Decimal quantities can be used, as, for example, 4.5 seconds. The limit 546 (about 9 minutes) is an absolute upper limit, beyond which the time counter simply won't work. (To see why, whip out your trusty calculator and multiply 546 by 60.)

The CALL LINK to SETTIM can be done at any place in the XB program. It sets the limit number in 60ths of a second, but  
(See Page 11)

## THE ART OF ASSEMBLY —

(Continued from Page 10)

doesn't start the count operating. To apply the time limit, insert a statement just before the input statement like this:

```
CALL LINK("ACT") :: INPUT "ANY STRING ":X$
```

This will "activate" the preset time limit for that input action. While the computer is waiting for input, it will cycle through interrupts, one of which will increment the counter in our user interrupt routine every 60th of a second. (SETTIM multiplies the number given in seconds by 60.) When the count equals or exceeds the limit, the interrupt will begin executing the code at label TIMEUP. This code puts the ASCII value for the Enter key into the key value address (>8375), and puts the value >20 into the GPL Status byte (>837C). The INPUT routine sees this condition and "assumes" the user has pressed Enter on the keyboard, so it exits from the INPUT statement. When we first started developing this routine, it worked perfectly with INPUT, but not with ACCEPT AT. ACCEPT AT would exit with an error. We discussed the matter with Harry Wilhelm, and he quickly figured out that the problem had to do with whether an actual key scan had just taken place before we tried forcing the Enter and >20 into place. As he has so many times, Harry came to our rescue, with a modification to our routine. Harry figured out how to check the GROM address to determine whether the TI has just done a key scan. Thus the forced "Enter" can be put in at exactly the right time, so no error occurs upon return from ACCEPT AT. With this modification, the timeout can be used with any kind of input routine, including CALL KEY. With this necessity to check the GROM address, the timing is not as accurate, as the Interrupt may have to wait several 60ths of a second before it hits the correct "window" to terminate the input cycle. Harry recommended that we discard the floating point calculations, since the accuracy won't be that good. We've left that part alone, however, so that the programmer can at least try for a fraction of a second time allowance. A SETTIM link with 4.5 seconds allowed will last longer, on average, than one with 4 seconds, and since it cost very few bytes to keep the floating point capability, we left it in. The sidebar is well annotated, so our regu-

lar readers should understand it without a line-by-line explanation. The business of checking the GROM is rather complicated, so we've taken Harry's word for how that works (see below). Two different XB Test programs are listed in the sidebar, one that uses both INPUT and ACCEPT AT, and another that uses CALL KEY.

### HARRY'S EXPLANATION

The idea behind the routine at TIMEUP is to determine if the GPL instruction SCAN has just happened. If it has, then loading >8375 with a key value and SOCing >837C with >2000 will be detected as a keystroke.

The GPL byte for SCAN is >03. Also during SCAN the current address of the GROM pointer is pushed onto the GROM substack, and then popped from the stack. If we find that one level deeper in the GROM substack points to the next byte scheduled to be read from GROM and that the byte before this byte is a >03 then SCAN has just occurred.

When we come to the routine at TIMEUP the GROM registers are set so that XB can read the next GROM byte from >9800 our program can read the address of that byte from GRMRA at >9802. When this address is read it is automatically incremented one byte higher than where the actual read would have taken place. Therefore, the address needs to be DECT'd to point to the byte that would have been read in the normal sequence of events. If the address is DECT'd it then points to the most recent byte to be read from GROM. Once that byte is read from >9800, the pointer will autoincrement to the same address normally expected by XB. One catch is that if reading the address at GRMRA can result in, say, >A001. With DECT this becomes >9FFF. With the autoincrement reading the GROM byte should result in the pointer becoming >A000; the expected result. However, after the autoincrement, the pointer is left pointing at >8000! Because of this you cannot read 1 byte lower than these addresses: >0000,>2000 >4000,>6000,>8000,>A000,>C000,>E000

Next month's topic is one that some of our readers have asked for, namely the Bit-Map mode of operation. We'll include a small sample program that gets nicely into and out of the Bit-Map mode, so you can see how that's done.

## Using BREAD XOP routine to read disk sectors

By JIM UZZELL  
©1994 DDI Software

The bread XOP routine basically allows you to read sectors

from a storage device. It also has an option that allows you to retrieve part of the file descriptor record, referred to as file ID.

This article and program is intended to illustrate the use of

BREAD XOP file ID option. When this program is run and an error occurs, the program will display zeros in all bytes. There are BBS' and user groups that have these XOP routines (See Page 12)

```
OBJECT CODE FILE BREAD-O
```

```
0009EDDIREA A0000B0006B0008B0001B0000B0A00B0000BF8F2B0000B00007F24FF      0001
A0012B0000B0000B0028A0018A0040B0000BC80BC0006B02E0BF000BC0207F324F      0002
A004CC0004B0201B0001B0202C0017B0420B2014BC1A0C0017B0586BD9A07F309F      0003
A0062C0040C0017B0200C0008B2C20C0002BD020C000AB1301B100BB02017F334F      0004
A0078BF8F2B0202C009EB0203B0100B06A0BF040BC2E0C0006B045BB02027F2C3F      0005
A008EC009EB0203B0100BDCA0C0040B0603B13F5B10FB7F5E8F                      0006
50042GO      7FDB8F                      0007
:            DDI SOFTWARE                  0008
```

## XOP—

(Continued from Page 11)

and are recommended if you are interested in programming in MDOS or creating assembly language programs for use in MY-BASIC. There are also articles that have been published over the years that provide detail information on the structure of file descriptor records.

In order to explain some of the displayed information, all references to "byte" will be the column on the left of the screen. A "general rule" that can be applied to certain bytes is

$$(byte\ 3) * (byte\ 4 \& 5) = (byte\ 8 \& 9)$$

The following are 2 examples of a "type" d/v 128 file;

1  $(byte\ 3 > 02) * (byte\ 4 \& 5 > 0010) = (byte\ 8 \& 9 > 0020)$

2  $(byte\ 3 > 02) * (byte\ 4 \& 5 > 000b) = (byte\ 8 \& 9 > 0015)$

(see pg 16 june 1994 for a hex to dec chart)

Example 1 matches but example 2 does not.

Example 2 says the final sector is not a full sector and by using byte 7 you can test for end of file. A "type" D/V 80 file does

not conform to this rule. In a D/V 80 file byte 8&9 will always be equal to  $(byte\ 3) * (byte\ 4 \& 5)$ , so we have to look at a different byte to find the end of file. Byte 6 will always have the number of bytes of the last sector. A zero would mean a full sector.

Now get out some different "type" files and do some exploring. A hardcopy feature is built in to the program so you can compare the printout to what you would see using a sector/editor on the file descriptor record for that file. The information on the right of screen is the location of the data in the file descriptor record.

I guess if there were a wish list, I would like to see this routine expanded to include the file location sector information, including fragmentation. With this added information, you could very quickly copy a file or do a selective backup of a hard drive or a directory of a specific "type" file, etc.

This program requires the object code file be in drive No. 1, or change line 120. Type in the object code file in MY-WORD and save as a fixed file with a name of BREAD-O.

## ASSEMBLY XOP

100 CALL GRAPHICS(3,1) :: DI	>0A	290 DATA DATE OF CREATION
M A(20),A\$(20)	200 DATA EXTENDED RECORD LEN	YYYY YYMM
110 DISPLAY AT(4,20)ERASE AL	GTH LSB	>14 U
L : "ASSEMBLY XOP >0A BREAD F	>0B	300 DATA DATE OF CREATION
ILE ID"	210 DATA TYPE OF FILE i.e. P	MMMd dddd
111 DISPLAY AT(7,20) : "ENTER	GM DV etc.	>15 A
PATH.FILENAME"	>0C	310 DATA TIME OF CREATION
112 ACCEPT AT(8,26) : X\$(1)	220 DATA NUMBER OF RECORDS P	hhhh hmmm
120 CALL INIT :: CALL LOAD("	ER SECTOR	>16 T
DSK1.BREAD-O")	>0D	320 DATA TIME OF CREATION
130 CALL LINK("GO",X\$( ))	230 DATA # OF SECTORS RESERV	mmms ssss
135 DISPLAY AT(1,3) : "BYTE";T	ED FOR FILE MSB	>17 E
AB(14);"XOP 0A BREAD FILE I	>0E F	330 DATA DATE OF LAST CHANGE
D";" of";TAB(40);X\$(1)	240 DATA # OF SECTORS RESERV	>18
140 FOR X=0 TO 19	ED FOR FILE LSB	340 DATA DATE OF LAST CHANGE
150 READ A\$(X) :: CALL PEEK(	>0F D	>19
9618+X,A(X))	250 DATA NUMBER OF BYTES IN	350 DATA TIME OF LAST CHANGE
160 DISPLAY AT(X+3,3) : X;TAB(	LAST SECTOR	>1A
8);">";SEG\$(HEX\$(A(X)),3,2)	>10 R	360 DATA TIME OF LAST CHANGE
;" ";A\$(X)	260 DATA LOGICAL RECORD LENG	>1B
170 NEXT X	TH	370 DATA MSB LEVEL 2 NOT AC
180 DISPLAY AT(24,10) : "ANY K	>11	TIVE CAN BE ANYTHING
EY TO EXIT";TAB(35);"PRT SCR	270 DATA FILE RECORD COUNT o	380 DATA MSB LEVEL 3 NOT AC
EEN IS ACTIVE"	r SECTORS USED (REVERSE	TIVE CAN BE ANYTHING
185 CALL KEY(0,K,S) :: IF S<	>12 E	
1 THEN 180 :: END	280 DATA FILE RECORD COUNT o	
190 DATA EXTENDED RECORD LEN	r SECTORS USED BYTE 8&9)	
GTH MSB	>13 Q	

## Former Bradenton president dies

William W. Saner, former president of the Bradenton TI Users Group in Florida, died in early September from prostate cancer.

At the time of his death, he was living in Springfield, Illinois. He had been a member of the Lima, Ohio, Users Group for a number of years.

Support  
MICROpendium  
Advertisers

# Extended BASIC

## Forget Me Knots and Reminders combine for useful computer memo system

By **ALFRED MALCOLM**

*The author is a member of the British Columbia 99er Users Group. The article and programs first appeared in the group's newsletter.—Ed.*

Forget Me Knots (FMK) is a handy memo program that will allow you to store and retrieve important dates (birthdays, doctor appointments, etc.). It also has limited graphics capability that can be used to frame significant events. Files created by the program Remind Me can be converted to the Forget Me Knots format. The companion program, Reminders, will read these files, and the ones you create with this program. Change the file name, Reminders, to the filename "LOAD," and your memos will be made available to you when you turn on your TI system.

System requirements to use these programs are a TI-99/4A, peripheral expansion box, disk controller, one disk drive, a monitor (or TV), and an Extended BASIC cartridge.

### THE FMK MENU

When Forget Me Knots is loaded, a menu will appear on your monitor. The flashing yellow cursor near the bottom left corner is waiting for your input. You have five choices.

#### 1. MONTHLY INPUT

This option will allow you to enter memos, etc., for any month of any year. Press the No. 1 key, followed by the Enter key. You will then be prompted to enter the month. Enter a number (1 to 12), and press the Enter key again. Now you must enter the year. Enter a 4-digit number, such as 1999. Press Enter. A file will now be created on the disk, under the format, `DSK1.MONTH/00/YEAR` (e.g. `DSK1.06/00/1994`). The "00" in the name will always indicate that this is a monthly reminder file.

The screen will change again, and is now ready for you to enter your reminders. There are 12 lines available. After each line you type, you must press the Enter key to save your information. Pressing the Enter key on an empty line will return you to the starting menu.

**Forget Me Knots (FMK) is a handy memo program that will allow you to store and retrieve important dates (birthdays, doctor appointments, etc.). It also has limited graphics capability that can be used to frame significant events.**

Each file created with FMK is 2 sectors long. When you press the enter key, the information is saved to DSK1. If you press enter on the first line, without entering any information, the file will not be saved.

Graphics can be used to enhance your memos. Hold down the CTRL key, and at the same time press any key from A to L (see graphics instructions below). Editing a file can be done with any word processor, such as TI-Writer.

#### 2. DAILY INPUT

Option 2 is for daily reminders. It will prompt you for the month (a number from 1 to 12), then the date (1 to 31), and lastly the year, (four numbers). When the information is entered, you will then have 12 lines to enter your reminders for that day. Pressing the Enter key will save the information to DSK1. The file will be saved as `DSK1.month/date/year` (e.g. `DSK1.06/30/1994`). Pressing the Enter key on a blank line, returns the program to the main menu. Graphics can be utilized by holding down the control key and pressing any key from A to L. As in option

1, editing may be done with a word processor.

#### 3. CONVERT REMIND ME

If you are using the program Remind Me, this option will convert Remind Me files to the FMK format. Before using this option, you must make sure your Remind Me Files are on DSK1, and that there is sufficient room for FMK files. You may also customize FMK to convert Remind Me Files from another source. More on that later. With the disk containing Remind Me Files in DSK1, press the No. 3 key. The program will now convert all Remind Me files to FMK files and then return to the main menu.

#### 4. DISPLAY A DATE

This option will display any FMK file. Press the No. 4 key and you will be prompted for the month. Enter a number from 1 to 12. Now you must enter the date (1-31 or 00 for monthly memos). Now enter a year in four digits.(1994,1998 etc.). If the month, date and year match a file created with FMK, that file will be displayed. If there are no reminders for the date you chose, FMK will let you know. When you have read the information displayed, press any key to return to the FMK menu.

#### 5. END PROGRAM

This option is self-explanatory. Press the No. 5 key and you will be returned to the TI title screen. By customizing FMK you can have the program load another program.

#### CUSTOMIZING FMK

If you plan to run this program from a hard drive or a RAMdisk, or any drive other than DSK1, you will need to make alterations. Remember, monthly reminders and daily reminders must be in the same location, so that FMK or Reminders can find and display them.

**Monthly Reminders** — If you want your monthly reminders written to some  
(See Page 14)

## FORGET ME KNOTS—

(Continued from Page 13)

place other than DSK1, carefully follow steps 1 to 6 below.

**Daily Reminders** — If you want your daily reminders written to or read from another source (not DSK1), then refer to steps 1 to 6 below. Follow directions carefully with one exception — type 430 instead of 230 (step 3).

**Converting Remind Me files** — If you have Remind Me files stored on a drive other than DSK1, then refer to steps 1 to 6 below. Follow directions carefully with one exception — type 530 instead of 230 (step 3).

**Displaying files** — If you are going to copy your FMK files to another source, then you will have to alter this program so

FMK can display them. Refer to steps 1 to 6 below. Follow directions carefully with one exception — type 860 instead of 230 (step 3).

**STEPS 1 TO 6**

1. Make a copy of your FMK disk. Place the copy in DSK1 without a write-protect tab. Load MFK.

2. With FMK loaded and running, hold down the FCTN key and press key No. 4.

3. Type 230. Do not press Enter. Hold down the FCTN key and press the X key (down arrow).

4. What you now see should be a program line of FMK. Hold down the function key again, and press the D key (left arrow). This will move the yellow cursor along the line. Stop the cursor over "DSK1.". If you passed over it, then hold down the function key again while pressing the S key (right arrow), to go back.

5. Type over DSK1. If your pathname is longer than DSK1. (e.g. HDS1.A1.), then press the function key and the No. 2 key together. Now you are in the insert mode. You can now type in the extra letters in your pathname. The line is very long, but it will accept more letters.

6. When you have your pathname typed in, press the Enter key. Now type SAVE "DSK1.LOAD" and press the Enter key again. Your modifications will be saved to your copy disk.

**Exiting to another program** — If you press key No. 5 from the FMK menu, the program will exit to the TI title screen. To exit to another program you must retype line No. 860. Following step 2 above, type 860 and press the Enter key. Type 860 again, then type RUN "PATH/FILE-HERE" (e.g. 860 RUN "DSK5.FWB"). Follow step 6 above to complete.

**USING GRAPHICS**

Graphics can be used to enhance your memos. After selecting option 1 or 2 from the FMK menu, you can begin to use graphics. Graphics are accessible by holding down the CTRL key and pressing any key from A to L. CTRL plus keys A to H produce a double-lined box. See Fig. 1 for

**Fig. 1****Graphics Control Keys**

CTRL plus A = Top left corner of frame  
 CTRL plus B = Top of frame  
 CTRL plus C = Top right corner of frame  
 CTRL plus D = Bottom left corner of frame  
 CTRL plus E = Bottom of frame  
 CTRL plus F = Bottom right corner of frame  
 CTRL plus G = Left side of frame  
 CTRL plus H = Right side of frame  
 CTRL plus I to L = various patterns

a list of graphics that you can create and the keypresses required.

**REMINDERS**

Reminders works in conjunction with a Triple Tech, MBP, or Myarc HFDC clock. Reminders reads the clock, then finds and displays files created (or converted) with the Forget Me Knots program.

To use Reminders you must have a TI system consisting of a TI-99/4A, peripheral expansion box, disk controller, disk drive, monitor, Extended BASIC cartridge, and a clock. Clocks include those supported by an MBP mounted inside a speech synthesizer or in the PEB, Myarc Hard and Floppy Disk Controller clock or Watchamacallit, or a Triple Tech card.

Load Reminders from DSK1. When Reminders is loaded, type RUN "DSK1.REMINDERS". If Reminders has been loaded correctly, you should see the monthly reminder screen. Your previously saved monthly reminders will now be dis-

played. When you have read your monthly reminders, press any key to load your daily reminders, or press the FCTN and No. 9 keys to bypass them.

**CUSTOMIZING REMINDERS**

Using a copy of the FMK/Reminders disk, load the Reminders program. If you want the Reminders program to read your reminder files from another drive, then you must change device reference in line No. 210. Change DSK1 in line 210 to whatever destination drive you want to use. Make sure you don't delete the quote marks.

If you want your Reminders program to load another program after completion, then you must change lines 400 and 330. You must also delete line 380. Type 400 RUN "DSK?.PROGRAM" (e.g. RUN "DSK5.MENU"). Press the Enter key. Delete line No. 380. Edit line 330 by changing THEN 380 to THEN 400. When you have done the changes, save the program to the disk you wish to load it from, using the filename REMINDERS. If you want your Reminders program to load automatically then save it as LOAD.

*Note* — If you like this program and find it useful, send a couple of bucks to the author's user group. The address is 216 10th Ave., New Westminster, British Columbia, Canada, V3L 2B2.

**4GETMEKNOT**

```

1 @=0 :: [=1 :: ]=2 :: _=3 ::
: \=4 :: GOTO 100 :: B$,C$,D
$,E$,F$,G$,H$,I$,J$,K$,L$ ::
A,B,C,D,E !067
2 CALL CHAR :: CALL CLEAR ::
CALL COLOR :: CALL HCHAR ::
CALL KEY :: CALL LOAD :: CA
LL SCREEN :: CALL VCHAR :: D
IM A$(12) :: !@P- !079
100 CALL CLEAR :: CALL SCREE
N(5) :: FOR A=[ TO 14 :: CALL
COLOR(A,16,5) :: NEXT A :: C
ALL COLOR(0,12,1) :: PRINT "
Forget-Me-Knots" :: :
: :: !197
120 PRINT " A Reminders Crea
tion Menu" :: : PRINT " 1
= Monthly Input " ::
2 = Daily Input " ::
3 = Convert Remind Me" !2

```

(See Page 15)

## FORGET ME KNOTS—

(Continued from Page 14)

```

27
130 PRINT " 4 = Display a
Date": " 5 = End Program"
:: PRINT : : : : :!203
150 B$="102854AA92442810" ::
CALL CHAR(123,B$):: CALL HC
HAR(9,_,123,27):: CALL HCHAR
(20,_,123,27):: CALL VCHAR(1
0,_,123,10)!255
160 C$="0000000000FF0000" ::
CALL CHAR(124,C$):: CALL HC
HAR(9,124,15):: CALL HCHAR
(5,9,124,15):: CALL VCHAR(10
,29,123,10)!033
162 CALL CHAR(129,"030404050
5050505"): :: CALL CHAR(130,"F
F00FF00000000000"): :: CALL CHA
R(131,"C02020A0A0A0A0A0"): ::
CALL CHAR(132,"0505050505040
403")!017
164 CALL CHAR(133,"000000000
0FF00FF"): :: CALL CHAR(134,"A
0A0A0A0A02020C0"): :: CALL CHA
R(135,"0505050505050505"): ::
CALL CHAR(136,"A0A0A0A0A0A0A
0A0")!128
166 CALL CHAR(137,"F00FF00FF
00FF00F"): :: CALL CHAR(138,"F
0AF0BF0CF0AF0BF"): :: CALL CHA
R(139,"1234567890ABCDEF"): ::
CALL CHAR(140,"AB0AB0AB0AB0A
B0A")!002
170 DISPLAY AT(\,6)SIZE(1):C
HR$(123):: DISPLAY AT(\,22)S
IZE(1):CHR$(123)!227
180 DISPLAY AT(22,7):"5 = Yo
ur Choice"!055
190 ACCEPT AT(22,7)SIZE(-1)B
EEP VALIDATE(DIGIT):B :: IF
B>5 THEN 180 !098
200 CALL CLEAR :: ON B GOSUB
230,370,520,790,1000 !215
210 GOTO 120 !199
220 ! (MONTHLY REMINDERS) !2
32
230 CALL CLEAR :: CALL A(F$)
:: CALL B(G$):: D$="DSK1." :
: E$=D$&F$&"/"&"00"&"/"&G$ :
: CALL CLEAR :: OPEN #]:E$ :
CALL C :: DISPLAY AT(21,7)
"Monthly Reminders" :: FOR
A=7 TO 18 :: ACCEPT AT(A,[])S
IZE(28):H$ :: PRINT #]:H$ !2
23
330 IF (H$="")OR(A=18)THEN C
LOSE #] :: GOTO 100 ELSE 340
!185
340 NEXT A :: CLOSE #] :: GO
TO 100 !136
360 ! (DAILY REMINDERS) !016
370 CALL CLEAR :: CALL A(F$)
:: DISPLAY AT(12,10):"DATE?"
:: ACCEPT AT(12,16)VALIDATE
(DIGIT)SIZE(1):I$ :: CALL B(
G$):: IF LEN(I$)=[ THEN I$="
0"&I$ !194
430 D$="DSK1." :: E$=D$&F$&
"/"&I$&"/"&G$ :: CALL CLEAR :
: OPEN #]:E$ :: CALL C :: DI
SPLAY AT(21,8):"Daily Remind
ers" :: FOR A=7 TO 18 :: ACC
EPT AT(A,[])SIZE(28):H$ :: PR
INT #]:H$ :: IF (H$="")OR(A=
18)THEN CLOSE #] :: GOTO 100
ELSE 510 !186
510 NEXT A !215
520 ! (CONVERTER) !172
530 CALL CLEAR :: CALL A(F$)
:: CALL B(G$):: D$="DSK1." :
: E$=D$&F$&"/"&G$ :: CALL CL
EAR :: CALL C :: DISPLAY AT(
12,10):"Files are Being Conve
rted": " Please Standby."
:: OPEN #]:E$ !190
630 FOR A=[ TO 12 :: IF EOF(
])THEN 760 !209
650 LINPUT #]:A$(A):: NEXT A
:: C=ASC(SEG$(A$(A-1),[,[,]))
:: J$=STR$(C):: IF LEN(J$)=[
THEN J$="0"&J$ !132
690 K$=D$&F$&"/"&J$&"/"&G$ :
: OPEN #]:K$ :: FOR A=[ TO 1
2 :: PRINT #]:A$(A):: NEXT A
:: CLOSE #] :: GOTO 630 !16
7
760 CLOSE #] :: GOTO 100 !04
7
780 ! (READ A FILE) !177
790 CALL CLEAR :: DISPLAY AT
(10,10):"MONTH? " :: ACCEPT
AT(10,17)VALIDATE(DIGIT)SIZE
(1):F$ :: IF LEN(F$)=[ THEN
F$="0"&F$ !059
830 DISPLAY AT(11,10):"DATE?
" :: ACCEPT AT(11,16)VALIDA
TE(DIGIT)SIZE(1):G$ :: IF LE
N(G$)=[ THEN G$="0"&G$ !132
860 DISPLAY AT(12,10):"YEAR?
" :: ACCEPT AT(12,16)VALIDA
TE(DIGIT)SIZE(1):I$ :: CALL
CLEAR :: D$="DSK1." :: J$=D$
&F$&"/"&G$&"/"&I$ :: OPEN #_
:J$ !096
910 IF EOF(____)THEN DISPLAY AT
(10,____):"No Reminders For Thi
s Date" :: CLOSE #_ :: DELET
E J$ :: GOTO 940 !066
920 FOR A=[ TO 12 :: LINPUT
#_]:A$(A):: PRINT A$(A):: IF
EOF(____)THEN CLOSE #_ ELSE NEX
T A !229
930 PRINT : : : : :!218
940 DISPLAY AT(23,____):" Press
Any Key When Ready" !086
950 CALL KEY(____,D,E):: IF E=@
THEN 950 !135
970 CALL CLEAR :: GOTO 100 !
006
1000 CALL LOAD(-31961,51)::
END !121
1001 !@P+ !062
1010 SUB A(F$):: DISPLAY AT(
10,10):"MONTH?" :: ACCEPT AT
(10,17)VALIDATE(DIGIT)SIZE(2
):F$ :: IF LEN(F$)=1 THEN F$
="0"&F$ !008
1011 SUBEND !168
1020 SUB B(G$):: DISPLAY AT(
14,10):"YEAR?" :: ACCEPT AT(
14,16)VALIDATE(DIGIT)SIZE(4
):G$ :: SUBEND !140
1030 SUB C :: DISPLAY AT(2,1
):" Forget-Me-Knots" :
: DISPLAY AT(4,1):"a program
by Alfred Malcolm" :: CALL
HCHAR(3,1,124,32):: CALL HCH
AR(5,1,124,32)!019
1040 SUBEND !168

```

## REMINDERS

```

100 @=0 :: [=1 :: ]=2 :: _=3
:: \=4 :: GOTO 110 :: A$,B$,
C$,E$,F$,G$,H$,I$,J$,K$,L$,
M$,N$ :: A,B,C,D,E,K,S :: CA
LL CHAR :: CALL CLEAR :: CAL
L COLOR :: CALL INIT :: CALL
KEY :: CALL SCREEN :: DIM D
$(12):: !@P- !205
110 CALL CHAR(129,"030404050
5050505"): :: CALL CHAR(130,"F
F00FF00000000000"): :: CALL CHA
R(131,"C02020A0A0A0A0A0"): ::
(See Page 16)

```

## FORGET ME KNOTS—

(Continued from Page 15)

```

CALL CHAR(132,"0505050505040
403")!017
120 CALL CHAR(133,"000000000
0FF00FF"):: CALL CHAR(134,"A
0A0A0A0A02020C0"):: CALL CHA
R(135,"0505050505050505")::
CALL CHAR(136,"A0A0A0A0A0A0A
0A0")!128
130 CALL CHAR(137,"F00FF00FF
00FF00F"):: CALL CHAR(138,"F
0AF0BF0CF0AF0BF"):: CALL CHA
R(139,"1234567890ABCDEF")::
CALL CHAR(140,"AB0AB0AB0AB0A
B0A")!002
135 CALL SCREEN(5):: FOR A=1
TO 12 :: CALL COLOR(A,16,5)
:: NEXT A :: FOR A=13 TO 14
:: CALL COLOR(A,9,5):: NEXT
A !117
140 CALL CLEAR :: CALL E(H$,
I$,J$)!Triple Tech !175
150 IF I$="" THEN 160 ELSE 2
10 !171
160 CALL F(H$,I$,J$)!Myarc C
lock !081
170 IF I$="" THEN 180 ELSE 2
10 !191
180 CALL G(H$,I$):: J$="94"
!MBP Clock !248
190 IF I$="" THEN 200 ELSE 2
10 !211
200 CALL CLEAR :: DISPLAY AT
(13,):"Error..No clock in s
ystem":" or clock not functi
oning.":" Is MBP Clock se
t?" :: END !230
210 K$="DSK1." :: A$="19" ::
N$=A$&J$ :: IF J$="00" THEN
A$="20" !168
220 L$=K$&I$&"/"&"00"&"/"&N$
:: DISPLAY AT(,6):"Monthly
Reminders" :: DISPLAY AT(,
11):I$&"/"&A$&J$ :: CALL C
: OPEN #]:L$ :: ON ERROR 240
!021
230 GOTO 250 !073
240 DELETE L$ :: CALL B :: C
ALL D :: GOTO 280 !073
250 FOR B=[ TO 12 :: LINPUT
#]:D$(B)!023
260 DISPLAY AT(B+6,())SIZE(32
):D$(B):: IF EOF(())THEN 270
ELSE NEXT B !221
270 CLOSE #] :: CALL D !038
280 CALL KEY(,K,S):: IF S=@
THEN 280 !010
290 IF K=15 THEN 390 !189
300 DISPLAY AT(23,())SIZE(28)
:"" !068
310 M$=K$&I$&"/"&H$&"/"&N$ :
: OPEN #_:M$ :: ON ERROR 330
!232
320 GOTO 340 !164
330 DELETE M$ :: CALL B :: G
OTO 380 !130
340 DISPLAY AT(,6):" Daily
Reminders" :: DISPLAY AT(,1
0):"For "&I$&"/"&H$ :: CALL
A !028
350 FOR B=[ TO 12 :: LINPUT
#_:D$(B)!025
360 DISPLAY AT(B+6,())SIZE(32
):D$(B):: IF EOF(())THEN 370
ELSE NEXT B !067
370 CLOSE #_ !252
380 CALL D :: CALL KEY(,K,S
):: IF S=@ THEN 380 !154
390 !@P+ !062
400 CALL LOAD(-31961,51)!108
410 SUB A :: FOR A=7 TO 19 :
: DISPLAY AT(A,1)SIZE(32):"
:: NEXT A :: SUBEND !146
420 SUB B :: DISPLAY AT(12,4
):"No Reminders Entered." ::
SUBEND !096
430 SUB C :: A$="F2F2F2F2F2F
2F2F2F2" :: CALL CHAR(123,A$
):: CALL HCHAR(5,1,123,32)::
CALL HCHAR(21,1,123,32):: S
UBEND !107
440 SUB D !174
450 DISPLAY AT(23,2):"Press
any key to continue" !215
460 SUBEND !168
470 SUB E(H$,I$,J$):: ON ERR
OR 490 !063
480 OPEN #9:"CLOCK" :: INPUT
#9:D$,E$,F$ :: CLOSE #9 ::
I$=SEG$(E$,1,2):: J$=SEG$(E$
,7,2):: H$=SEG$(E$,4,2)!064
490 SUBEND !168
500 SUB F(H$,I$,J$):: ON ERR
OR 520 !095
510 OPEN #9:"TIME",INTERNAL,
FIXED :: INPUT #9:D$,E$,F$,H
$,I$,J$ :: CLOSE #9 !140520
SUBEND !168
530 SUB G(H$,I$):: CALL INIT
:: CALL PEEK(-31158,A,B,C,D
,E):: C=C-6*INT(C/16):: E=E-
6*INT(E/16):: H$=STR$(C):: I
$=STR$(E)!152
540 IF LEN(H$)=1 THEN H$="0"
&H$ !224
550 IF LEN(I$)=1 THEN I$="0"
&I$ !227
560 SUBEND !168

```

## Fest West rewards early bird vendors

Fest West vendors sending in fees before Dec. 31 will receive two tables and two admission tickets for \$10 with \$10 for each additional table, according to information from the Southern California Computer Group, sponsors of the Feb. 18 event.

Vendors sending fees after that time will receive two tables and two admission tickets for \$20, with a \$10 fee for each additional table. Tickets for Fest West are \$5 apiece.

Fest West will be held at the Fabulous Inn, 2485 Hotel Circle

Place, San Diego, CA 92108. Special rates are offered to Fest West visitors Feb. 17-20. A single bed will be \$44 and two beds will be \$49, plus tax. Call 1-800-647-1903 (California) or 1-800-824-0950 (U.S.) and mention Fest West.

A number of prizes are offered to visitors to the event.

For further information, write the SCCG, P.O. Box 15235, San Diego CA 92195 or call the group's BBS at (619) 263-9135, User No. 25, password FEST.



## Chicago TI Faire

**PC99 shows great improvement**

By GARY COX

This year's Chicago TI Faire was not quite like the previous fairs. Attendance was not as heavy and few new products were presented. However, many of the vendors at the fair had a variety of used equipment at bargain-basement prices, and at least a few new products were presented.

One interesting item which caught my eye was the improvement in PC99 — a program which emulates the TI99/4A on a PC-compatible. The biggest improvement was in its speed. One example shown at the fair was the TI Demonstration Module running on a 50Mhz 486 at speeds faster than what the TI99/4A can produce. PC99 also has improved sound emulation for those who have Sound Blaster cards in their PCs. This allows PC99 to emulate the three voices of the TI sound chip. PC99 now can also make use of the Review Module Library feature built into a TI99/4A where up to 16 modules can be online and ready for use.

Also added is compatibility with Plato disks, as well as the addition of TI p-code card emulation. PC99 also now emulates the TI RS232 port so that programs that use the TI RS232 can operate through PC99. Finally a very neat debugger program has been added. It displays the TI screen in a small window on the PC display where access to all the debugging and editing features are available and updates are applied immediately to the running application. While this particular version of PC99 was not ready for release, Mike Wright of CaDD Electronics said the release of this version will be coming soon.

New from Tim Tesch of S&T Software and available from Cecure Electronics is CYA. CYA is a MDOS configuration program that allows the user to customize his MDOS operating system without resorting to a sector editor.

As for the SCSI controller project, a beta version of the DSR software for the Geneve has been distributed by Mike Maksimik to those on the beta test team.

Maksimik said that the object of the first official release of the SCSI DSR is simply to get a SCSI hard drive to function, fol-

lowed by the addition of a DSR for a SCSI CD-ROM, a SCSI tape backup and to eventually go with all SCSI devices in the system. David Nieters and Brad Snyder are continuing work on the DSR for the TI99/4A.

New from Beery Miller of 9640 News was MDOS version 2.20 which fixes some minor bugs such as RS232 problems and now adds support for "oddball" keyboards as well as support for PFM devices. Beery was also pushing his involvement on Gene (an online service) with the TI Round Table!

Elsewhere Bud Mills of Bud Mills Services had his usual variety of RAMdisks, P-GRAM cards and other products which were available in both in kit form and fully assembled form. Bud also had a variety of used TI equipment available.

Don Walden of Cecure Electronics said that Cecure has a variety of new projects in the works for the TI99/4A and Geneve. Cecure is now the only Texas Instruments authorized repair center for TI99/4A and CC40 equipment (it is also the authorized Myarc repair center). Special projects in the works include the possibility of remanufacturing (producing new) Myarc floppy and hard drive controller cards. Don stated though that the major problem in such a project is locating the chips needed to make the cards. Another project in the works is the possibility of manufacturing Hexbus-based equipment for the TI99/4A as Texas Instruments had intended.

Dave Nieters of Western Horizon Technologies demonstrated the installation of the AT-Keyboard and ROM upgrade kit whereby a standard AT keyboard can be used with the TI. Not only can you use the AT keyboard, the TI keyboard also remains active at the same time. Other products available from WHT include the Horizon Mouse, Turbo video for the Geneve, Digi-Port, monitors and PC-based equipment as well as repair services.

Elsewhere Larry Conner of L.L. Conner had his usual assortment of hard-to-find hardware, software and parts. Ted Kieper of Competition Computer had an extensive array of equipment and software for

sale, including a hard-to-find TI99/4 (not 4A) which I purchased myself. Ricky Botoms of RBD Enterprises had an assortment of hardware and software for sale. Rob Markus of Ramcharged Computers had his usual array of hardware and software available, including some hard-to-find software. Others having a variety of software and equipment include Program Innovators, the Chicago TI Users Group, Hoosier Users Group, Milwaukee Users Group, Mid-South (Memphis) TI Users Group, Will County Users Group, Computers & Crafts (Kevin Keller) and Aaron Busch from St. Louis.

One interesting item which caught my eye was from Berry Harmsen of the Dutch TI Users Group who brought over an interesting 80-column card made by Michael Becker in Germany. The 80-col-

(See Page 18)

**BUY - SELL - TRADE  
HARDWARE - SOFTWARE**  
TI Buyers Guide \$2

**WANTED; Disk Manager, MBX, Yahatzee,  
X-Basic, TI Recorders, Geneve, Hard Disk  
Controller, RS-232 Cards & Rare Items!**

**Dual ½ HT Kit complete with  
cables, screws, drill template  
& instructions \$59**

**PE Box Complete \$129**  
**Corcomp DD Cont \$180**  
**Star NX-1001 Printer \$175**  
**2400 Baud Modem \$79**  
**TI-99/4A Console \$45**  
**Rare - Myarc Micro**  
**Expansion with 32k, PIO,  
RS232, 2 ½ Ht DD Drives in  
1 case. 720k Storage! \$250**

**800-471-1600**  
(Nationwide & Canada ORDERS ONLY)  
**414-672-1600 Local/tech Support**  
**Huge Genuine TI Inventory Since 1982**  
**Competition Computer**  
2219 S Muskego Milwaukee WI 53215 Fax 414-672-8977

**Open Daily 9-5 Sat 10-3**  
**Bankcards, Discover, Checks & UPS/COD**

## CHICAGO—

(Continued from Page 17)

umn card which is called the EVPC (Enhanced Video Processor Card) incorporated not only a video port but a mouse port as well. It looked very well built and of ex-

tremely good quality.

This year's John Birdwell Award for outstanding commitment to the TI community was given to Tim Tesch for his involvement in supporting the TI community.

Finally, Charles Good of the Lima Ohio TI Users Group (P.O. Box 647, Venedocia, OH 45894) was promoting the next Multi-User Group conference to be held April 28-29 in Lima Ohio.

## Bud Mills says Geneve version of SCSI 'imminent'

By JOHN KOLOEN

Bud Mills spoke confidently that the long-awaited SCSI card is ready for the market, at least for the Geneve. He's taking orders because only one piece of software remains to be delivered — a formatter.

"The SCSI reads and writes fine," he said at the Chicago TI Faire. "But I'm not going to sell SCSI cards to anybody until we have the formatter that runs with the Geneve." The formatter is required to initialize the disk. However, according to Mills, owners of the Myarc Hard & Floppy Controller who also have a Western Horizon SCSI card in their PEB, can copy the contents of their disk to an unformatted SCSI drive and can then write and read to the SCSI. "Those who have a Myarc hard and floppy can take the MDOS Mike Maksimik has rewritten and, using one of his standalone programs, back up the hard and floppy drive to the SCSI drive and that will automatically format the drive." The standalone program is called SCSI Back. Mills doesn't recommend this because he wants users to format their SCSI drives using software that comes with the SCSI. Arrival date of the formatting software? "Imminent," Mills says.

Geneve users who want to buy a SCSI will also have to upgrade their MDOS software. A version that supports the SCSI, V2.50S, will be available from Mills as well as bulletin boards. This version of MDOS is specifically modified for the SCSI card.

Maksimik also spoke about the SCSI card and outlined additional requirements for use. These include:

- CRU address of >1200 (other devices using this address must be switched to a different address).
- 32K of static RAM, in addition to the 32K that's on the card, for 64K total (most Geneve's have this).

• A SCSI hard drive (virtually any size can be used, but Maksimik recommends not going overboard, especially for those who have never had a hard drive on their Geneve).

The SCSI supports both SCSI-1 and SCSI-2 protocols. Depending on the size of memory cache used by a SCSI, Maksimik predicts that a SCSI hard drive will be very close to a RAMdisk in terms of how quickly it accesses data and programs.

Mills predicts that a version of the SCSI for the TI will be ready by February. The problem isn't hardware so much as software. Maksimik said that MDOS provided much to build on while the TI presents more of a problem. Also, since he's more of a Geneve user than a TI user, his real incentive is to get the SCSI working on the Geneve before getting it to work on a TI.

Mills is selling the SCSI card for \$170. If you want to control SCSI floppy drives, add another \$100 for an FCI board. This board attaches to the SCSI card. There should be no problem running any other disk controller card along with the SCSI, as long as only the SCSI card is set for a >1200 CRU address.

### CYA DEMOED

Tim Tesch gave a demonstration of his remarkable CYA program by the end of which several observers were reaching for their checkbooks. Distributed by Cecure Electronics, CYA is a Geneve program that not only simplifies the task of configuring the Geneve but provides the user with as much flexibility as most users will probably ever need.

Don Walden, of Cecure, said that CYA "saves us a lot of time when we get a Geneve card sent in for repairs." Through CYA, a user can customize device names and designations without touching any hardware. Once the user has finished configuring his system, CYA writes the

changes to the SYSTEM/SYS file, making the new configuration the new default. At \$15, this program looks like something any Geneve user would want to have. A version for the SCSI version of MDOS is also expected.

### PC99 EMULATOR

Mike Wright showed his PC99 Emulator running on a 33Mhz 486 and a 90Mhz Pentium. The demonstrations were quite impressive, both in terms of the speed of the emulations and uncanniness of seeing a TI running on a PC. TI modules running on the emulator worked exactly as they do on a TI. One thing you won't see on a TI99/4A, however, is a TI99/4 title screen. PC99 not only displayed the 99/4 title screen, but ran the equation calculator that appears on the original 99/4 as one of three menu selections. TI dropped the calculator when it introduced the 4A. What this feat demonstrated was the authenticity of the emulation.

The only drawback to PC99 is that a user needs a fairly high-powered PC. The faster, the better.

### AT KEYBOARD DEMO

David Nieters showed the Western Horizon AT Keyboard interface, a circuit board that is installed in the TI console and allows the user to connect an AT-style keyboard directly to the console. Although he bravely demonstrated how to install the board, he had difficulty getting it to work. However, he assured the audience that the board works well. It should be noted that the system he had planned to use to demonstrate the board lacked a proper video cable and that at the last minute he was forced to try to install a board from scratch into a borrowed keyboard.

In addition to supporting an AT-keyboard, the interface also corrects several small bugs found in the AVPC card. The interface sells for \$65 without a keyboard.

# Investing for retirement

## An Extended BASIC program to help you make decisions

By BILL GASKILL

(This is reprinted from the September 1994 issue of Tidbits, the newsletter of the Mid-South 99 Users Group in Memphis, Tennessee.)

Long-time MICROpendium readers will remember a program and article that I wrote almost five years ago that dealt with determining the rate of return one was realizing on a mutual fund investment. In that article I talked briefly about *risk* and how it affected the choices you make in putting your investment portfolio mix together. The program that accompanies this article is designed to help you determine the mix of your investment portfolio, based upon the risk you are willing to accept and the number of years you have left until you plan to retire. But retirement is just a goal that I use. You could substitute the goal of retirement with kid's college education; vacation in Mazatlán; new car purchase; or any other short or long term goal that drives your desire to invest your money. Before you go out and blow your life savings based upon what the program tells you, though,

*Always consult with a finance professional before investing what to you is a substantial amount of money. I am not a finance professional, so please do not blindly accept what this program suggests as being the best advice for your situation.*

Investing for Retirement asks you a series of questions designed to determine whether your tolerance for risk in investing is low, moderate or high. A person with a low tolerance is not willing to accept much risk where money is concerned, where a person with a high tolerance is a risk taker, willing to gamble a little in order to chance greater returns on investments.

Once you have answered the questions involving risk, you need only provide the number of years until you plan to retire (or send the kids to college, take that trip to Mazatlán or buy that new car). Investing for Retirement will then display a suggest-

ed "mix" for your investment portfolio that will meet your risk level within the number of years you have left until your planned retirement. The mix is presented in terms of:

- SAFETY type investments, which are investments that all but guarantee a return, with little or no risk,
- INCOME type investments, which are those with a little risk, but a higher rate of return, designed to provide the investor with a flow of cash,
- CONSERVATIVE type investments, which are investments with a higher risk rating still, but with a rate of return that traditionally outperforms either Safety or Income type investments, and lastly,

**When you have completed the six screens that make up the program you will receive a suggested mix that is tied to the risk level you are willing to take, plus the number of years until your investment goal is to be realized.**

- AGGRESSIVE type investments, which are those that have the highest risk, but the greatest rate of return in the short-est amount of time.

All input into the program requires only a one-keypress response. The questions used to determine risk and retirement years are answered by pressing the A, B or C keys listed next to the question.

The Investing for Retirement screen is divided into two windows. The questions asked appear in the top three-quarters of the screen. The bottom quarter of the screen displays the directions needed to decide your answer. That's all there is to it.

When you have completed the six screens that make up the program you will receive a suggested mix that is tied to the risk level you are willing to take, plus the number of years until your investment goal is to be realized. For example, if you

are a person in your 20s, with a high tolerance for risk and 10-30 years to planned retirement, Investing for Retirement is going to suggest an investment portfolio mix that is a balance of Aggressive growth type investments designed to make higher returns, coupled with Conservative growth type investments that let you protect (conserve) some of your money to a greater degree than the Aggressive growth investments would.

The theory behind the program is that the more time you have until a goal is to be realized, the more risk you can afford to take in order to pursue higher investment returns. The reason being, you have more time to recover from a "bad year" or more time to absorb any fees that you may have to pay your investment broker or mutual fund manager.

### INVEST

```

1 !invest 10/18/93
   Bill Gaskill !218
100 CALL CLEAR :: CALL CHAR(
126, "00FF", 129, "FF0000000000
0000010101010101010100000000
000000FF8080808080808080") !1
37
110 CALL CHAR(140, "003844447
C444444", 141, "00782424382424
78", 142, "00384444040404438") !
219
120 CALL SCREEN(5) :: FOR I=0
TO 13 :: CALL COLOR(I, 16, 5)
:: NEXT I !154
130 FOR I=14 TO 14 :: CALL C
OLOR(I, 5, 16) :: NEXT I :: GOS
UB 860 !234
140 DISPLAY AT(2, 1) BEEP: "INV
ESTING FOR RETIREMENT-1" ::
! ON BREAK NEXT !012
150 FOR I=2 TO 30 :: CALL VC
HAR(4, I, 36, 15) :: NEXT I !092
160 DISPLAY AT(21, 3): "PRESS
ANY KEY TO BEGIN" :: CALL KE
Y(3, K, S) :: IF S=0 THEN 160 !
188
170 FOR I=30 TO 2 STEP -1 ::
CALL VCHAR(4, I, 32, 15) :: NEX
T I !198

```

(See Page 20)

## INVEST—

(Continued from Page 19)

```

180 DISPLAY AT(5,1):"This program will suggest a mix for your investment portfolio that is based on the level of investment" !111
190 DISPLAY AT(9,1):"risk you are willing to accept, and the number of years you have left before retirement." !142
200 DISPLAY AT(14,1):"Make sure that you consult with a finance professional before investing based upon" !174
210 DISPLAY AT(17,1):"the suggested portfolio mix produced by this program." !182
220 DISPLAY AT(21,2):"PRESS ANY KEY TO CONTINUE" :: CALL KEY(3,K,S):: IF S=0 THEN 220 !250
230 FOR I=2 TO 30 :: CALL VCHAR(4,I,32,15):: NEXT I !088
240 DISPLAY AT(5,1):"INVESTMENT CATEGORIES USED -----
-----" !113250
DISPLAY AT(7,1):"SAFETY-money market and guaranteed income funds." !061
260 DISPLAY AT(10,1):"INCOME-bonds and government type funds." !128
270 DISPLAY AT(13,1):"CONSERVATIVE GROWTH-equity and index type funds." !228
280 DISPLAY AT(16,1):"AGGRESSIVE GROWTH-equity or common stock funds." !157
290 DISPLAY AT(21,2):"PRESS ANY KEY TO CONTINUE" :: CALL KEY(3,K,S):: IF S=0 THEN 290 !065
300 FOR I=30 TO 2 STEP -1 :: CALL VCHAR(4,I,32,15):: NEXT I !198
310 DISPLAY AT(20,1):"Choose the option that most closely describes you right now. Press A, B or C." !202
320 DISPLAY AT(5,1):CHR$(140)&"-age 20-30, just beginning career, starting a family, saving for home, car, etc." !234
330 DISPLAY AT(9,1):CHR$(141)&"-age 30-45, saving for the kid's college education, paying for home, starting to invest for retirement." !214
340 DISPLAY AT(14,1):CHR$(142)&"-age 45-60, children are on their own, you're saving seriously for retirement." !166
350 CALL KEY(3,K,S):: IF S=0 THEN 350 ELSE IF K=15 THEN 100 ELSE IF K<65 OR K>67 THEN 350 !217
360 IF K=65 THEN V1=5 ELSE IF K=66 THEN V1=3 ELSE IF K=67 THEN V1=1 !030
370 DISPLAY AT(2,26)BEEP:"2" :: FOR I=3 TO 30 :: CALL VCHAR(5,I,32,14):: CALL VCHAR(20,I,32,4):: NEXT I !222
380 DISPLAY AT(20,1):"Choose the option that best describes what you would do with a tax free $100,000 lottery prize." !091
390 DISPLAY AT(5,1):CHR$(140)&"-invest in a home you can comfortably afford and that you'll live in for the rest of your life." !010
400 DISPLAY AT(10,1):CHR$(141)&"-pay off your current home mortgage and buy a newer, more upscale residence." !241
410 DISPLAY AT(14,1):CHR$(142)&"-invest in resort property that the developer promises will one day be a great success." !212
420 CALL KEY(3,K,S):: IF S=0 THEN 420 ELSE IF K=15 THEN 300 ELSE IF K<65 OR K>67 THEN 420 !046
430 IF K=65 THEN V2=1 ELSE IF K=66 THEN V2=3 ELSE IF K=67 THEN V2=5 !033
440 DISPLAY AT(2,26)BEEP:"3" :: FOR I=30 TO 2 STEP -1 :: CALL VCHAR(5,I,32,14):: CALL VCHAR(20,I,32,4):: NEXT I !076
450 DISPLAY AT(20,1):"Which scenario makes you feel best about your money?" !059
460 DISPLAY AT(5,1):CHR$(140)&"-you double your money in a stock fund in the first year." !161
470 DISPLAY AT(9,1):CHR$(141)&"-over the long term, your overall mix of investments protects you from loss and it outpaces inflation." !096
480 DISPLAY AT(14,1):CHR$(142)&"-your money market fund saves you from losing half your money in a market downturn." !001
490 CALL KEY(3,K,S):: IF S=0 THEN 490 ELSE IF K=15 THEN 370 ELSE IF K<65 OR K>67 THEN 490 !000
500 IF K=65 THEN V3=5 ELSE IF K=66 THEN V3=3 ELSE IF K=67 THEN V3=1 !036
510 DISPLAY AT(2,26)BEEP:"4" :: FOR I=2 TO 30 :: CALL VCHAR(5,I,32,14):: CALL VCHAR(20,I,32,4):: NEXT I !223
520 DISPLAY AT(20,1):"Of the following investment strategies, which do you consider most important?" !057
530 DISPLAY AT(5,1):CHR$(140)&"-avoid any loss of money, even if it means you don't earn the highest possible return." !086
540 DISPLAY AT(10,1):CHR$(141)&"-take a moderate risk for the chance to earn higher returns on your investment." !098
550 DISPLAY AT(14,1):CHR$(142)&"-tolerate the possibility of short-term losses in return for better long-term growth potential." !182
560 CALL KEY(3,K,S):: IF S=0 THEN 560 ELSE IF K=15 THEN 440 ELSE IF K<65 OR K>67 THEN 560 !212
570 IF K=65 THEN V4=1 ELSE IF K=66 THEN V4=3 ELSE IF K=67 THEN V4=5 !039

```

(See Page 21)

## INVEST—

(Continued from Page 20)

```

580 DISPLAY AT(2,26)BEEP:"5"
  :: FOR I=30 TO 2 STEP -1 ::
  CALL VCHAR(5,I,32,14):: CAL
L VCHAR(20,I,32,4):: NEXT I
!078
590 RF=V1+V2+V3+V4 !187
600 IF RF>0 AND RF<9 THEN RF
$="low risk investor." !223
610 IF RF>8 AND RF<15 THEN R
F$="moderate risk investor."
!025
620 IF RF>14 THEN RF$="high
risk investor." !080
630 DISPLAY AT(20,1):"How ma
ny years left until you reti
re?" !062
640 DISPLAY AT(5,1):CHR$(140
)&"-0 to 5." !108
650 DISPLAY AT(7,1):CHR$(141
)&"-5 to 10." !161
660 DISPLAY AT(9,1):CHR$(142
)&"-10 to 30." !211
670 CALL KEY(3,K,S):: IF S=0
THEN 670 ELSE IF K=15 THEN
580 ELSE IF K<65 OR K>67 THE
N 670 !061
680 IF K=65 THEN V5=1 :: V5$
="0 to 5 years before" !221
690 IF K=66 THEN V5=2" V5$=
"5 to 10 years before" !086
700 IF K=67 THEN V5=3 :: V5$
="10 to 30 years before" !06
6
710 DISPLAY AT(2,26)BEEP:"6"
  :: FOR I=3 TO 30 :: CALL VC
HAR(5,I,32,14):: CALL VCHAR(
20,I,32,4):: NEXT I !226
720 DISPLAY AT(5,1):"The cho
ices indicate you're a ";RF$
!080
730 DISPLAY AT(8,1):"With ";
V5$:"retirement, coupled wit
h that level of risk, you
r" !162
740 DISPLAY AT(11,1):"invest
ment portfolio should probab
ly consist of:" !075
750 IF RF<9 AND V5=1 THEN DI
SPLAY AT(14,1):"70% Safety,"
:"30% Income." !233
760 IF RF<9 AND V5=2 THEN DI
SPLAY AT(14,1):"40% Safety,"
:"50% Income,":"10% Conserva
tive growth." !003
770 IF RF<9 AND V5=3 THEN DI
SPLAY AT(14,1):"20% Safety,"
:"50% Income,":"30% Conserva
tive growth." !004
780 IF RF>8 AND RF<15 AND V5
=1 THEN DISPLAY AT(14,1):"30
% Safety,":"30% Income,":"40
% Conservative growth." !068
790 IF RF>8 AND RF<15 AND V5
=2 THEN DISPLAY AT(14,1):"15
% Safety,":"20% Aggressive g
rowth,":"35% Income,":"30% C
onservative growth." !124
800 IF RF>8 AND RF<15 AND V5
=3 THEN DISPLAY AT(14,1):"35
% Aggressive growth,":"40% I
ncome,":"25% Conservative gr
owth." !185
810 IF RF>14 AND V5=1 THEN D
ISPLAY AT(14,1):"15% Safety,"
:"15% Aggressive growth,":"
30% Income,":"40% Conservati
ve growth." !103
820 IF RF>14 AND V5=2 THEN D
ISPLAY AT(14,1):"40% Aggress
ive growth,":"20% Income,":"
40% Conservative growth." !1
55
830 IF RF>14 AND V5=3 THEN D
ISPLAY AT(14,1):"50% Aggress
ive growth,":"50% Conservati
ve growth." !233
840 DISPLAY AT(20,1):"Press
FCTN 8 to do another retire
ment scenario." :: DISPLAY A
T(23,1):"Copyright 1993 Bill
Gaskill" !137
850 CALL KEY(3,K,S):: IF S=0
THEN 850 ELSE IF K=6 THEN 1
00 ELSE END !146
860 CALL HCHAR(1,1,129,30)::
CALL VCHAR(1,1,130,4):: CAL
L HCHAR(3,2,131,29):: CALL V
CHAR(1,31,132,4)!228
870 CALL HCHAR(19,1,129,30):
: CALL VCHAR(1,1,130,24):: C
ALL HCHAR(24,2,131,29):: CAL
L VCHAR(1,31,132,24):: RETUR
N !194

```

## 1994 TI FAIRS

## OCTOBER

**9th International TI-Meeting**, Oct. 14-16, Kirch l. Gemein-dehaus Roshorf, German, sponsored by TI-Club Goettingen. For information, contact Jörg Kirstan, Mengershäuser Weg 5, D-37124 Rosdorf, Germany, tel. 01551/781153; Reinhard Obuch, Keplerstr. 5, D-37085 Göttingen, Germany, tel. 0551/46405; or Hans-Hartmut Kortry, Grüner Weg 10, D-37181 Hardegsen, Germany, tel. 05505/1470.

## NOVEMBER

**The TI International World's Faire**, Nov. 12, Holiday Inn, Gurnee, Illinois. Sponsored by Chicago and Milwaukee users groups. For information, contact Don Walden (414) 679-2336.

## 1995 TI FAIRS

## FEBRUARY

**Fest West '95**, Feb. 18, Fabulous Inn, San Diego, California. Contact Southern California Computer Group, P.O. Box 152535, San Diego, CA 92195, or call the SCCG BBS, (619) 263-9135, User No. 25, password FEST

## APRIL

**Lima Multi Users Group Conference**, April 29, Reed Hall, Ohio State University at Lima. Contact Lima Users Group, P.O. Box 647, Venedocia OH 45894, or call Charles Good (evenings) at (419) 667-3131 or Internet cgood@lima.ohio-state.edu.

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.

# Vega

## Can you figure out how to communicate with the Vegans?

By **JIM PETERSON**

Vega was written by the late Jim Peterson. It's an example of the kind of educational and entertaining program that he wrote so well.

The storyline finds you crashed on a strange planet. While the inhabitants are friendly, you have to figure out a way to communicate with them. This is the challenge of the game.

Vega requires Extended BASIC and looks best on a color monitor.

### VEGA

```

1 DATA 128,7E81BDA5BDBD8166,
129,2424242424242424,130,242
42424242424E7,131,7F80808080
808080,132,DB24,133 !133
2 DIM N(35),CC(35)!079
10 GOTO 100 !179
11 K,S,J,Q$,ST,PL,RP,RW,TC,P
T,P2,M$,K$,N(),CC(),D,RPC,S$,
T$,TT$,TG$,LG,X,W,WR(),PR,P
C,P$,RC(),X$,A,B,C,RR !206
30 CALL CLEAR :: CALL VDPUTI
L2 :: CALL KEY :: CALL CHAR
:: CALL SCREEN :: CALL COLOR
:: CALL VCHAR :: CALL SOUND
:: CALL HCHAR !053
40 !@P- !064
100 CALL CLEAR :: CALL VDPUT
IL2 :: CALL KEY(3,K,S):: ON
WARNING NEXT :: CALL CHAR(94
,"3C4299A1A199423C")!106
110 REM VEGA programmed b
y James W. Peterson 2/83 rev
ised 6/83, converted to VDP
TIL2 XBasic 7/86 !122
120 REM COPYRIGHT 1983 Tiger
cub Software TCX-1086 156 C
ollingwood Ave., Columbus Oh
io 43213 !124
130 REM REPRODUCTION PROHI
BITED. DELETION OF COPYRIGHT
NOTICE PROHIBITED. !108
140 !1131
150 CALL SCREEN(11):: FOR J=
1 TO 15 :: CALL COLOR(J,2,11
)!100
160 NEXT J :: CALL COLOR(16,
11,11):: CALL VCHAR(1,31,1,9

```

```

6):: CALL SCREEN(5):: DISPLA
Y AT(3,12):"VEGA": : : " TC
X-1086":TAB(11);" ^ Tigercub
Software" !198
170 DISPLAY AT(12,1):" Do yo
u need instructions?": " Type
Y(yes) or N(no) " :: ACCEPT
AT(13,23)VALIDATE("YN")SIZE(
1):Q$ :: CALL VCHAR(1,3,32,6
72):: IF Q$="N" THEN 270 !08
0
180 DISPLAY AT(1,1):" Your
space ship has": " crashed o
n the 3rd planet": " of Vega,
and you are": " marooned for
life unless": " you can repa
ir it." !242
190 DISPLAY AT(7,1):" The
Vegans are friendly": " and i
ntelligent, and will": " help
you if you can commu-" !036
200 DISPLAY AT(10,1):" nicat
e with them. They": " have al
ready learned your": " langua
ge. However, the" !136
210 DISPLAY AT(13,1):" Vegan
language consists": " only o
f tones, and they": " can onl
y pronounce your": " language
as tones." !148
220 DISPLAY AT(17,1):" The
se strange creatures": " also
have a panel on their": " fo
rehead, which flashes" !058
230 DISPLAY AT(20,1):" with
colors corresponding": " to t
he tones they speak.": " To
uch any key to turn page" !0
00
240 CALL KEY(0,K,ST):: IF ST
<1 THEN 240 !222
250 CALL VCHAR(1,3,32,672)::
DISPLAY AT(3,1):" The Veg
an teacher will": " pronounce
and display any": " word you
type into the" !129
260 DISPLAY AT(6,1):" comput
er. Then you must": " recogni
ze the word among a": " rando
m selection of other": " word
s." !210

```

```

270 DISPLAY AT(12,1):" Do
you want to study": " (1) b
y yourself?": " (2) with yo
ur crew members?": " Type 1
or 2" :: ACCEPT AT(18,14)VA
LIDATE("12")SIZE(1):PL :: IF
PL=1 THEN 290 !042
280 DISPLAY AT(20,1):" How m
any crew members?" :: ACCEPT
AT(20,25)VALIDATE("12345678
")SIZE(1):PL :: PL=PL+1 !240
290 DISPLAY AT(12,1):" How m
any times can you ask": " to
have a word repeated?": " :
: : : : : :: ACCEPT AT(13,
27)VALIDATE(DIGIT)SIZE(1):RP
!086
300 DISPLAY AT(12,1):" How m
any random words will": " be
used?": " : : ACCEPT AT(13,11
)VALIDATE("12345678")SIZE(1
):RW !095
310 DISPLAY AT(12,1):" Do yo
u want to practice": " (1)
Tone and color?": " (2) Tone
only?": " (3) Color only?":
: : : DISPLAY AT(18,1):" Type
number of choice" !107
320 ACCEPT AT(18,24)VALIDATE
("123")SIZE(1):TC :: DISPLAY
AT(12,1):" Wait, please....
": " : : : : : : : :: GOSUB 86
0 !116
330 IF PL=1 THEN 390 !216
340 PT=PT+1 :: IF PT<=PL THE
N 360 !102
350 PT=1 !092
360 P2=PT+1 :: IF P2<=PL THE
N 380 !054
370 P2=1 !058
380 M$=" Player # "&STR$(PT)
&" type word for player #
"&STR$(P2)&" and Enter" ::
GOTO 400 !161
390 M$=" Type word and Enter
" !025
400 CALL COLOR(16,11,11):: G
OSUB 840 :: GOSUB 830 :: ACC
EPT AT(23,4):K$ :: GOSUB 840
:: GOSUB 420 !143

```

(See Page 23)

## VEGA—

(Continued from Page 22)

```

410 GOTO 530 !099
420 FOR J=1 TO LEN(K$)!240
430 IF TC=3 THEN 450 !017
440 N(J)=ASC(SEG$(K$,J,1))*6
  :: IF TC=2 THEN 460 !200
450 CC(J)=INT((ASC(SEG$(K$,J,1))-58)/2)!003
460 NEXT J !224
470 FOR J=1 TO LEN(K$):: IF
TC=3 THEN 490 !171
480 CALL SOUND(-99,N(J),0)::
  IF TC=2 THEN 520 !155
490 CALL COLOR(16,CC(J),11):
: IF TC=1 THEN 520 !046
500 FOR D=1 TO 20 !101
510 NEXT D !218
520 NEXT J :: CALL COLOR(16,
11,11):: RETURN !177
530 IF RP>0 THEN 550 !128
540 M$=" Hit any key except
Enter" :: GOTO 560 !239
550 M$=" Hit Enter to repeat
or any other to continue"
!126
560 GOSUB 840 :: GOSUB 830 !
174
570 CALL KEY(0,K,ST):: IF ST
<1 THEN 570 !042
580 IF K<>13 THEN 620 !099
590 RPC=RPC+1 :: IF RPC>RPT
HEN 620 !119
600 GOSUB 470 :: IF RPC=RP T
HEN GOSUB 840 :: M$="Hit any
key to start" :: GOSUB 830
!163
610 GOTO 570 !139
620 GOSUB 840 :: S$=K$ :: T$
="AZGYUJBNMOILPRQWEGYBADRCFG
RUYHJBNMOIQWESDXCZFTYBGHNBJI
UTYOIAWQAQAWSZDXVBHYNJKKHGCF
GYMKLKZAQWRFCGVHBJYH" !043
630 TT$="KIJUYUNJBHVGRFDTQZO
PCOZYUBMJQEDHUYUNKLPQWAGHCJR
KIVBGERXGJKNVCASEQWRTPONMDFU
HZAMPLPODFGHBHJDRYYYYHUIBCD"
  :: TQ$=T$&TT$ :: LG=LEN(S$):
: RANDOMIZE :: X=INT((RW+1)*
RND+1)!151
640 FOR W=1 TO X-1 :: K$=SEG
$(TQ$,INT(180*RND+1),LG):: C
ALL HCHAR(2,16,W+48):: GOSUB
420 !016
650 NEXT W :: K$=S$ :: CALL
HCHAR(2,16,W+48):: GOSUB 420
!041
660 FOR W=X TO RW :: K$=SEG$
(TQ$,INT(180*RND+1),LG):: CA
LL HCHAR(2,16,W+49):: GOSUB
420 !004
670 NEXT W :: M$=" Type numb
er of word" :: GOSUB 840 ::
GOSUB 830 !229
680 CALL KEY(0,K,ST)!015
690 IF (ST<1)+(K<49)+(K>57)T
HEN 680 !082
700 CALL HCHAR(2,16,32):: GO
SUB 840 :: IF K-48=X THEN 76
0 !237
710 M$=" Wrong! Word was # "
&STR$(X):: CALL COLOR(16,2,1
):: CALL SOUND(1000,30000,30
,30000,30,410,30,-4,0):: GOS
UB 840 :: GOSUB 830 :: WR(PT
)=WR(PT)+1 :: PR=4+PT !166
720 IF PT<P2 THEN 740 !200
730 PR=4 !093
740 PC=26 :: P$=STR$(WR(PT))
:: GOSUB 1030 !021
750 GOTO 790 !104
760 M$=" RIGHT!" :: CALL COL
OR(16,14,11):: GOSUB 840 ::
GOSUB 830 :: GOSUB 1060 :: R
C(PT)=RC(PT)+1 :: PR=4+PT ::
  IF PT<P2 THEN 780 !174
770 PR=4 !093
780 PC=23 :: P$=STR$(RC(PT))
!034
790 GOSUB 1030 !090
800 IF RC(PT)>11-PL THEN 108
0 !041
810 FOR D=1 TO 500 !153
820 NEXT D :: CALL HCHAR(9,7
,135):: CALL HCHAR(9,8,136):
: RPC=0 :: GOTO 330 !044830
DISPLAY AT(20,4):M$ :: RETUR
N !011
840 FOR J=20 TO 24 :: CALL H
CHAR(J,4,32,27)!034
850 NEXT J :: RETURN !234
860 !!131
870 DATA FE01010101010101,13
4,8080808080808080,135,00000
0030C302020,136,000000C0300C
0404,137,8080402020408080 !2
25
880 DATA 138,80804020204080F
F,139,0106080808080601,140,0
00000003C424281,141,00000000
000000FF,142 !082
890 DATA 8040402020180601,14
3,80402010080810E0,144,80404
0C00808080F,152,3F7FFFFFFFFF
7F3F !021
900 DATA 153,FFFFFFFFFFFFFFF
F,154,FCFEFFFFFFFFFEFC !221
910 DATA 155,2020300C03,156,
04040C30C0 !043
920 DATA 3,6,128,3,9,128,4,6
,129,4,9,129,5,6,130,5,9,130
,6,5,131,6,7,132,6,8,132,6,1
0,133 !205
930 DATA 7,5,134,7,6,152,7,7
,153,7,8,153,7,9,154,7,11,13
4,8,5,134,8,11,134,9,5,134,9
,7,135,9,8,136 !152
940 DATA 9,11,134,10,5,137,1
0,11,137,11,5,134,11,11,134,
12,5,137,12,11,137,13,5,134,
13,11,134 !130
950 DATA 14,5,137,14,11,137,
15,5,134,15,11,134,16,5,137,
16,11,138,16,12,140,16,13,14
1,16,14,140 !252
960 DATA 16,15,141,16,16,140
,16,17,141,16,18,140,17,4,13
9,17,10,139,17,19,142,18,5,1
42,18,6,141,18,7,141 !196
970 DATA 18,8,141,18,9,141,1
8,10,141,18,11,144,18,12,141
,18,13,140,18,14,141,18,15,1
40,18,16,141 !059
980 DATA 18,17,140,18,18,141
,18,19,140,18,20,143,3,20,80
,3,21,76,3,22,35,3,24,82,3,2
7,87 !180
990 FOR J=1 TO 22 :: READ X,
X$ :: CALL CHAR(X,X$)!010
1000 NEXT J :: CALL VCHAR(1,
3,32,672):: FOR J=1 TO 67 ::
  READ A,B,C :: CALL HCHAR(A,
B,C)!125
1010 NEXT J :: RR=4 :: FOR J
=1 TO PL :: CALL HCHAR(RR,21
,J+48):: RR=RR+1 !152
1020 NEXT J :: RETURN !234
1030 FOR J=1 TO LEN(P$)!245
1040 CALL HCHAR(PR,PC+J,ASC(
SEG$(P$,J,1)))!143
1050 NEXT J :: RETURN !234
1060 CALL HCHAR(9,7,155):: C
ALL HCHAR(9,8,156):: FOR S=1
TO 6 :: CALL SOUND(-99,INT(
3000*RND)+1000,5)!090

```

(See Page 24)

# VEGA—

(Continued from Page 23)

```

1070 NEXT S :: RETURN !243
1080 M$="PLAYER # "&STR$(PT+
1) & " HAS LEARNED THE LANGU
AGE! NOW YOU CAN GO HOME!
" :: GOSUB 840 :: GOSUB 830
:: FOR D=1 TO 1000 !057
1090 NEXT D :: END !231
1100 !"VDP UTILITY II" !183
1110 !BY JOHN BEHNKE !058
1120 !CHICAGO IL !038
1129 !@P+ !062
1130 SUB VDPUTIL2 :: CALL CL
EAR :: CALL INIT :: CALL LOA
D(8196,63,232)!084
1134 GOTO 1140 !199
1137 CALL CLEAR :: CALL INIT
:: CALL LOAD !251
1138 !@P- !064
1140 CALL LOAD(16360,80,79,7
5,69,82,32,38,12,80,79,75,69
,86,32,37,164,80,69,69,75,86
,32,37,36)!175
1150 CALL LOAD(9491,100)!168
1160 CALL LOAD(9508,2,224,37
,20,3,0,0,0,2,0,0,100,200,0,
37,18,4,192,2,1,0,1,4,3,2,32
,12,4,32)!082
1170 CALL LOAD(9536,32,24,18
,184,192,32,131,74,2,1,37,0,
208,160,131,18,9,130,2,34,25
5,255,4,32,32,44)!080
1180 CALL LOAD(9562,4,197,20
9,34,36,255,9,132,19,21,4,19
5,60,224,37,18,200,5,131,76,
200,5,131,78,200,5)!198
1190 CALL LOAD(9588,131,80,2
,5,64,0,161,68,2,131,0,1,17,
6,2,5,65,0,161,67,6,196,200,
4,131,76)!188
1200 CALL LOAD(9614,200,5,13
1,74,4,192,192,66,5,129,4,37
,254)!154
1210 CALL LOAD(9636,2,224,37
,20,3,0,0,0,4,192,2,1,0,1,20
0,1,37,18,4,32,32,12,4,32,32
,24,18,184)!139
1220 CALL LOAD(9664,200,32,1
31,74,37,0,184,32,131,18,37,
19,2,3,0,2)!150
1230 CALL LOAD(9680,4,192,19
2,67,4,32,32,12,4,32,32,24,1
8,184,216,224,131,75,37,0,5,
131,136,3)!090
1240 CALL LOAD(9704,37,18,22
,242,192,32,37,0,2,1,37,2,19
2,131,2,34,255,254,4,32,32,3
6)!037
1250 CALL LOAD(9726,4,192,21
6,0,131,124,2,224,131,224,4,
96,0,112)!150
1260 CALL LOAD(9740,3,0,0,0,
4,192,2,1,0,1,4,32,32,12,200
,32,131,74,37,18,2,1,0,2,4,3
2,32,12,4,32)!112
1270 CALL LOAD(9770,32,24,18
,184,192,32,131,74,208,32,37
,19,4,32,32,48,4,91):: CALL
LOAD(8194,39,04)!246
1271 !@P+ !062
1272 SUBEND !168
1280 SUB CHAR(A,A$):: L=LEN(
A$):: A$=A$&RPT$( "0",16-L)::
FOR I=1 TO 16 STEP 2 :: A1$
=SEG$(A$,I,1):: A2$=SEG$(A$,
I+1,1):: IF A1$<" " THEN A1=
VAL(A1$)*16 ELSE A1=(ASC(A1$
)-55)*16 !031
1290 IF A2$<" " THEN A1=A1+V
AL(A2$) ELSE A1=A1+ASC(A2$)-5
5 !135
1300 CALL LINK("POKEV",767+8
*A+(I+1)/2,A1)!083
1310 NEXT I :: SUBEND !009
1320 SUB COLOR(A,B,C):: CALL
LINK("POKEV",2063+A,(B-1)*1
6+C-1):: SUBEND !165

```



**Want to contact  
MICROpendium by phone?  
Call us Saturday mornings,  
from 9 a.m. to noon, Central  
Standard Time.  
The number is  
512-255-1512.**



### UPGRADES

Includes parts and labor

EXTRA 64K VIDEO-ON SOCKETED MEMORY	\$25
EXTRA 64K VIDEO-NON SOCKETED MEMORY	\$35
RESET SWITCH	\$10
CLOCK AND BATTERY HOLDER	\$7.50
HARD DRIVE BYPASS for 4A	\$15
<small>Eliminates HARD DRIVE time out, when HARD DRIVE not connected</small>	
HFDK 250 KB upgrade to 500KB	\$20
<small>NOTE: This is used for 1.44 MEG FLOPPY and tape back-up QIC 40/80</small>	

### PFM

**PFM \* Programmable Flash Memory \* PFM  
UPDATE YOUR OWN BOOT BIOS WITH PFM  
PFM BOOT BIOS VER.#2.0 SYSTEM/SYS\$75  
PFM+ WITH 128K FLASH DISK VER. 3.0 \$125  
MEMORY**

**SALE • SALE • SALE • SALE • SALE**  
128K X 8 LP 80ns MEMORY HITACHI \$17.95  
Look again...\$17.95

Quantities limited to 8 per customer.  
SORRY, NO DEALER SALES.

### CRYSTAL SOFTWARE

MIDI MASTER 99 VER. 2.3 W/CABLE	\$45
MIDI ALBUM 99	\$10
SONG DISKS 1-4 SSSD	EACH\$ 4
ALL 4 SONG DISKS	\$15
<small>NOTE: MIDI 99 REQUIRES XB, EA OR MINI MEMORY and 32K AND RS232 MIDI ALBUM REQUIRES MIDI MASTER 99 &amp; SUPERCART, MINI MEMORY OR GRAM DEVICE</small>	

### SALE • SALE • SALE • SALE • SALE CRYSTAL SOFTWARE

TEAC 720K/1.44M 3.5" MODEL FD 235	\$44.95
5.25 MOUNTING KIT	\$4.95

VOICE 414 679-4343 BBS 414 442-9669  
MASTER CARD or VISA ORDERS USE OUR TOLL FREE ORDER ONLY PHONENUMBER:

**1-800-952-9640 ORDERS ONLY!  
SECURE ELECTRONICS INC.**

South 74 West 17000 JANESVILLE ROAD P.O. BOX 132, MUSKEGO, WI 53150-0132



## MICRO-REVIEWS

# Newsletter Editor and Formatter, USVBA Power Volleyball

By CHARLES GOOD

## Newsletter Editor and Formatter by Bill Gaskill

This software can create and print newsletters one page at a time. Text is printed in two columns with optional right justification, and each page is numbered at the bottom. Designated text lines can be printed double strike (or "emphasized") and thus made to stand out on the printed page, and you can have a full width enlarged centered headline printed at the top of any or all pages. Each page, with all these features, is printed with a single pass through your printer. You get good-looking nicely formatted all text (no graphics) newsletter pages without cutting and pasting.

NEF, as the author calls this software, has its own text editor. The software is written in Extended BASIC, but because it uses Brad Snyder's 40-column assembly routines you get a 40-column screen in which to enter your text. The actual text you enter and print is a maximum of 36 columns wide. To put your text into memory for formatting and printing you can either press "L" to load in a text file you previously saved using NEF, import (press FCTN/I) a TI-Writer file, or type your text directly into NEF's very basic text editor. Using the NEF editor you get word wrap and you can insert blank lines between or delete lines from existing text lines. You can also insert or delete characters within a line, but, unlike in TI-Writer, insertions and deletions within a line do not affect text in other lines of your document.

If you don't like the limitations of the NEF text editor, you also have the option of importing text created with a word processor. NEF imports word processor files, and saves text back to disk in DV80 format. The only limitation to importing word processor files into NEF is that each imported DV80 file must be no more than

36 columns wide. DV80 files greater than 114 lines must be broken down into several smaller 114-line files before the text can be loaded into NEF. It is easy to break down such large text files using TI-Writer type word processors. When using NEF to print pages I prefer using the Funnelweb v5.01 word processor to create right justified newsletter articles before importing them into NEF for formatting and printing. The Funnelweb v5.01 word processor lets you do right justification directly from the editor without having to use a separate formatter. The main reasons I don't like using the NEF text editor are that NEF's text editor tends to lose a character at word wrap even with my moderate touch typing speed, and its full screen editor is rather slow to respond to up and down cursor movement.

Whether you are entering an article using the NEF text editor or importing a disk or hard drive file, NEF handles a maximum of 114 lines of text at a time. This is exactly enough text to fill one double-column page, with enough room left for a top-of-page headline and a bottom-of-page page number. If you have fewer than 114 text lines, the right column on the printed page will not be completely filled with text. You load in or type in the text for one page, print that page, then load in and print the text for the next page, etc. NEF creates newsletters one page at a time. The results are quite impressive, and you can make them better. NEF is set up to print normal (pica) text in your printer's "draft" font, sometimes using double strike to make selected lines appear darker. Since NEF is written in Extended BASIC, you might be able to figure out how to alter the XB code so that NEF prints text in your printer's "NLQ" font instead of "draft" font for even better looking text.

Once your text is typed into or loaded in from a file you get to decide which lines to right justify and which to print in double strike or "emphasized" for extra darkness on the printed page. The cursor appears to

the right of each line and you move down your document one line at a time marking lines for right justification, which you see immediately on screen, and extra dark. Unlike the TI-Writer formatter, this program contains no global right justification. With NEF, if you want right justification you must do this one line at a time. You probably don't want to right justify the last line of a paragraph. Blank lines, and text lines with only one word are automatically not right justified.

Two software products in the TI world similar to NEF will print two column newsletter pages with one pass through the printer. Art Gibson has written one that allows you to mix graphics with text. I have not used his program, but I understand from others that Art's program does not have a WYSIWYG display and uses control codes in a TI-Writer type of file to switch printer fonts and to mix graphics and text. The results of Art's newsletter printing software that I have seen in the Knoxville and the Cleveland newsletters look really good.

These days most TI user group newsletters that aren't cut-and-paste are created with Page Pro. This software gives you approximately a what-you-see-is-what-you-get screen display, allows you to mix text and graphics, and lets you print a nice border all around the page. My main complaint about Page Pro is that its text is printed using graphics mode. Page Pro has many different fonts, but on a 9-pin dot matrix printer they all look grainy and many are difficult to read. NEF doesn't allow "one pass through the printer" mixing of text and graphics, has no WYSIWYG display, and you don't get a fancy border. (NEF, by the way, can be used to create a disk file of text formatted in two columns that can be loaded into and printed with Page Pro.) Because NEF produces such good-looking text hard copy I prefer NEF over Page Pro for creating newsletter pages with columns.

(See Page 26)

## MICRO-REVIEWS—

(Continued from Page 25)

Of course the ultimate in total control of what is on a printed page is a careful neat cut-and-paste job. In creating a cut-and-paste master you should use "magic" (frosted) tape to secure all the edges of all smaller papers you secure to the surface of your master page. Edges so taped will not be seen when the master page is copied on a Xerox copier. My user group's newsletter is created this way.

Send \$1 to me at P.O. Box 647, Venedocia, OH 45894 and I'll send NEF to you on a SSD disk to play with. Bill Gaskill is asking a \$15 fairware fee that includes a royalty payment to Brad Snyder for using his 40-column routines.

### USVBA Power Volleyball by Program Innovators

This "arcade action" game written in assembly can be loaded from Extended BASIC. It can be played one against the computer or as a two-player game. USVBA, of course, stands for United States Volley Ball Association. Considering the possible

complexities of emulating a fast action team game on a computer screen, Power Volleyball is really quite well done.

On screen you see the net, the ball, and one player on each side of the net. This is a one-on-one game. These two on-screen players are all you get. Using the joystick(s) you move your player toward or away from the net and you make your player into either a server or any of three ball hitting modes. A "bumper" takes the newly served ball and pops it up into the air. A "setter" moves the ball in an arc somewhat forward so that it can be clobbered by the "spiker" who jumps way up in the air and bangs the ball over the net. You may switch your man between any or all of these modes, but the recommended method of dealing with a ball newly arrived on your side of the net is bump, set, spike. I find it fun to try to immediately spike from near the net a newly arrived ball.

Several levels of play are characterized by faster ball movement. They are all hard! Even at the slowest level it is diffi-

cult to score points against the computer in a you-against-the-computer game. This is really my only complaint. The computer is too good at this game! The lowest skill level should let bimbo players such as I am win once in a while, but I have yet to win a game against the computer even at the lowest level. You need practice to figure out exactly how to position your player with respect to the incoming ball. If you don't judge your position correctly you will miss hitting the ball, which will drop at your feet. Spiking is particularly difficult. The ball is coming down and your man on screen is going up.

Power volleyball really is a game of skill. The on-screen action, even at slow speed, is fast and quite realistic. I give it a thumbs up! It comes on a SSD disk with both on-disk and hard copy documentation. Cost is \$10 including postage. Send your order to Program Innovators, 4122 Glenway, Wauwatosa WI 53222. Ask about their stock market analyst/advisor software and their other arcade action games.

### Funnelweb V5.00 40-Column Editor

# Users who haven't upgraded miss a lot

By COL CHRISTENSEN

(This review appeared in the August 1993 issue of BUG-Bytes, the newsletter of the Texas Instruments Brisbane Users Group.)

If you're not yet using this latest version of Editor you are missing out on a number of valuable features which help to make the task of word processing so much easier. Documentation for this version is included on the disk under the filenames FWDOC/ED40, FWDOC/ED41 and FWDOC/DE42. These documents should be read, too, as they enlarge on some of the items mentioned here.

#### THE BASELINE EDITOR

The 5.00 Editor version is run from the filenames ED and EE. There are a number of versions of editors that use these filenames but this one can be recognized in a disk directory because the ED file occu-

pies 33 sectors and the EE file takes up 21 sectors. Earlier versions had a smaller EE file that this while the enhanced version mentioned later has 31 sectors for its EE file.

To run this editor it is imperative that version 4.40 or later of Funnelweb be used. The editor will not load if you use earlier versions of Funnelweb, as I finally found out after attempts from time to time of loading it with my earlier version of F'web.

On first loading, Funnelweb 4.40 goes straight into the Disk Review utility which is useful if you wish to work on a file stored on disk or RAMdisk, but it can be a nuisance having to press CTRL/= to escape from it to start some document from scratch. This is no great problem when you remember to hold down the spacebar while Funnelweb is loading to bypass

Disk Review and the Funnelweb option screen will appear. From it you can select option 1, the Text Editor or toggle to the next option screen by tapping the spacebar. The program editor available from this set of options is a non-wordwrap editor useful for programming in either assembly and C languages.

When the Editor is loaded, a caption near the bottom of the screen confirms that you are using version 5.00 and whether you are in Text Editor or Program Editor by the abbreviations W/P or P/E.

The cursor will be at the top of the screen as usual and the editor will be in the command mode waiting for some command to execute. The usual Edit, Files, Lines, Search, Tabs and Quit prompts are displayed and their respective sub-groups of two-letter commands are still available.

(See Page 27)

# FUNNELWEB EDITOR—

(Continued from Page 26)

But there are a number of additions to those to which we have become accustomed.

A very nice feature is an addition where, instead of typing "E" and pressing <enter> to move into the edit mode, you now only have to press <enter>. Another good feature is the Showline ability to just type a line number and press <enter> to get to that line. There is no need for the Show command first. In the Files group, are some new commands.

**LoadTemporary:** the "temporary" filename may be entered directly or marked in Show Directory with the "T" key. This allows for loading the temporary file (you must save the workfile first) or for inserting all or a selected part of the temporary file into the workfile without disturbing the workfile's name in the mailbox. Both filenames seem to have their own mailbox storage for later display of the appropriate filename.

► **DirectoryPrinter:** This allows entry of a name other than the PrintFile name (PIO) such as the name of a disk file where the directory listing can be stored and edited or incorporated into other text.

In the Search group the WildCard command allows setting the wild card symbol to be used during searches and replacements. The default wild card is "\*".

The Tabs function has been enhanced to allow not just two sets of tabs but now three. You can set up one set for normal text, one for, say, outdenting and one for typing tables in columns. When you type the "T" command you are offered the choice of modifying Tabsets 1-3. The three tab settings are saved with the file to disk and restored when the file is reloaded.

A new command replaces the old Quit which required two keypresses and <enter>s. The new command is QuickQuit which will take you directly back to the Funnelweb menu screens provided the file in memory has not been edited in any way since being loaded or previously saved, otherwise you are given the option to save the changed file first.

Help is a new command. This is very useful in that, during typing sessions, the 10 help files can be loaded and paged sequentially to refresh your memory of par-

ticular keystrokes and other info. You page through the screens with the "Q" and the "A" keys and escape through FCTN/9. The first two screens show lists of Editor functions and the keys to use. The others show Formatter dot commands, Extended PrintFile options, Box drawing characters for the all-chars mode, 9900 processor opcodes for assemblers and a quick reference for system addresses and color table.

**New Load/Save Functions.** The first to be mentioned is the progress line count number displayed at the top right of the screen while a file is being saved or loaded. The second is in the provision of further options for the PrintFile command. The printer name suitable changed to DSKn.FILENAME may then be preceded by one of the following:

**M** The file is output in DIS/FIX 128 TI file format with MS-DOS end-of-line (CR and LF) separators and CTRL/Z end-of-text marker. These files, I presume, can then be exported by modem or other means to a PC.

**U** Unix format with LF separators and CTRL/D at end of text.

**P** Sends a preset printer code before any text. (The print code sequence must have been installed through the CON/ED file and INSTALL/ED program.)

**Q** Sends a printer reset sequence after text is completed if installed.

**A** Appends the text to the end of the file already on a disk.

Note that the other options displayed on the fourth help screen prefixed to the printer name are still available.

As mentioned above, the printer options of "P" and "Q" need to be installed before they are operational. Here is a short description of the process of configuring the editor program. Have your editor disk in a known drive. It must have the files ED, EE, INSTALL/ED and CON/ED on it. Load an editor and load the CON/ED file. It should have 11 very short lines of text. The document file, FWDOC/ED42, gives full details of how to set up a configure file. For now you should make it:

```
N
11
11
1
1
```

```
1
N
0
0
P
Q
```

The P must be followed, if need be, by the required printer control code sequence typed using CTRL/U where necessary to set the printer to NLQ or bold or whatever you prefer.

The Q must be followed by the printer reset code that cancels the above.

Save the file under the name CON/ED. Go into Funnelweb/loaders/option 2 GPL and load the INSTALL/ED file. Then enter the pathname of the particular ED/EE file you wish to configure, the configure filename, CON/ED, that you just edited and finally the ED filename to save to.

Some keypresses have new special functions in command mode. You can use the up and down arrow keys while in command mode to scroll both the screen display of text and the associated line numbers. Not only that, you can use the FCTN/4 and FCTN/6 keys to page the screen display up or down. When you exit from command mode back to edit mode by pressing <enter> or CTRL/2, it returns you to the point from where you left off editing. However, if you exit from command mode by pressing CTRL/1, you are returned to the top line of the current screen display.

There are a number of Edit mode key combinations and you need to read the documents to find out which they are as I shall mention just three that I find most useful.

CTRL/H displays the beginning of the text starting at line 1.

CTRL/J "jumps" to the end to show the last screen of text.

CTRL/F places a horizontal line across the screen below the cursor and freezes the text below the cursor. The text above the cursor can be scrolled up or down line by line or page by page. This can be useful when you wish to refresh your memory on what had been mentioned in an earlier section of the workfile. Another press of CTRL/F restores the status quo to the as-you-were condition.

(See Page 28)

# USER NOTES

## An answer to a question

This comes from Dan Eicher. In the May 1994 issue Bruce Harrison asked about:

```
XMLLNK
VPUSH EQU >0E
VPOP EQU >10
```

While I can't find a definite reference to VPUSH EQU >0E, VPOP and VPUSHG are completely defined on page 49 of the Mini-Memory manual. The source code to these routines can be found on page 74 and 75 of the Intern.

The Mini-Mem manual was written with the benefit of hindsight. Also includ-

ed in the Mini-Mem manual, but left out of the Editor/Assembler manual, is information on special calling conventions that must be observed when talking to the RS232 unit ( see page 51 ).

```
The reference to:
VGWRITE EQU >34
GVWRITE EQU >36
```

The source code to these two routines follows. This source code was retrieved from an archive of the XB-ROM source code as uploaded by Rich Gilbertson to GENie.

```
*****
AORG >7FDA
TITL 'GVWITES'
```

- \* Move data from ERAM to VDP
- \* @GSRC : Source address where the data stored on ERAM
- \* @DEST : Destination address on VDP
- \* @BCNT3 : byte count

(See Page 29)

## FUNNELWEB EDITOR—

(Continued from Page 27)

### THE ENHANCED EDITOR

This editor has all the functions of the baseline editor except that TI Eurowriter, PC character graphic capability and path-name functions for hard drive users have been added.

The filenames on disk for this editor are named ED/AEH and EE/AEH. As they only run out of Funnelweb, their filenames must be renamed ED and EE after being copied to the appropriate disk. This editor especially would be best run from a hard drive or a RAMdisk to speed up proceedings because, with the increases in functions provided, the F'web kernel can no longer be stored in VDP memory for instant reappearance on exit but must be reloaded from the boot disk. Disk accesses also occur to run some support features such as Help.

The big feature of this editor, I find, is the All-chars mode where a full 8-bit set of PC character graphics is supported in the ASCII range from 0 to 255. These characters are supported by most modern printers and are printed through the PF command directly to the printer. Files containing ASCII characters above 127 should not be sent through the formatter which strips them out of the text. Look up your printer

manual and see all the characters above ASCII 127 that we can't normally print.

For this editor, you need a character definition file to suit and a suitable one is supplied on disk as CHAR@1. This file must be on the drive from which Funnelweb is loaded. The CON/ED file must be changed to make use of the all characters mode. The second line needs to be changed from 11 to 31 and the INSTALL/ED program has to install the change to the ED files as mentioned for the baseline editor.

Now how do you type those line graphic characters that are shown on the fifth, sixth and seventh Help screens when the keys are normally only interpreted as ASCII 32 to 127? If you hark back to the CTRL/U mode of TI-Writer, this keypress toggles between a condition where the key pressed is interpreted normally or its ASCII value is reduced by 32. With its use, we can type characters from ASCII 0 to 31 or so. Similarly, to get the ASCII values above 127 we use the keypress CTRL/, which also has a toggling function. By its use, 128 is added to the normal keypress value. So to get character 205 that is a double horizontal line, all we have to do is press CTRL/, to toggle the upper

set in and then SHIFT/M that is normally ASCII 77. The fifth Help screen shows which characters convert to the box drawing characters. If you are wondering how to type characters just above ASCII 127, then the combination of CTRL/, (+128) and CTRL/U (-32) does the trick. Remember to toggle them both off afterwards. Probably because of the above situation where both conversions can be toggled at the one time, no provision has been provided for a special cursor to indicate the CTRL/, mode but this presents no real problem.

Show Directory has been modified in the enhanced editor version. One difference is that instead of entering SD and the drive number from the command line of the editor, only the SD is entered which brings up directory screen on which the drive number is to be entered. The difference, I find, is instead of entering in quick succession SD-enter-1-enter I must change it to SE-enter (short pause while directory screen is painted) 1-enter.

I have been using the baseline version and later the enhanced version for many months now and find them to be very friendly and useful tools for word processing.

# USER NOTES

(Continued from Page 28)

```

GVWRITE MOV @DEST,R2      VDP address
MOV @R2LB,*R15           LSB of VDP address
ORI R2,WRVDP             Enable VDP write
MOV R2,*R15             MSB of VDP address
MOV @GSRM,R3            ERAM address
GV$1  MOV *R3+,@XVDPWD    Move a byte
DEC @BCNT3              One less to move
JNE GV$1                If not done, loop for more
AORG >7FFE
DATA >9226

```

```

*****
*****
AORG >7FC0
TITL 'VGWITES'

```

\* Move data from VDP to ERAM  
 \* @ADDR1 : Source address where the data stored on VDP  
 \* @ADDR2 : Destination address on ERAM  
 \* @BCNT1 : byte count

```

VGWRITE EQU $
MOV @ADDR1,*R15          LSB of VDP address
MOV @ADDR2,R2           Address in ERAM
MOV @ADDR1,*R15          MSB of VDP address
NOP
GV$1  MOV @XVDPWD,*R2+    Move a byte
DEC @BCNT1              One less to move
JNE GV$1                If not done, loop for more
RT                      Return

```

```

*****
*****
AORG >7ECA
TITL 'GWRITES'

```

\* Write the data which is stored in CPU to ERAM  
 \* @GDST : Destination address on ERAM where data is going  
 \* to be stored  
 \* @CSRC : Source address on CPU where data stored  
 \* @BCNT2 : byte count  
 GWRITE LI R3,BCNT2 Count  
 LI R2,GDST Destination  
 LI R1,CSRC Source  
 JMP GW\$1

\* Write the data which is stored in CPU to ERAM  
 \* @ADDR1 : Destination address on ERAM where data is going  
 \* to be stored  
 \* @ADDR2 : Source address on CPU where data is stored  
 \* @BCNT1 : byte count  
 GWRITE LI R3,BCNT1 Count  
 LI R2,ADDR1 Destination  
 LI R1,ADDR2 Source

\* Common routine to copy from CPU to ERAM  
 GW\$1 EQU \$  
 MOV \*R2,R4 Get destination address  
 MOV \*R1,R1 Get CPU RAM address  
 AI R1,PAD0 Add in CPU offset  
 GW\$2 MOV \*R1+,\*R4+ Move a byte  
 DEC \*R3 One less to move, done?  
 JNE GW\$2 No, more to move

```

*****
*****

```

Hope this helps.

## New 80-column card released in Germany

A new 80-column card for the TI made its debut at the Göttingen TI fair, Oct. 14-16. Some 70 TI enthusiasts from Austria attended the fair.

The new PEB card, called the Enhanced Video Processor Card (EVPC), was designed by Michael Becker. Co-designer was Jürgen Stelter. The DSRs and software were written by Roland Meier and Harald Glaab. Sven Dyrhoff also worked on the project.

(See Page 30)

## READER TO READER

□ Joe Stomiany, 14948 Templar Dr., La Mirada, CA 90638, (714) 521-2114, writes:

Help, help help! I have a Grand RAM Disk from Databiotics which doesn't seem to recognize the RAMdisk when I try to CONFIGR it. The very end of the load of CONFIGR comes on the screen for a split second but does not stay on the screen.

I had Don Walden (Cecure) and Don O'Neil (Western Horizon Technology) try to repair this problem but they had no success. If anyone out there can help me to get my configure program to stay on so I can configure it, please let me know. Please write to me. I would be very grateful to anyone who can help me.

□ Allen Masden, 4608 38th Ave. Ne., Salem, OR 97305-1607, writes:

Could someone who has the instructions for the Foundation 128K card please make and send me a copy?

**Reader to Reader is a column to put TI and Geneve users in contact with other users. Address questions to Reader to Reader, c/o MICROpendium, P.O. Box 1343, Round Rock, TX 78680. We encourage those who answer the questions to forward us a copy of the reply to share with readers.**

# USER NOTES

(Continued from Page 29)

The card runs at CRU >1400. According to Gerd Weissmann, who is marketing the card, the EVPC can display 256 colors from a palette of 256,000. This can be expanded to 16 million, he says. The card requires modification to the TI console.

The device includes a socket for a sound chip from the console, 128K of video RAM and a 64K DSR-ROM. It can support up to eight sprites simultaneously.

Weissmann says the group of developers is working on a "Super GPL Card." This device will hold the contents of up to 16 modules in flash EPROMs. He says this will allow BASIC programs to run at 10 times normal speed. Also, a new motherboard for the PEB is being developed. This board would eliminate the need for the flex cable and would be based on the TMS9900 processor.

Pricing of the EVPC is \$254, based on an exchange rate of one U.S. dollar to 1.5 Deutschmarks.

For more information, contact Weissmann at Koenigstrasse 17-19, D-67655 Kaiserslautern, Germany. Telephone/fax at 0631/12169.

## Maze Maker offers challenges at many levels

The following article and program were written by Steve Karasek. It appeared in Wordplay, the newsletter of the Portland (Oregon) Users of Ninety-Nines.

Maze Maker is an Extended BASIC program that prints mazes for you to solve. It asks for the number of mazes to print, then for the level of difficulty, which can be from 0 to 9. Level 0 is a trivial maze that can serve as a child's first maze. Level 9 is fairly challenging. The level number is printed at the top of the maze.

No matter what level you select, the maze will be printed to fill as much of the page as possible. This means that the lower level mazes will have wider pathways, which are easier for young children to follow. There will always be exactly one path from Start to Finish.

The higher level mazes take a while to

compute. In particular, level 9 mazes take over 20 minutes each. You can always start up the program and come back a few hours later. The program keeps track of how far it has gone in computing each maze by displaying a line of the form M/N on the screen, where N is the number of squares in the maze and M is the number of squares the program has computed a path to. When M equals N, the maze is done and is sent to the printer.

If your printer is not named "PIO," change the name in line 110. The last part of this line sets the printer line spacing to 7/72 inch. If you do not have an Epson-compatible printer, you will have to change this to the codes needed by your printer to set the line spacing. If you can't set it to 7/72 inch, set it to eight, or preferably 10, lines per inch.

### MAZE MAKER

```

100 RANDOMIZE :: OPTION BASE
110 DIM M(39,39):: INPUT "
HOW MANY MAZES? ":Z :: PRINT
!223
110 INPUT "LEVEL OF DIFFICUL
TY(0-9)? ":L :: IF L<0 OR L>
9 THEN 110 ELSE OPEN #1:"PIO
",OUTPUT :: PRINT #1:CHR$(27
);"A";CHR$(7);!131
120 N=INT(L+1)*4+(L=4 OR L=9
):: X=80/N :: S=INT(X):: S=S
+(X=S)!138
130 PRINT #1:"Start";TAB(30)
;"Level";L :: FOR X=1 TO N :
: FOR Y=1 TO N :: M(X,Y)=0 :
: NEXT Y :: NEXT X :: IF N=3
9 THEN 150 !174
140 FOR X=1 TO N :: M(N+1,X)
,M(X,N+1)=16 :: NEXT X !203
150 C,X,Y=1 :: DISPLAY ERASE
ALL AT(12,12):"1 /";N*N ::
ON ERROR 290 !059
160 W=INT(RND*4):: DX=X+(W=0
)-(W=1):: DY=Y+(W=2)-(W=3)::
K=M(DX,DY):: IF K THEN 160
!229
170 M(X,Y)=M(X,Y)+2^W :: IF
INT(W/2)*2=W THEN W=W+1 ELSE
W=W-1 !125
180 X=DX :: Y=DY :: M(X,Y)=M
(X,Y)+2^W :: C=C+1 :: DISPLA
Y AT(12,9)SIZE(4):USING "###

```

```

#:C :: IF C=N*N THEN 240 !0
53
190 IF X<N THEN IF M(X+1,Y)=
0 THEN 160 !198
200 IF Y<N THEN IF M(X,Y+1)=
0 THEN 160 !199
210 IF Y>1 THEN IF M(X,Y-1)=
0 THEN 160 !117
220 IF X>1 THEN IF M(X-1,Y)=
0 THEN 160 !116
230 X=INT(RND*N)+1 :: Y=INT(
RND*N)+1 :: IF M(X,Y) THEN 19
0 ELSE 230 !248
240 ON ERROR STOP :: PRINT #
1 :: PRINT #1:"#";TAB(S+1);R
PT$("#",S*(N-1)+1):: S=S-1 :
: S$=RPT$(" ",S):: X$=RPT$("
#",S)!069
250 M(N,N)=M(N,N)+8 :: FOR Y
=1 TO N :: FOR W=1 TO S :: P
RINT #1:"#";:: FOR X=1 TO N
:: PRINT #1:S$;!076
260 IF M(X,Y) AND 2 THEN PRIN
T #1:" ";ELSE PRINT #1:"#";!
084
270 NEXT X :: PRINT #1 :: NE
XT W :: PRINT #1:"#";:: FOR
X=1 TO N :: IF M(X,Y) AND 8 T
HEN PRINT #1:S$;ELSE PRINT #
1:X$;!244
280 PRINT #1:"#";:: NEXT X :
: PRINT #1 :: NEXT Y :: S=S+
1 :: PRINT #1 :TAB(S*N-4);"
Finish":CHR$(12);:: Z=Z-1 ::
IF Z>0 THEN 130 ELSE END !0
20
290 ON ERROR 290 :: RETURN 1
60 !159

```

## Success with NEW in Extended BASIC

This comes from John C. Johnson of Cedar Rapids, Iowa, via the Internet. He writes:

I read with great interest the User Notes item on page 26 of the September issue about the fake Extended Basic NEW command. I have three Horizon RAMdisks and have gotten from the John Johnston Menu to XB by using a one-liner from John Parkins of the Spirit of 99 newsletter, November 1987. His call also addresses

(See Page 31)

# USER NOTES CLASSIFIEDS

(Continued from Page 30)

-31952 but only enters 255,255,0. While this will get you into XB it requires you to do a CALL NEW after you are there. Failure to do so will result in a crash.

I have succeeded in using the NEW call to allow me to get to XB "clean" with no DSK1 read. Here is the three line XB program.

```
100 CALL PEEK(8198,A) :: IF A
<>170 THEN CALL INIT
110 FOR I=1 TO 50 :: NEXT I
120 CALL LOAD(-31952,255,231
,255,231) :: END
```

I am not sure that line 110 is necessary. If you can convince yourself it isn't, take it out. I tried it without it and it seemed okay. I must give credit to Oliver Hebert, Jerry Keisler and MICROpendium for the information that allowed me to make this program. Many thanks.

## Reverse linefeed solves problem

The following is credited to Jim Peterson. We saw it in the TISHUG News Digest.

We all know that the TI-Writer formatter insists on giving us five blank lines at the top of the page and three at the bottom. If your printer supports a reverse linefeed, you can back those lines at the top by beginning the page with a line CTRL-U and RJRJRJRJ. However, I can't find a practical way to print the three lines at the bottom.

## FOR SALE

### ALL THIS FOR THE CC40

8K Ram cartridge (for 6K computer)	35.00
16K Ram cartridge (adds 16K)	35.00
CC40 Computers 6K	50.00
CC40 Computers 18K	70.00
Electrical Engineering package (cart)	20.00
Finance Packages (cart. & book)	10.00
Games #1 (1 used \$5)	10.00
Hexbus Interface Manual	10.00
Mathematics Package (cart. & book)	20.00
Memo Processor (word processor)	20.00
Modem 300 baud	50.00
Modem 300 baud (USED)	25.00
RS232 User's manual	5.00
Power Supply for CC40 and periph.	15.00
Printer Plotter 4 color (pen set)	15.00
RS232	50.00
RS232 with parallel printer cable	75.00
RS232 parallel printer cable	20.00
Statistics Package (cart & book)	20.00
Wafertape Drive User's Manual	10.00
Wafertapes	10.00
Word Processor (cart & book) used (also has a modem program)	10.00
S&H 10% Minimum \$3	214 821 9274
Jim Leshner, 722 Huntley, Dallas Tx 75214	

chine Language Games) \$10.00 Cutthroat Cribbage, Tltris, Cockroaches, Snomobile, Martian Missiles, \$10.00 Multiple Orders 10% discount  
4122 Glenway, Wauwatosa WI 53222-1116

### "THE WATCHAMACALIT"

From DBM TECHNOLOGIES

ATTENTION MYARC HFDC USERS!  
LIMITED SUPPLY \$29.95 Each. GET YOUR HFDC UPGRADED NOW! From DBM TECHNOLOGIES 17725 22nd Street Ogden, Utah 84401-2112 Phone: (801) 782-1004 or (801) 394-6815. "THE WATCHAMACALIT" is designed to Back-Up the clock on the Myarc HFDC. Reviewed in MICROpendium in August 1993. Comes complete with software

v11n10

### FOR SALE

2 complete TI systems & many accessories for sale. For price list, write Larry Harpring, 645 Five Notch Rd., North Augusta, SC 29841 or call (803) 278-4606 after 6 p.m.

v11n11

### PROGRAM INNOVATORS SOFTWARE SALE

TOUCHDOWN: NFL Football Predictor \$10.00 WALLSTREET: Investment Package \$30.00 USVBA Power Volleyball (ML-Game) \$10.00 Klingon Invaders, Destroy Klaatu, Desert Rat (Ma-

### FOR SALE

TI99/4A complete system, all or part, including 3 consoles, PEB with 32K card, RS232 card, disk controller, 2 DSDD disk drives, original TI color monitor. System comes with 50+ disks of programs, including TI-Artist, Writer, Multiplan, & Editor/Assembler Extended BASIC. All comes with instruction books & a complete set of MICROpendium from December '83 to December '92. Make offer. Call Tim Blakley, (916) 788-7468.

v11n10

## What a deal!

Prices slashed  
on MICROpendium  
Classifieds!

Classified ads are now available for 10 cents per word, a reduction of more than 50 percent.

If you want to sell or buy something, advertise it in MICROpendium Classifieds.

Simply write your ad on a separate sheet of paper, count the words (a phone number counts as one word) and send it, along with payment, to:

**MICROpendium Classifieds**  
P.O. Box 1343  
Round Rock, TX 78680.

# The ONLY monthly devoted to the T199/4A

## Subscription Fees

- 12 issues, USA, \$35       12 issues, Mexico, \$40.25
- 12 issues, Canada \$42.50     12 issues, other countries surface mail, \$40.00
- 12 issues, other countries, air mail, \$52.00

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

## Address Changes

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

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

Check/MO      (check one)

Card No. \_\_\_\_\_

Expiration Date \_\_\_\_\_  
(Minimum credit card order is \$9)

Signature \_\_\_\_\_  
(Required on credit card orders.)

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

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

Name \_\_\_\_\_

Address \_\_\_\_\_

City \_\_\_\_\_

State \_\_\_\_\_ ZIP \_\_\_\_\_

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

## Disks, Etc.

Back Issues, \$3.50 each. List issues: \_\_\_\_\_

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

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

- 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
- MICROpendium Index II** annual disks ordered separately (1 disk per year, 1984-1992); each .....\$6.00

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 2.0 (req. DSSD or larger (for floppy & hard drive systems) .....\$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 SETCOLR, SHOWCOLOR, FIND, XUTILS, REMIND .....\$4.00

### GENEVE PUBLIC DOMAIN DISKS

These disks consists of public domain programs available from bulletin boards. If ordering DSSD, 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