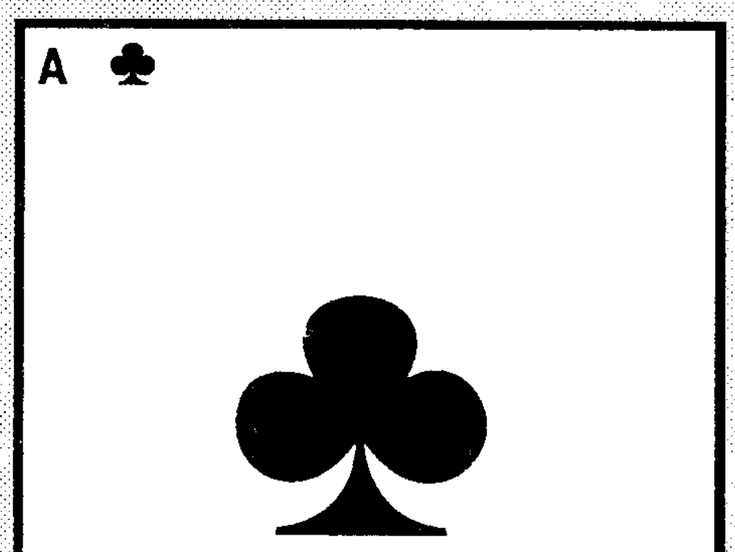
Covering the TI99/4A and the Myarc 9640

MICROpendium

Volume 11 Number 11 | December 1994 | $3.50



Montecolor

It's like Solitaire, but without the cards

See Page 7

PLUS:

- √ Bit map in assembly
- √ A VRAM analyzer for the Geneve
- √ Static electricity hassles
- √ The CC40
- √ Reviews of CYA, Disk of Medieval Times and Risk
- √ TIPS files catalogued

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.

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 TI NET: MICROpendium
GEnie: J.Koloen
Internet E-mail: jkoloen@io.com

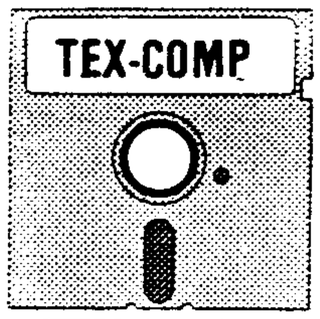
John Koloen.....Publisher
Laura Burns.....Editor

*READ THIS

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

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

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



ANNOUNCING NEW LOW PRICES ON FREEWARE DISKS FROM $2.95 TEX COMP $2.95

TEX-COMP HAS SLASHED PRICES ON THE BIGGEST & BEST COLLECTION OF FREEWARE 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

GRAPHICS, MUSIC & ANIMATION
#1 THE SINGING TI VOL. 1 (S)(2)
#4 PRINTART (2)(P)
#5A MUSIC/GRAPHICS
#6 EXBASIC MUSIC (2)
#7 SPACE SHUTTLE MUSIC/GRAPHICS
#9 MONA LISA PRINTOUT (P)
#11 ANIMATED XMAS (WOODSTOCK)
#14 FIGURE STUDIES (P)(PG)
#32 EXBASIC XMAS MUSIC (2)
#41 VIDEO GRAPHS (M)
#52 ANIMATION 99' (2)
#69 PLAYER PIANO/KEYBOARD ANALYSIS
#93 KBGB GIRLIE CALENDAR (4)(P)
#103 SORGAN THE TI ORGAN
#107 STARTREK MUSIC ALBUM
#111 POP MUSIC & GRAPHICS
#114 PANORAMA
#115 GRAPHICS DESIGN SYSTEM
#120 BITMAC (2)(P)
#230 THE SINGING TI VOL. 2 (S)(2)
#231 THE SINGING TI VOL. 3 (S)(2)
#246 THE SINGING TI VOL. 4 (S)(2)
#313 3-D WORLD (CAD for the TI)

COMPUTER UTILITIES, PRINTER UTILITIES & PROGRAMMING LANGUAGES
#3 DUMPIT (E/A)
#15 STAR/EPSON PRINTER DEMO (P)(2)
#16 SIDEWAYS PRINTOUT (P)(2)
#18 TI DIAGNOSTIC (MM)(2)
#28 LOADERS & CATALOGERS
#30 HOUSEHOLD BUDGET PRINTOUT (P)
#35 PROGRAMMING AIDS & UTILITIES I
#42 FUNNELWEB FARM (SHELL UTILITY)
#53 HACKER CRACKER
#55 SCREEN DUMP (P)
#62 DISK MANAGER II (M)
#75 DISK CATALOGER
#76 PROGRAMMING AIDS & UTILITIES II
#78 ARICON GRAPHIC CONVERSION
#79 DISK MANAGER 1000
#80 BIRDWELL DISK UTILITY (2)
#85 AUTOBOOT UTILITY
#86 COLUMN TEXT III (P)
#87 ARCHIVER III
#89 PROCALC DECIMAL/HEX CONVERTER
#96 STATISC & SORT ROUTINES
#97 MEMORY MANIPULATOR
#101 ENHANCED DISPLAY PACKAGE
#108 FUNLPLUS (PLUS!)
#110 DISK+ AID
#117 UNIVERSAL DISASSEMBLER
#119 RAG LINKER CONVERSION
#127 PIX GRAPHICS UTILITY
#243 OS/99 (GD)
#251 PC TRANSFER TI/IBM (DD)
#253 THE EXPLORER/DM1000
#254 NIBBLER/TURBO
#260 TI FORTH (DISK ONLY- add $8 for manual)
#17 TI FORTH DEMO
#116 TI FORTH TUTORIAL
#5079 FORTH SOURCE CODE
#104 C99 COMPLIER & LIBRARY
#5007 TEACH YOURSELF TI BASIC
#5019 TEACH YOURSELF EX-BASIC
#5067 BEGINNING BASIC TUTOR
#307 GRAPHICS CODE GENERATOR
#3912 SUPER BUGGER
#3913 BETTER BANNERS
#3914 CERTIFICATE 99
#3915 HOROSCOPE MAKER
#3916 GRAPHX+ PRINT SHOPPE
#3917-3720
GRAPHX COMPANIONS 1-4
#3721 MAC FLICK (G)
#3722 PRINTING TO GO (G)
#3723 GRAPHX DINOSAURS (G)

BUSINESS, ACCOUNTING, WORD PROCESSING, DATA BASE HOME OFFICE & HOME
#10 GOTHIC PRINTOUT (P)
#19 TI WRITER/MULTIPLAN UPGRADE
#20 ACCOUNTS RECEIVABLE (2)
#21 DATA BASE DEMO
#23 WILL WRITER
#29 LABEL MAKER I (P)
#36 STRICTLY BUSINESS (2)
#56 SPREAD SHEET
#58 PR BASE (database)
#59 GRAPH MAKER
#74 LABEL MAKER II (P)
#77 MICRodex 99 (database)
#81 HOME ACCOUNTING SYSTEM
#83 HOME APPLICATION PROGRAMS (2)
#90 JET CHECKBOOK MANAGER
#92 HOUSEHOLD INVENTORY
#109 TI WRITER MINI MANUAL
#112 INVOICE PACK
#113 LABEL MAKER III
#129 CASH DRAWER (point of sale)
#130 THE ORGANIZER
#147 CALENDAR-NOTEPAD
#177 HOUSEHOLD BUDGET MANAGEMENT (M)
#30 HBM DATA PRINTOUT (P)
#221 PERSONAL REAL ESTATE (M)
#249 MAPMAKER
#252 99 WRITER II (TI WRITER)(P)
#253 AMA MAILING LIST
#255 TRAK-A-CHECK
#257 DAILY DIARY
#257A EXER-LOG
#23 WILL WRITER
#310 SELF HELP TAX CUT

TELECOMMUNICATIONS (MODEM)
#57 TELCO
#57D TELCO (FOR SYSTEMS WITH DS DRIVES)
#118 FAST TERM

EDUCATION & PERSONAL DEVELOPMENT
#22 ASTROLOGY
#24 ENGINEERING CALCULATIONS (2)
#25 MEDICAL ALERT
#27 KIDS LEARNING I (2)
#31 MORSE CODE TRAINER
#37 LAPD COOKBOOK (2)
#40 ARTIFICIAL INTELLIGENCE (ELIZA)
#54 ASTRONOMY
#66 HEBREW TYPEWRITER
#67 GENEALOGY (2)
#71 KIDS LEARNING II (2)
#95 WEATHER FORECASTER +
#138 FIREHOUSE COOKBOOK
#142 TOUCH TYPING TUTOR (M)
#184 FACE MAKER
#194 ST. NICK/GHOSTLY SPELLING (M)
#195 TINY LOGO
#199 MILLIKEN ADDITION (M)
#200 MILLIKEN DECIMALS (M)
#203 MILLIKEN FRACTIONS (M)
#204 MILLIKEN INTEGERS (M)
#205 MILLIKEN LAWS OF MATH (M)
#211 MIND CHALLENGERS (M)
#212 MINUS MISSION (M)
#213 MILLIKEN PERCENTS (M)
#214 STORY MACHINE (M)
#215 BEGINNING GRAMMAR (M)
#216 METEOR MULTIPLICATION (M)
#217 HANGMAN (M)
#222 MUSICMAKER (M)
#223 PHYSICAL FITNESS (M)
#225 ALIEN ADDITION (M)
#226 ALLIGATOR MIX (M)
#227 DEMOLITION DIVISION (M)
#228 DRAGON MIX (M)
#270 TRIGONOMETRY
#271 CALCULATORS & CONVERSIONS (2)
#272 HIGHER MATH (2)
#273 ASTRONOMY 2
#306 SPEAK & SPELL II (S)(EX)

GAMES
#2 WH. OF FORTUNE, BLKJACK, POKER
#3 TI TREK
#8 LOTTO PICKER
#13 STRIP POKER (PG)
#26 R-RATED NOVELTY GAME
#33 CHECKERS & BACKGAMMON
#34 SOLITAIRE & SCRABBLE
#38 GREAT TI GAMES VOL 1
#39 GREAT TI GAMES VOL 2
#43 BEST OF BRITAIN GAMES VOL 1
#45 BEST OF BRITIAN GAMES VOL 2 (LEGEND OF CARFAX ABBY GRAPHIC-INTERACTIVE ADVENTURE)
#46 SUPER TRIVIA 99
#47 INFOCOM RAPID LOADER
#48 GHOSTMAN (from U.K.)
#49 DEMON DESTROYER (from FRANCE)
#50 OH MUMMY! (HIT from GERMANY)
#51 BERLIN WALL (from CANADA)
#60 FREDDY (HIT from GERMANY)
#61 THE MINE (from GERMANY)
#62 ASTROBLITZ & MAZOG
#64 MAJOR TOM & SPACE STATION FRETA
#65 PERFECT PUSH (HIT)
#68 CHESS (SARGON)
#70 TI RUNNER II (HIT)
#72 CEBERUS (HIT SPACE GAME)
#73 CRYTO (GRAM)
#82 CROSSWORD (PUZZLES)
#84 GALACTIC BATTLE & SPY ADVEN.
#88 AUSSIE GAME COLLECTION VOL 1
#91 THE MAZE OF GROG (WOODSTOCK II)
#94 GREAT TI GAMES VOL 3
#98 DAYS/DOORS OF EDEN (BIBLE ADV)(AM)
#99 GREAT TI GAMES VOL 4
#100 ASSAULT THE CITY (TD)
#102 COLOSSAL CAVES (ADVENTURE)
#105 KINGS CASTLE (M)
#106 QUEST (D&D)
#121 SUPER YAHTZEE & WHEEL II
#122 ADULT ADVENT & GAMES (PG)
#123 GREAT TI GAMES VOL 5 (2)
#124 GREAT TI GAMES VOL 6 (2)
#125 BLACKJACK & POKER (M)
#126 VIDEO CHESS (M)
#128 TETRIS (HIT from RUSSIA)
#131 COMPUTER CRAPS
#132 AMBULANCE (M)
#133 DRIVING DEMON (M)
#134 ROTO-RAIDER (M)
#135 ARTURUS (HIT-ZARGON)
#136 ANT-EATER (M)
#137 CROSS-FIRE (M)
#139 MOONMINE (M)
#140 MASH (M)
#141 MOONSWEEPER (M)
#143 CONGO BONGO (M)
#144 STAR TREK (M)
#145 BUCK ROGERS (M)
#148 KENO & SLOTS
#149 GREAT TI GAMES VOL 7 (with HIT BLOCKBUSTER)
#150 ULTIMATE TRIVIA
#151 JUNGLE HUNT (M)
#152 POLE POSITION (M)
#153 DONKEY KONG (M)
#154 PROTECTOR II (M)
#155 PAC MAN (M)
#156 CENTIPEDE (M)
#157 DEFENDER (M)
#158 SHAMUS (M)
#159 MS. PAC MAN (M)
#160 DIG-DUG (M)
#161 PICNIC PARANOIA (M)
#162 MOON PATROL (M)
#3711 ARCADE SPECIAL (4 GAMES)
#3712 THREE GREAT GAMES
#3713 PRO TENNIS+

INFOCOM ADVENTURE BACKUPS
#163 ZORK I
#164 ZORK II
#165 ZORK III
#166 HITCHIKER'S GUIDE
#167 WITNESS
#168 ENCHANTER
#169 INFIDEL
#170 PLANETFALL
#171 SORCERER
#172 DEADLINE
#173 CUTTHROATS
#174 SUSPENDED
#175 STARCROSS
****
#176 AMAZING (M)
#178 DEMON DESTROYER (M)
#179 POPEYE (M)
#180 QUEBERT (M)
#181 METEOR BELT (M)
#182 BLASTO (M)
#183 CAR WARS (M)
#184 FACE MAKER (M)
#185 SUPER FLY (M)
#186 SPACE BANDITS (M)
#188 KILLER CATERPILLER (M)
#190 BLACK HOLE & SPACE AGRESS. (M)
#191 GREAT TI GAMES VOL 8
#192 GREAT TI GAMES VOL 9
#193 SPY'S DEMISE (HIT) (M)
#194 ST. NICK+ (M)
#196 JOTTO
#197 PRO TENNIS+ (HIT)(M)
#198 TI INVADERS/TOMBSTONE CITY (M)
#202 CONNECT FOUR (M)
#205 HOPPER (M)
#206 TREASURE ISLAND (M)
#206B SLYMOIDS (M)
#207 OTHELLO (M)
#208 PARSEC (M)
#209 SOCCER (M)
#210 SEWERMANIA (M)
#218 HUSTLE/FOOTBALL (M)
#219 CHISOLM TRAIL (M)
#220 ZERO ZAP (M)
#224 ATTACK (M)
#229 4A FLYER (FLIGHT SIM.)(M)
#232 TUNNELS OF DOOM (MOD BACKUP PLUS 2 NEW ADVENTURES)(M)
#233 MS ADVENTURES (3 ADV-EXBASIC)
#248 STRIKE THREE BASEBALL (M)
#248 GREAT TI GAMES VOL 10
#250 BARRAGE/SPOTSHOT
#318 THREE GREAT GAMES
#319 ARCADE SPECIAL
#317 BEANSTALK ADVENTURE
****
SCOTT ADAMS ADVENTURES (USE WITH ADVENTURE MODULE)
#349 TI ADVENTURES 1-13+
#350 TI ADVENTURES 14-16+

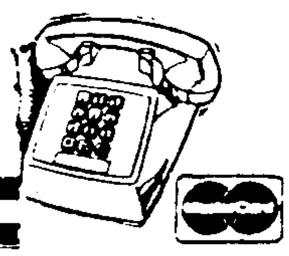
P = Printer required
G = Graphx required
S = Speech required
M = Module Backup
MM = Mini Memory req.
E/A = Editor Assem.
Exbasic and 32k mem req. for most programs.

Authorized Dealer

TEX COMP

America's Number One TI computer retailer

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



ORDER BY PHONE

(818) 366-6631 24 HOURS A DAY

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

Waiting for SCSI

I'm waiting for a SCSI card to arrive from Bud Mills. I'd like to review it as soon as possible. However, there may be some readers who already have it running on their Geneves. It that's the case, I invite you to send us a report of what you think of it.

As a Geneve owner, the SCSI card is high on my wish list. Unlike some users I've spoken with, I'm not interested in its abilities to access CD-ROM drives. Having it available to control a SCSI-1 and SCSI-2 hard drive is plenty for me. For now, at least. If Santa Bud is listening, this would be a great Christmas present.

Meanwhile, if you've got one running on your Geneve, tell us about it so we can tell everyone else.

GOING OFF OF COMPUSERVE

After January, readers won't be able to reach via CompuServe. TI subscribers to CompuServe know that its TI forum is the least active of all the electronic services. There are good reasons for this, the first of which is the cost. We can still be reached via Delphi, GEnie and the Internet, so there's no problem with accessibility. We're just trying to save a few bucks.

A GREAT LITTLE LAPTOP

We're publishing a lengthy treatise on the forgotten notebook computer, the TI CC40. Written by Charles Good, the article presents a convincing case for the tiny, incredibly efficient notebook. Even though TI never developed the potential of this computer, it's hard not to be intrigued by its possibilities. Charlie seems to enjoy it immensely. And why not. The CC40 will go a couple of hundred hours without replacing batteries, never mind recharging.

If you've ever seen one at a flea market or computer fair and wondered whether it was worth the few dollars that was being asked, the article may resolve your question into a resounding "YES."

HAPPY HOLIDAYS

Laura and I want to extend holiday greetings to all of you as our 11th year slowly comes to a close. It's hard to believe that we've been meeting like this for that long. Thanks for all your support. It never fails to put a smile on our faces.

—JK

"FABULOUS"

FEST WEST 95

SAN DIEGO

SATURDAY, FEBRUARY 18, 1995

PRODUCTS!

A wide selection of hardware, software, and accessories
for the TI 99/4A & Myarc Geneve 9640 computers.

VENDORS!!

Comprodine, Tex-Comp, WHT, Bud Mills Services, and More!

PRIZES!!!

Myarc HFDC w/20MB Hard Drive, Horizon 4000 RamDisk,
SCSI card, complete TI system, & much more!

& MORE...

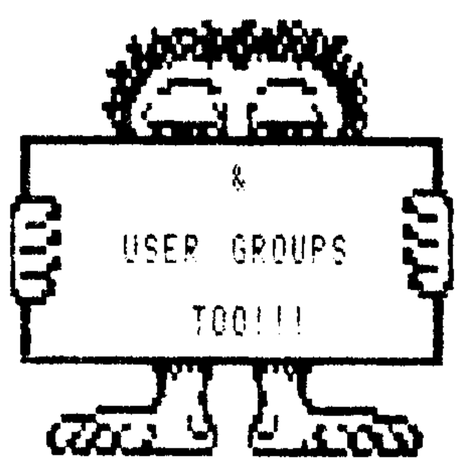
Sea World, San Diego Zoo, Wild Animal Park, The Beach,
Harbor Cruises, "Fabulous" Shopping Malls, & much more!

FEST WEST '95 is being held at the FABULOUS INN.
For reservations and related information, please write
or call: FABULOUS INN, 2485 Hotel Circle Place,
San Diego, CA 92108. In CA, TOLL FREE 1 (800) 647-1903
or in U.S., TOLL FREE 1 (800) 824-0950.

Admission is $5 per person. Raffle tickets are ONLY $1 each.
Buy ten (10) raffle tickets & receive one (1) FREE ADMISSION.
For more information about TI Fest West '95, please write:

Southern California Computer Group
P.O. BOX 152535
San Diego, CA 92195

Or Call the S.C.C.G. Data Line at:
(619) 268-9135 300-2400 BPS 8N1
User No.: 25 Password: FEST



FEEDBACK

'Hoax' a misnomer

As an ex-user of the 99/4A, 99/8 and the Geneve 9640, I still stay current with the TI community. The reason I'm writing is this: I was told by a subscriber of your publication that there was an article about a *hoax* regarding Texas Instruments developments, namely the 99/5 "Waxwing 5," the 99/8 "Armadillo" and so on. Well, I used to own a 99/8 with all the Hex-Bus peripherals (wafertape, printer/plotter, 80-column printer, 5.25-inch DS/DD disk drive/controller, 300-baud modem, RS232 interface and the Video Interface for the CC40) and the schematics for each including the 99/8 itself. These were sent to me by Mike Bunyard of the Bunyard Manual fame with his own personal notes regarding the "Armadillo" project. I've held in *my* hands one of the *very few* "Waxwing 5" motherboards and have a copy or the schematics and I have photographs of the motherboard. I've owned such peripheral cards as the IEEE-488, EPROM programmer card, the Super Modem card, the 128K card and TI's DS/DD card.

So, as far as a hoax goes, I don't know about that! It seems to me that someone is very misinformed or I was told the wrong thing by my friend.

Steve Eggers
Abilene, Texas

Reunion a hit

Just wanted to drop you a note to say "Thanks" for the excellent coverage you gave the Dallas TI Home Computer Group (DTIHCG) regarding our Reunion Oct. 29 at the Infomart in Dallas. By publicizing our Reunion in the September 1994 issue of MICROpendium (page 4) and again in the October issue (page 12) you really helped us to get the word out. Several attendees came up to me and said they read about our Reunion in the MICROpendium. Your write-up really helped. We had a turnout of between 65 and 75 dedicated 99ers. And, as a result, we had several former members rejoin our group and also picked up four new members.

We feel the Reunion was a great success. Many former members attended and we all enjoyed the visiting each other and reminiscing about the old days. Plus, this gave us an opportunity to show how far the TI99/4A home computer has come from the early '80s to the present "State-of-the-Orphan" that it is today. We accomplished this by demonstrations and displays: Genealogy histories printed via the 99/4A and Beyond Wordwriter cartridge; displays of T-shirts with graphics from Page Pro; hard drives for the 4A; Cor-Comp sidecars; and a demonstration of the TI-Emulator conversion program. And in between demonstrations and displays, there were several different kinds of cookies, coffee, cold drinks, etc.

I've enclosed a photo of Mike Stanfill who came dressed as a 99/4A sitting atop a file cabinet, P-Box and disk files. This *was* a highlight, with Mike going around the room saying, "Hey, what are you looking at?" or "Did you lose something?" We all got a big laugh out of Mike's shenanigans.

So, in closing, let me thank you again for the great coverage you gave our Reunion; we all really appreciated it.

Dan M. Lowe, President
Dallas TI Home Computer Group
Garland, Texas

Newsletter notes

In the November issue you published a review by Dr. Charles Good of Newsletter Editor and Formatter by Bill Gaskill. In the review Dr. Good rightly stated that Art Gibson's Newsletter Printer software does not have a WYSIWYG (what you see is what you get) screen. Persons not familiar with Art's software might consider that a shortcoming, when in fact Art's software has a feature called "View Pages" which emulates the WYSIWYG screen with some additional features that are beneficial to a newsletter editor in adjusting the location of text and graphics before printing. One can readily recognize what a paper saver this tool proves to be.

If the editor needs to compare the space required for different printing options he/she needs only to load the text into the Editor and enter or alter the options as desired: ".P1" for pica one column; ".P2" for pica two column; ".C1 for condensed one column; or ".C2" for two-column condensed. Also, special character codes permit the choice of elite type in any of these four stated options. All these options can be exercised at any point in the text file as often as desired. When finished, one saves the text file to disk, leaves the editor and selects Option 2 on the screen, which is the printer program menu. Now select Option 3 for configuration and then Option 2 for VIEW PAGES. The file will be displayed (imitated) on screen and depicts the page space required for the options exercised.

A format file can be created which links several files together including graphics and permits one to view or print continuous pages of an entire newsletter. As I view the entire program, its only limitation is that all graphics must be processed in the pica one-column mode and this limits the use of graphics in two-column options.

We here at K-Town have used this program for years and would find it difficult to believe it to be inadequate in any manner. True, errors sometimes show up in our printed copy, which is no fault of the software. These errors are the result of poor editing combined with time limitations relative to publishing deadlines.

If anyone has any problems with Art's Newsletter Printer software (not to be confused with First Draft) I will be glad to help resolve the problem if the individual will contact me after reading the instructions packaged with the software. Also, I will send anyone a copy of the software for the sum of $3 to cover copying and mailing fees. Be sure to include a legible name and address. For either offer contact me, E.M. Smith, at 3506 Garden Dr., Knoxville, TN 37918. This is the same address used by the K-Town Users Group

E.M. Smith
Knoxville, Tennessee

MONTECOLOR

Like solitaire, but without the cards

By LUCIE DORAIS

Another year... my fifth... After a quiet summer? Not exactly, as I purchased a PC laptop last May. And while I did spend much time playing with it, I found that it renewed my interest in Tex. How? Just as I was really running out of inspiration for my column, I got ideas from PC programs.

One shareware disk I got for the PC is SOLMENU1, a collection of solitaire card games by Karl Meadows. I had always wanted to write solitaire games, but it seems that Regena did them all in MICROpendium. The ones I found in a book I own I did not find interesting enough, or too easy to play. Well, SOLMENU provided me with five new games to play; I especially like CANFIELD, so I wrote a TI version.

Another game, MONTE-CARLO, I found easy to program, but a bit boring to play, until I read about a shareware PC version that replaces the cards with colored squares (called SOLISQUARE). You have to remove cards from the tableau in pairs; the cards must be adjacent either vertically, horizontally, or diagonally. The colored squares make it easier to play, since is it easier to spot colors than card values; but, more importantly, it allows the game to be played by young children. Full instructions are at lines 820-870 (GOSUBed from line 140).

Since a card deck has four suits of 13 cards, we use 13 colors, starting at No. 4 to avoid middle green. That makes the use of text a bit hard. After much planning, I devised a colorful menu by redefining the needed letters in different color sets (one set for each menu item). In DATA lines 120-130, the first number is the character to be CHARPATed, the second is the new character. The letters are redefined in that order: (line 120) K,T,M,O,V,E,A,R; (line 130) N,E,W,D,A,L,F,S,H,). We do K first because we use it for T right after, so we can display "INIT" in line 190 (I and N are not changed, being in the right color set). Lines 220-240 redefine more graphic characters: Arrows (140-143) and space bar (139) for the menu, cursor (137) and a flag (81) that shows a card has been marked. A$ and HD$ are here temporary variables.

The deck, four cards for each suit/color, is built by lines 200-210, by using the first character of each set and adding it four times to DK$. The menu is displayed by lines 260-280 at the right of the screen (meaning in English given as remarks. The screen will show it in uppercase.).

You have to remove cards from the tableau in pairs; the cards must be adjacent either vertically, horizontally, or diagonally. The colored squares make it easier to play, since is it easier to spot colors than card values; but, more importantly, it allows the game to be played by young children.

The shuffle routine comes from another TI solitaire game I own, "Solitaire" by Ben Elizer (*Compute*, January 1986). I am not sure how it works, but it is much faster that randomly finding 52 cards, checking each time if it has been found before, a routine which drags towards the end. Here, the random function is called exactly 52 times. The shuffled deck is built into A$, then copied into DK$. From it we take the hand (HD$) of 25 cards that will be first dealt on screen. PD is the position of the next card in DK$ to be taken when we ask for a new deal. T1 is a temporary variable for line 360, R1 and C1 temporarily hold the cursor position after a shuffle (upper left card). G is the number of the game: After each session you are given back a summary of the games played. The maximum number of games per session is 60, but you can change the DIM GP(60) in line 140 for more (!). OK$ will hold the number of pairs found in each game. The dots displayed at the bottom of the screen by line 340 show the total pairs you must find to win (26); each time you find one, a dot will be replaced by a "gold coin" in the string OK$.

To draw the 25 cards on the screen, HD$ is read character by character (card by card), then we CALL GET the corresponding row and column for each card, derived from its Xth position in HD$. Then the card character K found at position X is put on the screen in a 3x3 square.

The rest of line 380 initializes some variables after each new deal: R,C are for the cursor, PIK tells Tex how many cards have been marked so it knows when to go and check for a pair. The main CALL KEY (line 400-410) and the cursor movement (lines 420-470) are easy to follow. When you move at the end of a row, you end up at the beginning of the next row (be careful, the check-key string in line 400 contains a space).

You mark a card, which you think forms a pair, by pressing the space bar. The CALL KEY will bring you to line 490. If the spot is empty (card removed), Tex beeps and returns you to line 400. If the card has already been marked, i.e. it has a flag (three arrows) left to it, the flag is erased, PIK is reset to zero, and back to the CALL KEY. (Here's a clue: If you mark a first card by error, just press space to erase its flag). If the card is valid, PIK is incremented to one or two, and Tex flags it in relation to the cursor: Next row, two columns to the left (the card positions are kept in R1,C1 and R2,C2). If only one card has been marked (PIK=1), we go back to CALL KEY for the second one. Otherwise, Tex automatically goes on to check if you marked a pair.

To successfully form a pair, the cards must be adjacent. The difference between

(See Page 8)

MONTECOLOR—

(Continued from Page 7)

their rows or columns must be four, since each card takes three rows, plus an empty row between the cards. To check that, we use temporary variables T1 and T2 to hold the difference between columns C1/C2 and rows R1/R2, respectively (line 530). Since the cards are not marked in any order, that difference can be positive or negative, hence the use of the ABS function (absolute value). Here T1 and T2 get a value resulting from a "relational expression," which explains the parentheses — they are set to zero if the relation is not true, or to -1 if it is true. These values will both be true if the cards are adjacent diagonally. If they are in the same row (R1=R2) or column (C1=C2), T1 or T2 will be zero, so line 540 checks all possibilities: same row and adjacent columns (T1), same column and adjacent rows (T2), or adjacent rows and columns (diagonals, T1 AND T2).

Note that we don't have to write IF T2<>0, since IF T2 automatically checks that T2 is not zero. If the pair is not adjacent, Tex goes to line 600 to erase the two flags and sound some sour notes (CALL S); PIK is reset to zero. If the cards are adjacent, Tex then checks the color: T1 and T2 take the value of the upper left characters of the two squares. Of course, if they form a pair, they will be the same. If not, more sour sounds.

Okay, a pair has been found. Tex erases both cards and their flags in two CALL ERs, and replaces those cards with asterisks in the hand string HD$ (CALL FLAG). OK$ is incremented by one "gold coin" (char. "f") and displayed for its length with a nice chime sound. The remaining dots stay on screen to remind you how many more pairs you need to find! If you found the 26 pairs (52 cards), Tex will go straight to line 720 to shower you with random sounds and minimal visual effects.

You can remove all the pairs you find in a deal before calling a N)ew Deal from the menu, but you don't have to, since you can call a new deal as often as you want. By starting to look for pairs from the bottom, it is sometimes possible to predict what the new screen will look like, and visualize new pairs at the bottom before you attack the top ones and, therefore, make the predicting more difficult! (Yes, there is some strategy to that game, it is not all boring, automatic play as you might first think.)

When you press "N" for a New Deal, the spots left in the hand (and on screen) by the removed pairs are filled up by the next card in the hand. Of course, if there is no empty spot, Tex knows it, since there is no asterisk in the hand yet, so it beeps and returns you to the CALL KEY (line 630). Otherwise, it erases the cursor, then the portion of the screen located after the first empty spot. It CALL GETs the row and column of that spot (T1 now holds the position of the first asterisk in HD$) and also GETs the row of the last card in the hand (last character in HD$, row put into T2; X is here a dummy for column value, of no use here).

Tex can now erase the squares at the right of the first empty spot for a length of K=22-C, then the remaining rows of cards, from the top screen row of the next full row of cards (R+4) to two screen rows below the top row of the last card (T2+2). R1 and C1 now take the value of the cursor position above the first empty card, in preparation for the screen redrawing. S will temporarily hold the starting position for the next routine, which will remove the asterisk flags from HD$ (lines 670-680). When that is done, we have our new hand, to which we add cards from the deck to make a total of 25. The deck flag PD is incremented accordingly. If the deck is empty, i.e. if its flag is beyond 52, no cards are added. In both cases, Tex goes back to redraw the screen from the first empty spot, i.e. from where it started to erase it. T1 still holds the position of that first empty spot in HD$, now position of first card to put back on screen.

We press "F" to give up, which happens when there are no more pairs to be found. The check in line 410 brings us to line 730 (we don't win, so Tex skips line 720). The array GP keeps the number of cards removed from the deck, which is twice the final number of gold coins/pairs in OK$. It then displays a choice: N)ew or E)nd. Pressing "N" for a new game goes back to the shuffling routine in line 300, after erasing the screen from rows 1 to 20.

If we E)nd the session, Tex clears the screen and colors it pale blue, then it resets the uppercase characters and the digits to their normal self with CALL CHARSET. It then prints the results of all the games we have played, i.e. how many cards were successfully removed from the deck. The printing pauses after each group of 20-game results.

Here are some notes on the user-defined subs: In GET, row R and column C are derived from position X in HD$. The screen displays the cards in five rows and five columns, hence the formulas. Since each row of cards takes four screen rows, C and R are eventually multiplied by four. ER erases both flag and square with four spaces on each row. FLAG is the opposite formula of GET — we have the row and column of the two cards in the pair (R1,C1 and R2,C2 passed as parameters), from which the sub finds the P position of the corresponding character in HD$, which it replaces by an asterisk. The other subs are easy to understand.

MONTECOLOR

```
100 ! **** MONTE-COLOR SOLIT
AIRE **** L.Dorais/Ottawa UG
/Aug. 1991 !243
110 !!131
120 DATA 75,111,84,75,77,106
,79,107,86,108,69,109,65,105
,82,110 !168
130 DATA 78,89,69,90,87,91,6
8,92,65,93,76,94,70,76,83,77
,72,74,41,129 !066
140 DIM SH(52),GP(60):: GOSU
B 820 :: CALL CLEAR :: CALL
SCREEN(2)!027
150 GOTO 170 :: C,C1,C2,G,K,
P,PD,PIK,R,R1,R2,S,T1,T2,X,A
$,DK$,HD$,OK$,P$ !083
160 CALL CHAR :: CALL HCHAR
:: CALL VCHAR :: CALL GCHAR
:: CALL KEY :: CALL COLOR ::
 CALL CHARPAT :: CALL CHARSE
T :: !@P- !115
170 FOR X=2 TO 14 :: CALL CO
LOR(X,X+2,1):: NEXT X :: P$=
CHR$(129)!239
180 FOR X=1 TO 18 :: READ K,
S :: CALL CHARPAT(K,A$):: CA
LL CHAR(S,A$)!140
```

(See Page 9)

MONTECOLOR—

(Continued from Page 8)

```
190 IF X=2 THEN DISPLAY AT(1
2,11):"I N I K" ! init !181
200 NEXT X :: A$="FFFFFFFFFF
FFFFFF" :: FOR X=40 TO 136 S
TEP 8 !229
210 CALL CHAR(X,A$):: DK$=DK
$&RPT$(CHR$(X),4):: NEXT X !
 deck !005
220 A$="18181818" :: HD$="00
"&A$&"7E3C18" !172
230 CALL CHAR(102,"10387CFE7
C3810",81,"00040607070604",1
37,HD$,139,"00FFFFFFFF")!204
240 CALL CHAR(140,"00183C7E"
&A$&HD$&"0010307F7F301000"&"
0004067F7F0604")!098
250 ! ===== menu ===== !250
260 DISPLAY AT(2,22):"jklm":
 :TAB(23);CHR$(140)&" "&CHR$
(141): :TAB(23);CHR$(142)&"
"&CHR$(143)! MOVE/arrows !09
4
270 DISPLAY AT(9,22):"jino":
:TAB(23);RPT$(CHR$(139),3):
" ! MARK/space bar !074
280 DISPLAY AT(15,22):"Y"&P$
&"Z[":TAB(24);"\Z]^": : : :T
AB(22);"L"&P$&"INIMJ" ! n)ew
 deal / f)inish !197
290 ! ===== shuffle ===== !0
50
300 DISPLAY AT(11,5)SIZE(-9)
:"MJ......." :: FOR X=1 TO 5
2 :: SH(X)=X :: NEXT X !190
310 RANDOMIZE :: A$="" :: K=
52 :: FOR X=1 TO 52 :: P=INT
(RND*K)+1 !041
320 A$=A$&SEG$(DK$,SH(P),1):
: SH(P)=SH(K):: K=K-1 :: NEX
T X :: DK$=A$ !053
330 HD$=SEG$(DK$,1,25):: PD=
26 :: T1,R1=1 :: C1=4 :: G=G
+1 :: OK$="" !241
340 DISPLAY AT(11,5)SIZE(-9)
:"" :: DISPLAY AT(23,2):RPT$
(".",26)!176
350 ! ===== redraw screen fr
om hand ===== !249
360 FOR X=T1 TO LEN(HD$):: K
=ASC(SEG$(HD$,X,1)):: CALL G
ET(X,R,C)!213
370 FOR S=C TO C+2 :: CALL V
CHAR(R,S,K,3):: NEXT S !205
380 NEXT X :: PIK=0 :: R=R1
:: C=C1 :: CALL HCHAR(R,C,13
7)!055
390 ! ===== play game/move c
ursor ==== !028
400 CALL KEY(0,K,S):: IF S=0
 THEN 400 ELSE P=POS("EXSD N
F",CHR$(K),1)!078
410 ON P+1 GOTO 400,420,420,
420,420,490,630,730 !189
420 CALL HCHAR(R,C,32):: ON
P GOTO 430,440,450,460 !143
430 R=R-4 :: IF R<1 THEN R=1
7 :: GOTO 470 ELSE 470 !237
440 R=R+4 :: IF R>17 THEN R=
1 :: GOTO 470 ELSE 470 !237
450 C=C-4 :: IF C<4 THEN C=2
0 :: GOTO 430 ELSE 470 !134
460 C=C+4 :: IF C>20 THEN C=
4 :: GOTO 440 !111
470 CALL HCHAR(R,C,137):: GO
TO 400 !000
480 ! ===== mark a card with
 <space> / check pair =====
!230
490 CALL GCHAR(R+1,C,K):: IF
 K=32 THEN CALL S(-3):: GOTO
 400 ! empty spot !041
500 CALL GCHAR(R+1,C-2,K)::
IF K=81 THEN CALL S(-3):: CA
LL EF(R1,C1):: PIK=0 :: GOTO
 400 ! already marked: erase
 first flag !095
510 PIK=PIK+1 :: IF PIK=1 TH
EN R1=R+1 :: C1=C-1 ELSE R2=
R+1 :: C2=C-1 !024
520 CALL VCHAR(R+1,C-2,81,3)
:: IF PIK=1 THEN 400 ! put f
lag=mark !234
530 T1=(ABS(C2-C1)=4):: T2=(
ABS(R2-R1)=4)! T1,T2=-1 if t
rue, zero if false !167
540 IF (R1=R2 AND T1)OR(C1=C
2 AND T2)OR(T1 AND T2)THEN 5
50 ELSE 600 ! adjacent? !083
550 CALL GCHAR(R1,C1,T1):: C
ALL GCHAR(R2,C2,T2):: IF T1<
>T2 THEN 600 ! same col? !01
0
560 ! ===== ok, pair found =
==== !083
570 CALL ER(R1,C1):: CALL ER
(R2,C2)! erase pair/flags !1
05
580 CALL FLAG(R1,C1,HD$):: C
ALL FLAG(R2,C2,HD$):: OK$=OK
$&"f" :: CALL S(2500)!091
590 DISPLAY AT(23,2)SIZE(-LE
N(OK$)):OK$ :: CALL S(3000):
: GOTO 610 !213
600 CALL EF(R1,C1):: CALL EF
(R2,C2):: CALL S(200):: CALL
 S(110)! no pair... !118
610 PIK=0 :: IF LEN(OK$)<26
THEN 400 ELSE 720 !184
620 ! ===== n)ew deal =====
!110
630 T1=POS(HD$,"*",1):: IF T
1=0 THEN CALL S(300):: GOTO
400 ! no empty spot !024
640 CALL HCHAR(R,C,32):: CAL
L GET(T1,R,C):: CALL GET(LEN
(HD$),T2,X):: K=22-C !073
650 FOR X=R TO R+2 :: CALL H
CHAR(X,C,32,K):: NEXT X ! er
ase end of row !135
660 CALL ES(R+4,T2+2):: R1=R
-1 :: C1=C+1 :: S=1 ! erase
end of screen !032
670 P=POS(HD$,"*",S):: IF P=
0 THEN 690 ! remove all flag
s from hand !047
680 HD$=SEG$(HD$,1,P-1)&SEG$
(HD$,P+1,25):: S=P :: GOTO 6
70 !114
690 IF PD>52 THEN 360 ELSE X
=25-LEN(HD$)! deck empty or
not !098
700 HD$=HD$&SEG$(DK$,PD,X)::
 PD=PD+X :: GOTO 360 ! add c
ards from deck !254
710 ! ===== f)inish game ===
== !169
720 FOR X=1 TO 15 :: DISPLAY
 AT(23,2):"" :: DISPLAY AT(2
3,2):OK$ :: S=500*(INT(RND*5
)+1):: CALL S(S):: NEXT X !
you won !185
730 GP(G)=2*LEN(OK$):: DISPL
AY AT(24,9)BEEP:"Y"&P$&"Z[
  Z"&P$&"Y\" !n)ew  e)nd !04
2
740 CALL KEY(0,K,S):: IF S=0
 OR(K<>69 AND K<>78)THEN 740
 !018
750 IF K=78 THEN DISPLAY AT(
23,1):"": :"" :: CALL ES(1,2
0):: GOTO 300 !195
760 ! ===== end / game resul
ts ===== !183
```

(See Page 10)

MONTECOLOR—

(Continued from Page 9)

```
770 CALL CLEAR :: CALL SCREE
N(8):: CALL CHARSET :: FOR X
=1 TO G :: IF GP(X)=52 THEN
A$="* WON *" ELSE A$="" !223
780 PRINT USING "GAME ##: ##
 CARDS  #######":X,GP(X),A$
:: IF INT(X/20)<>X/20 THEN 8
00 ELSE PRINT : :"< PRESS A
KEY TO CONTINUE >" !207
790 CALL KEY(0,K,S):: IF S=0
 THEN 790 ELSE DISPLAY AT(23
,1):"" !053
800 NEXT X :: END !251
810 ! ===== instructions ===
== !122
820 DISPLAY AT(1,4)ERASE ALL
:"MONTE-COLOR Solitaire": :"
The cards are big colored":"
  squares (4x13 colors)." !1
47
830 DISPLAY AT(6,1):"You mus
t try to remove all":"  52 c
ards in pairs; cards":"  mus
t be adjacent either" !120
840 A$="     " :: DISPLAY AT
(9,1):A$&". horizontally":A$
&". vertically":A$&". diagon
ally": :"When done, ask a Ne
w deal:" !021
850 DISPLAY AT(14,1):". empt
y spots will disappear. next
 card will fill them  . new
cards will be added" !057
860 DISPLAY AT(17,3):"from d
eck if not empty": :"k|  <EX
SD> MOVE the cursor":"e| <SP
ACE> MARK a card" !033
870 DISPLAY AT(21,1):"y|"&A$
&"<N> NEW DEAL of hand s|"&A
$&"<F> NEW GAME / END": :A$&
" < press a key >" !013
880 CALL KEY(0,K,S):: IF S=0
 THEN 880 ELSE RETURN !192
890 !@P+ ===== user-def. sub
s ===== !038
900 SUB GET(X,R,C):: R=INT(X
/5):: IF R<>X/5 THEN R=R+1 !
178
910 C=4*(X-5*R+5)-1 :: R=4*R
-2 :: SUBEND ! get r,c from
hand position !137
920 SUB ER(R,C):: FOR X=R TO
 R+2 :: CALL HCHAR(X,C-1,32,
4):: NEXT X :: SUBEND ! eras
e square/flags !129
930 SUB FLAG(A,B,A$):: R=(A+
2)/4 :: C=(B+1)/4 :: P=5*(R-
1)+C !125
940 A$=SEG$(A$,1,P-1)&"*"&SE
G$(A$,P+1,25):: SUBEND ! put
 flag in hand !121
950 SUB ES(A,B):: FOR R=A TO
 B :: CALL HCHAR(R,2,32,20):
: NEXT R :: SUBEND !012
960 SUB EF(R,C):: CALL VCHAR
(R,C-1,32,3):: SUBEND ! eras
e a flag !037
970 SUB S(X):: CALL SOUND(-1
00,X,0):: SUBEND !099
```

THE ART OF ASSEMBLY — PART 42

At long last, bit map

By BRUCE HARRISON

Seems we're forever thanking people in this column, and this month we're going to put the thanks right up front. Thanks to John C. Johnson of Cedar Rapids, Iowa. John is a regular reader of our column, and a very skilled programmer. He read our complaint about not being able to get Bit Map mode to work, and supplied some immediate help. His help was in the best possible form, as source code files on disk. These files included a correct way of setting up the Bit Map mode and a plotting subroutine so that we can turn on any selected pixel on the screen. FANTASTIC! This was just exactly what we were hoping to be able to do! Today's column will be the first of two parts devoted to Bit-Map mode.

AN OLD STORY

John's package to us included a very old article from the IUG magazine, in which Bill Gronos explained the operation of Bit Map mode, and supplied the subroutines that John incorporated into his own "sample" programs. Long-time readers of this column will remember that way back, near the beginning of this series, we gave you a series of subroutines that could be used in Graphics Mode to do things like put strings on the screen and such. This month, we will do much the same for the Bit-Map

(See Page 11)

Sidebar 42

```
0001  * SIDEBAR 42
0002  * THREE FILES FOR BITMAP EXPERIMENTS
0003  * FIRST, THE MAIN FILE - BITEXP/S
0004  *
0005  * Bitmap source code     01/15/94
0006  * Loads from EA3 or funnelweb choice number 4.
0007  * code derived from John C. Johnson and Bill Gronos
0008  * modified and combined by Bruce Harrison
0009  * PUBLIC DOMAIN
0010  * BITEXP/S
0011        DEF  START        DEFINE ENTRY POINT
0012        REF  VWTR,KSCAN,VMBW,VMBR,VSBW,VSBR
0013  KEYADR EQU  >8374       KEY-UNIT ADDRESS
0014  START  LWPI WS          LOAD OUR WORKSPACE
0015        LI   R0,>380      POINT AT COLOR TABLE
0016        LI   R1,SAVCLR    AND AT STORAGE SPACE
0017        LI   R2,32        32 BYTES TO GET
0018        BLWP @VMBR        READ COLOR TABLE INTO STORAGE
0019        MOV  @>8370,R0    GET VDP ADDR FROM >8370
0020        LI   R1,ANYKEY+1  POINT AT STORAGE BUFFER
0021        LI   R2,6         SIX BYTES TO READ
0022        BLWP @VMBR        READ THOSE INTO BUFFER
0023        LI   R0,>800      POINT AT CHARACTER TABLE
0024        LI   R1,CHRTBL    AND AT BUFFER STORAGE
0025        LI   R2,256*8     256 CHARACTER DEFINITIONS
0026        BLWP @VMBR        STASH CHARACTER DEFS
0027        CLR  @KEYADR      CLEAR KEY-UNIT
0028  MENU   LI   R1,MENDAT   POINT AT MENU DATA
0029        LI   R0,32+6      ROW 2, COL 7 OF SCREEN
0030        BL   @DISSTR      DISPLAY MENU TITLE
0031        LI   R4,4         FOUR ITEMS IN MENU
```

THE ART OF ASSEMBLY—

(Continued from Page 10)

mode. The column itself will be fairly short, as the sidebar is a complete program that does some things to illustrate the use of the subroutines. This program, among other things, goes back and forth between the Graphics and Bit Map modes to show that this can be done smoothly. We've also used some slick moves at the beginning of the program to capture the character definitions and color tables that were in use while we're in graphics mode. The character definitions are then used while we're in Bit Map mode to print legends on the screen.

THE SOURCE CODE

This month's sidebar is a complete program, organized into three parts. The main program is called BITEXP/S. This is supported by subroutines in BITSUBS and by a data section called BITDATA. The main file uses the others with copy directives. The heart of all this is the BITSUBS file, where we've supplied subroutines to get you gracefully into and out of Bit Map mode, to plot lines or curves pixel-by-pixel, and to put text strings or just single ASCII characters on screen. The coordinate system for the single-pixel plotting is similar to the Dot-Row and Dot-Column system you'd use for placing sprites. For characters or text, the Row and Column are used just like for Graphics Mode. In both cases, the subroutine will accept a color instruction placed in R9 before the BL. If R9 is clear, no change will be made to whatever color that part of the screen is already set up for. For characters or strings, R9 specifies in its left byte the foreground and background colors. For single pixel plotting, the left nybble of R9 specifies the foreground color for that pixel, with the background remaining whatever it was for that part of the screen.

THE PROGRAM

It doesn't do a whole lot, but here's what: It puts a menu on-screen in Graphics Mode, with four choices offered. Any keystroke outside the 1-4 range will be ignored. Selecting 1 through 3 will cause a switchover to the Bit Map mode, which will set up as black-on-white colors. For BOXES, a series of single-pixel boxes will appear, each one inside the previous one, until there's room for no more. All these boxes are made blue on white, since R9 was set to >4000 before we started drawing. Then a legend "BOXES" is written as a string in white on blue at the bottom of the screen. Once that's done, the computer just waits for a keypress, then returns to the Graphics Mode and displays the menu. The graphics mode colors and characters are put back where they belong in VDP Ram as part of the SETGM routine.

Selection 2 makes a Bit Map rectangular spiral on the screen, changing colors for each leg of each time around the screen. The vertical lines are all black, lines on the bottom are red, while lines across the top are blue. The word SPIRAL gets displayed at the bottom of the screen in white on dark green. Pressing a key returns to the menu.

Selection 3 uses those many data entries in BITDATA to draw sine curve on the screen. The X-axis is labeled at 0, 90, 180, 270, and 360 degree spots, then "A SINE WAVE" appears twice — near the bottom of the screen in black on white, then again near the top in white on dark green. This is done just to illustrate

(See Page 12)

```
0032        LI   R0,32*7+6    ROW 8, COL 7 OF SCREEN
0033  MENU0  BL   @DISSTR     DISPLAY AN ITEM
0034        AI   R0,64        MOVE DOWN TWO ROWS
0035        DEC  R4           DECREMENT COUNT
0036        JNE  MENU0        IF NOT ZERO, REPEAT
0037        LI   R0,22*32+8   ELSE POINT ROW 23, COL 9
0038        BL   @DISSTR      DISPLAY SELECT LEGEND
0039  MENKEY BL  @KEY         GET A KEYSTROKE
0040        MOV  @KEYADR,R8   MOVE KEY'S VALUE TO R8 AS WORD
0041        AI   R8,-49       SUBTRACT ASCII FOR "1"
0042        JLT  MENKEY       IF LESS THAN ZERO, REJECT
0043        CI   R8,3         COMPARE TO THREE
0044        JGT  MENKEY       IF GREATER, REJECT
0045        JEQ  EXIT         IF EQUAL, EXIT
0046        SLA  R8,1         ELSE DOUBLE NUMBER IN R8
0047        MOV  @LUT(R8),R12 GET BRANCH ADDRESS INTO R12
0048        BL   @SETBM       SET UP BIT MAP MODE
0049        B    *R12         BRANCH TO SELECTED FUNCTION
0050  EXIT   MOV  @>8370,R0    GET BACK >8370 ADDRESS
0051        LI   R1,ANYKEY+1  POINT AT BUFFER STORAGE
0052        LI   R2,6         SIX BYTES
0053        BLWP @VMBW        WRITE THOSE BACK TO VDP
0054        CLR  R0           CLEAR OUR R0
0055        LWPI >83E0        LOAD GPL WORKSPACE
0056        B    @>6A         RETURN TO GPL INTERPRETER
0057  SINWAV LI   R12,SINDAT   POINT AT SINE DATA
0058        LI   R9,>C000     SET COLOR TO DARK GREEN
0059  POINT  MOV  *R12+,R7     MOVE DOT COLUMN NUMBER TO R7
0060        AI   R7,20        ADD 20
0061        MOV  *R12+,R8     MOVE DOT ROW DATA TO R8
0062        CI   R12,CRSOVR   CHECK FOR COLOR CHANGE POINT
0063        JLT  POSWV        IF LESS THAN, SKIP AHEAD
0064        LI   R9,>6000     ELSE SET COLOR FOR DARK RED
0065  POSWV  NEG  R8          MULTIPLY BY -1
0066        AI   R8,96        ADD COORDINATE ORIGIN
0067        BL   @PLOT        DRAW ONE PIXEL
0068        CI   R12,ENDDAT   ARE WE PAST DATA?
0069        JLT  POINT        IF NOT, REPEAT FOR NEXT PIXEL
0070        LI   R8,96        POINT AT DOT-ROW 96
0071        LI   R7,255       DOT-COLUMN 255
0072        LI   R9,>4000     SET COLOR FOR DARK BLUE
0073  LINELP BL  @PLOT        PLOT ONE PIXEL AT DOT-ROW, DOT-COLUMN
0074        DEC  R7           DEC COLUMN COUNT
0075        JNE  LINELP       IF NOT ZERO, WRITE ANOTHER
0076        LI   R8,14        ROW 14
0077        LI   R7,3         COL 3
0078        LI   R6,48        CHARACTER 48 ("0")
0079        LI   R9,>4F00     COLOR DARK BLUE ON WHITE
0080        BL   @CHAR        PLACE THE CHARACTER
0081        LI   R7,9         COL 9
0082        LI   R12,LEG0     STRING LEG0
0083        BL   @BITSTR      DISPLAY STRING ("90")
0084        LI   R7,16        COL 16
0085        BL   @BITSTR      DISPLAY NEXT STRING ("180")
0086        LI   R7,23        COL 23
0087        BL   @BITSTR      NEXT STRING ("270")
0088        LI   R7,30        COL 30
0089        BL   @BITSTR      NEXT STRING ("360")
0090        CLR  R9           USE EXISTING COLORS (BLACK ON WHITE)
0091        LI   R8,23        ROW 23
0092        LI   R7,11        COL 11
0093        BL   @BITSTR      NEXT STRING "A SINE WAVE"
0094        LI   R12,LEG4     POINT BACK AT "A SINE WAVE"
0095        LI   R9,>FC00     COLOR WHITE ON DARK GREEN
0096        LI   R8,2         ROW 2
0097        LI   R7,14        COL 14
0098        BL   @BITSTR      DISPLAY STRING
0099        BL   @KEY         WAIT FOR KEYPRESS
0100        BL   @CPDT        CLEAR THE PATTERN TABLE
0101        BL   @SETGM       RE-SET TO GRAPHICS MODE
0102        B    @MENU        RETURN TO MENU
0103  SPIRAL LI   R12,20      STARTING DOT-COL 20
0104        LI   R13,10       STARTING DOT-ROW 10
0105        LI   R14,180      STOP ROW 180
0106        LI   R15,240      STOP COLUMN 240
0107        MOV  R13,R8       PUT START ROW IN R8
0108  LINE   MOV  R12,R7      STARTING COL IN R7
0109        AI   R12,10       ADD 10 TO START COL
0110        CLR  R9           COLORS BLACK ON WHITE
```

THE ART OF ASSEMBLY—

(Continued from Page 11)

an important difference between the normal Graphics mode and the Bit Map mode. The same characters can be set into different colors on the same screen, provided only that the characters are written in a different place.

Selection 4 exits gracefully back to Editor/Assembler, but will also exit gracefully to Funnelweb if that's how it was loaded. The EXIT routine replaces six bytes at the location pointed to by >8370 so that DSR routines will work upon return to E/A or Funnelweb. We stashed away those six bytes as part of our opening section, so that on return to E/A, everything in E/A would work correctly.

THE SUBROUTINES

First in the subroutines is SETBM. This is the means of getting from Graphics mode into Bit Map. We re-arranged the order of performing these steps from the original, so that the transition to bit map is made at the very end, after all the conditions have been set. This makes the transition appear "seamless," so no glitches appear during the changeover. We also took the part that clears out the character definition table into a separate subroutine so that it can be used by itself as a "clear screen" while you're in the Bit Map mode. We also use that subroutine to clear things before our transition back to graphics mode, again to minimize any cosmetic problems in that transition.

The PLOT subroutine was copied as is from what Johnson sent, but we added the ability to color the pixel just written. The CHAR feature was added to show that it could be done. Inventive readers will find this works for any characters, even those taken from revised character sets, CHARA1 files, and such. Now that we have cracked the door open, we'll try to pass along some more techniques and tips for bit map operations in future columns. As we write this, we're working on a little "toy" program to draw pictures and such. The TI world doesn't need any more such programs, but we learn a lot more by writing them than by using the existing programs, and learning is what this column is all about.

Because this month's sidebar is so large, we're going to save the explanation of how all this works for next month. Keep this one handy, so next time you'll be able to study how all this little "example" program works.

```
0111 LOOP1  BL   @PLOT       DRAW ONE PIXEL
0112        INC  R8          MOVE DOWN ONE ROW
0113        C    R8,R14      COMPARE TO LIMIT
0114        JL   LOOP1       IF LOW, REPEAT
0115        LI   R9,>6000    COLOR DARK RED
0116 LOOP2  BL   @PLOT       DRAW ONE PIXEL
0117        INC  R7          INC COLUMN
0118        C    R7,R15      COMPARE TO LIMIT
0119        JL   LOOP2       IF LOW, REPEAT
0120        CLR  R9          COLOR BLACK ON WHITE
0121 LOOP3  BL   @PLOT       DRAW ONE PIXEL
0122        DEC  R8          DEC ROW
0123        C    R8,R13      COMPARE TO TOP LIMIT
0124        JH   LOOP3       IF HIGH, REPEAT
0125        MOV  R13,R8      PUT LIMIT IN R8
0126        AI   R13,10      ADD 10 TO TOP LIMIT
0127        LI   R9,>4000    COLOR DARK BLUE
0128 LOOP4  BL   @PLOT       DRAW ONE PIXEL
0129        DEC  R7          DEC COL
0130        C    R7,R12      COMPARE TO LEFT LIMIT
0131        JH   LOOP4       IF HIGH, REPEAT
0132        AI   R14,-10     SUBTRACT 10 FROM LEFT LIMIT
0133        AI.  R15,-10     AND FROM STOP COLUMN
0134        C    R13,R14     COMPARE TOP AND BOTTOM LIMITS
0135        JLT  LINE        IF LESS, BACK TO START
0136        LI   R8,24       ROW 24
0137        LI   R7,15       COL 15
0138        LI   R9,>FC00    COLOR WHITE ON DARK GREEN
0139        LI   R12,SPISTR  STRING "SPIRAL"
0140        BL   @BITSTR     DISPLAY THAT
0141        BL   @KEY        ELSE WAIT FOR KEYSTROKE
0142        BL   @CPDT       CLEAR PATTERN TABLE
0143        BL   @SETGM      SET GRAPHICS MODE
0144        B    @MENU       RETURN TO MENU
0145 BOXES  LI   R12,20      STARTING COLUMN
0146        LI   R13,10      STARTING ROW
0147        LI   R14,180     STARTING BOTTOM LIMIT
0148        LI   R15,240     STARTING RIGHT LIMIT
0149        LI   R9,>4000    COLOR DARK BLUE
0150 BOX0   MOV  R12,R7      PUT COL IN R7
0151        MOV  R13,R8      PUT ROW IN R8
0152 BOX01  BL   @PLOT       DRAW ONE PIXEL
0153        INC  R8          INC ROW
0154        C    R8,R14      COMPARE TO BOTTOM LIMIT
0155        JL   BOX01       IF LOW, REPEAT
0156 BOX02  BL   @PLOT       DRAW ONE PIXEL
0157        INC  R7          INC COL
0158        C    R7,R15      COMPARE TO RIGHT LIMIT
0159        JL   BOX02       IF LOW, REPEAT
0160 BOX03  BL   @PLOT       DRAW ONE
0161        DEC  R8          DEC ROW
0162        C    R8,R13      COMPARE TO LIMIT
0163        JH   BOX03       IF HIGH, REPEAT
0164 BOX04  BL   @PLOT       DRAW ONE
0165        DEC  R7          DEC COL
0166        C    R7,R12      COMPARE TO LIMIT
0167        JH   BOX04       IF HIGH, REPEAT
0168        AI   R12,10      ADJUST START COLUMN
0169        AI   R13,10      ADJUST START ROW
0170        AI   R14,-10     ADJUST BOTTOM ROW
0171        AI   R15,-10     ADJUST BOTTOM COL
0172        C    R13,R14     COMPARE TOP, BOTTOM LIMITS
0173        JLT  BOX0        IF LESS THAN, DRAW NEXT BOX
0174        LI   R8,24       ROW 24
0175        LI   R7,15       COL 15
0176        LI   R9,>F400    COLOR WHITE ON DARK BLUE
0177        LI   R12,BOXSTR  STRING "BOXES"
0178        BL   @BITSTR     DISPLAY THAT
0179        BL   @KEY        WAIT FOR KEYSTROKE
0180        BL   @CPDT       CLEAR PATTERN TABLE
0181        BL   @SETGM      SET GRAPHICS MODE
0182        B    @MENU       BACK TO MENU
0183        COPY "DSK1.BITSUBS" COPY IN SUBROUTINES
0184        COPY "DSK1.BITDATA" COPY IN DATA FILE
0185        END
0186 * END OF BITEXP/S
0187 *
0188 * SECOND FILE - BITSUBS
0189 * 15 JAN 1994
0190 *
0191 * SUBROUTINES FOR HANDLING BIT-MAP
0192 * OPERATIONS AND TRANSITIONS
0193 *
0194 * FOLLOWING SECTION SETS COMPUTER INTO BIT-MAP MODE
0195 *
0196 SETBM  LI   R0,>206     SET TO WRITE VDP REGISTER 2
0197        BLWP @VWTR       SIT TO >1800  (SCREEN IMAGE TABLE)
0198        LI   R0,>403     SET TO WRITE TO VDP REG. 4
0199        BLWP @VWTR       PDT TO >0000  (PATTERN DESCRIPTOR TABLE)
0200        LI   R0,>3FF     SET TO WRITE TO VDP REG 3
0201        BLWP @VWTR       CT TO >2000   (COLOR TABLE)
0202        LI   R0,>607     SET TO WRITE VDP REG 6
0203        BLWP @VWTR       Sprite descritor table to >3800
0204        LI   R0,>570     SET TO WRITE VDP REG 7
0205        BLWP @VWTR       Sprite atribute list to >3800
0206        LI   R0,>58      INITIALIZE SCREEN IMAGE TABLE (SIT) (AT >1800)
0207        MOVB R0,@>8C02   WRITE LOW BYTE VDP ADDRESS
0208        SWPB R0          SWAP R0
```

THE ART OF ASSEMBLY—

```
0209          MOVB R0,@>8C02   WRITE HIGH BYTE VDP ADDRESS
0210          LI   R0,3        THREE TABLES OF 256 BYTES EACH
0211          CLR  R1          START WITH ZERO
0212  SIT     MOVB R1,@>8C00   WRITE TO VDP (SELF-INCREMENTING)
0213          AI   R1,>100     ADD 1 TO HIGH BYTE R1
0214          JNE  SIT         IF NOT ZERO, REPEAT
0215          DEC  R0          ELSE DEC COUNT
0216          JNE  SIT         IF NOT ZERO, REPEAT
0217          LI   R0,>60      INIT COLOR TABLE  (CT) AT >2000
0218          MOVB R0,@>8C02   WRITE LOW BYTE OF ADDRESS
0219          SWPB R0          SWAP R0
0220          MOVB R0,@>8C02   WRITE HIGH BYTE OF ADDRESS
0221          LI   R0,>1800    >1800 BYTES TO WRITE
0222          LI   R1,>1F00    COLORS ALL BLACK ON WHITE
0223  CT      MOVB R1,@>8C00   WRITE ONE BYTE
0224          DEC  R0          DEC COUNT
0225          JNE  CT          IF NOT ZERO, REPEAT
0226          MOV  R11,R14     STASH RETURN ADDRESS
0227          BL   @CPDT       CLEAR PATTERN TABLE
0228          LI   R0,2        SET R0 TO WRITE 2 TO VDP REGISTER ZERO
0229          BLWP @VWTR       SET TO M3 MODE (BIT MAP)
0230          B    *R14        RETURN
0231  CPDT    LI   R0,>40      CLEAR PATTERN DESCRIPTOR TABLE  (PDT) AT >0000
0232          MOVB R0,@>8C02   WRITE LOW BYTE ADDR
0233          SWPB R0          SWAP
0234          MOVB R0,@>8C02   WRITE HIGH BYTE ADDRESS
0235          LI   R0,>1800    >1800 BYTES TO WRITE
0236          CLR  R1          ALL ZEROS
0237  PDT     MOVB R1,@>8C00   WRITE ONE
0238          DEC  R0          DEC COUNT
0239          JNE  PDT         IF NOT ZERO, REPEAT
0240          RT
0241  *
0242  * FOLLOWING SETS COMPUTER BACK TO NORMAL GRAPHICS MODE
0243  *
0244  SETGM   LI   R0,>1E0     SET TO WRITE VDP REG 1
0245          BLWP @VWTR       WRITE
0246          LI   R0,>200     SET TO WRITE VDP REG 2
0247          BLWP @VWTR       WRITE
0248          LI   R0,>401     SET TO WRITE VDP REG 4
0249          BLWP @VWTR       WRITE
0250          LI   R0,>30E     VDP REG 3
0251          BLWP @VWTR       WRITE
0252          LI   R0,>600     VDP REG 6
0253          BLWP @VWTR       WRITE
0254          LI   R0,>506     VDP REG 5
0255          BLWP @VWTR       WRITE
0256          LI   R0,>380     POINT AT COLOR TABLE
0257          LI   R1,SAVCLR   AND AT SAVED COLOR DATA
0258          LI   R2,32       32 BYTES
0259          BLWP @VMBW       WRITE THE COLOR TABLE BACK
0260          LI   R0,>800     POINT AT GRAPHICS CHAR TABLE
0261          LI   R1,CHRTBL   AND AT STORED CHARACTER DATA
0262          LI   R2,256*8    256 CHARACTERS
0263          BLWP @VMBW       WRITE CHARACTER DEFS BACK
0264          CLR  R0          PREP TO WRITE VDP REG 0
0265          BLWP @VWTR       WRITE THAT TO REMOVE BIT MAP
0266          RT               RETURN
0267  *
0268  * FOLLOWING WRITES ONE PIXEL TO SCREEN AT LOCATION POINTED BY
0269  * R8 (DOT ROW) AND R7 (DOT COLUMN)
0270  *
0271  PLOT    MOV  R7,R3       MOVE DOT COLUMN TO R3
0272          MOV  R8,R4       AND DOT ROW TO R4
0273          MOV  R4,R5       DOT ROW ALSO IN R5
0274          ANDI R5,7        R5 HAS DOT ROW MODULO 8
0275          SZC  R5,R4       SO DOES R4
0276          SLA  R4,5        MULTIPLY R4 BY 32
0277          A    R5,R4       ADD R5, SO R4 HAS DR MOD. 8 * 32 + DR MOD 8
0278          MOV  R3,R0       MOVE DOT COL TO R0
0279          ANDI R0,>FFF8    R0 HAS DC - DC MOD 8
0280          S    R0,R3       R3 HAS DC MOD 8
0281          A    R4,R0       ADD R4
0282          SWPB R0          SWAP BYTES
0283          MOVB R0,@>8C02   WRITE LOW ADDRESS BYTE
0284          SWPB R0          SWAP
0285          MOVB R0,@>8C02   WRITE HIGH ADDRESS BYTE
0286          NOP              WASTE TIME
0287          MOVB @>8800,R1   READ THE BYTE
0288          SOCB @M(R3),R1   OVERLAY MASK FROM TABLE M
0289          ORI  R0,>4000    SET THE 4000 BIT IN R0
0290          SWPB R0          SWAP
0291          MOVB R0,@>8C02   WRITE LOW BYTE OF ADDRESS
0292          SWPB R0          SWAP
0293          MOVB R0,@>8C02   WRITE HIGH BYTE OF ADDRESS
0294          NOP              WASTE TIME
0295          MOVB R1,@>8C00   WRITE MODIFIED BYTE BACK TO VDP
0296          MOV  R9,R9       IS COLOR TO BE SET?
0297          JEQ  PLOTX       IF NOT, JUMP AHEAD
0298          ANDI R0,>3FFF    STRIP OFF "4" FROM R0
0299          AI   R0,>2000    ADD >2000 TO POINT AT COLOR TABLE ENTRY
0300          BLWP @VSBR       READ THAT BYTE INTO R1
0301          MOVB R1,R2       MOVE THE BYTE TO R2
0302          ANDI R2,>F000    STRIP ALL BUT LEFT NYBBLE
0303          CB   R2,R9       COMPARE TO LEFT BYTE R9
0304          JEQ  PLOTX       IF EQUAL, COLOR ALREADY SET
0305          ANDI R1,>0F00    ELSE STRIP OFF LEFT NYBBLE R1
0306          AB   R9,R1       REPLACE WITH LEFT NYBBLE R9
0307          BLWP @VSBW       THEN WRITE COLOR BYTE BACK
0308  PLOTX   RT               RETURN
0309  BITSTR  MOV  R11,R15     STASH R11
0310          MOV  R7,R13      SAVE COLUMN IN R13
0311          MOVB *R12+,R4    GET STRING LENGTH BYTE IN R4
0312          JEQ  BITSX       IF ZERO, SKIP PROCESS
0313          SRL  R4,8        RIGHT JUSTIFY
0314  BITST0  MOVB *R12+,R6    MOVE ONE BYTE OF STRING TO R6
0315          SRL  R6,8        RIGHT JUSTIFY
0316          BL   @CHAR       DISPLAY THAT CHARACTER
0317          INC  R7          INC COLUMN
0318          DEC  R4          DEC LENGTH COUNT
0319          JNE  BITST0      IF NOT ZERO, REPEAT
0320          MOV  R13,R7      PUT COLUMN BACK IN R7
0321  BITSX   B    *R15        ELSE RETURN
0322  CHAR    MOV  R8,R0       PUT ROW COUNT IN R0
0323          DEC  R0          DEC TO ZERO-BASE NUMBER
0324          LI   R2,8        PUT 8 IN R2
0325          SLA  R0,5        MULTIPLY R0 BY 32
0326          A    R7,R0       ADD COLUMN
0327          DEC  R0          DEC TO ZERO-BASE COLUMN
0328          SLA  R0,3        MULTIPLY R0 BY 8
0329          MOV  R6,R1       PUT CHARACTER FROM R6 INTO R1
0330          SLA  R1,3        MULTIPLY BY 8
0331          AI   R1,CHRTBL   ADD START OF STORED CHARACTER DEFINITIONS
0332          BLWP @VMBW       WRITE 8 BYTES TO VDP RAM
0333          MOV  R9,R9       CHECK FOR COLOR CHANGE
0334          JEQ  CHARX       IF NONE, SKIP AHEAD
0335          AI   R0,>2000    ELSE ADD COLOR TABLE OFFSET
0336          MOVB R9,R1       MOVE COLOR BYTE TO R1
0337  CHCL    BLWP @VSBW       WRITE ONE BYTE
0338          INC  R0          POINT AT NEXT LOCATION
0339          DEC  R2          DEC COUNT IN R2
0340          JNE  CHCL        IF NOT ZERO, REPEAT
0341  CHARX   RT               RETURN
0342  KEY     BLWP @KSCAN      SCAN KEYBOARD
0343          LIMI 2           ALLOW INTERRUPTS
0344          LIMI 0           THEN TURN THEM OFF
0345          CB   @ANYKEY,@>837C KEY STRUCK?
0346          JNE  KEY         IF NOT, SCAN AGAIN
0347          RT               ELSE RETURN
0348  DISSTR  MOVB *R1+,R2     GET STRING LENGTH INTO R2
0349          SRL  R2,8        RIGHT JUSTIFY
0350          BLWP @VMBW       WRITE STRING TO SCREEN
0351          A    R2,R1       ADD LENGTH TO POINTER
0352          RT               RETURN
0353  * END OF BITSUBS
0354  *
0355  * THIRD FILE - BITDATA
0356  * 15 JAN 1994
0357  *
0358  *
0359  * DATA SECTION
0360  *
0361  WS      BSS  >20         OUR WORKSPACE
0362  M       DATA >8040,>2010,>0804,>0201  MASK DATA
0363  *
0364  * FOLLOWING IS DATA FOR THE SINE WAVE
0365  * EACH PAIR OF WORDS IS ONE POINT IN DOT ROW, DOT COLUMN (ZERO BASED)
0366  *
```

THE ART OF ASSEMBLY—

```
0367  * THIS WAS CREATED
USING AN XB PROGRAM
0368  * TO CORRECTLY SCALE
THE DATA FOR BIT-MAP SCREEN
0369  *
0370  SINDAT DATA 0,0
0371         DATA 1,2
0372         DATA 2,5
0373         DATA 3,8
0374         DATA 4,11
0375         DATA 5,14
0376         DATA 6,17
0377         DATA 7,20
0378         DATA 9,23
0379         DATA 10,26
0380         DATA 11,29
0381         DATA 12,32
0382         DATA 13,34
0383         DATA 14,37
0384         DATA 15,40
0385         DATA 16,43
0386         DATA 18,45
0387         DATA 19,48
0388         DATA 20,50
0389         DATA 21,53
0390         DATA 22,55
0391         DATA 23,58
0392         DATA 24,60
0393         DATA 25,62
0394         DATA 27,64
0395         DATA 28,67
0396         DATA 29,69
0397         DATA 30,71
0398         DATA 31,73
0399         DATA 32,74
0400         DATA 33,76
0401         DATA 35,78
0402         DATA 36,80
0403         DATA 37,81
0404         DATA 38,83
0405         DATA 39,84
0406         DATA 40,85
0407         DATA 41,87
0408         DATA 42,88
0409         DATA 44,89
0410         DATA 45,90
0411         DATA 46,91
0412         DATA 47,91
0413         DATA 48,92
0414         DATA 49,93
0415         DATA 50,93
0416         DATA 51,94
0417         DATA 53,94
0418         DATA 54,94
0419         DATA 55,94
0420         DATA 56,94
0421         DATA 57,94
0422         DATA 58,94
0423         DATA 59,94
0424         DATA 60,94
0425         DATA 62,93
0426         DATA 63,93
0427         DATA 64,92
0428         DATA 65,92
0429         DATA 66,91
0430         DATA 67,90
0431         DATA 68,89
0432         DATA 70,88
0433         DATA 71,87
0434         DATA 72,86
0435         DATA 73,84
0436         DATA 74,83
0437         DATA 75,81
0438         DATA 76,80
0439         DATA 77,78
0440         DATA 79,77
0441         DATA 80,75
0442         DATA 81,73
0443         DATA 82,71
0444         DATA 83,69
0445         DATA 84,67
0446         DATA 85,65
0447         DATA 86,63
0448         DATA 88,60
0449         DATA 89,58
0450         DATA 90,56
0451         DATA 91,53
0452         DATA 92,51
0453         DATA 93,48
0454         DATA 94,46
0455         DATA 96,43
0456         DATA 97,40
0457         DATA 98,38
0458         DATA 99,35
0459         DATA 100,32
0460         DATA 101,29
0461         DATA 102,26
0462         DATA 103,23
0463         DATA 105,21
0464         DATA 106,18
0465         DATA 107,15
0466         DATA 108,12
0467         DATA 109,9
0468         DATA 110,6
0469         DATA 111,3
0470         DATA 112,0
0471  CRSOVR DATA 114,-3
0472         DATA 115,-6
0473         DATA 116,-9
0474         DATA 117,-12
0475         DATA 118,-15
0476         DATA 119,-18
0477         DATA 120,-21
0478         DATA 121,-24
0479         DATA 123,-27
0480         DATA 124,-29
0481         DATA 125,-32
0482         DATA 126,-35
0483         DATA 127,-38
0484         DATA 128,-41
0485         DATA 129,-43
0486         DATA 131,-46
0487         DATA 132,-48
0488         DATA 133,-51
0489         DATA 134,-53
0490         DATA 135,-56
0491         DATA 136,-58
0492         DATA 137,-61
0493         DATA 138,-63
0494         DATA 140,-65
0495         DATA 141,-67
0496         DATA 142,-69
0497         DATA 143,-71
0498         DATA 144,-73
0499         DATA 145,-75
0500         DATA 146,-77
0501         DATA 147,-79
0502         DATA 149,-80
0503         DATA 150,-82
0504         DATA 151,-83
0505         DATA 152,-85
0506         DATA 153,-86
0507         DATA 154,-87
0508         DATA 155,-89
0509         DATA 156,-90
0510         DATA 158,-91
0511         DATA 159,-92
0512         DATA 160,-92
0513         DATA 161,-93
0514         DATA 162,-94
0515         DATA 163,-94
0516         DATA 164,-95
0517         DATA 166,-95
0518         DATA 167,-95
0519         DATA 168,-95
0520         DATA 169,-95
0521         DATA 170,-95
0522         DATA 171,-95
0523         DATA 172,-95
0524         DATA 173,-95
0525         DATA 175,-94
0526         DATA 176,-94
0527         DATA 177,-93
0528         DATA 178,-93
0529         DATA 179,-92
0530         DATA 180,-91
0531         DATA 181,-90
0532         DATA 182,-89
0533         DATA 184,-88
0534         DATA 185,-87
0535         DATA 186,-85
0536         DATA 187,-84
0537         DATA 188,-83
0538         DATA 189,-81
0539         DATA 190,-79
0540         DATA 192,-78
0541         DATA 193,-76
0542         DATA 194,-74
0543         DATA 195,-72
0544         DATA 196,-70
0545         DATA 197,-68
0546         DATA 198,-66
0547         DATA 199,-64
0548         DATA 201,-62
0549         DATA 202,-59
0550         DATA 203,-57
0551         DATA 204,-55
0552         DATA 205,-52
0553         DATA 206,-50
0554         DATA 207,-47
0555         DATA 208,-44
0556         DATA 210,-42
0557         DATA 211,-39
0558         DATA 212,-36
0559         DATA 213,-33
0560         DATA 214,-31
0561         DATA 215,-28
0562         DATA 216,-25
0563         DATA 217,-22
0564         DATA 219,-19
0565         DATA 220,-16
0566         DATA 221,-13
0567         DATA 222,-10
0568         DATA 223,-7
0569         DATA 224,-4
0570         DATA 225,-1
0571  ENDDAT EQU  $           END OF SINE DATA
0572  LUT    DATA BOXES,SPIRAL,SINWAV  MENU BRANCH TABLE
0573  MENDAT BYTE 21
0574         TEXT 'BITMAP SELECTION MENU'
0575         BYTE 16
0576         TEXT '1.  BITMAP BOXES'
0577         BYTE 17
0578         TEXT '2.  BITMAP SPIRAL'
0579         BYTE 20
0580         TEXT '3.  BITMAP SINE WAVE'
0581         BYTE 15
0582         TEXT '4.  EXIT TO E/A'
0583         BYTE 16
0584         TEXT 'SELECT BY NUMBER'
0585  LEG0   BYTE 2
0586         TEXT '90'
0587  LEG1   BYTE 3
0588         TEXT '180'
0589  LEG2   BYTE 3
0590         TEXT '270'
0591  LEG3   BYTE 3
0592         TEXT '360'
0593  LEG4   BYTE 11
0594         TEXT 'A SINE WAVE'
0595  BOXSTR BYTE 5
0596         TEXT 'BOXES'
0597  SPISTR BYTE 6
0598         TEXT 'SPIRAL'
0599  ANYKEY BYTE >20          COMPARISON BYTE FOR KEYSTROKE
0600         BSS  6           STORAGE FOR DSR DATA FROM VDP RAM
0601  SAVCLR BSS  32           STORAGE FOR GRAPHICS COLOR TABLE
0602  CHRTBL BSS  256*8        STORAGE FOR GRAPHICS CHARACTER DE-
FINITIONS
```

GENEVE

VRAM analyzer lets user determine VRAM memory address

By JIM UZZELL
©1994 DDI SOFTWARE

In order to do any serious programming on the Geneve that involves the 9938 video chip, you must have a copy of the V9938 manual, which is available from 9640 News.

Readers who have the manual know that it is not written in a manner that is easy to understand, so I wrote the VRAM Analyzer to allow me to quickly ascertain the VRAM memory address based on mode and display page.

This program uses a XOP6 routine to

(See Page 15)

VRAM ANALYZER—

(Continued from Page 14)

capture the table of addresses built into MDOS. I believe these tables do not accurately reflect the VRAM memory locations based on the display page of the mode selected, but you can be the judge of that.

Regardless of mode, each page starts with page 0 through one less than what is displayed on the screen for that mode.

All files must be on drive 1, unless you change program line 120.

This program uses an object code file (V9938-O). Use MY-Word to type it in and save it as a fixed file with a name of V9938-O.

VRAMTEST1

```
1 !V9938 ANALYSIS
2 !DDI SOFTWARE
3 !COPYRIGHT 1994
4 !PGM NAME VRAMTEST1
100 CALL GRAPHICS(3,1) :: CL
S
110 DIM A(24),B(94),A$(11),B
$(11),R$(46),C$(46)
120 CALL INIT :: CALL LOAD("
DSK1.V9938-O")
130 M=1 :: P=0 :: TBL$(1)=""
 :: TBL1$(1)=""
140 FOR X=0 TO 10 :: READ A$
(X) :: NEXT X
150 FOR X=0 TO 10 :: READ B$
(X) :: NEXT X
160 DISPLAY AT(5,34):"INSIDE
 THE V9938"
170 DISPLAY AT(7,28):"VIDEO
MODE          MAX PAGES"
180 FOR X=0 TO 10 :: DISPLAY
 AT(X+9,28):A$(X);TAB(50);B$
(X); :: NEXT X
190 DISPLAY AT(20,34):"ESC=E
XIT";
200 DISPLAY AT(22,34)BEEP :"
WHICH MODE" :: DISPLAY AT(24
,1):"DDI SOFTWARE COPYRIGHT
1994" :: GOSUB 670
210 IF K=155 THEN 640
220 IF K=65 THEN M=10 :: GOT
O 240
225 IF K>57 THEN 200
230 M=K-48
240 DISPLAY AT(22,34)BEEP :"
which PAGE 0"; :: ACCEPT AT(
22,45)VALIDATE(DIGIT)SIZE(-2
):K :: GOSUB 690 :: P=K
250 CALL LINK("V9938",(M),(P
),TBL$(),TBL1$())
260 CALL GRAPHICS(3,1)
270 FOR X=0 TO 23
280 A(X)=ASC(SEG$(TBL$(1),X+
1,1))
290 NEXT X
300 FOR X=0 TO 93 :: B(X)=AS
C(SEG$(TBL1$(1),X+1,1)) :: N
EXT X
310 Y=0 :: FOR X=0 TO 93 STE
P 2 :: R$(Y)=SEG$(HEX$ (B(X+
1)),3,2) :: Y=Y+1 :: NEXT X
320 BIN$="00000001001000110100
0010101100111100010011010101
1111001101111011111"
330 FOR X=0 TO 46 :: FOR R=1
 TO 2
340 PS=POS("0123456789ABCDEF
",SEG$(R$(X),R,1),1)
350 C$(X)=C$(X)&SEG$(BIN$,(4
*PS)-3,4) :: NEXT R :: NEXT
X
360 CT$=SEG$(HEX$ (A(0)),3,2
)&SEG$(HEX$ (A(1)),3,2)
370 PT$=SEG$(HEX$ (A(4)),3,2
)&SEG$(HEX$ (A(5)),3,2)
380 PNT$=SEG$(HEX$ (A(8)),3,
2)&SEG$(HEX$ (A(9)),3,2)
390 SPRA$=SEG$(HEX$ (A(12)),
3,2)&SEG$(HEX$ (A(13)),3,2)
400 SPRP$=SEG$(HEX$ (A(16)),
3,2)&SEG$(HEX$ (A(17)),3,2)
410 SPRC$=SEG$(HEX$ (A(20)),
3,2)&SEG$(HEX$ (A(21)),3,2)
420 DISPLAY AT(1,1):"ANALYSI
S OF MODE  ";A$(M);
430 DISPLAY AT(3,1):"DEFAULT
 VRAM MEMORY LOCATION";
431 DISPLAY AT(4,14):"PAGE "
;P
432 IF M=1 OR M=4 OR M=5 OR
M=10 THEN 440 ELSE 450
440 DISPLAY AT(5,1):"COLOR T
ABLE           >";CT$;" ";
VALHEX(CT$)
450 DISPLAY AT(6,1):"PATTERN
 GENERATOR TABLE >";PT$;" ";
VALHEX(PT$)
460 DISPLAY AT(7,1):"PATTERN
 NAME TABLE     >";PNT$;" "
;VALHEX(PNT$)
470 IF M=0 OR M=1 OR M=10 TH
EN 520
480 DISPLAY AT(8,1):"SPRITE
ATTRIBUTE TABLE  >";SPRA$;"
```

(See Page 16)

```
OBJECT CODE FILE V9938-O

0014C       A0000B0006B0000B0018A0006A001EB005EA0020A007EB002F7F337F       0001
A0080B0000B0000B0000BC80BC0002B02E0BF000B0200B0000B0201B00017F347F         0002
A0096B0420B200CB0420B2018B12B8BC820B834AC0082B0200B0000B02017F30DF         0003
A00ACB0002B0420B200CB0420B2018B12B8BC820B834AC0084B0200B00007F2F7F         0004
A00C2BC060C0082B2C20C0000B0200B0004BC060C0084B0202B0000B2C207F31AF         0005
A00D8C0000B0200B0037B0201C0006B2C20C0000B0200B0001B0201B00037F35EF         0006
A00EEB0202C0005B0420B2010B0205B0000B0200B0036BC060C0080B2C207F32AF         0007
A0104C0000BC940C0020B05C5B05A0C0080B8820C0080C007EB1301B10F07F2F9F         0008
A011AB0200B0001B0201B0004B0202C001FB0420B2010B0200B0000B02017F36BF         0009
A0130B0001B2C20C0000B0200B0004B0201B0000B0202B0000B2C20C00007F370F         0010
A0146BC2E0C0002B045B7FB6FF                                                   0011
50086V9938 7FD73F                                                           0012
:       9640 AS
```

VRAM ANALYZER—

(Continued from Page 15)

```
";VALHEX(SPRA$)
490 DISPLAY AT(9,1):"SPRITE
GENERATOR TABLE  >";SPRP$;"
";VALHEX(SPRP$)
500 IF M=2 OR M=3 OR M=4 THE
N 520
510 DISPLAY AT(10,1):"SPRITE
 COLOR TABLE      >";SPRC$;"
 ";VALHEX(SPRC$)
520 DISPLAY AT(12,1):"REGIST
ER VALUES FOR THIS MODE"
530 FOR X=0 TO 9 :: DISPLAY
AT(X+13,1):X;"  ";R$(X);" ";
C$(X); :: NEXT X
540 FOR X=10 TO 11 :: DISPLA
Y AT(X+13,1):X;" ";R$(X);" "
;C$(X); :: NEXT X
550 FOR X=12 TO 23 :: DISPLA
Y AT(X+1,20):X;" ";R$(X);" "
;C$(X); :: NEXT X
560 FOR X=32 TO 43 :: DISPLA
Y AT(X-19,39):X;" ";R$(X);"
";C$(X); :: NEXT X
570 FOR X=44 TO 46 :: DISPLA
Y AT(X-31,58):X;" ";R$(X);"
";C$(X); :: NEXT X
575 FOR X=24 TO 31 :: DISPLA
Y AT(X-7,58):X;" ";R$(X);" "
;C$(X);" ?"; :: NEXT X
580 FOR X=1 TO 79 :: CALL HC
HAR(26,X,ASC(SEG$("HARDCOPY=
PRT SCREEN------------ANY KE
Y FOR MENU                DDI
 SOFTWARE 1994",X,1))) :: NE
XT X
590 CALL KEY(0,K,S) :: IF S<
1 THEN 590
600 CALL HCHAR(26,1,32,79)
610 CALL MEMSET(A$(),"") ::
CALL MEMSET(B$(),"") :: CALL
 MEMSET(R$(),"") :: CALL MEM
SET(C$(),"")
620 CALL MEMSET(A(),0) :: CA
LL MEMSET(B(),0)
630 CLS :: RESTORE :: GOTO 1
20
640 CLS :: END
650 DATA 0 TEXT 1,1 TEXT 2,2
 MULTICOLOR,3 GRAPHIC 1,4 GR
APHIC 2,5 GRAPHIC 3,6 GRAPHI
C 4,7 GRAPHIC 5,8 GRAPHIC 6,
9 GRAPHIC 7,A TEXT 2-26
660 DATA 32,16,32,32,8,8,4,4
,2,2,16
670 CALL KEY(0,K,S) :: IF S<
1 THEN 670
680 RETURN
690 IF M=0 AND K>31 THEN 800
700 IF M=1 AND K>15 THEN 800
710 IF M=2 AND K>31 THEN 800
720 IF M=3 AND K>31 THEN 800
730 IF M=4 AND K>7 THEN 800
740 IF M=6 AND K>3 THEN 800
750 IF M=7 AND K>3 THEN 800
760 IF M=8 AND K>1 THEN 800
770 IF M=9 AND K>1 THEN 800
780 IF M=10 AND K>15 THEN 80
0
790 RETURN
800 DISPLAY AT(22,34)BEEP :"
PAGE ERROR" :: FOR X=1 TO 10
00 :: NEXT X :: GOTO 240
```

Static electricity

Low temperatures and humidity increase problems for computer users

By DENNIS HATHAWAY

The following article appeared in HUGgers, the newsletter of the Hoosier Users Group.—Ed.

Fall is here, winter is just around the corner.

With winter comes low temperatures and extremely low relative humidity (RH). Low RH presents us with the problem of dealing with very high levels of static electricity. So it is timely for a discussion of static.

What is static? What can it do to our hardware? What precautions can we take to reduce problems?

First, what is it? Every person, animal and machine carries with them an electrostatic charge (Q), except when in direct contact with a conductive path to ground. We accumulate this charge as we move around. For example, walking on a carpet, touching things or others that carry a charge greater than ours, rubbing together of items of clothing, especially clothing of man-made materials, even the friction of air flowing past us.

Depending on the materials involved, we can become charged either positively or negatively with respect to ground. When we touch something with a different level of charge, we often feel a sharp tingle or minute burn. Oddly, we don't often feel anything if the potential is less than 2300 volts, because then the total static power involved is small. This sensation is caused by the discharge of the static we were carrying, its Electro Static Discharge (ESD). More on ESD later.

The charge we carry (Q) is given by Q=CV, where Q is in coulombs, C is in microfarads and V is in volts. For a human being, C is usually around 100 to 250 picofarads, depending mainly on overall bulk, size of feet and type of footwear. Because women's shoes often have thin soles, their capacity (C) is often higher than that of a man. Typically, Q ranges up to five microcoulombs.

A calculation will show that a person whose Q equals three microcoulombs and whose C equals 200 picofarads will be at 15 KV with respect to ground, where one kilovolt equals 1000 volts.

In extreme cases, this voltage can be as high as 40 KV. Beyond this point, the charge will bleed off through the air and sometimes may form a corona. The drier the air, the less the bleed-off. This is why we should be concerned when the relative humidity is low.

In upper Wisconsin, during the fall several years ago, I measured 23 KV on a m walking toward me over a vinyl-tiled floor. He was wearing shoes with crepe soles. The measurements were made with

(See Page 17)

STATIC ELECTRICITY—

(Continued from Page 16)

a grounded electrostatic meter. At the same location, the plastic cover on a notice board was at 45 KV. Even allowing for meter inaccuracy, this still gives an indication of the level of voltages out there during the winter months. We must not let this cause us to think that we generate static only when the relative humidity is low. This is not the case. We generate static all year long, but when the relative humidity is high, the charge bleeds away more readily.

I am sure that all of us have at some time touched the screen of our monitor/TV while working at the keyboard and have heard the air crackling as we became charged by some 12 to 15 KV. When we next touched the keyboard, we partially discharged into the console. Texas Instruments was fully aware of this problem when it designed the 99/4A and almost certainly took suitable precautions to prevent damage to the console when it is zapped in this way.

But what can happen to our equipment if we pass our charge to it in a manner TI did not anticipate?

Actually, we share our charge with the equipment until both ourselves and it are at an equal potential, then current flow ceases. In practice, a typical discharge is found to be somewhere on the order of 20 millijoules. A joule equals one watt per second.

When we discharge into our equipment, it takes place very rapidly, depending on our capacitance (C), and our skin resistance, usually around 1000 ohms. The effect is that we can be feeding some 20 amps into the equipment for some 100 nanoseconds. Doing this to the connector contacts on a module or a Peripheral Expansion Box cards will damage to them. In some cases, the damage will be obvious because the system will quit operating. However, usually we end up with a device that is only partially damaged. I refer to these devices as "walking wounded." The device continues to work but it is likely to die at any time, perhaps weeks later after we've completely forgotten that we'd zapped it with 20 millijoules.

When we discharge into our equipment, it takes place very rapidly, depending on our capacitance (C), and our skin resistance, usually around 1000 ohms. The effect is that we can be feeding some 20 amps into the equipment for some 100 nanoseconds.

It is general knowledge that MOS (Metal Oxide Semiconductor) devices used in many TI assemblies are susceptible to this kind of harsh treatment. However, bipolar transistors, diodes and other devices are also affected in a similar way, but to a somewhat lesser degree.

How do we prevent static damage to our 99/4A? As we are about to sit down at our terminal we should bear in mind that every item around us has some capacity to ground, so by touching them briefly we share our charge with them. Do this to a couple of items (preferably large metallic ones) like a chair or filing cabinet, etc. Personally I use an exposed screw head on my grounded power distribution expansion strip which sits on top of my PE box. If we are working on a printed circuit (PC) board outside of our PEB, or of our printer, then a more elaborate method is needed. Consider, while sitting at the bench/table we may raise our potential by as much as 1,000 volts each time we reach out to pick up a tool or device. An aid known as a wrist strap will help here. A conductive band goes on the wrist, and is connected via a 1.5 Megohm resistor to ground. This continually bleeds away any static as fast as we generate it. The resistor limits current if we accidentally touch something with mains potential on it. These wrist straps are available from most electronic parts vendors for only a few dollars.

While the above discussion is directed to work on the 99/4A, it applies equally to other electronic items — radio, TV, VCR, etc. Static can ruin most electronic equipment.

It would be well to ensure that one's spouse and children are made aware of this static problem, so they will not nullify our efforts to prevent damage to our equipment.

CC40

TI gave up too soon on this great little computer

By CHARLES GOOD

The CC40 (which stands for "Compact Computer 40") was in early 1983 TI's first ever entry into the portable computer market. It is in many respects a little brother to the 99/4A, so much so that Funnelweb's senior author Tony McGovern calls the CC40 "Little Tex." This article, based on my own experience using the CC40 system, describes the CC40 computer and its tiny peripherals. Some of these are rare collector's items. The article also lists current sources of supply where you can purchase the CC40, its software, and important peripherals.

The CC40 computer is battery powered, very small, (smaller than most modern laptops), and it was offered with a host of small peripherals, most of which are also battery powered. Without the need to plug into an external power source, a CC40 system allows truly portable computing and printing anywhere. This little orphan is of interest to owners of 99/4A computers for two reasons:

1. The syntax of its built-in CC40 BASIC language is almost identical to the 99/4A's Extended BASIC.

2. TI intended the 99/2, the 99/8, the

(See Page 18)

CC40—

(Continued from Page 17)

99/4A, the CC40, and all of its tiny peripherals to be physically cabled to each other and to talk to each other using a proprietary bus connector called a "HexBus" that is found on all these machines except the 99/4A. To make HexBus devices work with the 99/4A, TI intended to sell a "HexBus interface," a peripheral that had a HexBus connector and that attached to the right side of a 99/4A console. The HexBus interface is pictured on the boxes that contained beige 99/4A consoles but it was never officially released. I own a HexBus interface and use it regularly as part of the 99/4A system on a little table next to my bed. The interface, when combined with tiny HexBus peripherals, permits an expanded 99/4A system to occupy very little surface area.

Although the CC40 is no longer manufactured by TI, the computer, cartridge-based software, and some of its tiny peripherals are still available from dealers such as those listed at the end of this article.

When it was introduced, the CC40 had a list price of $250. Sales were not good in 1983 and 1984 because no mass storage device was made available by TI. The promised cheap Wafertape Digital Tape Drive turned out to be exactly that, cheap. It was unreliable and thus never released, and at that time TI had no other inexpensive CC40-compatible mass storage device to offer the public. In 1984, production of this fantastic little computer ceased. In May 1990 I paid $95 for my new CC40. New CC40s are available now for $49 from Jim Lesher, and used computers are available from several of the dealers listed below. For an extra $20-$25 you can purchase an expanded memory CC40 or have dealer installation (by L.L. Conner Enterprise) of the necessary chips to bring the CC40's internal RAM to the maximum 18K, up from the 6K RAM found in the typical CC40. Conner will also sell you the RAM chips if you want to do the job yourself. This extra memory increases the CC40's internal buffer capacity to around five double-spaced pages of word processing text.

The CC40 measures about 9x6x1 inches, the size of a small textbook. It uses a 2.5MHz TMS70C20 8-bit processor and has 34K of ROM and 6K (expandable to 18K internally) CMOS RAM. There is a "solid state cartridge" port, and the internal RAM can be further expanded with 8K or 16K memory expansion cartridges. Software cartridges, such as the Memo Processor word processing cartridge, can also be inserted into the cartridge port. The ROM includes a very powerful and familiar-looking BASIC. Both upper- and true lowercase letters (not just small uppercase letters) are provided. Error and system messages can be displayed in either English or German.

I have no idea what the "40" in CC40 refers to, certainly not the CC40's display. The LCD display shows 31 characters of a single 80-character line. You need to move the display left/right to view the entire line. Four dedicated cursor keys allow you to scroll up/down to view other lines or left/right within a line of text or program code. The LCD display includes special indicators for such things as low-battery, the status of the shift function and control keys, uppercase lock, and special math functions. Some LCD display indicators are user programmable. A control on the left side of the CC40 regulates the contrast (intensity) of the LCD display.

A very important feature of the CC40 is that any BASIC program or any word processing document entered into the CC40's RAM stays there even after the computer is turned off.

The CC40's keyboard consists of chiclet keys. Alpha numeric keys are arranged in a 44-key qwerty typewriter layout, with number keys on the top row. It looks similar to the 99/4A keyboard arrangement. No, you can't easily touch-type. The alpha keys are just too close together. One finger pecking is the usual method of laptop data entry while holding the CC40 steady with your other hand. It is not necessary to press two keys at once. For those features, such as one-time capital letters that require the use of the Space, FN (function), or CTL (control) keys, either press both keys at once or press the special key first and see an indicator on the LCD display turn on. You then press the second key — for instance Shift and then D to display an uppercase "D," or FN and then tilde ( ) for insert — and the special LCD display indicator turns off. A separate numeric keypad is to the right of the qwerty alphanumeric keys. The number keys on the top row of the qwerty layout are duplicated in this keypad. Special keys are included for cursor movement (four dedicated keys), Break, Run, On, Off, and reset.

200 HOUR BATTERIES

A very important feature of the CC40 is that any BASIC program or any word processing document entered into the CC40's RAM stays there even after the computer is turned off. Four alkaline AA cells are said to provide enough power for 200 hours of operation. My experience shows that these batteries will last many months of "computer off" time. Compare this to the 2-4 hours most "modern" laptops will run using their batteries. The CC40 and all its small battery powered peripherals can also be powered with an AC adapter.

The BASIC that comes as standard equipment on the CC40 closely resembles TI Extended BASIC, but lacks most of the 99/4A's graphic, color, and sound features. There are no sprites and only one kind of programmable Beep. Multi-line statements up to 80 characters in length are supported, as are user defined subprograms with variables independent of the main program. Seven (ASCII 0-6), can be user defined with CALL CHAR on a 5x8 pixel grid. CALLs relating to assembly code include POKE, LOAD (an assembly subprogram from an external device), PEEK, and EXEC (starts an assembly language program). Two dimensional arrays are supported.

Typing BASIC code into the CC40 is made easy with automatic line numbers (NUM) as in Extended BASIC. Delete

(See Page 19)

CC40—

(Continued from Page 18)

will delete one line number or a specified group of line numbers from the middle of a BASIC program. You can type the words for BASIC functions and commands with the alpha keys one letter at a time. However, many BASIC commands and functions can also be displayed on screen by pressing only one or two keys. A plastic keyboard overlay that comes with the CC40 shows these special keypresses, most of which involve pressing the CTL or FN key followed by another key.

A particularly powerful feature you can access from command mode or from a running BASIC program is CALL DEBUG, which brings up a built-in assembly language monitor and memory manager. This is designed to be used with the CC40's Editor Assembler Module (never officially released), but can also be used by itself. When in the DEBUG monitor you can display, modify, or copy any memory in hex. You can also change the microprocesso's program counter, stack pointer, and status register. You can set break points, single step through assembly code, start execution at a given address, and control paging in and out of system ROM and cartridge ROM. DEBUG is very powerful, and it is built into the CC40 for use whenever needed.

User defined hot keys can be set up, and remain in battery backed memory even after the CC40 is turned off. FN + 1-9 are the potential hot keys. These can, for example, be set up for commonly entered BASIC code, number sequences used in math calculations, or short text memos such as names and addresses.

No little calculator can do a better job than the CC40 for the display of chain number calculations. I routinely use the CC40 to balance my checkbook and to calculate student grades from a series of numerical student exam scores. You can type in up to 80 characters of mathematical numbers and symbols (such as 112.56+56.35-45-54.95+12) and then scroll left/right to make sure that all your numbers are correctly entered before pressing Enter to display the answer. Pressing "play back" will redisplay the numbers of the chain calculation that gave you that answer. If your chain is greater than 80 characters, you can enter part of the chain and press Enter for an intermediate answer. Then, starting with the intermediate answer, enter the rest of the numbers of the chain and press Enter to get the complete mathematical answer to the entire chain calculation.

You can also use the CC40 as a scientific calculator by typing in your calculations directly, rather than writing a BASIC program to do the calculations. Calculation accuracy is 13 significant figures, with 10 significant figures usually showing on the CC40's display. Scientific notation is supported, allowing the CC40 to deal with numbers as small as +/-1E-128 or as large as +/-9.9999999999999E+127. PI, SQR, any other power or root, log (base 10, and base E), sine, cosine, tangent, arcsine, arccosine, and arctangent are all supported with special keypresses. Angles are calculated in either degrees, radians, or grads. A special indicator on the LCD display (DEG, RAD, or GRAD) shows which kind of angle is in effect. RAD is the powerup default. You could easily spend $30 for a hand-held scientific calculator, and you would still not have a 31-column display or a scrolling 80-column data field. For a few more dollars you can have a CC40, which is a real programmable computer as well and not just a calculator.

For an investment of $125 ($50 for the CC40 and $75 for the Printer 80), you can have a totally portable, battery powered word processing system

WORD PROCESSING

For me the most practical use of the CC40 is as a portable word processor. When used as a word processing system, the following CC40 hex bus peripherals are important:

1. Memo Processor, a CC40 software cartridge; $20 new with an extensive instruction book. Actually I prefer to use my own BASIC CC40 word processing program, which does not require a cartridge and which is more stable in the CC40's memory. Send me a self-addressed stamped envelope and I will send you a hard copy of this program.

2. The HexBus RS232; about $30-50 used. This is a very important peripheral. You can use it to print to a regular parallel printer from the CC40 or to send word processing text or other data to another computer. To send word procesing text from a CC40 to a 99/4A use a HexBus cable to connect the CC40 to a HexBus RS232 peripheral and run a serial cable from it to the RS232 port of your 99/4A. You can then send text directly into TI-Writer or the Funnelweb editor without using a terminal emulator program or null modem on the 99/4A. Here's how. From TI-Writer type "LF" (load file) and specify "RS232.CR" as the file name. Then, using either Memo Processor or my own CC40 BASIC word processing program, tell the CC40 to SEND its text. Text will flow out of the CC40 and into the TI-Writer edit buffer. When the computer lights stop flashing press FCTN/4 on the 99/4A and your text originally entered into the CC40 will be displayed on the 99/4A's monitor ready for further editing and saving to a TI disk. The HexBus RS232 is the only HexBus peripheral that is not battery powered. It needs an AC adapter.

3. The HexBus Printer 80; around $100 new or used. This small (about 13x6x2 inches) 80-column thermal dot-matrix printer is powered by 4 "D" batteries or an AC adapter. It uses small ribbon cartidges to print on ordinary 8.5 x 11-inch typing paper, or you can print on rolls of 8.5-inch wide fax paper without the ribbon cartridge.

Thus, for an investment of $125 ($50 for the CC40 and $75 for the Printer 80), you can have a totally portable, battery powered word processing system. For an extra $70 ($50 for the HexBus RS232 and $20 for Memo Processor) you can have everything you need for a complete word processing package.

I am composing this article on my CC40. This paragraph is being written while sitting on a bench in the quadrangle of the O.S.U. Lima Campus enjoying the sun. Other paragraphs will be written later

(See Page 20)

CC40—

(Continued from Page 19)

today sitting on my fromt porch at home and laying in my bed watching the evening news on TV. Then I will dump the text, via my HexBus RS232, to the Funnelweb (TI-Writer) editor on my 99/4A and save it from there to a 99/4A disk that I will send to *MICROpendium*. This is truly portable word processing! A CC40 system is absolutely the cheapest word processing system it is possible to purchase anywhere. Compare these prices to the cheapest laptop computer advertised in *Computer Shopper* or a non-battery powered dedicated word processor/printers with little flip up screens (Brother, Smith/Corona, and similar brands) sold in retail stores and in discount catalogs. Price-wise there is no comparison.

OTHER PERIPHERALS

In addition to the peripherals described above, the following two HexBus peripherals are sometimes still available new or used from dealers. All HexBus peripherals should be purchased with a HexBus cable. Make sure you get one with each peripheral you purchase. You daisy chain the needed peripherals together with such cables and connect the first peripheral in the chain to the CC40. Most HexBus peripherals measure about 6x4.5x1.5 inches and are designed to neatly stack on top of each other.

• Hexbus Printer Plotter — This cute little printer prints on adding machine paper. There are four little ballpoint pens, each a different color. Replacement pens can still be purchased at Radio Shack stores. You can program the X-Y axis movement of each pen as you print multicolored graphs and drawings. Several different text sizes from teeny-tiny to about one inch tall are available. Text can be printed in any direction — vertically facing either left or right, horizontally, and even upside down. Although this printer does have some unique features, it is not really useful in printing documents. Also, it has some reliability problems. There is an internal plastic gear that has a history of breaking (Cecure has a metal replacement gear), and its alkaline battery is soldered in and cannot easily be replaced. If the battery fails to hold a charge you are out of luck even if you use the optional AC adapter.

• Hexbus Modem — This is a 300 baud direct connect modem with rear connectors for two HexBus cables and two RJ11 phone cables. I am told that electronically it has properties that are identical to the 99/4A's acoustic "telephone coupler" modem. It works well, but today would probably be considered little more than a toy. It has been a long time since computer data crawled along phone lines at a speed of only 300 baud. Many information services and BBS systems do not support such a slow speed any more.

AVAILABLE SOFTWARE:

The following official TI software cartridges for the CC40 are available new for $20 each from Cecure Electronics and sometimes less from other dealers listed below. Each cartridge comes with a well-written user guide. They include: Learn Pascal, Memo Processor, Finance, Elementary Engineering, Statistics, Math and Games.

I have about 20 BASIC programs which I will be glad to send you either as hard copy listings or on a "quick disk." Some of these programs take advantage of the special features of various HexBus peripherals. Either send me a quick disk (see below) and a paid return mailer or send me $2 for the hardcopy listings. Your cash pays for return postage and my copying costs.

THE MASS STORAGE PROBLEM

Lack of mass storage options is why the CC40 failed commercially in 1984/85, and this is still a big problem for CC40 owners today. Since I use the CC40 mostly for word processing, I can usually get along without mass storage. Text I enter into my BASIC word processing program for the CC40 or into Memo Processor is conserved for weeks or months in the battery backed RAM of the computer until I can dump the text to my 99/4A system via the HexBus RS232. The following mass storage options are possible:

• 8K Memory Expansion — About $30 used. Functionally this resembles the 99/4A's Mini-Memory cartridge. The 8K CC40 cartridge is battery backed and can be used either for program storage or as memory expansion, but not both. These 8K battery backed cartridges are not very common anymore, but some are still available from dealers listed below. You can purchase a bunch of these and store one BASIC program in each cartridge. Program storage only works if you have a 6K CC40. If you are using an enhanced 18K CC40, the 8K cartridge can only be used for memory expansion.

Combined use of the battery backed cartridge for program loading and the non-battery backed 16K cartridge for RAM expansion works very well with my BASIC word processing program. 16K cartridges are still commonly available for about $30-40 from dealers. First I plug in an 8K cartridge that contains my word processing program and transfer that program to to the RAM of my 6K CC40. I then unplug the 8K cartridge and plug in the 16K RAM expansion cartridge. Executing a CALL ADDMEM adds the 16K to the 6K already in the CC40, giving me 22K of RAM to store text (6-7 double-spaced pages) using my word processing program. You can't do this using the Memo Processor cartridge, which must remain inserted in the CC40 while in use. This is one of the reasons I prefer my BASIC word processing program.

• TI's PC Interface — $60 new, sold directly by TI. This small peripheral, known as the PCIF, plugs into a PC parallel or LPT port and allows BASIC programs and data files in a CC40 to be stored on or loaded from a floppy disk or the hard drive of an IBM-compatible computer. The IBM computer then becomes your mass storage. Sounds great doesn't it! Unfortunately, it is a bit tricky to hook the PCIF to the CC40. The PCIF was made for use with the TI74, which is a more modern and somewhat smaller version of the CC40. Although the PCIF is electrically compatible with the HexBus, the 10-pin holes arranged in one straight line on the PCIF's female connector will not directly plug into a HexBus or a HexBus cable. The HexBus has eight pins arranged in two rows of four. I cut common paper clips to make short wires that stick snugly into the holes in the end of a female HexBus cable and the corresponding holes in the female connector of the PCIF, filling eight of the 10 PCIF connector holes. The remaining two PCIF connector holes are for power, six volts in and out. The CC40 has no way of delivering this needed power to the

(See Page 21)

CC40—

(Continued from Page 20)

PCIF. You have to modify a Radio Shack black cube AC-to-6vDC power adapter so you can plug the adapter into the last two pins of the PCIF. Connecting my CC40 and power adapter as described here to the the PCIF allows me to store CC40 software on PC disks.

• Mechatronic QuickDisk peripheral — This small disk drive is the only HexBus peripheral I have ever heard of that is not made by TI. It was made by a German company specifically for the CC40. I find it to be very fast, reliable, and easy to use for data file and program mass storage. The peripheral is fairly small (7x5.5x3 inches), not battery powered, and uses 2.8-inch disks (not the common 3.5-inch disk size) to store up to 64K on each side of a flippy disk. In 1990 I paid $110 for a new QuickDisk drive. The drive is now out of production and there apparently are no new QuickDisk drives gathering dust on dealer's shelves. If you can find a used, working QuickDisk drive then buy it! Used QuickDisk drives are hard to find.

• Wafertape Digital Tape Drive — This was going to be TI's cheap, portable mass storage device. It ran on batteries or an AC adapter and used a small continuous-loop tape cartridge. Although data was stored serially, it had many of the characteristics of a random access device. For example, programs and data files can be loaded by file name from a wafertape that contains several different files. I own one of these rare devices (serial number 0000007) and several official TI wafer cartridges that have a TI logo on the label. My wafertape drive is not very reliable. Many times I have saved and verified data files or BASIC programs to wafertape only to find that later I can't load this information back into my CC40. Reliability problems are probably why TI never released this peripheral to the public.

• Hexbus Floppy Disk Drive Controller — This also was never released by TI, probably because the CC40 and its peripherals were marketed as an inexpensive alternative to other 1983 computer systems, and the HexBus floppy drive was not inexpensive. This is the rarest and probably the most useful of the HexBus peripherals. The controller worked with IBM-compatible 360K drives and 5.25-inch disks, formatting DSDD at 16 256K sectors per track just like TI's never released DSDD disk controller for the 99/4A. I know of four working HexBus floppy controllers in the whole world. Its too bad one of them isn't mine, yet!

SOURCES OF SUPPLY

• Cecure Electronics. P.O. Box 222, Muskego WI 53150; phone 800-959-9640. This is the official TI service and exchange center for the CC40 and its peripherals. They don't sell the computer or peripherals but they do repair them on a flat fee exchange basis. They sell the following CC40 cartridges new: 16K expansion RAM ($40), Memo Processor and other software cartridges listed above ($20 each). They also have new "user guides" for those who have the computer but no book and a "Learn Basic" book published by McGraw Hill specifically for the CC40.

• Jim Lesher, 722 Huntley, Dallas TX 75214; phone 214-821-9274. A nice selection of used CC40s ($50 for a 6K CC40), HexBus peripherals, and rare documentation. Write or call for a current product list. He is the only source I know of for 8K battery backed RAM cartridges. Jim also sells software cartridges and 16K expansion RAM cartridges and has the two books mentioned above.

• L.L. Conner Enterprise, 1521 Ferry St. Lafayette IN 4790;. phone 317-742-8146, fax 317-423-4879. A source of used and occasionally new CC40 computers, HexBus peripherals, and cartridge software. Phone almost anytime for a list of what is currently in stock. Larry Conner will upgrade CC40s from 6K to 18K of internal RAM or sell you the chips to do it yourself. He will also make the serial cable to hook a HexBus RS232 to the 99/4A RS232.

• Texas Instrument; phone 800-TI-CARES and have your credit card ready to order the PCIF, which is considered by TI-CARES representatives to be a TI74 or TI95 product. TI is the only source I know for this peripheral. It is part number 1065751-0001 and costs $60, plus shipping and state sales tax. TI also sells an AC adapter you can use instead of batteries to power the CC40 and some of its peripherals. This is called the AC9201, part number 1055601-8900, and costs $18.95. TI now refers all inquiries about sales and repair of CC40 products and HexBus peripherals to Cecure Electronics.

• Charles Good, P.O. Box 647, Venedocia OH 55894; phone 419-667-3131. That's me, the author of this article. I will send you what I have in the way of CC40 BASIC software as described earlier in the article.

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.

SEPTEMBER

10th International TI-Meeting, Sept. 22-24, Wohlfahrtsgebäude der Wiener E-Werke (Welfare Building of the Vienna Electricity Board), Wachaustr. 28, A-1020 Vienna, Austria. For information write Kurt Radowisch, TI- and Geneve User Group Vienna, Fugbachgasse 18/17, A-1020 Vienna, Austria.

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.

10th European meet slated for Austria

The TI- and Geneve User Group Vienna is organizing the 10th International TI- and Geneve User Meeting in Vienna Sept. 22-24.

This is the first meeting of this kind to be held in Austria. Site of the event is the Wohlfahrtsgebäude der Wiener E-Werke (Welfare Building of the Vienna Electricity Board), Wachaustr. 28, A-1020 Vienna.

An entrance fee of ATS 100 will be charged. Overnight lodging including breakfast is available at the site. Accommodations with two are three beds are available for ATS 250 per person. As only 27 beds are available, attendees are advised to make reservations early. Kurt Radowisch, facilitator for the event, says he can send exact descriptions of the arrival route and maps locating the event to anyone needing them. He also says he would appreciate a short note from TI and Geneve users, even those unable to attend.

To make a reservation or for further information, contact Kurt Radowisch, TI- and Geneve User Group Vienna, Fugbachgasse 18/17, A-1020 Vienna, Austria. Phone 266 95 84 (private), 76 56 25 638 (office); fax 76 56 25 642.

Correction to The Art of Assembly

The following source code was left out of the Sidebar 41 from November's Art of Assembly column titled "It's about time." These lines precede the lines that started on page 9. The entire listing was intact on the November MICROpendium disk.

```
*  SIDEBAR 41
*  "IT'S ABOUT TIME"
*
* FIRST, TODAY'S SOURCE CODE
*
* TIME DELAY INTERRUPT (TIMER/S)
* FOR USE WITH XB
* PUBLIC DOMAIN
* by Bruce Harrison and Harry Wilhelm
*
* REQUIRED EQUATES
*
XMLLNK EQU  >2018        XML LINK VECTOR
NUMREF EQU  >200C        NUMERIC REFERENCE
GPLWS  EQU  >83E0        GPL WORKSPACE
FMUL   EQU  >0E88        FLOATING POINT MULTIPLY
FAC    EQU  >834A        FLOATING POINT ACCUMULATOR
ARG    EQU  >835C        ARGUMENT
GSTAT  EQU  >837C        GPL STATUS BYTE
*
* BEGIN CODE SECTION
*
       DEF  SETTIM,ACT       DEFINE ENTRY POINTS
*
* ACT ACTIVATES (STARTS) THE TIMEOUT COUNT
*
ACT    MOV  @CHKON,@>83C4 PLACE ADDRESS OF INTERRUPT
       CLR  @CUMNUM      CLEAR COUNTER
       CLR  @GSTAT       CLEAR GPL STATUS
       RT               RETURN
*
* SETTIM SETS THE LIMITING TIME FOR THE USER
* THROUGH A CALL LINK FROM XB
* ALLOWED TIME IS GIVEN IN SECONDS
*
SETTIM LWPI WS           USE OUR WORKSPACE
       CLR  R0          CLEAR R0 - NOT AN ARRAY
       LI   R1,1        FIRST PARAMETER
       CLR  @GSTAT       CLEAR GPL STATUS BYTE
       BLWP @NUMREF      GET PARAMETER
       LI   R9,SIXTY     POINT AT F.P. NUMBER 60
       LI   R10,ARG      AND AT ARGUMENT
       LI   R4,8         EIGHT BYTES TO MOVE
MOV1   MOVB *R9+,*R10+   MOVE A BYTE
       DEC  R4           DEC COUNT IN R4
       JNE  MOV1         IF NOT ZERO, REPEAT
       CLR  @GSTAT       CLEAR GPL STATUS
       BLWP @XMLLNK      USE XML LINKAGE
       DATA FMUL         MULTIPLY SECONDS BY SIXTY
       CLR  @GSTAT       CLEAR GPL STATUS
       BLWP @XMLLNK      USE XML LINK
       DATA >12B8        CONVERT NUMBER TO INTEGER
       MOV  @FAC,@LIMNUM MOVE TO LIMIT NUMBER
       LWPI GPLWS        LOAD GPL WORKSPACE
       B    @>6A         GO TO GPL INTERPRETER
*
* TIMER IS THE INTERRUPT ROUTINE
*
TIMER  LIMI 0            PREVENT INTERRUPTS
       CB   @>8375,@HX0C   look for signs of life at the key-
board
       JGT  TIMEU5       IF KEYS BEING PRESSED, JUMP
TIMER1 INC  @CUMNUM       INCREMENT TIME COUNT
       C    @CUMNUM,@LIMNUM  COMPARE TO LIMIT
       JGT  TIMEUP       IF GREATER, JUMP AHEAD (TIME EXPIRED)
```

Tone dialing through a decadic service

Doing it the TI way

By GEOFF WARNER

Warner belongs to the TI Users of Perth (Australia) user group. We found the article in the TIsHUG News Digest.—Ed.

I had occasion to access a remote computer system from my home recently and started to think of ways to enable me to go the "TI way" rather than use the PC laptop that I cart about with me for employment-related purposes — and the odd demonstration at TIUP meetings.

The problem was the remoteness of the system and my unwillingness to go through the hassle of making the call at great personal expense and then recouping the cost through the "system." I have been issued a Telecon Telecard for occasions such as these, but I don't have tone dialing connected at my house. I was tired of the impersonal interface with a faceless voice after what seems to be an inordinately long wait for an operator. I hated holding a "tone commander" up to the telephone handset and entering the appropriate codes that way. I wanted to automate everything. But above all, I wanted to prove that it could be done on the TI!

EXPLANATION FOR OUR FRIENDS IN THE USA

The Australian Telecard system requires you to dial an access code to get into the system, whereupon you wait for instructions, voice prompts to use the correct jargon. You then dial your Telecard number, followed by your Personal Identification Number (PIN) and wait further instructions. Then you dial the number you wish to reach, preceded by the area code and followed by a "#" to indicate to the system that you are ready to connect.

This is all very well, if you are connected to tone dialing. Unfortunately, it doesn't work very well if you use decadic, or pulse, dialing at your abode or place of work. If this is the case, you need to wait and talk to an operator after dialing the first four-digit access code.

The answer eventually came to me while reading through the documentation for Telco — I could use the macro facility of Telco to access the Hayes-compatible modem that comes as a set with the aforementioned laptop in my setup procedure.

What I would do is assign a macro for the access code to the Telecard system and a semicolon at the end of the string to prevent the modem from going online when the call is answered by the remote Telecard computer.

The possibilities are limitless, and you can use your TI to achieve what your mates with more expensive computers can do at a fraction of the cost.

Then I'd assign a macro for the Telecard number and PIN number — and here is the secret to it all — prefix the string with "ATDT," the command that tells the modem to tone dial the numbers that follow. We must also not forget to finish with another semicolon.

Now assign a macro for each of the numbers you wish to dial, not forgetting the preceding ATDT and the trailing number sign to tell the Telecard computer system to get on with it.

In order to operate in this way, you must first have a Hayes-compatible modem with a built-in speaker and a terminal program that supports macros. All of my tests so far have been with Telco. Of course, you can enter each of the strings that I have assigned to macros individually while in command mode, but we have computers to make our lives easier, not harder. Besides, sometimes you may type too slowly for the system and end up with the network operator trying to have a conversation with your modem. I know that I have.

Here is the operating procedure:

Load Telco or your preferred telecommunications program.

Enter terminal mode.

Execute the macro to dial the Telecard service.

Listen for the response in your modem's speaker.

When prompted for your Telecard number and PIN number, execute the macro to tone dial your Telecard and PIN numbers.

Listen for the response from the modem.

When prompted for the number you wish to call, execute the macro for the number of the service you wish to connect to.

Log on and enjoy your session.

Obviously, there are other uses for this method, and one that immediately springs to mind is the use of your computer to autodial numbers for you, and then enter PINs and account numbers and the like.

The possibilities are limitless, and you can use your TI to achieve what your mates with more expensive computers can do at a fraction of the cost.

NEWSBYTES

Categorized TIPS files available on disk

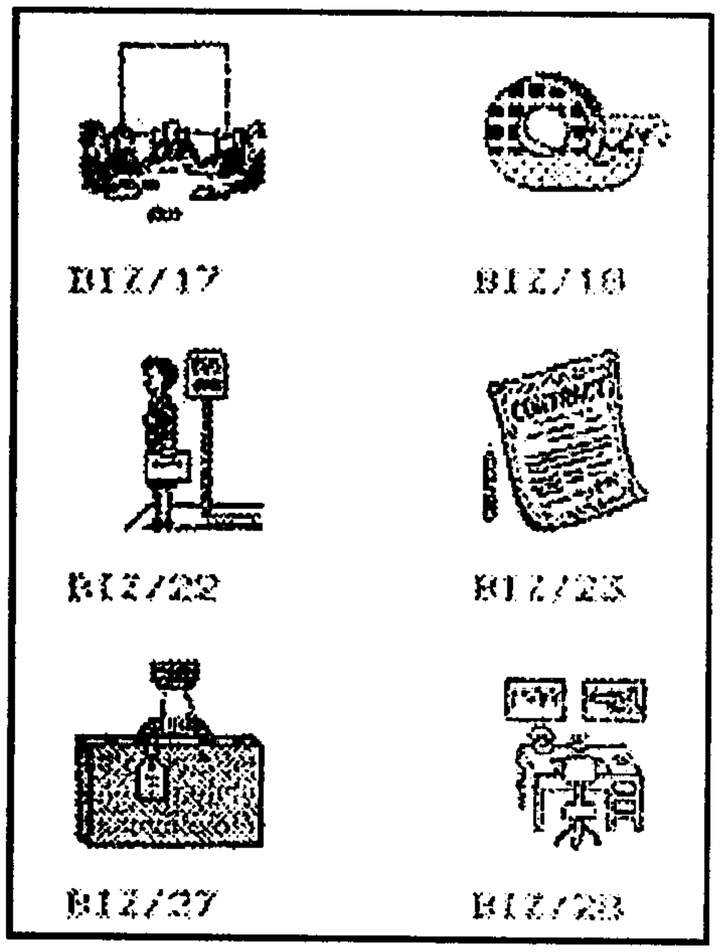
More than 6,000 TIPS images are available on a 52-disk set complete with index.

Raymond Frantz of Phoenix, Arizona, says he spent six weeks separating and recording the images onto disk with use of Patrick Powell's TIPS Manipulator. He says the disks include all images known to him, after checking with Earl Raguse, who added some images.

Disks are categorized according to topic. Images of cats and dogs, for instance, support two disks, while religious images are together on a single disk. Duplicates have been eliminated and images have been renamed to conform to a system. Files are numbered consecutively so that users have a numbered filing system they can use.

Frantz offers the entire 52-disk set (with index and four-column printer) in either single or doublesided format, for $49.95 including shipping. He suggests that each users group get its own set and allow its membership to copy it for a slight fee.

Checks should be made out to Raymond Frantz, 502 N. 51st. St. #2, Phoenix, AZ 85008.



CD-ROM project fizzles

Fred Moore of the LA Computer Group has announced that his effort to put all TI99/4A program from all users groups on a CD-ROM has ended.

"I could not get the other users groups to send me their disks although I would do all of the work and return back their disks," he states.

However, he notes that in the procedure he has archived the complete LA99 library on about 100 DSDD disks, approximately 1,000 DSSD disks of programs. Price for the complete set is $50, including tax and shipping, payable by check to Fred Moore, 7730 Emerson Ave., Los Angeles, CA 90045-1117; phone is (310) 670-4293.

Send information about your products and services to MICROpendium Newsbytes, P.O. Box 1343, Round Rock, TX 78680.

Relief from eyestrain is just a blink away

While computers save users time and money by performing an amazing array of functions, they can also cause eyestrain and other similar problems.

Persons who spend long stretches at a computer may suffer from headaches, fatigue and serious eyestrain. If your eyes feel irritated or itchy at the end of the work day, or you suffer occasional headaches, your computer could be the likely culprit.

The computer usage experts at Quill Corporation, a direct marketer of office products, recommend these tips to alleviate potential eye problems and ease tension:

• **Have your eyes examined regularly.** Tell your eye doctor how long you sit in front of the computer monitor each day. Since tinted eyeglasses sometimes help eyestrain, let your doctor know what color screen and typeface you work with so that the correct color lenses can be prescribed.

• **Keep glare to a minimum and use a glare screen.** Both outside and inside sources can cause glare on your screen. Glare screens are available that offer maximum glare reduction while still maintaining the brightness of the monitor screen.

• **Don't position your screen in front of a window.** The contrast caused by trying to focus between outside light and your terminal could cause tension headaches.

• **Change your field of vision periodically.** Gazing at a distant object forces your eyes to readjust, while maintaining a short, fixed focus too long causes stiffness and tension.

• **Take frequent breaks.** A good rule of thumb is to spend 15 minutes away from the computer terminal for every two hours in front of it.

• **Exercise your eyes.** Slowly roll them clockwise three or four times, then reverse the direction.

• **Blink frequently.** Blinking moistens the eyeballs and minimizes itching sensations caused by tired eyes. Also lower the pressure on your eyeballs by cupping your hands and place them over your eyes for one minute every half hour.

XB program simplifies creation of arc file catalogs

By W. LEONARD TAFFS

Most catalog programs provide the option of saving directories in D/V80 format. This makes them mergeable with a program editor if you wish to create an extended file of many directories. These merged directories usually do not append the name of the disk where each file is found, so such a directory is only useful viewed disk by disk. It means looking through each disk directory, one by one, until you find your file or program. This merged directory would be much more useful if it was sorted. However, such an alphabetical version would not be of much use if it did not give the name of the disk on which the file could be found.

There are some programs that will do this, such as the early Disklister, though I haven't seen many. Masterdisk is the one I have used, which is not to be confused with Diskmaster. Masterdisk is a good program but it is slow in executing — especially as its file grows in size. Manipulating files made by Masterdisk is difficult in some key respects and these files, in my experience, can be lost if one inadvertently attempts to load a disk with a blown directory or any embedded file glitch is present in a directory. Also, any files "deleted" by you in Masterdisk are not truly deleted. This has to be considered when viewing file records of a spoiled Masterdisk directory with a sector editor.

Wishing for something more flexible than Masterdisk, I wrote my own cataloging program some years ago (DSCNTODSCN) which appends disknames to filenames. I continue to use it today. Using Masterdisk has the advantage of being a single step process to produce an extended directory, while using a program such as DSCNTODSCN requires three steps:

1. Make individual directories, in my case using DSCNTODSCN.
2. Load these (merge) into a program editor, not a text editor.
3. Sort the composite file.

This is not a complicated process and is worth the time, as it is far more flexible than Masterdisk. Incidentally, with regard to step 2, if you have a large number of archived file catalogs, it might be quicker to use my MASSREAD2 program to merge these files, as compared to loading them one by one into an editor. MASSREAD2 is much easier to use than the earlier version reprinted in MICROpendium.

CREATING A COMBINED ARCHIVER CATALOG DIRECTORY

For those who need or are obsessed with large directory catalogs, here is a program that may be of use. It works with files archived using Barry Boone's Archiver program. The more archived programs or catalogs of arc files you've got, the more useful it will be. If you don't maintain a composite file of all your arced programs, the more archived programs you collect the more difficult it becomes to find particular programs. Generally, without a composite catalog, you have to read many catalogs or have to load Archiver and then catalog each arc file until you find what you're looking for. It would be nice to have an extended directory of arc files that shows the names of the archived files and the names of the files in each archived file.

Archiver will produce a disk file of any archived file catalog if, when you select the "Catalog Arc File" option, you enter a disk number and filename designation at the "Device" prompt. This comes up when you select "Y" when prompted for a printout. The format of these catalogs is slightly different from the conventional disk catalog.

With regard to making a larger, composite file of several such catalogs, this difference is quite significant in terms of the amount of disk space used. Three lines of each archived file can be eliminated — a considerable savings of space when creating very large files. The program below deletes these lines and preserves the integrity of the last line by incorporating the number of files and catalog size in the file header. These arc file headers are preceded by an exclamation mark "flag" to separate them from filenames when the composite file is sorted.

As the D/V80 arc file catalogs created by Archiver do not append the name of the arc file beside the individual filenames — necessary if you intend to assemble several and then sort them — the program below does it for you. Curiously, arc files do not show disk sector capacity, probably only a minor inconvenience to most users.

To use the program, you must first have the arc file catalogs saved to disk using Archiver. Then run these file catalogs through the program below. The program does not create a composite file. This is accomplished by merging the files in TI-Writer or other editor. You can load as many such files as your editor or sorting program will handle. Forgive me for mentioning MASSREAD2 again, but MASSREAD2 could make this work much easier by creating this composite arc catalog for you. If using MASSREAD2, just be careful not to make your composite too large for the program you use to do the sorting. The same precaution applies when using an editor.

The following program has been a great convenience to me, so I am glad to share it.

ARCFILTRAN

```
1 REM [ARCFILTRAN] 8-30-94 A
RCFILE CATALOG DV80 FORMAT !
022
10 DIM A$(127)!193
20 CALL CLEAR :: DISPLAY AT(
3,1):"ARCHIVER CATALOG TRANS
LATOR": :"By W.Leonard Taffs
, SW99ers": :" 8-30-90" !025
30 DISPLAY AT(12,1):"This pr
ogram will create new Arc Fi
les in D/V 80 format":"From
Existing Arc Files and":"wil
l append Arc File Catalog" !
055
35 DISPLAY AT(16,6):"name to
 each file" :: DISPLAY AT(23
,1):" Press <ENTER> to Conti
nue" !036
40 CALL KEY(0,K,S):: IF S<1
```

(See Page 26)

ARCFILTRANS—

(Continued from Page 25)

```
THEN 40 !110
50 CALL CLEAR :: INPUT "Arc
File Name? ":RCF$ :: PRINT :
: INPUT "Read from Dsk#: ":D
SC2$ :: PRINT :: RCF$="DSK"&
DSC2$&"."&RCF$ !120
60 INPUT "SAVE as: ":FN$ ::
PRINT :: INPUT "SAVE to Dsk#
: ":DSC$ :: PRINT :: FN$="DS
K"&DSC$&"."&FN$ !141
70 PRINT :: INPUT "OK? (Y/N)
 ":OK$ :: IF OK$<>"Y" AND OK
$<>"y" THEN 50 :: CALL CLEAR
 !207
80 OPEN #1:RCF$,INPUT !143
90 FOR A=1 TO 127 !155
100 IF EOF(1)THEN 200 !246
110 LINPUT #1:A$(A):: IC=IC+
1 !124
120 CK$=RPT$(" ",20):: IF SE
G$(A$(A),1,20)=CK$ THEN 110
!198
130 P=POS(A$(A),"ArcFile:",1
)!102
140 IF P THEN B$=" ! ArcFile
: "&SEG$(A$(A),P+9,10):: BB$
=SEG$(A$(A),P+9,10):: G$=BB$
 :: IF P THEN P=0 !196
150 IF POS(A$(A),"Files",1)T
HEN C=VAL(SEG$(A$(A),17,3))!
224
160 IF POS(A$(A),"Size",1)TH
EN D=VAL(SEG$(A$(A),35,4))!1
37
170 PRINT A$(A):: IF C THEN
B$=B$&" F "&STR$(C)&" sz "&S
TR$(D)!113
180 IF C THEN OPEN #2:FN$,OU
TPUT :: PRINT #2:B$ :: GOTO
200 !239
190 NEXT A !215
200 REM ** READ ARRAY AND MA
KE FILE !090
210 FOR I=1 TO IC-3 !136
220 IF I=1 THEN 270 !013
230 IF A$(I)="" THEN 270 !17
2
240 SV$=SEG$(A$(I),1,36)&G$
!161
250 PRINT #2:SV$ !022
260 SV=SV+1 :: PRINT "Saving
:";SV:SV$ !056
270 NEXT I !223
280 REM ** DONE ** CLOSE FIL
ES !241
290 CLOSE #2 :: CLOSE #1 !17
7
300 PRINT :RCF$: :" ArcFile
has been redone" !002
310 PRINT :"Saved as: ";FN$
!207
320 END !139
```

CYA

A utility for every Geneve users toolkit

By CAL ZANELLA

REVIEW

CYA is a Geneve MDOS-native utility program. The purpose of CYA is to aid the Geneve user to simplify system configuration by supplying a useful tool to individually tailor SYSTEM/SYS to his needs without the use of a sector editor. For many Geneve users, the thought of having to sector-edit the the SYSTEM/SYS file can be an exercise in frustration or confusion. CYA greatly simplifies this task!

CYA is a copyrighted program written by Tim Tesch. It is basically a menu system that is loaded into memory from the MDOS CLI (command line interface) prompt. Once loaded into memory, the CYA menu system allows you to perform a host of tasks that would otherwise require the use of a text editor, a sector editor and much time and forethought. CYA takes care of all this through a n elegant menu system that requires minimal keyboard input from the user.

Take note that CYA will only work with MDOS versions 2.20 and 2.21. Future versions of MDOS may or may not require a CYA upgrade, depending on code changes within MDOS. Future upgrades to CYA, if necessary, will be made available at a nominal cost.

This program can be described as allowing the user to embed most AUTOEXEC commands, as well as some hardware configuration parameters, directly into the SYSTEM/SYS file. What follows is a quick rundown on what and how it is accomplished.

The first task is to load CYA into memory. CYA requires 200K of free memory, so if you just have a basic Geneve with no additional memory cards you will need to disable TI MODE in your AUTOEXEC configuration. CYA can be loaded from a RAMdisk, hard drive, floppy drive or a PFM Flashdisk. The documentation that is shipped with the CYA package is quite descriptive, so I intend only to cover utilitarian advantages of the the program.

Once loaded into memory, the CYA main menu appears. The menu appears as:

(B)atch commands
(D)rive remap/assign
(H)FDC Information
(T)ests
(C)alculate CRC
(S)ave SYSTEM/SYS
(L)oad SYSTEM/SYS
(R)eset MDOS configs
(M)emory

Each menu item can be accessed by keying in the the corresponding letter enclosed in parentheses. Each letter will display a submenu of additional selections that are specific to the initial selection. Menus are either hierarchical or terminate at an input prompt.

The necessary first action involved loading a copy of SYSTEM/SYS into a buffer area set aside by CYA, hence the need for 200K of free memory. SYS

(See Page 27)

CYA—

(Continued from Page 26)

TEM/SYS can be loaded from virtually any external device. You can load in the file along with CYA at the MDOS prompt or you can load CYA first and select the letter (L) at the CYA main menu, type the path and filename at the submenu prompt, press Enter and SYSTEM/SYS will load in after the CRC integrity check is successfully completed. If the version of MDOS that you are attempting to load has been sector-edited previously or is corrupted, CYA will issue a warning and will not load the file into memory. "If SYSTEM/SYS has been modified or corrupted, intentionally or otherwise, CYA will NOT be able to use it" according to CYA documentation). The reason for this lies in the fact that all future releases of MDOS will have the CRC code (Cyclic Redundancy Check) embedded in the MDOS code to help identify the receipt of a possibly corrupted version.

Once MDOS is successfully loaded, all menu options are available. Since MDOS is by default hard-coded for the most basic system configuration, this is where CYA begins to shine. Browsing through the menu system will give you an idea of how flexible and useful CYA can be. AUTOEXEC-driven batch commands can be modified and patched back into the SYSTEM/SYS file. These include the prompt characters, file search path designations, upper/lowercase sensitivity and foreground/background default colors. The DSK1 subdirectory emulation default can be turned on or off. The default location of the AUTOEXEC file can be specified. Other submenus allow you to change physical and logical drive ASSIGNments, drive REMAPping, LASTDRIVE designation, default drive letter and provision to set hard drive headstep rates.

There are four menus that provide system information or perform tests on external hardware devices. If you are using a Myarc HFDC card, the (H)FDC submenu provides information on the card configuration, i.e. CRU address, EPROM version number, DIP switch settings for floppy drives and HFDC buffer memory capacity.

The (T)ests menu will give you access to hardware test routines for your joysticks, bus mouse port and sound generators.

A (M)emory menu lists the memory configuration of your Geneve. Slow and fast CPU memory are displayed as well as video memory configuration. If you have a PFM Flashdisk installed, this will also be displayed.

If you happen to change several settings during the current session and decide you would like to cancel the changes to start over with a clean slate, simply return to the main menu and select (R)eset MDOS configs. This will reset all configurations to the state that they were when you initially loaded MDOS.

When you finish with all configurations that you intend to make, the final step is to (S)ave SYSTEM/SYS. You have the option of saving to any external device. When you do a save, a new CRC value is calculated and saved back into MDOS. This will allow you to reload this modified version back into CYA should you decide to make any further changes.

At this point, you may want to edit your AUTOEXEC file to remove any configuration commands that were embedded into MDOS. You should find, as I did, that you will probably end up with a very minimal AUTOEXEC file.

I expended considerable effort trying to crash the program, only to be defeated at every turn! This is a well done and most useful utility that complements my Geneve toolkit. Now that I have it, I wouldn't part with it. Documentation that is shipped with the disk is provided on laser printed paper and is clear and complete. Even a novice should have no problem following the concepts presented.

Two final notes:

1. Tim Tesch is adding yet another configuration setup to CYA to allow setting the floppy drive head step rates. This is useful for those who are using a Myarc HFDC as their only controller card, since SETDSK will not work with a HFDC.

2. Tim is also squashing a minor cosmetic bug in the routine that reports the head step rate of floppies connected to a Myarc HFDC.

CYA sells for $15. For information or to order contact: Cecure Electronics Inc., P.O. Box 132, Muskego, WI 53150-0132; phone: 414-679-4343.

MICRO-REVIEWS

Disk of Medieval Times and Risk

By CHARLES GOOD

Starting with this month's column I will list all appropriate names and addresses at the end of the column under the heading "Access." This is where you look to find out where to send for the software reviewed here.

DISK OF MEDIEVAL TIMES
By Ken Gilliland

This $15 commercial software is the latest of Ken's theme disk sets. This one is about the middle ages and is full of knights in armor, castles, dragons, jousts, and fair maidens locked up in towers. I just finished reading Arthur Conan Doyle's "The White Company" and this disk set reminds me of the events in Doyle's story. (Yes,

(See Page 28)

MICROREVIEWS—

(Continued from Page 27)

Doyle wrote about other things besides Sherlock Holmes.)

Medieval Times comes on four SSSD disks and is designed to run out of DSK1 using Extended BASIC. You also get a nicely illustrated, 16-page user guide. The main menu is displayed from the first disk as follows:

A — Tales of Chivalry
B — Legends of Valor (Game)
C — History of the helmet
D — Castles, Keeps, & Towers
E — Exit Medieval Times.

When you select these items, you are sometimes prompted to insert another of the four disks into DSK1 in order to continue.

A — Tales of Chivalry displays text files which include legends relating to King Arthur and his knights as well as scholarly discussions concerning the history of these stories. Separate stories about King Arthur, Sir Bors, Sir Gareth, Sir Gawain, Sir Percival, Sir Lancelot, and Sir Erec are included. My favorite texts are the two dealing with the known history of these tales, answering questions such as what is the oldest known reference to King Arthur. As you probably know, the Arthur stories are, in all probability, a mixture of ancient fiction and fact. Very little independent information about Arthur and his knights exists. I was particularly intrigued with the "Holy Grail" legend. Supposedly, the cup Christ used in the last supper was brought to England where it exhibited various magical powers. Lots of time, effort, and money was expended by medieval English important people looking the grail. The source materials Ken used in researching these text files and in creating his medieval artwork are listed in the documentation that accompanies Medieval Times.

B — The Legends of Valor game is actually three games tied loosely together with a story. The queen is locked up in a tower and you have to rescue her. However, first you have to prove you are worthy of such a rescue attempt by competing in various medieval tests of courage. The three games are supposed to be played in order, and you have to win each in order to go on and play the next game. All three games are written in Extended BASIC.

The first test of valor is a joust on horseback against a great big knight. You position your shield and lance high/middle/low and try to knock the other knight off his horse. This requires several tries, with you or the other knight getting wounded each time. When somebody is wounded enough that he can't continue the game ends and, if you are the least wounded, you then heal quickly and go on to the next game. There really isn't any skill involved here, only luck. There is about a 50 percent chance of winning. The joust is slightly modified from a joust game originally published in *99er Magazine* in the ancient medieval past of the 99/4A, back in 1983.

The second test of valor is similar to the joust except that your opponent is a fire-breathing dragon. You adjust your lance and shield position and try to skewer the dragon with your lance as you pass on your horse. The hope is that your shield prevents the dragon's fiery breath from burning you. As with the first test, there doesn't seem to be much skill involved. Chances of slaying the dragon are about 50 percent.

The third test of valor really is original and fun. You have to approach the castle in which the queen has been locked, climb the tower, and rescue her. All along the way you have to dodge the arrows and swords of the defenders. Some of these defenders pop out of the ground at sudden and irregular intervals to do you in. Screen graphics are very well done. You get two simultaneous displays. Most of the screen shows you (as a knight) and your immediate surroundings. This display constantly changes as you move from left to right towards the queen's location. In the lower left of the screen you see a map of the whole castle area, with your location on this map indicated. The closer you are to the queen's location the faster the action. There is a fair amount of skill in this game since you need to know just when to jump or dodge arrows.

My main complaint about these tests of valor is that you are supposed to complete them in order and you only get one chance (one life) to complete all three games. Once killed you go to a high score screen where your chosen "Sir _____" name and score are permanently recorded and are then returned to the Medieval Times main menu. If you want to attempt another queen rescue, you have to start again from the beginning of the first test. What is needed is a "save game" feature. The third test of valor is really neat, but you can waste a lot of time trying to get to this third test and if you are killed you have to start all over again with the first test. Since Medieval Times is largely written in Extended BASIC I figured out a way of cheating so that I can start directly with the third test at any time. Each of the valor tests is a stand alone XB program. You can OLD and RUN the third test directly from XB's command mode.

C — History of the Helmet shows you a series of TI-Artist pictures, each of which shows several types of medieval helmets. A short description of each helmet accompanies each picture.

D — This gives you a slide show of several castles.

E — From the main menu will return you to reality and display the 99/4A title screen.

Two of the four disks are completely full of Ken Gilliland's own artwork related to medieval times. The disks are full of TI-Artist pictures, instances and fonts. You really get a lot of artwork for your money because some of the art is archived. What if you don't own TI-Artist? If you don't want to purchase TIA from a dealer, you can use Bruce Harrison's Drawing Program which is public domain and which I will send you if you mail me a buck. TIA instances and fonts can be loaded into and displayed by Drawing Program.

Ken's theme disks are totally 99/4A and Geneve creations. The artwork and software were all created on a 99/4A and the documentation was created using TPA for MDOS. I know of nothing in the IBM-PC world that resembles the unique combination of scholarship and original artwork found in the theme disk sets. Probably anyone can learn from and enjoy Disk of Medieval Times.

(See Page 29)

USER NOTES

Getting more lines out of the formatter

The following article originally appeared in newsletter of the Cedar Valley 99ers User Group. It was written by Bob Heiderstadt.—Ed.

Many will remember the article subtitled "Formatter Frustration" that appeared in many TI99/4A newsletters on how to increase the number of lines printed on a page. The article how to increase the number from 58 to 66 using the TI-Writer formatter. The article was written by John Owens of the JUG 99'ers.

His method does the job, but requires eight steps to print a page consisting of up to 66 lines, the number of lines the formatter is set for. Print with one-eighth inch spacing, 88 lines, and you get the same 66 lines with a space left over at the bottom.

While producing three genealogy books, I thought that a lot of paper was wasted while using the standard TI-Writer formatter. John's method worked, but it

(See Page 30)

READER TO READER

❑ Laurence Topliffe, 2A Jeanette Dr, DeLand, FL 32720, writes:

Can someone please tell me, out of all the checkbook balancing software for the TI, which one is the best — not simplest, but can handle checking account withdrawals, overdrafts and charges and check charges? I function as an independent salesman.

❑ Jerome A. Noel, 209 Martins Creek Rd., Bernardsville, NC 28709, writes:

Is there a way to size the stack memory within a program, e.g., one that loops through an array and size until the size indicates a predetermined value? I have tried with little success. The Editor/Assembler Manual, page 410, refers to this problem; however, this is not in an assembly program but an Extended BASIC program.

❑ Alfred Slovak, Fugbachgasse 18/17, A-1020, Vienna, Austria, writes:

Since I use MDOS 2.00 I have trouble using Archiver 3.03g and Birdwell's Disk Utilities 4.2.

Archiver won't let me work from DSK2 to DSK1. The opposite direction works fine. Also, doing all operations on DSK2 works. But using DSK1 as target, archiving or extracting, causes a red screen and Error Code 3. Sometimes extracting works, I don't know why.

DSKU 4.2 does not allow any more the deleting of files. The program displays "Deleting File:FILENAME" but there is no increase of free sectors, and when I catalog again the files are still present. Furthermore, the option View File does not work. It stops with Error , press any key, without even having tried a disk access.

These misfunctions only occur with MDOS 2.00. Earlier versions (1.,53) work fine.

If you or one of the readers can help, I will be grateful

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.

MICROREVIEWS—

(Continued from Page 28)

RISK
By Oliver Arnold

Risk is a Parker Brothers board game I enjoyed playing with my friends when I was a young teenager. The object of the game is to conquer the world with your armies, physically occupying all countries of the world displayed on board game's world map. There is at least one "official" Risk computer version and I have seen a couple of fairware Risk versions for IBM-PCs in CD-ROM collections. Now we 99/4A owners have a nice adaptation of Risk we can use on our machines.

The assembly language (EA5) game is for two to six players. It is a game of strategy, not fast action. A joystick is required. The gameboard displays an amazingly detailed, multi-colored mercator projection of the world. The game begins with players entering their names in the upper right corner of the screen. The computer randomly divides the countries of the world among the various players and then allows each player to place his surplus armies (each country already has one army) in the countries under his control. Players attack countries next to the ones already under the player's control, with a roll of the computer dice deciding the outcome of each battle.

Player names, country names, and computer prompts for input are all displayed in teeny-tiny, multi-colored, 64-column text. Each letter is only three pixels wide, with one pixel between letters. This is hard to read on a TV screen. You had better use a monitor. Cute background music plays all the time. You can turn it off by moving the cursor next to the pictured gramophone (the "His master's voice"-type with a big horn-like speaker) and pressing the fire-button.

Risk is an excellent educational tool for learning world geography. You run into some strange country names, particularly within what is now Russia. Several autonomous areas within Russia are treated by Risk as separate countries.

I really like this game and am amazed at the detail that is simultaneously squeezes onto the screen. My only complaint is that sometimes all this detail is hard to see. Players 1 and 2 have their countries colored white and gray, colors hard to distinguish from each other. Sometimes the text is hard to read because it is so small.

Risk is fairware. Send me $1 and I'll send it to you to try out. Oliver asks whatever you think it is worth as a fairware donation.

ACCESS

Ken Gilliland (Notung Software): 7647 McGroarty St., Tujunga CA 91042.

Olivar Arnold: Hauptstrasse 44, 69517 Gorxheimertal, Germany. Internet address: oliver@thorin.swb.de.

Charles Good: P.O. Box 647, Venedocia OH 45894. Phone 419-667-3131. Internet address: cgood@lima.ohio-state.edu.

USER NOTES

(Continued from Page 29)

was cumbersome. I started experimenting. I found what I believe is an easier method to accomplish the same goal. Plus, allowing many more lines if you use less line spacing and a smaller typestyle, elite or superscript, for example.

The method is essentially a three-step process. Details follow:

1. Defeat the formatter's wasting the top five and bottom three lines of paper.

2. Print the first 58 lines, or fewer, lines of heading and text.

3. Print the remaining lines needed to fill out the page as close to the bottom as you desire.

You know that the header (.HE) prompt will print on the third line, and then skip to the sixth line. The problem with the header prompt is nothing except dot commands (.) can precede it. Add a transliterate command (.TL) and a symbol to activate the command, such as expanded type, ahead of the .HE and you lose the header. Most of the time you don't want a header. What to do?

Step 1: Make up a simple three-line "dummy" program you will use with each page you want to print using more than 58 lines. Give it a simple file name and save it on each disk for use as needed.

The dummy program is as follows:

Line 1: (CTRL-U, Shift-Q, CTRL U)
Line 2: CR (carriage return)
Line 3: (CTRL-U, Shift-S, CTRL-U)

That's it. This program, when run through the formatter, will space down six blank lines and turn off the printer motor. You then roll back the paper to where you want printing to start.

Step 2: Call up the formatter again and run the text you have prepared. Your text, including all formatter and printer commands, heading, etc. is the same as you normally would prepare when using the formatter. Add a CTRL-U, Shift-S, CTRL-U on the first line of the screen to turn the motor on again. Add a CTRL-U, Shift-S, CTRL-U to turn the motor off as the last line of text. When the text goes through the formatter, the motor turns on and starts printing your file. When done, the motor stops.

Note that if you have carriage returns at the start of the text, as mentioned above, other than after .TL commands (such as the .TL activator symbol following by a carriage return), that symbol's carriage return will insert a blank line before starting to print. So position your paper to allow for that first linefeed.

Step 3: Call up the formatter again, run Step 3 of your text, prepared the same was as in Step 2. Use the same format, i.e. start motor, add formatter and printer commands, add text, and finish with the stop motor command. This keeps the printer from running out a sheet of blank paper.

It is best, and easier, to stop the Step 2 text at the end of a paragraph where you want a following blank line.

When preparing text, I include 58 lines of text in Step 2 and 17 lines in Step 3 to add up to 75 lines, with 10/72 inch line spacing. You could do 84 lines with 1/8 inch line spacing. I like to leave two lines at the top and bottom. Experiment to find what you like best.

I use NLQ elite on my Star NX1000 printer. The commands are compatible with the Epson printer.

Making electronic conversation

We found the following item in Tic Toc, the newsletter of the Rocky Mountain 99ers.—Ed.

When you are "talking" to a friend using email, you can't see each other's faces. Because of this you miss the smiles, winks and frowns that add meaning to conversation and lend context to cold words. Longtime navigators of the information highway developed a substitute called "emoticon." Emoticons are icons reflecting emotion. They are created using combinations of colons, hyphens and other characters and are viewed sideways. Here are some emoticons to get you started:

:-)	Smile
:-(	Frown
;-)	Wink
:-D	Big smile
:-O	Open-mouthed amazement
:-Q	Tongue hanging out in nausea or disgust
:-{)	Smile (user has a moustache)
8-)	Smile (user wears glasses)
)-:	Smile (user is left-handed)

In addition to emoticons, experienced emailers use a variety of shorthand abbreviations to save time and keystrokes in expressing frequently repeated concepts. Here are a few:

BRB	Be right back
BTW	By the way
FWIW	For what it's worth
FYI	For your information
GD&R	Grinning, ducking and running (after a snide remark)
IANAL	I am not a lawyer
TIA	Thanks in advance
TTFN	Ta-ta for now
IMHO	In my humble/honest opinion
OTOH	On the other hand
PMJI	Pardon my jumping in
ROFL	Rolling on floor, laughing
GD&WVVF	Grinning, ducking & walking very, very fast
IMO	In my opinion

Computer users say the funniest things

The following item was written by Earl Raguse and appeared in the User Group of Orange County ROM newsletter. He excerpted it from a column written by Vince Demers in the New Hampshire 99ers newsletter. He in turn got the information from an article in the Wall Street Journal.—Ed.

The exasperated help-line caller said she couldn't get her new computer to turn on. A technician at the computer company asked her to make sure the computer was plugged in and then asked the woman what happened when she pushed the power button.

"I have pushed and pushed on this foot pedal and nothing happens," the woman said.

"Foot pedal?" the technician asked.

"Yes," the woman said. "This little white foot pedal with the ON switch."

The "foot pedal" turned out to be the computer's mouse.

(See Page 31)

USER NOTES CLASSIFIEDS

(Continued from Page 30)

There were many more equally stupid calls reported. A woman called to find out how to insert the batteries in her new laptop. When told that the instructions were on the first page of the manual, she said, "I just paid $2,000 for this damn thing and I am not going to read a book."

Many calls were made by people who couldn't find the "any" key. One man was insulted when the computer said "bad command" and "invalid." The technician had to tell him not to take it personally.

One person called to complain that his keyboard no longer worked, even after he had carefully cleaned it by soaking it in the bathtub over night.

Another called to say he was unable to "fax" anything using the computer's fax modem. It turned out he was holding a piece of paper in front of the monitor and hitting the print screen key.

Chicago TI Faire Vendor List

Here is a list of vendors who attended the Chicago TI Faire in November. The list wasn't included last month for space reasons.

9640 News, c/o Beery Miller, P.O. Box 752465, Memphis, TN 38175-2465, (901) 368-1169, BBS (901) 368-0112
Aaron Busch, 3629 Boswell Ave., St. Louis, MO 63114
Bud Mills Services, 166 Dartmouth Dr., Toledo OH 43614, (419) 385-5946, BBS: (419) 385-7484
CaDD Electronics, Mark Van Coppenolle, 81 Prescott Rd., Raymond, NH 03077 (603) 895-0119
Cecure Electronics Inc. c/o Don Walden, P.O. Box 222, Muskego, WI 53150 1-800-959-9640, (414) 679-4343, BBS: (414) 679-3736
Chicago TI Users Group, P.O. Box 7009, Evanston, IL 60204-7009
Competition Computer Products, 2219 S. Muskego Ave., Milwaukee, WIS. 53215 (414) 672-1600 or 1-800-471-1600
Computers & Crafts, Kevin Keller, 1418 Biscayne Dr., Little Chute, WI 54140, (414) 788-6656
Hoosier TI User Group, P.O. Box 2222, Indianapolis, IN 46206-2222
L.L. Conner Enterprise, 1521 Ferry St., Lafayette, IN 47901 (317) 742- 8146
Midsouth (Memphis) TI99/4A User Group, P.O. Box 38522. Germantown, TN 38183-0522
Milwaukee TI Users Group, 4122 N. Glenway, Wauwatosa, WI 53222
Program Innovators - Arcade Action Software, 4122 Glenway, Wauwatosa, WI 53222
RAMcharged Computers, 6467 E. Vancy Dr. Brook Park, OH 1-800-669-1214 or (216) 243-1244
RBD Enterprises, c/o Ricky Bottoms, 643 Fair Ave., Shelbyville, IN 46176 1-800-464-8851
S&T Software (Tim Tesch), 3804 North 75th St., Milwaukee, WI 53216
Western Horizon Technologies, 3297 Woody Lane, San Jose, CA 95132 (408) 934-0352
Will County TI Users Group, 1400 Caton Ave., Joliet, IL 60435

FOR SALE

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

Hardware For Sale

Volksmodem 1200 baud modem (needs TI cable), $20; Signalman Mark XII 1200 baud modem (incl. TI cable, no docs), $20; Signalman III-TI 300 baud modem (incl. TI cable, no docs), $10; TI RS232 interface (no docs), $35; Power Supervisor, fits under monitor, includes switches for 5 devices plus master switch, $25; Commodore 1702 color composite monitor, incl. cables for TI99/4A (measures 13" diagonally), $100; 5.25" Mitsumi 360K floppy drive (new), no docs or cables, $25. All prices OBO. Will split shipping cost. Call 512-255-1512.

WANTED TO BUY

WANTED

Manual and software for Mini-Memory, P-code card (only have the hardware). Cerebral Software, Inc. 708/392-3860.

The ONLY monthly devoted to the TI99/4A

Subscription Fees

❑ 12 issues, USA, **$35**

❑ 12 issues, Mexico, **$40.25**

❑ 12 issues, Canada **$42.50**

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

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

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

Address Changes

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

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

Check/MO ❑ MasterCard ❑ VISA ❑ (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. Credit card orders add 5%.

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 versions(s); includes SETCOLR, SHOWCOLOR, FIND, XUTILS, REMIND $4.00

GENEVE PUBLIC DOMAIN DISKS

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

	SSSD	DSSD	DSDD
❑ **Series 1**	$9.00	$7.00	$5.00
❑ **Series 2**	$9.00	$7.00	$5.00
❑ **Series 3**	$9.00	$7.00	$5.00
❑ **Series 4**	$9.00	$7.00	$5.00
❑ **Series 5**	$9.00	$7.00	$5.00
❑ **Series 6**	$9.00	$7.00	$5.00

SECOND CLASS

T 1/95
ALLEN GOOD
. BOX 647
EDDICTA OH 45874