Covering the TI99/4A and the Myarc 9640

MICROpendium

Volume 11 Number 2 March 1994 $3.50

Fest West Better than ever!

XBASIC programs

MASSREAD

Noteworthy

A tribute to Regena

Reviews

Who's behind the

Mexican UFOs?

DIGISYNT

Hardware projects

TI/non-TI serial connection

Hooking up disk drives

4 Quest RAMdisks in a P-Box

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.

Display advertising deadlines and rates are available upon request.

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

Foreign subscriptions are $40.25 (Mexico); $42.50 (Canada); $40.00, surface mail to other countries; $52 airmail to other countries.

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

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

TEX★COMP Earthquake Clearance Sale!

Our Northridge, California, warehouse is about five blocks from the center of the January 17 earthquake. While our building stayed up, it has taken us three weeks to pick everything up. If we survived 1983, when TI pulled the plug, we can certainly handle a 6.8 earthquake. There is no better way to see what you are overstocked on than when you have to pick up. To celebrate our survival, we have slashed prices.

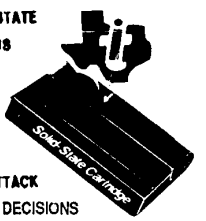
There may never be a better time to buy!

Modules 99¢ each—NO LIMIT!

Subject to availability. Effective 3-1-94

PHM 3027 ADDITION & SUBTRACTION I
PHM 3028 ADDITION & SUBTRACTION II
PHM 3093 MILLIKEN DIVISION
PHM 3051 NUMERATION II
PHM 3094 MILLIKEN INTEGERS
PHM 3025 MIND CHALLENGER
PHM 3021 WEIGHT & NUTRITION
PHM 3002 EARLY LEARNING FUN
PHM 3003 BEGINNING GRAMMAR
PHM 3112 PARSEC
PHM 3007 HOUSEHOLD BUDGET MANAGEMENT
PHM 3044 PERS. REPORT GENERATOR**
PHM 3053 TI INVADERS
PHM 3194 JAWBREAKERS II

PHM 3016 TAX INVESTMENT RECOMD KEEPING***
PHM 3022 PERSONAL REAL ESTATE
PHM 3029 HUNT THE WUMPUS
PHM 3031 THE ATTACK
PHM 3220 MICROSURGEON
PHM 3067 MUNCHMAN
PHM 3110 CHISHOLM TRAIL
PHM 3219 SUPER DEMON ATTACK
PHM 3006 HOME FINANCIAL DECISIONS
PHM 3035 TERMINAL EMULATOR II
PHM 3052 TOMBSTONE CITY
PHM 3109 LOGO II* →
PHM 3055 EDITOR ASSEMBLER →
PHM 3010 PHYSICAL FITNESS



Plus $4 additional shipping & handling

Special Purchase

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

Personal Record Keeping
PHM 3013
$4.95 (Regular $9.95)

*ADD AN ADDITIONAL $4.00 EACH ON THESE TITLES DUE TO WEIGHT
**FOR USE WITH PERSONAL RECORD KEEPING ***DISK DRIVE REQUIRED
PERSONAL RECORD KEEPING AVAILABLE AT $4.95

99¢ BOOK BLOWOUT!!! NO LIMIT!

PUBLISHER'S CLEARANCE ON BOOKS FOR THE 99/4A

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

*Add $3 for Games TIs Play disk or cassette
** Add $3 for disk or cassette for disk or cassette for Starter Pack Game Writer Series.

→ SPECIAL
Kids and the TI — $2.95
The TI in Bits and Bytes with disk — $4.95

SPECIAL PACKAGE AND PRICING ON Adventure

THE ADVENTURE SERIES...

SPECIAL PACKAGE AND PRICING ON THE ADVENTURE SERIES... ON DISK OR CASSETTE (specify)

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

NEW LOWER PRICE ONLY $4.95 FOR ALL 13

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

NEW LOWER PRICE ONLY $4.95 FOR ALL6

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

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

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

$4.95 Includes disk version of Editor/Assembler

ADVENTURE EDITOR

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

SUPER BUY!

TEX COMP is proud to announce the 99/4A Adventure Editor — an all new program which allows you to write your own adventure programs — Scott Adams move over! This disk or cassette based program allows:

1. Any existing adventure for the Adventure command module/s, to be edited, altered, listed and copied from any storage medium to any other storage medium (i.e. tape to disk or vice versa).
2. New adventure games can be created freely using a template game as a start up step. These games can be converted using the full power and capabilities of the TI Adventure command module.

The program is written for Mini Memory or Editor Assembler command modules (cassette or disk) and offers a full screen editor and a special easy to use mnemonic language called A.P.L. (Adventure Programming Language) and comes with a complete and detailed 75 page manual.

TEX COMP is also organizing a new TI Adventuring Users Group which will allow low cost exchange of user written adventures between members.

$4.95 Available NOW from TEX-COMP NEW LOW PRICE
Specify Mini Memory (Cassette) or Editor Assembler (Disk) versions

$4.95 Per Disk

SPECIAL Buy one, get one free Any disk or cassette below

EDUCATION

DISKETTE PROGRAMS

PHD 5009	Music Skills Trainer	4.95
PHD 5011	Computer Music Box	4.95
PHD 5018	Market Simulation	4.95
PHD 5030	Speak & Spell II (Ex Basic req.)	9.95
PHD 5031	Speak & Math (TE-II req.)	4.95
PHD 5042	Spell Writer (TE-II req.)	4.95
PHD 5026	Bridge Bidding I	4.95
PHD 5039	Bridge Bidding II	4.95
PHD 5041	Bridge Bidding III	4.95
SPECIAL!!	BRIDGE BIDDING I,II&III	9.95
PHD 5020	Music Maker Demo (use with module!)	4.95
PHD 6010	Mystery Melody (stop the music quiz)	4.95

CASSETTE PROGRAMS*
*see disk versions for requirements i.e. TE-II

PHT 6009	Music Skills Trainer	4.95
PHT 6010	Mystery Melody	4.95
PHT 6011	Computer Music Box	4.95
PHT 6018	Market Simulation	4.95
PHT 6031	Speak & Math	4.95
PHT 6042	Spell Writer	4.95
PHT 6026	Bridge Bidding I	4.95
PHT 6039	Bridge Bidding II	4.95
PHT 6041	Bridge Bidding III	4.95
SPECIAL!!	BRIDGE BIDDING I,II&III	9.95
PHT 6020	Music Maker Demo (use with module)	4.95

MANAGEMENT AND BUSINESS

DISKETTE PROGRAMS

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

CASSETTE PROGRAMS

PHT 6003	Personal Financial Aids	4.95
PHT 6038	Lease/Purchase Decisions	4.95

HOME ENTERTAINMENT

DISKETTE PROGRAMS

PHD 5002	TI-TREK (TE-II req. for speech)	4.95
HD 5010	Mystery Melody	4.95
PHD 5015	Oldies But Goodies I	4.95
PHD 5017	Oldies But Goodies II	4.95
****SPECIAL	Oldies But Goodies I & II	7.95
PHD 5025	Sat. Night Bingo (Ex-Basic & Speech)	4.95
PHD 5037	Draw Poker (Ex-Basic)	4.95

CASSETTE PROGRAMS

PHT 6002	TI-TREK (TE-II req. for speech)	4.95
PHT 6010	Mystery Melody	4.95
PHT 6015	Oldies But Goodies I	4.95
PHT 6017	Oldies But Goodies II	4.95
****SPECIAL	Oldies But Goodies I & II	7.95
PHT 6036	Sat. Night Bingo (Ex-Basic & Speech)	4.95
PHT 6037	Draw Poker (Ex-Basic)	4.95

COMPUTER PROGRAMMING AIDS

DISKETTE PROGRAMS

PHD 5007	Teach Yourself 99/4A Basic	4.95
PHD 5019	Teach Yourself Extended Basic	4.95
PHD 5004	Programming Aids I	4.95
PHD 5005	Programming Aids II	4.95
PHD 5077	Programming Aids I,II,III	9.95
PHD 5067	Beginning Basic Tutor	4.95
PHD 5076	Text to Speech (Ex-Basic Speech)	4.95
PHD 5098	TI Forth & manual (Ed/Assem req.)	19.95
PHD 5078	TI Forth Demo Disk (Ed/Assem)	4.95
PHD 5079	TI Forth Source Code (2 disks)	4.95

CASSETTE PROGRAMS

PHT 6006	Programming Aids I	4.95
PHT 6007	Teach Yourself 99/4A Basic	4.95
PHT 6019	Teach Yourself Extended Basic	4.95
PHT 6067	Beginning Basic Tutor	4.95

MATH AND ENGINEERING

specify disk or tape with order

Math Routine Library	4.95
Electrical Engineering Library	4.95
Graphing Package	4.95
Structural Engineering Library	4.95
AC Circuit Analysis	4.95

***SPECIAL--ALL 5 OF THE ABOVE ON DISK OR TAPE---17.95

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

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

TERMS: **Add $3 S&H to all orders.** California orders add 8.25% sales tax. Canada, Hawaii and overseas shipping extra.

Send $2 for our catalog

AUTHORIZED DEALER

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

COMMENTS

More on the SCSI

Now that the SCSI card is available from Western Horizon Technologies, I'm only a little less enthusiastic than I was several months ago when I mistakenly thought it would run with SCSI-1 devices. The word is that it works only with SCSI-2 devices. The unfortunate news is that SCSI-1 devices are in plentiful supply. They've been used for years on Macintoshes. For example, an 85-megabyte SCSI hard drive could be had used for less than $100. SCSI-2 is something new and the cost of these devices is likely to be considerably more expensive than SCSI-1, not to mention harder to find. And you definitely won't find a "small" SCSI-2 hard drive. I'm talking megabytes here, not physical dimensions. A small SCSI-2 hard drive would probably start at about 150 megabytes.

But don't get me wrong. I'm happy to see the SCSI card reach the marketplace, and I'll be even happier when the final DSRs are completed. For more information on the status of the SCSI card, see the story on page 17.

FEST WEST SUCCESSFUL AGAIN

Fest West '94 concluded successfully in February, with some 140 visitors taking it in. I've been told there were more people at Fest West than attended the Chicago TI fair last November. The word is that many vendors did well at Fest West, which augers well for future fairs. See page 5 for a report by Tom Wills.

NEW INTERNET ADDRESS

We've moved to a new Internet address. The new address is JKOLOEN@INDIAL.IO.COM. We will be phasing out our old address in April.

SHAREWARE TI EMULATOR

A shareware TI emulator that runs on 386 PCs has recently become available. The program was written by Edward Swartz of Georgetown, Texas. Minimum requirements are an 80386 system with EGA monitor, at least 256K of free RAM, and a 360K floppy drive. The shareware fee is $25. It is available from a number of bulletin boards. I've seen some promotional material on the boards that state the program will operate on a 286 PC. It doesn't.

MDOS 2.00 AVAILABLE

Also available is the long-awaited final version of MDOS. Version 2.00 includes a variety of improvements over previous versions. This version is compatible with the hard drive and the floppy only systems. The ""F" and "H" designations for floppy and hard drive versions are no longer used. In addition to bug fixes, a number of new commands have been added. There is also support for high-density Myarc HFDC floppy disks and 3.2 megabyte Horizon RAMdisks. MDOS 2.00 also includes Jim Schroeder's FORM3MEG, update notes, and information on using high-density disk drives. This final version will be mailed to Geneve contributors and then to all other Geneve users registered with 9640 News. Mailings are expected to begin in April or May. The program is 516 sectors long in archived format. MDOS 2.00 is available on GEnie and other boards and is also available from MICROpendium for $4.

—JK

BUGS & BYTES

Other TI orphans have a home

Sometimes users groups and dealers get inquiries for other TI products, such as the TI-Pro, or even about items which are not "orphaned" such as TI's laser printers.

One source to which you can refer them is TI-Biz, c/o Andmore Companies, 3207 Ashfield Dr., Houston, TX 77082-2205, 1-800-445-9980 or (713) 558-9900.

Oh, for such a problem!

The Brisbane Users Group in Australia's February 1994 newsletter says that the group has a substantial amount of money in the bank "and not much to spend it on."

The club is soliciting advice from members as to how the club can use the money to help them.

Mid-South fixes equipment

The Mid-South Users Group in Memphis, Tennessee, has started a project whereby they will be repairing computer equipment and then donating it to local schools. The group has received three complete systems.

Fest West '94

SCSI card sells like hot cakes; visitors come from all over

By TOM WILLS

Possibly, the best TI Fair of 1994 has already taken place in Tucson, Arizona. Fest West 1994, which was hosted by the SouthWest Ninety Niners User Group of Tucson, was a huge success. With 140 people attending, there was plenty to do for everyone. The fair was dedicated to the memory of Jim Peterson, who died Jan. 12.

MANY GROUPS REPRESENTED

Visitors from coast to coast were in attendance at Fest West. Included in those who were present, was Berry Harmsen of Amsterdam, Holland, and Jack Mathis, who is currently stationed in Korea. Jack planned his leave time. in such a way as to not miss this Fest.

User Groups represented included Chicago User Group, Milwaukee User Group, Ozark 99ers, Southern California Computer Group, the Rocky Mountain 99ers, the Los Angeles 99ers, the North County User Group, the Odgen TI User Group, the TI Slaves User Group, NOVA, Brea 99ers, Club 99, Pittsburgh User Group, Cin-Day User Group, Lima User Group, Pomona Valley User Group, South Bay User Group, and the Toledo User Group.

SCSI RELEASED

Several new products were shown at Fest West. The products of most interest were all from Western Horizon Technologies. They included the long awaited SCSI Disk Controller. The SCSI, pronounced SCUZZY, and standing for Small Computer System Interface, is the opening of another door through which TIers will pass, putting us into another era.

The biggest of the Fest was an event that occurred early on the first day of the two day event. It happened when Don O'Neil of Western Horizon Technologies walked into the hall pushing a luggage cart full of SCSI hard drive controllers and drives, along with his AT Keyboard and ROM upgrade for the TI-99/4A.

The scene resembled the feeding frenzy an aquarium when the sharks are fed. Don never made it to the table with his product. The crowd of people who were interested in the SCSI controller and drive, surrounded Don and were purchasing the products as fast as Don could collect the money and write up the receipts.

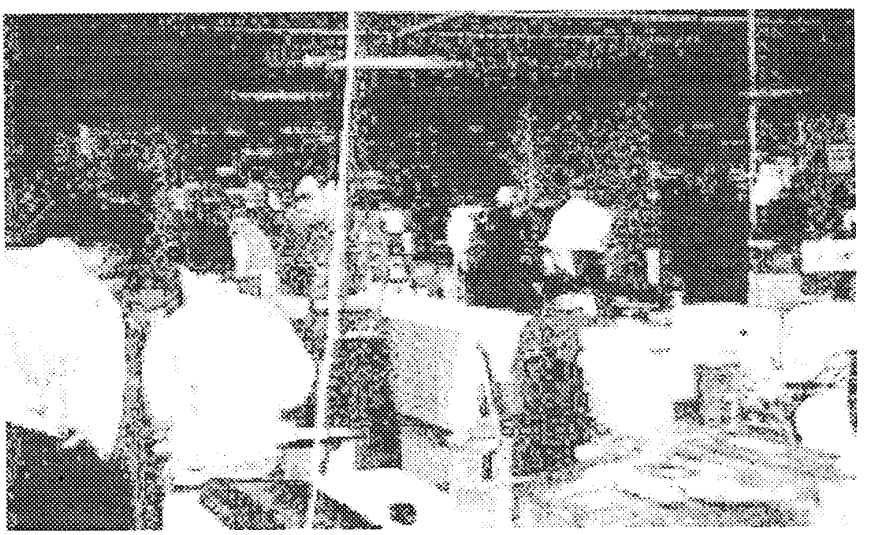
Thus it became official that the WHT SCSI controller and drives were available, and not vaporware as many has supposed. Don later said "I wish I had brought more with me!" Don stated he thought he would have problems selling the few he brought with him, not sell out in 45 minutes!

For the remainder of the Fest, Don and Bud Mills of Bud Mills Services were busy putting together additional SCSI cards for the many TIers who were demanding this revolutionary product. For those who received the controller, they had to wait for Don and his volunteers to burn more chips. Don and his volunteers reportedly were up until 4:30 AM burning more EPROMs to fill the orders. Those who also wanted the 44-megabyte hard drive direct from WHT, a special offering from WHT, would have to wait for Don to mail them out after Don returned home from the Fest. When Fest drew to a close on Feb. 20, it was one very tired Don O'Neil who went home. Tired, but still beaming from ear to ear.

But, Don O'Neil wasn't the only vendor at Fest West. Competition Computer of Milwaukee was there with the largest assortment of TI modules and accessaries this author has seen since the mid-80's. Ted Keifer as usual had some great deals for the TIer who wanted these hard to find products. In addition, Competition also had disk drives, printers, and numerous other hardware items on display.

Bud Mills Services was present with his Horizon 4000 Ram Disk, and all the other products he carries. Bud was working with Western Horizon Technologies and was kept as busy as Don O'Neil.

Ken Gilliland of Notung Software was present to display his amazing collection of TI graphics and and related products.



Some 140 visitors made for a busy weekend at Fest West '94. (Photo courtesy of Tom Wills)

Other vendors present included the Father and Son team of Don and Tom Shorock, The Ogden TI User Group, the Chicago TI User Group, the North County TI User Group who ran the consignment table, Cal Zanella, You Never Know!, Richard and Shawn Baron, and the host group, the SouthWest Ninety Niners User Group.

Mike Maksimik, of Crystal, in cooperation with Western Horizon Technologies, Cecure Electronics and 9640 News, was showing off the Digi-Port Digital Sound Player, which will be sold by WHT and Bud Mills Services. As usual, anything

(See Page 6)

FEEDBACK

Teamwork has let TI live on

I am from San Diego, California, and attended the Fest West in Tucson. I had a great time as well as everyone else I know of here in the San Diego area.

The atmosphere was very busy and the people all seemed to be having a good time. As usual, there were many very good people there. The debut of the SCSI card certainly helped and I know that many volunteers helped in filling the orders and even in stuffing a few boards for the new keyboard adapter. The folks from Western Horizon Technologies are all great people and I am certainly glad to have met them and had a chance to talk with them. All the hard work and teamwork that was evident at the Fest has certainly let the TI live on, and I hope that it continues for many years to come.

Daniel Leavy
San Diego, California

Assembly basics

Please encourage Bruce Harrison to go back to basic things often — one column on loading Assembler and others would even help, and what commands to enter for the file prompts, etc.

I'm having trouble with all of it.

Bill Buckeyne
Redford, Michigan

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

Cactus Patch BBS listed as 'preferred'

The Cactus Patch, official BBS of the SouthWest Ninety Niners User Group in Tucson, Arizona, has been included on the "Tucson Area Preferred (Family) BBS List," according to Tom Wills, sysop.

The Christian Connection BBS in Tucson released the listing Feb. 19.

Co-sysops for the Cactus Patch are Mike Doane and BJ Mathis.

The main paragraph for the listing reads, "This is a list of Tucson Area BBSs which, as far as we know, do not have adult areas, do not allow profanity and do not exchange pirated software or engage in other illegal or immoral activities."

According to Wills, the list contains 26 of the approximately 100 Tucson BBSs, and the Cactus Patch is the only non-IBM BBS on the list. He calls the listing, "a high honor, at least in my book."

The Cactus Patch can be reached by calling (602) 290-6277 using an 8N1 protocol. The BBS runs on the Paradigm 99 BBS software, written by Wills, Mike Kimble and Travis Watford, with assistance from Brad Snyder and Dave Deheer.

Wills says that the Paradigm 99 BBS software has been released to public domain as of Jan. 1.

He adds, "I'd really like to thank all of those who have used the Paradigm 99 BBS software for their support over the years. As far as shareware software goes, Paradigm 99 was a success, with virtually everyone who used it actually paying for it. This shows that not all TIers are skinflints. Paradigm BBS sysops are a great bunch of TIers."

FEST WEST—

(Continued from Page 5)

Mike touches turns to a quality product. The sound from this system was excellent.

MANY INTERESTING SPEAKERS

Talks were presented by Don Shorock, Don O'Neil, Bill Gaskill, Ken Gilliland, Bud Mills, Mary Phillips, Dave Deheer, and Tom Shorock. The most eagerly anticipated speaker was Don O'Neil who spoke about his just released SCSI controller and AT keyboard interface. In addition to speaking on these products, Don also spoke about the soon to be released 4A Memex Jr, a VGA 80-column card, a new PE Box, and the RXB module, all offered by Western Horizon Technologies.

Bill Gaskill, in his usual jovial manner, gave those in attendance a variety of subjects to dwell on. Bill spoke about Collecting Cartridges, various TI Base products, and TI Trivia in general. Bill had so much to say, and so many wanting to hear him, he could have had the speakers room all day and not run out of interesting subjects, or listeners!

PRIZE WINNERS

Fest West concluded with its annual drawing. The prizes and prize winners are:

Asgard Memory System — Gary Taylor

WHT SCSI card and drive — VAST User Group

WHT Keyboard— John Campbell

WHT Keyboard — Ben Lucere

Horizon 4000 RAMdisk—Dan Baker

Over/Under floppy drive — Les Neff

Myarc HFDCC and drive — Wallace Knight

Miscellaneous TI modules — Mike Wright

The other two prizes, a 2400 baud modem won by Jack Mathis and the WHT Turbo Video chip won by Wes Eng, brought some unexpected activity. Mathis already had a 2400 baud packet modem and didn't need another one, while Eng did not have any use for a Geneve video chip on his TI-99/4A. Before Jack could claim his prize, a bidding war broke out, Jack ending up selling the prize for $34. The same thing happened to Wes' prize. A most interesting and fun end to Fest West '94.

Hardware project

The TI/Non-TI Serial Hardware Connection

By BARRY TRAVER
©1994 B. Traver

Since TIers live in a world where other computers exist in addition to the TI99/4A, we sometimes find ourselves in situations where we want to get our TI to communicate via RS232 either to a computer hardware device not specifically designed to work with the TI or to an IBM (or to another non-TI) computer. This is not always easy to do. (Someone, I think, in fact referred to this task as a "serial killer"!)

This month's column deals with a simple homemade hardware device that can greatly simplify the operation and increase likelihood of success. As some of you know, I am a genuine technoklutz. Fortunately, however, I have a good friend, Allan Silversteen, who not only made such a gadget for me but also agreed to share the instructions with me for publication in this article!

Before we move on such a description, however, let me mention how I myself found it to be very useful. In our home, we have a TI-99/4A and an IBM sitting next to each other, and I have been doing my best to get the two on good speaking terms. Among other things, this means figuring out how to modify TI Extended BASIC programs so that they will run in QuickBASIC on the IBM (not too difficult, as long as sprites, speech, or multi-voice music are not involved). The very first necessary step, however, is to get a text LISTing of the TI XB program from the TI to the IBM, because you can't do any modifying of the program on the IBM until you've gotten the program to the IBM!

Here's the procedure. On the TI, load in a TI XB program that you want to port over to the IBM, and LIST it to disk. Now you need to get that text over from a TI disk to an IBM disk. There are a number of ways to do this. One is to use a program by Mike Dodd called PC-Transfer (currently available from Beery Miller of 9640 News). Bruce Harrison of Harrison Software also offers an inexpensive and useful product for bringing text from the TI to the IBM. A third way is to use a special cable to connect a TI and an IBM, RS232 to RS232, running terminal emulation programs on each (e.g., Fast-Term on the TI and ProComm on the IBM) and doing an "ASCII upload" from the TI to the IBM.

As I said, I'm a hardware technoklutz, but my good friend and hardware guru Allan Silversteen made for me a special connector that I use to connect the two normal cables that I already have hooked up to our TI and our IBM clone. Here's the way the computers are normally set up:

TI RS232>———> Modem #1 Modem #2 <———IBM RS232

And here's the way it looks when I want to send a TI XB LISTing from the TI to the IBM:

[RS232 >----------> > Specially Built Connector < <----------< IBM RS232

Note: the two computers are communicating with no need of a modem! What this means is that I can upload text from the TI to the IBM at 9600 baud with no need of a 9600 baud modem (and no need of any modem at all!). If you're used to uploading or downloading material at 1200 or 2400 baud, it is amazing to see how fast that material can move across at 9600 baud! (On the IBM clone, I just capture the incoming material in a buffer and log it to disk, editing the log at leisure later.)

If you look at the preceding paragraph, you'll note that the connector I needed for this particular application involved the need of a "sex change" (hence Alan's title for his hardware description, which follows). In order to hook the two male ends of the cables together, what was necessary was to use two female DB25 connectors in the gadget he made for me. Also included in that gadget is a slide switch with two positions, "null" and "normal." As a hardware klutz, I never remember which position should be used for which applications, but if it doesn't work one way, I just switch the switch to the other setting and the problem is normally solved. (Allan reassures me that I can't do any harm to my computer if I don't get it right the first time.)

Which DB25 connectors you use, of course, will depend upon what your intended application is. In my situation, two female DB25 connectors were needed, but equally useful might be a gadget with one male and one female DB25 connector (plus the same "null"/"normal" switch, of course). I think Allan mentioned, for example, the ability of hooking up the TI to a terminal using this gadget, but doubtless many possible uses involve hooking up the TI to other hardware through the serial port. This project is relatively simple, but one that you well may find invaluable.

In the next article, I hope (if there is interest in the subject) to say more on the topic of converting TI Extended BASIC programs so that they will run in QuickBASIC on the IBM. I've made substantial progress since some earlier comments published here in MICROpendium. (For example, I now have worked out a QuickBASIC equivalent to the CALL CHAR statement in TI XB! Also, I have written a couple of "processing" programs that do a lot of the work for you, so far as modifying the program is concerned.) Again, it is often fairly easy to get TI XB programs in a runnable form in IBM QuickBASIC (unless, as I said, the programs involve sprites, speech, or multi-voice music), although I've had to put in some hard work getting the process to become easy! <grin> Just remember, however: even if you use a different computer, keep on compuTIn'!

Sex Change.... Is it Normal?

By ALAN SILVERSTEEN

There were and are many times that I have tried to attach a non-standard RS-232-C (serial) device to my TI-99/4A. The 4A has its own protocol that sometimes requires a special cable just to see if

(See Page 8)

SEX CHANGE—

(Continued from Page 7)

a device will work.

To eliminate the multitude of cable requirements, I made several connectors that do gender switching and also allow me to switch from a null to normal configuration on pins 2 & 3. All parts used are available from Radio Shack; the cost there is about $5 per connector, or, if you find a surplus dealer, the price is about $1 per connector.

Each connector consists of two DB25 connectors (either solder pot or crimp/insert type) and one double pole double throw (dpdt) slide switch. I standardized on the DB25 because the IBM DB-9 RS-232-C connector can and usually is converted to a DB25 with a cable or connector for use with printers or modems.

To effect a gender change, use two male or two female DB25 connectors. To just have a null to normal switch, use one male and one female DB25 connector.

The normal RS-232-C uses pin 2 for a transmit or receive data signal and pin 3 for the opposite (I don't remember which is which, and it's the reverse if the system is acting as a host or satellite). The DPDT switch makes a direct 2 to 2 and 3 to 3 connection in the normal mode and a 2 to 3 and 3 to 2 connection in the null mode.

Pins 1 and 7 are directly attached to their mating pin numbers in the other connector. For continuity I usually attach pins 6 to 6 and 20 to 20, but this is unnecessary except for some modem operations. The direct connect pins are attached with about 2" of 18 to 26 gauge solid conductor wire. The 2 and 3 pins are attached with about 3"of the same grade wire that comes from the switch.

After testing for continuity of the connections, I hot glue the two DB25 connectors together back to back with a gap of about 1/4 inch between them. When the hot glue has set, I glue the slide switch to the top center of this package over the 1/4 inch gap, potting the legs of the switch at this time. Care must be taken not to get hot glue in the switch mechanism, for this will prevent the switch from working properly.

The end result of all of this technical stuff is that after this connector is attached to the computer and the RS-232-C device and the appropriate device commands are given, the device should work. If it doesn't, then the null/normal switch is toggled and there is about a 95 percent chance it will now work. Best of luck!

Diagram 1

(Slide Switch Position)
(Normal)

Pin 2 of 2nd DB25<-------o1 4o------>Pin 3 of 2nd DB25
also attached to leg 6

Pin 2 of 1st DB25<-------o2 5o------>Pin 3 of 1st DB25

Pin 3 of 2nd DB25<-------o3 6o------>Pin 2 of 2nd DB25
also attached to leg 4

(Null)
(Slide Switch Postion)

This is a bottom view of the switch, showing the destination of the wires.

Diagram 2

```
1st DB25 connector          Switch          2nd DB25 connector
Pin No.                     Leg No.             Pin No.
1 <--------------2"-------------------------------------------> 1
2 <---------------3"-----------> 2
                                  1 <---------------3"-----------> 2
3 <-----------------3"----------> 5
                                  4 <---------------3"-----------> 3
                                 1<-->6
                                 3<-->4
6 <--------------2"-------------------------------------------> 6
7 <---------------2"------------------------------------------> 7
20<----------------2"------------------------------------------>20
```

<----------> = Represents wire and connection points.
(Note: Use the manufacturer's numbers found on the DB25 Connectors.)
This is an actual wire-pinout, showing the length of the wires needed. The The 1<-->6 and 3<-->4 are short pieces that just cross over the underside of the switch in an x shaped pattern without shorting to one another.

MPII index for 1993 available

The 1993 installment of the MICROpendium II index by Bill Gaskill is now available. The price for the disk is $6, which includes shipping.

The index covers January through December of 1993. It requires Microdex 99. MICROdex requires Extended BASIC, expansion memory and a disk system. A printer is recommended. The 1993 instalment of the MPII index may be ordered from MICROpendium Index II, P.O. Box 1343, Round Rock, TX 78680.

THE ART OF ASSEMBLY — PART 33

More Instructions

By BRUCE HARRISON

Today we'll continue with the mini-series for beginners in assembly language. Last month we covered the use of some of the TI's machine instructions, and today we will cover more of the most often used ones, as well as closing some topics we began last month.

JUMP RANGE

We mentioned last month that the range of jump instructions is limited, without elaboration. Now's the time to explain ourselves. A jump instruction itself occupies only one word in memory. Within that single word are bits to indicate the kind of jump that's intended (JEQ, JNE, JGT, etc.) and the address offset to which that jump is to take place. The space for this offset is only eight bits (one byte). That means the number itself can't go beyond the range of -128 through +127. In hex, that's the range from >80 through >7F. The microprocessor interprets this number as a differential address (in words, not bytes), starting from the present state of the program counter. "The *what*?", you ask. Inside the microprocessor chip are special registers that we can't access directly. (Some hardware person will probably write us a letter to say something about whether these registers are "on-chip" or not. Save your ink, we don't care!) The three important ones for the present discussion are the Workspace Pointer, the Program Counter (or pointer) and the Status Register. We already discussed the Workspace Pointer, which simply keeps track of where the set of registers we're using is located in memory. The Program Counter keeps track of where the next Instruction to be performed is located in memory. This advances in word steps, always pointing at even-numbered addresses. Each time an instruction is fetched from memory, the program counter is incremented by two, so it points to the next possible location for an instruction.

We know this is already getting pretty heavy, but there's just no substitute for really explaining what happens so that you'll understand how Jump instructions work. Let's look at a concrete example. Suppose the computer is executing the following sequence of instructions:

```
        LI    R4,5          place five in register
4
REPEAT  MOVB  *R9+,*R10+    move a byte
        DEC   R4            subtract one from R4
        JNE   REPEAT        if not zero, jump
back.
        INCT  R4            (next operation)
```

For the moment, we will ignore most of these instructions, and start with what happens when the microprocessor is executing the instruction JNE REPEAT. Once that instruction has been fetched from memory for execution, the chip will advance the program counter by two bytes, so the program counter now points to the location in memory where INCT R4 is located. The computer executes the JNE instruction by comparing the last completed operation to zero. Actually, that's not quite accurate. It does the comparison to zero as part of the execution of the DEC instruction, so that the status register bits are already set before the JNE instruction is fetched from memory. If the DEC R4 resulted in R4 becoming zero, the "EQUAL" bit will be set (made to equal 1) in the Status Register. If R4 did not become zero, then the EQUAL bit in the Status Register will be zero. This latter condition, where the EQUAL bit is "turned off", will cause the microprocessor to look at the second byte in the JNE instruction, and to add twice that "offset" to the contents of the Program Counter. In this particular case, the second byte in the JNE instruction will contain the value -3, so that when this is doubled and added to the program counter, the next instruction to be fetched will be the one at label REPEAT, instead of the instruction INCT R4. Thus a loop operation will be performed until the number in Register 4 decrements to zero. On that pass through the loop, the jump will not happen because the "EQUAL" bit in the status register will have been turned on, so the computer will go ahead to the next instruction after JNE, or INCT R4.

If there were a couple of pages of source code between label

(See Page 10)

THE ART OF ASSEMBLY—

(Continued from Page 9)

REPEAT and the JNE instruction, the offset byte stored in the JNE instruction word might have to be a larger negative number than -128. If that's the case, the Assembler will tell us that by issuing an error during the Assembly process. The error will be reported as "RANGE ERROR in line XXX". This condition is hard to predict when looking at source code, because the number of words in memory that each instruction line occupies may vary from one word to three. (In the example shown above, all of the instructions used were one-word kinds, so our calculation of the jump offset as -3 was easy.)

When one encounters this situation, the normal "workaround" is to switch over the logic involved, then use a B or Branch instruction to go back to label REPEAT. The revised code would look like this:

```
        LI    R4,5
REPEAT  MOVB  *R9+,*R10+  (lots more instructions
included in loop)
        DEC   R4
        JEQ   NEXTOP      if zero, jump ahead
        B     @REPEAT     else branch back to RE-
PEAT
NEXTOP  INCT  R4
```

What we've done is to invert the logic of our Jump so that it will make the short jump ahead to label NEXTOP when R4 becomes zero, else it will perform the operation of an unconditional branch to repeat the loop. Because the TI measures jump ranges in words, not bytes, the acceptable range for jumps is actually quite broad. (On PC computers, jump ranges are in bytes, which means they have only half the range potential of those on the TI. This is not so for the unconditional JMP on the PC, which has a range anywhere within a 64K byte segment.)

Our "normal" practice, when there's any doubt about the range of a jump, is to put the jump instruction in anyway, then let the Assembler tell us whether that's okay. This sometimes has bad results, but it often saves us bytes in our programs by running the risk on jump ranges.

You'll notice that this revised source code needs more instructions, and one more label than the first example. The instruction B @REPEAT takes four bytes in itself. That's because there is a two-byte word just for the B instruction, then another two-byte word for the address of the label REPEAT. This leads us nicely into another topic for today.

THE BRANCH INSTRUCTIONS

There are three important forms of Branch instructions, so let's cover all three here. Each is easy to spot, because it starts with the letter "B". The plain and simple Branch has just that letter as its opcode. The other two forms are the BL (Branch and Load) and the BLWP (Branch and Load Workspace Pointer) opcodes. Each requires just one operand, which is an address to which the branch is to happen. Any of the forms can branch to anywhere in the 64K address space. (This, while true, must not be taken too literally, as branching to some unknown place in the memory space can have disastrous results.)

Now that you know what the Program Counter is, we can discuss these branches in the more correct sense of how the computer executes them. For the unconditional Branch, the computer takes the second word of the B instruction, places that in the program counter, then continues execution with whatever instruction is at that location. (There are situations where the B instruction or the BL instruction occupies only one word in memory. We'll get to one of those in a few moments.) No "return address" is saved anywhere, so there's no automatic way of "returning" from a B. In most senses, this is a direct equivalent of the Basic GOTO instruction.

The second form, called BL, is roughly equivalent to the Basic GOSUB instruction. The return address is stashed in Register 11 of the current workspace, and the program counter is loaded with the word following the BL instruction, then execution continues with the instruction at that location. The difference here is that a "return" to the instruction following the BL is possible by having an RT (return) instruction at the end of the subroutine.

Before going on to BLWP, let's digress for just a moment into the alternate ways of providing the address for a Branch operation. This can be, as we've shown in the example above, simply a label preceded by the @ symbol. It can also be an address contained in one of the workspace registers. This allows the program to dynamically change the place to which the branch goes. Let's suppose that we have a situation in which, depending on what has

(See Page 11)

THE ART OF ASSEMBLY—

(Continued from Page 10)

happened in the program, there may be a need to BL to any of several different subroutines. (BASIC equivalent is ON-GOSUB) If we arrange to place the address of the chosen subroutine in, say, register 5, then we can write a source instruction like this:

```
BL    *R5
```

The asterisk tells the computer to use whatever number is in Register 5 as the address for the BL instruction. Needless to say, this is a very powerful form of the instruction, because it allows us to perform the equivalent of ON-GOTO or ON-GOSUB operations based on what happens during program execution.

We'll now go headlong into the really complex case of the BLWP instruction. As we pointed out in an earlier part of this mini-series, the BLWP allows us to change the workspace pointer for the duration of a "subprogram". This is similar in many respects to the use of subprograms in Extended Basic, but not with respect to "variables" in the data portion of the program.

To really understand a BLWP, we must now define another term, the Vector. A Vector is nothing but two words of DATA in memory, which provide the information the computer needs to execute a BLWP. Let's look at an example:

```
UTVEC  DATA  UTWS,UTSUB  subprogram workspace &
code addresses
UTWS   BSS   32          32 bytes set aside for
workspace
             UTSUB       (first instruction in
subprogram)
(subprogram code continues)
             RTWP        return from subprogam
USEUT  BLWP  @UTVEC      use the subprogram spec-
ified at UTVEC
```

The BLWP is a two-word instruction, where the second word is the address of the VECTOR, and is not the address at which program execution will continue. The vector itself is two words of data. The first of these is the address of the workspace to be used by the subprogram, and the second is the address of the first instruction in the subprogram.

As the name implies, the instruction BLWP both branches and loads the workspace pointer. It actually does more than that. If we BLWP @UTVEC, as above, many things happen all in that one instruction. First, the microprocessor takes the first word at the vector address to use for its workspace pointer. Next it takes the current contents of the workspace pointer and places that number into R13 of the new workspace. It takes the current contents of the program counter (pointing to the instruction after the BLWP's two words), and places that in R14 of the new workspace. Next it takes the current contents of the status register and places that in R15 of the new workspace. Once all this is done, the workspace pointer is changed to the new workspace location, and the program counter is loaded with the second word of the vector, so that execution will continue with the first instruction in the subprogram.

MANY HAPPY RETURNS

Both the BL and the BLWP instructions set up means for the program to return to whatever comes after the branching instruction's one or two words. The return instructions are different, with RT being used in the BL case, and RTWP in the BLWP case.

In the case of BL, the "return address" is placed in R11 of the workspace, so that a subroutine can be ended with the instruction RT (return). RT actually is just a pseudonym for the instruction B *R11. Both source statements (RT and B *R11) produce exactly the same result when assembled. That is the Hexadecimal number 045B. The 045 part means Branch to the address in a register, while the B tells the computer that register 11 (B is the number 11 in hex notation) contains the address to which it's to branch. For this return method to work properly, it's important that the subroutine should not tamper with the contents of Register 11 during its execution, so that the return address is preserved until it's needed.

In the BLWP case, the instruction RTWP causes a whole bunch of things to happen. The old Workspace Pointer is taken from R13, the program counter is re-loaded with the contents of R14, and the status register is re-loaded with the contents of R15. Execution then continues at the instruction that follows the BLWP instruction. As in the case of BL subroutines, it's important that the contents of R13, R14, and R15 not be tampered with during the subprogram's execution.

In both situations, there are cases in which the contents of these "return" registers can safely be altered, but those are beyond the scope of our "beginner's" lessons. Earlier columns in this series have dealt with ways of handling such alterations of the registers for the more advanced student.

That's a lot of material for one sitting, so perhaps we'd better quit writing now. Next month we'll conclude this little digression into the beginner's realm, and after that it'll be back to the more advanced and esoteric stuff. Of course some of you may consider this member of the series to be advanced and esoteric, but we can't help that. There's just no substitute for taking the time to understand what things really do.

Avoiding disaster with your Geneve

By GREG KNIGHTES

Yeah, I know, you've heard it many times before: "Make sure you back up your hard drive!". I'm repeating it because I have had quite a bit of experience with that phrase recently! The most recent occurrence left me down for more than two weeks because it was early December and with the holidays fast approaching and overtime available at work, I didn't have the time I wanted to get back up and going. Just a little background, I have a 40-megabyte hard drive as No. 1 and recently added a couple of 20-megabyte drives that I alternate as No. 2.

First, if you do experience a "crash," it is best to have a "recovery kit" available. I

(See Page 12)

AVOIDING DISASTER—

(Continued from Page 11)

like to keep on hand and on floppy disks SYSTEM/SYS, LOAD/SYS, the GPL files, Autoexec, MDM5, Shell, Backup Miser, Sector One, Archiver 3.03G, ROMPAGE or OLDDSR, Xcopy and Disk Manager. They all can come in handy. Make sure that your floppy with SYSTEM/SYS boots! I made one once that didn't boot and I lost an hour trying to get it to work! I finally gave up and rummaged for another copy. By that time I had dip switches on my HFDC all messed up. I didn't know if I needed my TI floppy controller card or what. So, the last thing you need is a set of printed directions on how everything should be set up so that you can recover fast. Keep all this in a safe place along with your backups!

Many utilities will ensure a quick and painless recovery. The first one is MDOS's COPY command, probably the cheapest and easiest to use but potentially the slowest. Simply place a floppy diskette in drive one. Log onto the subdirectory of the hard drive that you want to back up and type COPY * A: This will copy all the files in that directory onto the floppy disk. The downside is that you most likely will run out of space on your floppy drive before you run out of files to copy. In that case, you have to back up the rest of the files manually one by one or small group by small group onto another floppy. Though time consuming, the plus is that you don't have to do a restore to start using those files again. They remain uncompressed.

Clint Pulley's Directory Manager (or DM for short) makes this task easier. Log onto the directory and tag the files you want copied. Keep an eye on the box in the upper right-hand corner and you will notice the size of the files to copy. Once you tag a file that exceeds the number of sectors on your floppy disk, untag the last file and you are ready to copy. You won't waste time while the last file tries to fit onto the floppy disk, but won't. One other downside to this method is that you can't copy files larger than the size of your floppy disk. They just won't fit!

Another method that allows you to get more files per disk is to use the copy method in conjunction with Barry Boone's Archiver. Copy the files to your RAMdisk or to a floppy if you have two floppy drives and archive all the files on the first drive to the second drive. This is also time consuming as files have to be copied twice, but more files can be stored on a floppy disk. Too bad Archiver doesn't allow you to use a hard drive in the path name! Again, this is also easier if you use DM to copy the files. Using a RAMdisk, either Horizon or Geneve internal, speeds up the copying and archiving appreciably.

The next option is to use MDM5 from Myarc. For a while, this was the only utility to back up and format your hard drive. To use MDM5, make sure that you have *plenty* of *formatted* disks on hand! Have more than you think that you will need! Once the backup starts, it doesn't end until the last file is backed up! If you run out of formatted disks, there's no stopping to format more. If you stop, you have to start all over. This can be quite painful if you're almost near the end! If MDM encounters an error in the backup, it will stop and you will have to start all over. One nice thing that MDM does is allow you to specify multiple floppy drives to back up to. If you have three drives, you can stick a fresh floppy in each, press enter and MDM will not pause for new floppies until the third one is used! This is handy if you have a number of drives and want to start the process and go do something else.

The backup process became much less painful with the advent of Al Beard's BackUp Miser. This is a Genbench Shell application that allows you to do complete backups and restores, partial backups, wild card backups and since date backups. You can specify that each floppy be formatted before use, swept clean, or files just added to the floppy without erasing anything. The backed-up files are compressed before being saved to disk and multiple directories can be saved to the same disk. Once a complete backup is done, new or changed files on the hard drive since the last backup can be saved the next time. This in effect saves a lot of time and saves the most recent files.

If you are lucky enough to have two hard drives of the same capacity, you can use two additional methods that are *much* faster and much more automated. One is a program by Tom Freeman called Hard Back. This program will back up a hard drive to another hard drive. I don't know too much about this program.

Another option is to use John Johnson's Xcopy program. This copies all the files on one hard drive to another. There is no compression done; it's like COPY HDS1.* HDS2.* but it also copies subdirectories, if you want. You can also tell it not to copy subdirectories. There is also an option to copy certain files. John provides good documentation and a short example that is all you really need to do backups.

I have two 20-megabyte hard drives at the moment and although Xcopy won't work for copying the large to the small, it works great the other way around. I wrote a couple of batch files to make directories and copy the contents of the first hard drive to the second hard drive. I copy the directories A-F and their associated subdirectories to one hard drive and G-Z to the other hard drive. When the first hard drive is done, I do a DIR >PRN for the hard drive and tape the output to it. I then unplug the hard dive and plug the other in. After letting it get up to speed, I use the other batch file to copy the second half of my main hard drive.

Every few days, I copy new or updated files to the appropriate secondary hard drive and I'm up to date. If anything happened to my main hard drive, I would reformat it and use Xcopy to restore the backups from the secondary hard drives to my primary hard drive. No need for batch files here.

Now that you have a backup that you can rely on, what do you do if your system does crash? So, your system is on but you can't do a DIR on a directory, can't find files to copy, get a "device error" message, etc. You come to the conclusion that you crashed. What do you do? First, don't turn your power off. This is especially true if you don't know where a bootable floppy is! Second, and even more important — *do not attempt to write any files to your hard drive!*

Why, do you ask? Because the possibility is good that subdirectories can still be accessed. Try it. Try a CD \subdirectc command and then DIR. There's a good chance that you will get a directory listing!

(See Page 13)

AVOIDING DISASTER—

(Continued from Page 12)

I've blown my root directory. Doing a DIR gave me a device error. Changing to a subdirectory let me copy files onto floppy. This is handy if your most recent backup isn't very recent and you have a lot of new or changed files since your last backup.

Using Sector One, follow the instructions for locating your subdirectories and the various files that belong there. You may be able to restore enough so that it is not a complete loss. At this point, you may have been able to do a directory on what you thought were lost directories and everything looks fine, just as if a crash never happened. You may be tempted just to continue. Don't do it! If you notice the sector count, it probably doesn't match exactly what you had before the crash. More or fewer sectors may be marked free. At this point, it's time for the next phase.

Load either MDM5 or Cform to do your reformat. Follow the proper steps to reformat your hard drive. At this point, *make sure* you have indicated the correct drive to reformat! Take the extra time to double and triple check everything, then check again before you type that word "FORMAT" and press enter! I have started to reformat a backup drive by being careless! I was able to use Sector One to recover the files. At this point, I unplugged the power to that drive! This is an extra step that you can take to insure that the wrong drive won't get reformatted.

A quick comparison of MDM5 and CFORM seems to be in order. MDM5 is free. It came with your Geneve (or should have!). MDM5 formats 32 sectors per track, with no option for setting step speed. It is sometimes buggy and unreliable. Not all versions work with your Geneve! Cform is a commercial product written by Mike Maksimik and distributed by Cecure Electronics. It is well written, makes use of the mouse interface, is GenBench Shell compatible (although GenBench Shell is not required), allows setting the step rate and provides a nice printed output showing all formatting specifications and lists errors found, if any. Cform also allows you to format 34 sectors per track if you wish. This effectively increases your storage capacity by 6.6 percent over MDM5! The three levels of sector verification range from very little to very extensive. I feel that Cform is by far the better product.

Now that you have your hard drive reformatted, it's time to restore your files so that you can get back to business. Use the restore files function of the program that you used to make your backup. If you have various generations, start with the oldest generation and work forward. This way, newer, more recent files will write over, older unwanted versions.

Let me make some other suggestions that may help things go smoother if your hard drive becomes sick for whatever reason. First, after you reformat your hard drive and before you write anything to it, load up Sector One and print a copy of sector zero and sector >20. This will help you reset your drive if you really screw up trying to recover something. Also, repeat this process after you have your directories and subdirectories made and your files restored. This will also help you find links to where a subdirectory belongs and where the files are that belong inside it. The directories tend to be pretty stable, once set, but files get erased, renamed, moved copied, modified, added to, etc. This won't help much with file recovery but it is a start.

A couple of other tips. Go through your backup diskettes (or hard drive(s)) and delete unwanted files, especially those files for which you have more up-to-date versions. This saves you from cluttering up your hard drive, as you will not inadvertantly restore them once you delete them. Also, with MDOS version 1.53H, a MIRROR command in your Autoexec file is supposed to make recovery easier. However, you are recommended to use this command only with hard drives that have been formatted with MDM5. I am sure that it would also work with a drive formatted with Cform and 32 sectors per track. I plan on testing this out and testing it with drives formatted 34 sectors per track. This is one advantage to using MDM5 to format your hard drive.

I have purposefully left out one method of backing up your files because this method has not been officially released to the Geneve community yet, tape backup. I understand that the software is done and works fine. Mike Maksimik will be releasing this as Uni-Manager with the next release of 9640 News This promises to make unattended backup much easier and I look forward to its release.

So, to review what you should do to make recovery from a disaster easier, you should:

a) Do complete backup followed by regular, frequent updates.

b) Have a disaster recovery kit made up of all the utility programs you think you may need on floppy disk in case a crash occurs.

c) Know how to use those programs to recover lost files.

d) Know how to reformat your hard drive.

e) Know how to restore your backed-up files to the hard drive effectively so you can be back in business as quickly as possible.

Noteworthy

An Extended BASIC game with a musical flair

The following Extended BASIC game program was written by Englishman Roland Trueman. The object of the game is to climb a series of floors while collecting musical notes. The S and D keys are used to move left and right and the P key is used to jump. The games requires a memory expansion.

NOTEWORTHY

```
100 CALL CLEAR :: CALL CHARS
ET :: CALL SCREEN(2):: RANDO
MIZE :: CALL MAGNIFY(3):: CA
LL CHARPAT(89,Y$):: CALL CHA
```

(See Page 14)

NOTEWORTHY—

(Continued from Page 13)

```
R(74,Y$)!251
110 CALL COLOR(3,7,11,4,7,11
,5,5,8,6,5,8,7,5,8,1,16,2,2,
13,11,8,4,15,11,2,16)!208
120 CALL CHAR(119,"080C0B083
878783",81,"",95,"",124,"",1
25,"0E1115111F38777F",126,""
,127,"7088A888F81CEEFE")!110
130 CALL CHAR(92,"1833777F7E
7C7839391C0E47A78FFF7F0080C0
E0202020A022257FFF0780C2FE")
!023
140 CALL CHAR(136,"000103070
404040544A4FEFFE001437F18CCE
EFE7E3E1E9C9C3870E2E5F1FFFE"
)!148
150 CALL CHAR(106,"18247E16F
7E7C7FF",40,"4B4B4B4B4B4B4B4
B012452ACB6DBBFFFF8A4DAFAE8D
AF4E110461AA4532E1B57",118,"
30787838080B0C08")!004
160 CALL CHAR(107,"18247E5ED
7C3F7FF",44,"089228D5564AAF6
DF0C4E9B4CAFBFDFF270B532F97B
7EFFF",116,"FFFAF2E2C2828282
")!217
170 CALL CHAR(36,"FF818181FF
000000FF81818181FF0000FF8181
818181FF00FF8181818181FF",
33,"FFFF",34,"FF81FF",35,"FF
8181FF")!146
180 CALL CHAR(88,"FFFDDB6D35
4A2480872F5B172F5B251FFFF7ED
E9F4CAB0E4FFBFDF532D97230F")
!252
190 CALL CHAR(117,"FF5F4F474
3414141",140,"01010101010101
01FF403F100F0403018080808080
808080FF02FC08F020C08")!066
200 CALL CHAR(60,"0E1115111F
3D777BFCFFFF7F7F3F1F0F7088A8
88F8B8EEDE3FFFFFFEFEFCF8E",1
04,"18247E76E7F73CFF",105,"1
8247E66F7EFECFF")!209
210 CALL CHAR(128,"071F3F716
EEFEDF7F9FFFF7F7F3F1F07E0F8F
CC6829B93C7FFFFE5C194FCFCE")
!080
220 CALL CHAR(132,"071F3F634
1D9C9E3FFFFA783293F3F07E0F8F
C8E76F7B7EF9FFFFFFEFEFCF8E")
!067
230 CALL CHAR(59,"6C547EC6FE
FE7C38",108,"03060C0C1F3F7F7
FFFFFFF7F7F3F1F07C020E0A0F8E
CFEDEE0FFFFFEFEFCF8E")!170
240 CALL CHAR(112,"030407050
71F377F7B07FF7F7F3F1F07C0603
030F8FCFEFEFFFFFFFEFEFCF8E")
!227
250 CALL CHAR(96,"0728382909
78F8FFBFC0F1F1F9FD7F2AE014DC
F4B0DE1FFFFD038F8F9FBFFE54")
!015
260 CALL CHAR(120,"07283B2F0
D7BF8FFBFC0F1F1F9FD7F25E0141
C94901E1FFFFD038F8F9FBFFE54"
)!024
270 CALL CHAR(58,"",100,"1E2
32333231F0373DB8F0703318BC77
F000000000080848AC8C8E8F8F8F
CFEFF")!156
280 SC,BO,CO=0 :: ME=3 :: LE
,WA=1 :: CALL CHAR(104,"0000
0000000121511313171F1F3F7FFF
78C4C4CCC4F8C0CEDBF1E0C08CD1
E3FE")!159
290 DISPLAY AT(2,1):"ZXXX[ Z
[ ZXXXX[ ZXXXX[ ZXXX[*+),Y *
Y *+)),Y -),+). *+)).*Y *Y *
Y *Y  *Y   *Y   *Y  *Y *Y *
Y *Y  *Y   *Y   *[X[" !117
300 DISPLAY AT(6,1):"*Y *Y *
Y *Y  *Y   *Y   *+). *Y *Y *
Y *Y  *Y   *Y   *Y  *Y *[XZ
Y *[XXZY   *Y   *[XX[-. -)))
. -)))).   -.   -))).." !153
310 CALL HCHAR(20,1,81,160)!
017
320 DISPLAY AT(12,1):"!
  !    !!! ''' ! ! ' '#
  #    # #  '  # # ' ' $
 $     $ $  &  $$$ & & %
 %  %  %%%  %  % % %%% !128
330 DISPLAY AT(16,1):"  & &
&  $ $ &&   $  & &  $   ' '
'  # # ' '          #   ''''
'   !  '  ' !!!!!!!!!" !173
340 DISPLAY AT(20,1):"QQQQQQ
FROMQQRQTRUEMANQQQQQQQQQQQQQQ
QQTHEQBADDIESQQQQQQQQQQ1QROBO
IDSQQQ2QSLUGQCREATURES3QGRUM
PIESQQ4QMADQDOGSQQQQQQ" :: G
OSUB 1090 !158
350 CALL KEY(0,K,S):: IF S=0
 THEN CALL SOUND(-90,(RND*20
)+110,19):: GOTO 350 !235
360 CALL KEY(0,K,S):: IF S=-
1 THEN CALL SOUND(-90,(RND*2
0)+110,19):: GOTO 360 ELSE
ALL HCHAR(20,1,81,160):: DIS
PLAY AT(20,1):"SQANDQDQTOQMO
VEQQLEAPQWITHQP" !182
370 DISPLAY AT(21,1):"WHENQU
NDERQLARGESTQPANELSQTOGOQTOQ
HIGHERQFLOORQQQPICKQUPALLQNO
TESQFORQAQBIGGERQBONUS" :: G
OSUB 1090 !248
380 CALL KEY(0,K,S):: IF S=0
 THEN CALL SOUND(-90,(RND*20
)+110,19):: GOTO 380 ELSE CA
LL CLEAR :: CALL COLOR(1,10,
16):: CALL SCREEN(16)!058
390 CALL VCHAR(3,1,42,22)::
CALL VCHAR(3,31,42,22):: CAL
L VCHAR(3,2,89,22):: CALL VC
HAR(3,32,89,22):: FOR Y=3 TO
 23 STEP 5 !151
400 CALL HCHAR(Y,2,91):: CAL
L HCHAR(Y+1,2,43):: CALL HCH
AR(Y,31,90):: CALL HCHAR(Y+1
,31,44):: NEXT Y !121
410 DISPLAY AT(1,1):"QQMEN";
TAB(14);"SCORE";SC:"LEVEL";L
E;TAB(14);"BONUS";LE*10 ::
ALL HCHAR(1,9,59,ME):: ON W
 GOTO 420,510,560,610 !079
420 DISPLAY AT(3,1):"XXXXXXX
XXXXXXXXXXXXXXXX[''ZX)))))))
))))))))))tu))).  -)
                      vv
 w v  w   vw  w  w   " !236
430 DISPLAY AT(8,1):"X[''ZXX
XXXXXXXX[$$ZXXXXXXXXX).  -))
tu))))))  -)tu))tu))
                     v
 wvv     w  v    v" !161
440 DISPLAY AT(13,1):"XXXX['
'&&%%$$ZXX[&&##%%ZXXXX)))).
      -)).     -))))
                     v
 w  w    v ww  w v w" !130
450 DISPLAY AT(18,1):"XXXXXX
XX[''Z[$$ZXXXX[!!''!!Z)))))
tu.  -.  -))tu.     -
                      w w
w   vv  w  v  v  wv" !219
460 RESTORE 930 !002
470 DISPLAY AT(23,1):"XXXXXX
XXXXXXXXXXXXXXXXXXXXXX))))))
))))))))))))))))))))" !1
480 FOR Y=2 TO 7 :: READ A,B
 :: CALL SPRITE(#Y,140,14,A*
```

(See Page 15)

NOTEWORTHY—

(Continued from Page 14)

```
8+1,B*8+1):: NEXT Y :: READ
A,B,D,FT,HT :: CALL SPRITE(#
8,HT,D,A,B,0,1+LE)!093
490 READ A,B :: M=A :: N=B :
: CALL SPRITE(#1,108,9,M,N)!
096
500 CALL KEY(0,K,S):: IF S=0
 THEN CALL SOUND(-90,(RND*20
00)+2000,10):: GOTO 500 ELSE
 710 !029
510 DISPLAY AT(3,1):"XXXXXXX
XXX[''ZXXXXXXXXXXXXXX)))))))
)tu.  -)))))))))))))
                    wv w v
ww    v    ww   v  v" !120
520 DISPLAY AT(8,1):"XXXXXXX
XXXXX[''&&%%$$####!!Z)))tu))
)))tu.              -
                     vv
wv w   vwww  v  w v" !105
530 DISPLAY AT(13,1):"XXXXXX
XXXXXXXXX[''ZX[$$&&ZXX))tutu
)))))))tu.  -).    -))
                     wv
wv w         v   vv" !208
540 DISPLAY AT(18,1):"X[''##
!!''&&ZXXXX[&&$$ZX[''Z).
     -)))).    -).  -
                     v w
 v   wwv       vv   w" !060
550 RESTORE 940 :: GOTO 470
!180
560 DISPLAY AT(3,1):"X[''ZX[
&&ZXXX[##%%ZXXXXXXXXX).  -).
  -))).    -)))))))))
                    wv   v
   wwvw   vvv  wv v" !029
570 DISPLAY AT(8,1):"XXXXXXX
XXXXXXXX[!!%%''%%!!ZX)))))))
tututu)).          -)
                     wvwvv
      wvv  w  w  v" !153
580 DISPLAY AT(13,1):"[##''%
%ZXXXXXXXXXXXXXXXXXXX.
 -))tu))tu)))))))))))
                       v
vv  www    vv v   www" !163
590 DISPLAY AT(18,1):"X[''ZX
XXXXXXXXX[''ZXXX[$$&&Z).  -)
)))))))))).  -)tu.    -
                      vw
vwv   w vv   wwv   v" !155
600 RESTORE 950 :: CALL SPRI
TE(#9,128,15,41,220,0,1+LE):
: GOTO 480 !147
610 DISPLAY AT(3,1):"XXXXXXX
XXXXXXXXXXXXXXX[''ZXX)))))))
)))tu)tu)tu)tu).  -))
                     v  wwv
  wv vw vv ww    wv" !064
620 DISPLAY AT(8,1):"X[''ZXX
XXXXXXXXXXXXXXXXXXXXX).  -))
))tu)))))))))tu))))))
                    wv  v
  w  v wv w  w  w v" !243
630 DISPLAY AT(13,1):"X[$$%%
ZXX[&&ZXXXX[##!!ZX[''Z).
-)).  -)))).    -).  -
                     w v
vw   wvw    wvwv" !052
640 DISPLAY AT(18,1):"X[''$$
ZXX[##Z[%%ZXXXX[!!%%ZX).
-)).  -.  -)))).    -)
                      wv
w  w v   wwv   vw  wv" !051
650 RESTORE 960 :: CALL SPRI
TE(#9,92,5,161,1,0,3+LE):: G
OTO 480 !011
660 CALL MOTION(#1,0,0):: CA
LL PATTERN(#1,60)!080
670 Y=(RND*18)+2 :: IF Y>7 T
HEN 690 ELSE CALL POSITION(#
Y,A,B):: CALL LOCATE(#Y,A+8,
B):: CALL PEEK(-31877,O):: I
F O AND 32 THEN 970 !024
680 CALL LOCATE(#Y,A,B):: GO
TO 710 !104
690 IF Y<17 THEN 710 :: IF Y
<18 THEN 700 :: CALL MOTION(
#8,0,-1+(LE-(LE*2)),#9,0,-1+
(LE-(LE*2))):: CALL PATTERN(
#8,FT,#9,FT):: GOTO 710 !131
700 CALL MOTION(#8,0,1+LE,#9
,0,1+LE):: CALL PATTERN(#8,H
T,#9,HT)!123
710 CALL PEEK(-31877,O):: IF
 O AND 32 THEN 970 !027
720 CALL KEY(0,K,S):: IF S=0
 THEN 660 ELSE ON POS("PpSDs
d",CHR$(K),1)+1 GOTO 660,770
,770,730,750,730,750 !079
730 CALL POSITION(#1,A,B)::
IF B<20 THEN CALL MOTION(#1,
0,0):: GOTO 670 ELSE CALL MO
TION(#1,0,-4):: CALL PATTERN
(#1,112)!070
740 K=INT(A/8)+1 :: Y=INT(B/
8)+1 :: CALL GCHAR(K,Y,O)::
IF O<118 THEN 670 ELSE GOSUB
 990 :: CALL HCHAR(K,Y,32)::
 GOTO 670 !097
750 CALL POSITION(#1,A,B)::
IF B>220 THEN CALL MOTION(#1
,0,0):: GOTO 670 ELSE CALL M
OTION(#1,0,4):: CALL PATTERN
(#1,108)!189
760 K=INT(A/8)+1 :: Y=INT(B/
8)+3 :: CALL GCHAR(K,Y,O)::
IF O<118 THEN 670 ELSE GOSUB
 990 :: CALL HCHAR(K,Y,32)::
 GOTO 670 !099
770 CALL MOTION(#1,0,0):: CA
LL POSITION(#1,M,N):: CALL G
CHAR(M/8-2,N/8+1,O):: IF (O>
33)*(O<40)THEN 780 ELSE 670
!102
780 CALL GCHAR(M/8-2,N/8+2,P
):: IF P<>O THEN 670 ELSE CA
LL COLOR(#1,10):: Y=O :: O=M
-42 :: T=M/8-2 !146
790 IF M<17 THEN 1050 ELSE M
=M-7 :: CALL LOCATE(#1,M,N):
: Y=Y-1 :: CALL HCHAR(T,N/8+
1,Y,2):: IF Y=33 THEN 800 EL
SE 790 !070
800 IF M=O THEN 830 ELSE CAL
L PEEK(-31877,K):: IF K AND
32 THEN 930 !004
810 FOR Y=M TO O+42 :: CALL
PATTERN(#1,108,#1,112):: CAL
L LOCATE(#1,Y,N):: CALL PATT
ERN(#1,60):: NEXT Y :: M=Y :
: CALL COLOR(#1,9)!104
820 GOTO 670 !239
830 CALL COLOR(#1,9):: M=M+2
 :: CALL LOCATE(#1,M,N):: GO
TO 670 !019
840 CO=11 :: CALL HCHAR(18,1
4,32,3):: CALL HCHAR(19,14,3
2,3):: CALL SPRITE(#1,60,9,1
21,141,#15,108,9,121,1)!152
850 CALL SPRITE(#14,92,8,121
,16,#13,128,11,121,32,#12,10
4,5,121,48,#11,96,14,121,64,
0,8)!214
860 CALL COINC(#CO,121,113,4
,O):: IF O=-1 THEN 870 ELSE
860 !003
870 CALL MOTION(#CO,16,0)::
FOR Y=610 TO 110 STEP -50 ::
 CALL SOUND(-60,Y,0):: NEXT
Y :: CALL DELSPRITE(#CO):: C
O=CO+1 !185
```

(See Page 16)

NOTEWORTHY—

(Continued from Page 15)

```
880 IF CO<15 THEN CALL MOTIO
N(#CO,0,8):: GOTO 860 ELSE C
ALL MOTION(#15,0,8)!062
890 CALL COINC(#1,#15,16,O):
: IF O=0 THEN 890 ELSE CALL
MOTION(#15,0,0):: CALL PATTE
RN(#1,112)!056
900 CALL CHAR(100,"1C3E7FFFF
FFFFFFF7F7F3F1F0F070100387CF
EFFFFFFFFFFFEFEFCF8F0E08")::
 CALL SPRITE(#9,100,7,100,13
2,-2,0)!081
910 ME=ME+1 :: FOR Y=1 TO 7
:: FOR T=610 TO 1110 STEP 50
 :: CALL SOUND(-50,T,0):: NE
XT T :: CALL HCHAR(1,9,32,8)
:: CALL HCHAR(1,9,59,ME)!164
920 NEXT Y :: CALL DELSPRITE
(ALL):: CALL CHAR(100,"1E232
333231F0373DB8F0703318BC77F0
000000000080848AC8C8E8F8F8FCF
EFF"):: GOTO 410 !010
930 DATA 3,20,8,9,8,26,8,22,
18,8,18,20,121,129,14,120,96
,161,81 !107
940 DATA 3,10,8,12,8,5,13,4,
13,6,13,15,161,1,5,100,104,1
61,209 !019
950 DATA 8,9,8,11,8,13,13,12
,13,16,18,22,41,100,11,132,1
28,161,192 !235
960 DATA 3,12,3,15,3,18,3,21
,8,11,8,22,121,192,8,136,92,
161,209 !092
970 Y=1 :: CALL SOUND(-90,-7
,0):: SC=SC+BO :: DISPLAY AT
(1,19):SC :: CALL MOTION(#1,
0,0):: CALL PATTERN(#1,124):
: BO,CO=0 :: ME=ME-1 !167
980 CALL COLOR(#1,(RND*13)+2
):: Y=Y+1 :: IF Y<20 THEN 98
0 :: CALL DELSPRITE(ALL):: C
ALL HCHAR(1,9+ME,32):: IF ME
=0 THEN 1000 ELSE 410 !104
990 BO=BO+(LE*10):: T=(RND*2
000)+1000 :: CALL SOUND(-50,
T,0):: RETURN !148
1000 CALL DELSPRITE(#1,#7)::
 FOR Y=10 TO 16 :: CALL HCHA
R(Y,10,81,14):: NEXT Y :: DI
SPLAY AT(11,10)SIZE(10):"GAM
EQQOVER" :: HI=MAX(HI,SC)!01
8
1010 DISPLAY AT(13,10):"HIGH
";:: DISPLAY AT(13,15):"Q"&S
TR$(HI)&"Q";:: DISPLAY AT(15
,9)SIZE(12):"REPLAJQJQ/QN" !
192
1020 CALL KEY(0,K,S):: IF K=
78 OR K=110 THEN CALL DELSP
ITE(ALL):: CALL CLEAR :: RUN
 "DSK1.MENU" ELSE IF K=121 O
R K=89 THEN 1030 ELSE 1020 !
065
1030 FOR Y=1 TO 20 :: CALL S
CREEN(RND*13+2):: NEXT Y ::
CALL SCREEN(16):: ME=3 :: SC
,BO,CO=0 :: LE,WA=1 !016
1040 CALL DELSPRITE(ALL):: C
ALL CLEAR :: GOTO 390 !030
1050 CALL DELSPRITE(#1,#8,#9
):: SC=SC+(LE*1000):: FOR Y=
1 TO 21 STEP 2 :: CALL SOUND
(-90,2000,Y):: NEXT Y :: DIS
PLAY AT(1,19):SC !072
1060 DISPLAY AT(2,19):BO ::
FOR Y=1 TO 10 :: CALL HCHAR(
2,21,32,8):: DISPLAY AT(2,19
):BO :: CALL SOUND(-10,(2000
)+2000,0):: NEXT Y !228
1070 SC=SC+BO :: DISPLAY AT(
1,19):SC :: BO=0 :: DISPLAY
AT(2,19):BO !092
1080 LE=LE+1 :: WA=WA+1 :: C
ALL DELSPRITE(ALL):: IF WA<5
 THEN 410 ELSE WA=1 :: GOTO
840 !023
1090 DISPLAY AT(24,1)BEEP:"Q
QPRESSQSOMETHINGQTOQSTARTQQ"
 :: RETURN !015
```

How to hook up disk drives

By MICHAEL MAKSIMIK

(This post from the Chicago User Group BBS was reprinted from the January 1994 Ozark 99er News)

Here are a few reminders on hooking up disk drives. If you own a TI:

1) *Any* 40-track drive will work, regardless of disk controller. Choose double-sided, double-density diskettes to get the most online storage.

2) The TI Disk Controller will operate in single density mode only. This can give up to 180K per diskette. Use nine sectors per track and use an interlace 2 factor, if applicable to your formatter.

3) The CorComp disk controller will operate in single and double density modes. So, you can get up to 360K per diskette. Also, faster head step settings are possible, to allow faster track to track movement of the read/write heads on drives that support them. All double-sided drives on the market today, with few exceptions, operate at a 6-millisecond head step, quietly, and at 12- or 20-millisecond head step, noisily.

A Myarc floppy disk controller on a Geneve does not change. It retains all features that are present with the card on the TI. Ideal for making disk media for interchanging between a TI system and a Geneve.

4) The Myarc Floppy Disk Controller equipped with the new 80-track EPROM will control:

360K 5.25"
720K 5.25"
720K 3.5" (or 1.44 meg drive, but only in 720K mode)

The controller's software can format and read 16- and 18-sector per track disks.

6) The Myarc Hard and Floppy Disk Controller. This can control all above-

(See Page 17)

DISK DRIVES—

(Continued from Page 16)

mentioned drives, plus 1.44 meg drives in true high density mode, with proper software. It can read all other disk formats and can format to 16/18 sectors per track with Myarc Disk Manager V, and can format to 36 sectors per track with the HFDC144 program.

On a Geneve, the master DSR program can make your current disk controller more versatile:

• A TI disk controller on a Geneve can control 80-track drives in single-density mode, giving true 630K storage.

• A CorComp controller on a Geneve can control 80-track drives in double-density mode, at 18 sectors per track only, giving 360K storage.

• A Myarc floppy disk controller on a Geneve does not change. It retains all features that are present with the card on the TI. Ideal for making disk media for interchanging between a TI system and a Geneve. Both computers can share the same media, with no problem converting between either computer.

• A TI double-density controller is not supported on the Geneve. Due to the absence of architect and schematic diagrams, plus software support, it is unlikely that this controller will work on a Geneve.

• A Myarc Hard and Floppy Disk Controller operates, as on a TI, with greater speed. Currently, you can operate the device in EPROM mode, or in master mode. Operating the HFDC on a Geneve in EPROM mode ensures compatibility with the previous TI programs that addressed the attached hard disk drives as "WDS1.filename."

Choosing "Master DSR" mode will enable most of the features of the floppy and hard disk drives. Currently, software can operate on the HFDC to the extent of "level 3 routines." These include OLD, SAVE, OPEN, CLOSE and DELETE from TI BASIC, or other languages.

Software does not operate properly with the low-level routines to access sectors directly on the hard disk through MDOS. This means that current software must be intelligent enough to switch to EPROM mode for these operations. Most new software, such as Hardback and Filezape, does this. Other operations, called "level 2 routines," or referred to as the "2x" routines, are being worked on. Also, the EPROM on the Geneve does not recognize the floppies connected to the HFDC as "bootable" floppies, or the drives where MDOS can be loaded if it cannot be found on RAMdisk or hard drive.

A new EPROM for the Geneve is forthcoming from Cecure Electronics, with MDOS burned into it permanently. This makes is unnecessary to load MDOS from any device, although it can be loaded if need be from any device with a keypress.

Currently, there are two hard disk controllers for the TI: the Geneve HFDC and the new Western Horizon Technologies SCSI controller.

The HFDC uses MFM hard drives. Most drives for PCs nowadays will not work with the HFDC. MFM drives are upgrading to the AT-style IDE hard drives for PCs only. MFM drives are most likely to be found for sale in good working condition at computer fairs, or on the bargain tables in computer specialty stores. They are also available in ads in Computer Shopper magazine. Many can be had for a reasonable price.

The WHT SCSI card interfaces the SCSI hard drives and peripherals. SCSI peripherals are popular on midrange IBMs (AS/400s) and on Apple Macintosh computers. In fact, Macintoshes uses SCSI de-

(See Page 18)

SCSI card up and running

The SCSI (small computer system interface) DSRs (device service routines) from Bud Mills Services and Western Horizon Technologies are scheduled for completion in mid-March, according to Bud Mills.

The team had expected to complete the DSRs during Fest West, Feb. 19-20, he notes, so the companies sold SCSI cards at that event.

The team completing the DSRs consists of Mike Maksimik, whom Mills describes as being turned loose "like a bull in a china shop, you give him the hardware and he throws commands in it," along with Jeff White, Brad Snyder and Tim Tesch.

The team is completing two versions of the SCSI DSR, Mills says, the initial one which "gets everything up and running" and the final version which "will contain all kinds of bells and whistles."

Right now, Mills says, Maksimik has optimized code for an NEC standalone drive, so it should work with most CD-ROM drives, but "we can't guarantee it. We don't recommend anyone going out and buying CD-ROMs until we have something."

The SCSI card is designed to support SCSI-2 standard format devices, Mills says.

The utility disk formatter programs utilities to set up folders for formatting the hard drive, Mills says, and can format TEAC-compatible drives, either 5.25-inch 1.2 meg, or 3.5-inch 1.44 meg. It can also format up to 3.5 meg on a super high density disk but this takes a special drive, Mills notes.

"We're doing it all up front so it will have the capability without having to revise anything," Mills notes. "One drawback is that in our floppy controller, due to a manufacturer's omission, you can't format TI double-sided, double-density or TI double-sided, single-density, but you can read or write to those disks that are already formatted. This is a problem we've referred back to the manufacturer. Most people who'll be using it are using it for the ability of higher density."

Mills says the companies will also ship a version of MDOS that has embedded the SCSI commands necessary. Rich Wendy XB by Richard Gilbertson will be included.

A cartridge using generic EPROMs for any number of programs, similar to the P-GRAM produced by the late John Guion, is available from Don O'Neil of Western Horizon Technologies "for a modest fee" Mills says.

For further information, contact Bud Mills Services, 166 Dartmouth Dr., Toledo, OH 43614-2911, (419) 385-5946 (voice), (419) 385-7484 (BBS, 8N1); or Western Horizon Technologies, 1025 Jean Ellen Dr., Gilroy, CA 95020, (408) 848-5947.

Don't overlook those LEDs for help in troubleshooting

By GLEN BERNASEK

The following article comes from the TI-Chips in Cleveland, Ohio.

The dreaded lock-up is usually the result of a poor or missing electrical connection somewhere in the computer system. Symptoms can include a frozen cursor, a screen filled with garbage characters, or unusual colors. The number one symptom is that no matter how much you pound on the keyboard, nothing happens.

As we all know, the computer has provided the environment for the creation of another addition to Murphy's Law. It goes something like this — a computer lock-up will occur at the worst time and just after a great amount of work and effort has been expended. (You did remember to save to disk or printer every 20 minutes or so, didn't you?)

However, no matter how traumatic and frustrating a lock-up is, a lot can be learned from the symptoms — what happened and when it happened — and especially from observing the system's light emitting diodes (LED).

Common lock-up times are:

• Using a word processor such as TI-Writer;

• Calling a disk drive (RAMdisks included) to write or read;

• Sending a signal to a printer or modem;

• Just turning on the system.

Most of these lock-ups can be analyzed by observing the LED indicator light for a particular operation. First, let's determine what is expected or not expected from the system's LEDs.

1. The TI99/4A has a read power LED indicator right up front. This LED should be on when the computer is on. Simple,, huh? The same goes for the printer.

2. Those with a Peripheral Expansion Box (PEB) and peripheral expansion cards have more LEDs to watch. Here's what you should see from the LEDs on various cards:

• The white PEB card LED should be lit as long as the PEB is turned on. A slight flickering is normal when the TI is sending or receiving commands to or from the PEB. If the light is not on, probably nothing is working. Of course, we have to assume that the LED itself isn't burned out.

• The 32K memory expansion card has a white LED that is lit all the time. A simple trouble-shooting test is to type "SIZE" and press Enter. If the 32K card is working, you will get the following message on-screen: "24400 Bytes of Program Space Free."

The RS232 card LED should never be lit all the time. This LED flashes or flickers only when the card is sending or receiving commands. A poor connection will cause this LED to glow continuously.

• The floppy disk controller card LED (CorComp) will remain on only when the CorComp main screen is active. Otherwise, this LED should blink only when commands are being sent to or received from the disk controller and disk drives. Regardless of the controller brand, the LED will light only when the controller is interacting with a drive. The LED on the disk drives should glow only when the particular drive is being addressed. (In the case of the CorComp controller card, if the LED doesn't turn off when the main CorComp screen is gone, then the card connection may be faulty.) In all cases, if the disk drive LED is on all the time, the cable connector is probably connected backwards.

• The RS232 card LED should never be lit all the time. This LED flashes or flickers only when the card is sending or receiving commands. A poor connection will cause this LED to glow continuously.

• The RAMdisk LED should never be lit all the time. This LED flashes only when the RAMdisk is being addressed.

Another thing to note on crashes or lock-ups is that this happens only when the computer was trying to do something it couldn't complete. It will continue to try to complete the operation until it is told to stop by either error-trapping or a power shutdown. A computer always works on one command at a time and must finish it before going onto the next. This is why the keyboard becomes inoperative during a lock-up. The computer is still working but it is unable to complete the command.

Generally, the odds are that if the computer isn't lost in an endless loop generated by software, look to missing or poor electrical connections for solutions. Always make sure that contacts and connections are clean and secure and you should be able to go about your business with confidence and little fear of an unexpected system crash.

DISK DRIVES—

(Continued from Page 17)

vices exclusively. SCSI cards are available for PCs and are a close, high-end power-user oriented upgrade for IDE users. Much programming support for the WHT SCSI card is being done, and it is truly the cards of the '90s for the TI and Geneve. Sources for low-cost SCSI peripherals are being sought. Commercial support for SCSI is mostly high-end with drives of less than 150 megabytes seldom advertised.

In addition to hard drives, SCSI can support tape drives, removable cartridge drives, optical or magnetico-optical, "floptical" media, hand scanners, printers, plotters, CD-ROM drives and even floppy disk drives. The WHT SCSI card can even operate in a shared mode, allowing two computers (a TI and a PC, or a TI and a Geneve) to share the same peripherals.

Use MASSREAD to create large text files

By W. LEONARD TAFFS

The following article appeared in the newsletter of the SouthWest Ninety-Niners.

The object of the following program, called MASSREAD, is to create a large D/V80 text file from reading a group of smaller text files. The technique for doing this is so simple that I have wondered why such information has not appeared more often. I have never seen this idea demonstrated in any of my TI books. All that is required is that you open one file (an output file) and then open as many input files as you wish to transfer (copy) to your output file. You just have to remember to close each input file as it finishes copying to the output file. You also have to pay attention to the amount of disk space you have to accommodate the size of the output file you are creating.

The scheme is:

1. Open OUTPUT file.
2. Open INPUT file No. 1.
3. Output INPUT file No. 1 information to the OUTPUT file.
4. Close INPUT file No. 1.

4a. Open INPUT File No. 2.

4b. Output INPUT file No. 2 information to OUTPUT file.

4c. Close INPUT file No. 2.

4d. Repeat steps 2 through 4d for each file you want to append to OUTPUT file.

5. Close OUTPUT file and you're done.

The program can be expanded to as many files as you wish, but keep track of the sizes of the files you are reading — add them up as you go along and keep track of how much disk space you have to work with for your OUTPUT file. Unless you have a quad-density drive or large RAMdisk, your limit will be 1440 sectors for double-sided double-density drives, or 360 sectors for single-sided single-density drives. Of course, you must be sure to substitute the correct names for your INPUT files.

Readers will have to make extensive changes in the drive designations and filenames included in the program listing. Instructions are doing this are included at the end of the program.—Ed.

MASSREAD

```
1 REM [*MASSREAD*] Reads fil
es to create single composit
e file. 11-2-91 by W. Leonar
d Taffs, SW99ers, Tucson, AZ
 !228
2 REM Module Lines 100-190 c
an be repeated for each sepa
rate file to be read. Last m
odule to close OUTPUT file !
219
10 CALL CLEAR :: DISPLAY AT(
7,1):"*MASSREAD* for READING
 Files": :" and Creating Com
posite File" :: DISPLAY AT(1
3,1):"By W. Leonard Taffs, S
W99ers" !046
15 DISPLAY AT(18,1):"WARNING
! OUTPUT FILE OPENED Line 90
" :: INPUT "Press <Enter> wh
en ready....":K$ :: CALL CLE
AR !228
90 ! CALL CLEAR :: OPEN #9:"
DSK7.MP8492A-Z,OUTPUT,DISPLA
Y,VARIABLE 80 !224
100 OPEN #1:"DSKC.MP8492AH",
INPUT !180
110 ON ERROR 190 :: LINPUT #
1:A$ :: CT=CT+1 :: F1=F1+1 !
022
120 DISPLAY AT(3,1):SEG$(A$,
1,8);CT :: DISPLAY AT(5,10):
F1;"  A/B":F1;!142
130 IF EOF(1)THEN 190 !236
140 PRINT #9:A$ !181
150 GOTO 110 !189
190 PRINT #9:A$ :: CLOSE #1
!206
200 OPEN #2:"DSK6.MP8492IZ",
INPUT !194
210 ON ERROR 290 :: LINPUT #
2:A$ :: CT=CT+1 :: F2=F2+1 !
126
220 DISPLAY AT(3,1):SEG$(A$,
1,8);CT :: DISPLAY AT(5,10):
F2;" C/E" !147
230 IF EOF(2)THEN 290 !082
240 PRINT #9:A$ !181
250 GOTO 210 !033
290 PRINT #9:A$ :: CLOSE #2
!207
300 OPEN #3:"DSKB.F/G",INPUT
 !111
310 ON ERROR 900 :: LINPUT #
3:A$ :: CT=CT+1 :: F3=F3+1 !
229
320 DISPLAY AT(3,1):SEG$(A$,
1,8);CT :: DISPLAY AT(5,10):
F3;" F/G" !153
330 IF EOF(3)THEN 900 !183
340 PRINT #9:A$ !181
350 GOTO 310 !134
390 PRINT #9:A$ :: CLOSE #3
!208
900 PRINT "ALL FILES READ" :
: CLOSE #9 :: PRINT "TOTAL C
OUNT WAS: ";CT !013
910 ! OPEN #4:"PIO" :: PRINT
 #4:TAB(5);"ALL FILES READ.
TOTAL CT=";CT: :F1,F2,F3,F4,
F5,F6,F7,F8 :: PRINT #4:TAB(
20);"MP8492A-Z ON DSK7":: CL
OSE #4 !010
```

PROGRAM EXPLANATION

Lines 10-15 are opening screen menu. Line 90 has been REMarked to make sure you avoid opening a file before you know the program is ready to proceed. Once you are "ready to roll," then unREMark (remove the exclamation point) this line.

Lines 100-190 read the INPUT file (filename you enter in line 100) and save this file to the filename you opened in line 90. Line 200 opens the next file to be read and saved. When this is done, line 300 opens the next file and saves that file. In effect, lines 200-290 and 300-390 are repetitions of the "module" of lines 100-190. In the illustration presented here, the program finishes when the third file is finished. Only at that point is the OUTPUT file, which was opened in line 90, closed. Obviously, the OUTPUT file must remain open until you have read all the files you wish to append to your "master file."

Examining the module of lines 100-

(See Page 20)

MASSREAD—

(Continued from Page 19)

190: line 100 opens the file you are reading. 110 LINPUT #1:A$ gets input information. The ON ERROR makes sure the program closes the file after printing the End of File. I have added two counters: CT and Fx. CT appears in every module to give me a total count of lines copies to the OUTPUT file. Fx gives me the line count of individual INPUT files, where "x" is the INPUT file number.

Line 120 can DISPLAY the full string A$ being INPUT or a part of it which, in this case, is SEG$(A$,1,8)) — this limits the amount of clutter in the screen display. Adjust to your own taste. The last DISPLAY AT in lines 120, 220 and 320 was added to give me a display on the screen to let me know the program has moved on to the next INPUT file. What you put in these quotes is up to you.

Line 130 is the usual EOF direction to be sure the INPUT file is closed at the end of file. Line 140 prints the INPUT file string information to the OUTPUT file. Line 150 loops back to input more from the INPUT file. Line 190 makes sure the last line of the INPUT file is saved to the OUTPUT file.

Much has been said about the proper closing of OUTPUT files, so I will not repeat all of it here. But if you have not read it before, failure to close OUTPUT files can result in more problems than just losing the entire file. Lastly, line 910 could be left out, or, you can enter the pertinent information you wish. As you can see, line 910 was left over from a run of eight consecutive files by the F1 through F8 counters.

Finally, if you eventually intend to sort the D/V80 files you are using, you must make sure that non of the lines is longer than 80 characters. If your lines exceed 80 characters, sorting will result in a jumble of any data exceeding the 80-character line length.

All of what MASSREAD accomplishes could also be done using a word processor. However, you would be limited to file sizes that fit in the word processor's tex buffer. An OUTPUT file that exceeds the buffer size, of course, could not be created using TI-Writer, for example. MASSREAD can be very handy for those who do not use Writer files. As a matter of fact, depending upon the circumstances, I have found MASSREAD to be more convenient and faster.

Once the master OUTPUT file has been created, it can be ready by any file-reader. Funnelweb Disk Review is terrific for reading large files. The advantage of Disk Review over something like DM-1000 to read the file is that Disk Review is capable of reading the file forward and backwards. With DM-1000-type readers you must start at the beginning of the file and scroll forward.

What about other types of files? Yes, this program can be modified to read different Display-type files. For example, it should be possible to apply MASSREAD to text files in Fixed format.

EXTENDED BASIC

A tribute to Regena

By DONALD STEFFEN

I am submitting this tribute to Regena. It is her "Farewell" program in BASIC (Novermber 1993) converted to Extended BASIC, as I have done with several of her programs, mostly to save a lot of repetitious typing. I have done this with many other programs written in BASIC and Extended BASIC to shorten them with subprograms and algorhythms.

I use a Star NX-1000 Rainbow printer and use the ~ in the remark lines to separate the comment, data, subprogram and program lines for easier reading.

A music program written in this manner will run at a little slower tempo than when written as Regena has done. If the subprogram PLAY were converted to an assembly language LINKed routine that tempo could be increased. I haven't yet become proficient in assembly language to do it.

The lines 10-24 are just to allow you to compare the DATA to the lines in Regena's program. The data line at 300 directs the program through the various sections and chorus repeats.

FAREWELL

```
5 ! ((C))1 *** DATA SONG ***
   BY B.X.ANEGER ((C))2 !16
0
6 ! THIS IS DONE AS A TRIBUT
E TO REGENA FOR THE MANY GOO
D BASIC PROGRAMS SHE HAS HAD
 PUBLISHED. !080
7 ! THANK YOU,REGENA FROM DO
N STEFFEN. I JUST HAD TO CON
VERT YOUR FAREWELL SONG TO
EXTENDED BASIC. !190
8 ! THE LINES 10-24 SHOW WHE
RE THE DATA CAME FROM. I ADD
ED THE SHORT SUBROUTINES IN
TO THE REGULAR SECTIONS.
 ((C))1 !074
9 GOTO 100 ! ~~~ ~~~~~~~~~~~
  DATA SELECT SECTION ((C))0
 !254
10 ! A 140-240+4020-4060 !06
8
11 ! B 260-690+3950-4000 !08
6
12 ! C 710-880+3850-3930 !09
8
13 ! D 900-1090+4020-4060 !1
27
14 ! E 1110-1310 !189
15 ! F 4080-4110+3050-3450 c
horus 1st half !091
16 ! G 3460-3830 chorus 2nd
half !101
17 ! H 1340-1540+3950-4000 !
183
18 ! I 1560-1730+3850-3930 !
199
19 ! J 1750-1940+4020-4060 !
```

(See Page 21)

FAREWELL—

(Continued from Page 20)

```
189
20 ! K 1960-2160 !212
21 ! L 2190-2320+3950-4000 !
188
22 ! M 2410-2580+3850-3930 !
202
23 ! N 2600-2790+4020-4060 !
192
24 ! O 2810-3010 !206
25 ! end of song !166
90 ! ((C))1  OPEN #1:"PIO" !
 REMOVE ! TO PRINT DATA ERRO
RS !028
100 ! ((C))1 **** FAREWELL T
O THEE FROM B.X.ANEGER
CONVERTED TO XBASIC FROM
REGENA'S FAREWELL IN MICROPE
NDIUM NOV93 ((C))0 !168
110 DIM F(35),W$(99)!070
115 ! GOTO 500 ! REMOVE ! FO
R CHECKING DATA STRING LENGT
H !119
120 CALL CLEAR :: CALL SCREE
N(2):: CALL MAGNIFY(2):: CAL
L SH :: FOR X=1 TO 35 :: F(X
)=103.8262*2^(X/12):: NEXT X
 :: F(0)=4250 !167
130 FOR X=1 TO 97 :: READ W$
(X):: NEXT X :: CALL DELSPRI
TE(ALL)!069
135 RESTORE 300 :: READ M,F$
 :: CALL CLEAR :: CALL SCREE
N(14)!169
140 FOR Y=1 TO M :: Z=ASC(SE
G$(F$,Y,1))-64 :: IF Z=12 OR
 Z=18 THEN RESTORE 380 ELSE
IF Z=14 THEN RESTORE 450 !21
0
150 READ N,D$,V$,S$,A$,B$,C$
 :: CALL PLAY(N,F(),W$(),D$,
V$,S$,A$,B$,C$):: NEXT Y ::
GOTO 135 !005
155 ! ((C))1 ~~~~~~~~~~~~~~
  DATA TABLE FOR WORDS ((C))
2 !172
160 DATA NOW,OUR,GOLD,EN,DAY
S,ARE,AT,AN,END,THE,PART,ING
,HOUR,IS,COM,SOON !086
170 DATA AND,WE,THINK,WHILE,
SWIFT,MO,MENTS,PASS,HOW,DE,L
IGHT,FUL,HAS,BEEN,OUR,FRIEND
,SHIP'S,BOON !111
180 DATA HAVE,FELT,THRILL,AU
T,UMN,SHARED,WIN,TER'S,COLD,
AS,WELL,WHEN,KNOW,MUST,SAY,G
OOD-,BYE !023
190 DATA ALL,SOR,ROW,NO,LANG
,UAGE,E'ER,CAN,TELL,SEEN,TO,
GETH,ER,SPRING,MADE,MIR,A,CL
ES,TREE,FLOW'R !022
200 DATA BUT,JOY,THAT,SUM,ME
R,BRO'T,US,LED,ON,TOWARD,THI
S,PEN,SIVE,OF !224
210 DATA FARE,WELL,THEE,BUT,
WILL,HOPE,FOR,BRIGHT,WHEN,SH
ALL,MEET,WITH !080
290 ! ((C))1 ~~~~~~~~~~~~~~~
    DATA TO SELECT VERSES T
O PLAY ((C))2 !224
300 DATA 19,ABCDEFGHIJKFGLMN
OFG !167
305 ! ((C))1 ~~~~~~~~~~~~~~~
    DATA FOR VERSES TO PLAY
  ((C))2 !132
310 DATA 16,2222222222222222
,3333322222333333,TTTTTTY]]]
[YXYVT,QQQQQQQTTTUVUVRQ,HJLO
MQQAHMQFJMRA,"
  " !166
320 DATA 41,3143143122224222
24222242222222211222222222,11
21231215322443422532222222233
33333333332 !045
325 DATA YXV[YX][AAQTYAATY]A
AQTYTY]]][YYXYVTTTTTT],TTRVV
OXRAHHQTAHHTYAHHQTQQTTTUVVUV
RQQQQQQY !144
330 DATA EEFCCHHHAAAHQAAHHTA
AAHQQQAHMQFJJMAAHJLMJH,"
                !""#  $% &
'()       " !239
340 DATA 19,2222222224222422
422,2222212224432333332,][[[
Z[[^][OX^bTX]`T,YXXXWXXVXROO
X^TTX]Q,AHHLHHHOHHOOOXTTTXQ,
"* +  ,-./,0        " !060
360 DATA 16,2222222222211222
,3333322233333333,TTTTTY]]][
YYXYVT,QQQQQQTTTUVVUVRQ,HJLO
MQAHMQFJJMRA,"    123 45 *67
8 " !087
370 DATA 11,22422222112,1121
2321122,YXV[YXX]][Y,TTRVVOOX
XRQ,EEFCCHHHHHM,"9:;<=>?@AB
" !044
380 DATA 27,2224444524452312
22222222224,2222222222223333
32323222222,YYYYTVY^VTY]YXVX
Y[[^^]]]]YY,TROQQRVVRQHTTOOO
ORRXXTTTTQQ !228
390 DATA KJIHMFFFJMHAEHHHJHH
OOAHJLOM,"     vw^xvw^x?#$%&
/,^()    " !065
400 DATA 21,2445222223242222
22444,333332323233332222222,
TVY^VTTYY]YXXY][[XYYY,QRVVRQ
QQQTTOOVXXXRQRQ !100
401 DATA MFFFJMMHHAEHHHHRHHA
AA," h2z{|}`%^~@ @   " !
232
410 DATA 18,2222222112222222
22,22223333333333332,TY]]][
YYXYVTTTTTT],QQTTTUVVUVRQQQQ
QQY,QQAHMQFJJMRAHJLMJH,"2CD
```

(See Page 22)

1994 TI FAIRS

FEBRUARY

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

MAY

Lima Multi User Group Conference, May 13-14, Ohio State University Lima Campus, Lima, Ohio. Contact Lima Ohio Users Group, P.O. Box 647, Venedocia, OH 45894.

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

FAREWELL—

(Continued from Page 21)

```
 *E FG%        " !135
420 DATA 19,2222222224222422
422,2222212222444444442,]]][[
Z[[^][OX^bTX]`T,YXXXWXXVXROO
X^TTX]Q,AHHLHHHOHHOOOXTTTXQ,
"1 H  *IJKLM        " !245
430 DATA 16,2222221122222222
,2222333333333333,Y]]][YYXYV
TTTTTT,QTTTUVVUVRQQQQQQ,QAHM
QFJJMRAHJLOM,"N2O  2! PQRS
  " !240
440 DATA 11,22422222112,1121
2321122,YXV[YXX]][Y,TTRVVOOX
XRQ,EEFCCHHHHHM,"T?UVWXYZ [\
" !028
450 DATA 18,2222222112222222
22,222223333333333332,TY]]][
YYXYVTTTTTT],QQTTTUVVUVRQQQQ
QQY,QQAHMQFJJMRAHJLMJH,"2C]
 ^_`9*a       " !064
460 DATA 19,2222222224222422
422,2222212222444444442,]]][[
Z[[^][OX^bTX]`T,YXXXWXXVXROX
X^TTX]Q,AHHLHHHOHHOOOXTTTXQ,
"b c  deuflg        " !229
470 DATA 16,2222221122222222
,2222333333333333,Y]]][YYXYV
TTTTTT,QTTTUVVUVRQQQQQQ,QAHM
QFJJMRAHJLOM,"h*i  jklm^n
  " !253
480 DATA 11,22422222112,1121
2321122,YXV[YXX]][Y,TTRVVUUX
XRQ,EEFCCHHHHHM,"onpqrst +,-
" !108
500 ! ((C))1 ~~~~~~~~~~~~~
    CHECK DATA FOR PROPER
STRING LENGTH ((C))0 !228
510 FOR X=1 TO 15 :: READ M
!227
520 FOR Y=1 TO 6 :: READ A$
:: L=LEN(A$):: IF M<>L THEN
PRINT #1:"DATA LENGTH WRONG
IN ";CHR$(X+64);Y:M;L;A$ !22
0
530 NEXT Y :: NEXT X :: STOP
 !121
540 ! ((C))1 IN THE SUBPROGR
AM PLAY THE VARIABLES ARE:
N=NUMBER OF NOTES IN VERSE
E$=DURATION CODES FOR VERSE
V$=VOLUME CODES FOR VERSE !1
46
550 ! S$=SOPRANO NOTE CODES
      A$=ALTO NOTE CODES
      B$=BASS NOTE CODES
      C$=WORD CODES FOR SONG
 !239
560 ! THE F() ARRAY HOLDS TH
E FREQUENCY VALUES FOR NOTES
  THE W$() ARRAY HAS ALL THE
 WORDS OF THE SONG IN IT. !2
31
570 ! BY USING STRINGS WITH
SINGLE CHARACTER CODES THE
SUBPROGRAM CAN PROCESS AN EN
TIRE VERSE AT A TIME. !233
1000 !((C))1 ### SUBPROGRAMS
 ### ((C))3 !088
1010 SUB SH :: H$=" FAREWELL
 TO THEE REGENA" :: FOR X=1
TO 24 :: CALL SPRITE(#X,ASC(
SEG$(H$,X,1)),RND*12+3,X*4+8
0,240,0,-12)!031
1020 CALL SOUND(300,X*110,X)
:: NEXT X :: SUBEND !142
1030 SUB PLAY(N,F(),W$(),D$,
V$,S$,A$,B$,C$):: T=240 :: K
=32 :: @=1 :: P=2 :: FOR X=@
 TO N !051
1040 D=VAL(SEG$(D$,X,@)):: V
=VAL(SEG$(V$,X,@)):: S=ASC(S
EG$(S$,X,@))-K*P :: A=ASC(SE
G$(A$,X,@))-K*P :: B=ASC(SEG
$(B$,X,@))-K*P :: C=ASC(SEG$
(C$,X,@))-K !237
1050 PRINT W$(C);" ";:: CALL
 SOUND(T*D,F(S),V,F(A),V*P,F
(B),V*P*P):: NEXT X :: SUBEN
D !074
```

Asgard Software discontinues program

Asgard Software has officially discontinued selling the program First Draft/Final Copy as of Feb. 18, according to Harry Brashear of the company.

Brashear says, "I deeply regret this decision, but after repeated attempts to get response from the author on two major bugs since June of 1993 and not getting any action on the problem, we are giving up. Though many people have found ways around the errors, my policy is that I will not knowingly sell a product with bugs."

He requests that anyone with questions regarding the program contact the author, "whose name and address appears on the first screen of the program." Previous advertising and announcements Asgard list the author as Arthur Gibson.

Brashear says, "I am taking steps to insure that this never happens again to any product I sell."

For information on other Asgard Software products, contact the company at 2753 Main St., Newfane, NY 14108, (716) 778-9104.

READER TO READER

❑ Vern Jensen, 817 Kingway Dr. E., Gretna, LA 70056, writes:

In the User Notes of the December 1993 issue, John Wineke said that we should use a BASIC compiler for file handling programs. Where can you get this compiler, and are there others out there? Is the compiler limited to file handling routines, or can it compile games, too?

❑ Olden Warren, 4016 Weber Way, Lexington, KY 40514, (606) 223-4599, writes:

I am trying to get a set of SSSD disk drives to work with my TI. The drives came out of a Digital 100+ computer that I bought and I have only been able to get them to respond to a file directory command with mixed results. Does anyone know if the cabling is "standard" or should some of the connections be "twisted" like some of the PCs?

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.

NEWSBYTES

Fest West '95 set for San Diego

The Southern California Computer Group will host Fest West in 1995, according to James D. Lanman. The fair is scheduled for the Presidents' Day weekend. Vendor tables will be free, Lanman says.

For further information, contact the Southern California Computer Group, P.O. Box 152535, San Diego, CA 92195.

AMS schematics now public domain

Asgard Peripherals has released the entire AMS (Asgard Memory System) schematics to the public domain as of Feb. 19, according to Jim Krych, Asgard Peripherals' director of research and development. The company has released the entire theory of operation and design features as well, he says.

Krych says that more than 10,000 manhours and more than $20,000 went into the project. He says the design team hopes that the release of the schematics to public domain will generate more interest in the AMS.

He says hardware experts "can easily figure out how to access the one-megabyte range of the AMS." The developmental software will take immediate advantage of all memory from 128K to 1024K, he says.

For further information, contact Asgard Peripherals, 1423 Flagship Dr., Woodbridge, VA 22192.

Bluegrass 99ers seek fair help

The Bluegrass 99 Computer Society in Lexington, Kentucky, is "seriously considering" hosting a TI fair in 1995, according to club president Olden Warren.

The group wants to establish contacts with groups who have successfully hosted fairs to share ideas on how to "pull them together," Warren says.

Warren says, "Although several of us have visited fairs in the past, none of us has any experience in hosting one. We are inexperienced but are eager to do our part in helping keep the TI alive!"

Persons who have information to share may call or write Warren at 4016 Weber Way, Lexington, KY 40514, (606) 223-4599; Dallas Phillips, Rt. 4, Box 310, La Grange, KY 40031; or the Bluegrass 99 Computer Society, P.O. Box 1237, Lexington, KY 40590-1237.

Dee Turner has new address

Dee Turner, distributor for The Printer's Apprentice by McCann Software, has a new address. He is at 641 S. Laughlin Lane, Merced, CA 95340. Phone is (209) 722-7352 (home); (209) 722-2053 (work).

Send your announcements about products and services of TI/Geneve users to MICROpendium Newsbytes, P.O. Box 1343, Round Rock, TX 78680.

New TI emulator software runs on 386

Looking for a program that will run some of your TI programs on a PC? A relatively slow PC?

Edward Swartz, of Georgetown, Texas, has written a TI emulator that runs on 386 PCs (SX or DX), requiring only an EGA monitor and 256K of free RAM. The program is shareware and is offered for $25. It's available on a number of electronic bulletin boards. The program runs on a PC and should be downloaded using a PC.

The program is claimed to support all standard TI graphics modes as well as TI key combinations using a 101-key keyboard. It also touts Adlib support for three-voice music. It also supports the TI disk system, including DSK1 through DSK3 and a hard disk. It also emulates a 32K memory expansion.

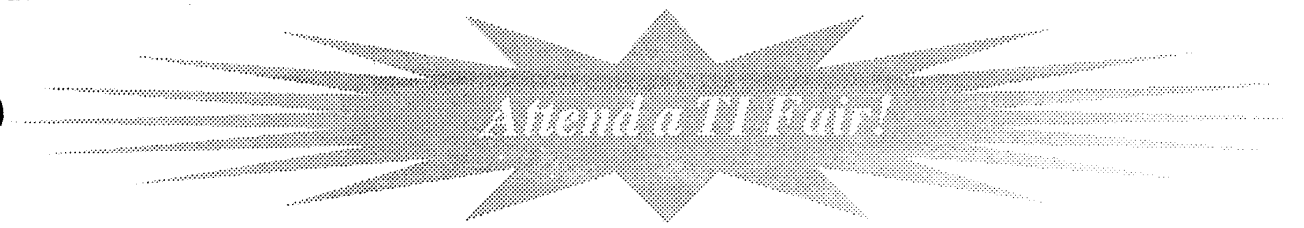
The program, called TI Emulator, doesn't include support for TI speech, printer, joystick or cassette routines. However, support for a mouse is included. It also does not support low-level routines such as sector read/write, formatting, etc.

According to the author, "TI Emulator is meant as a diversion for anyone who ever used the actual TI computer."

The emulator includes several demonstration modules. Registered users receive additional modules and information about transferring data from a TI99/4A to the TI Emulator.

Source code is also available for $50. It is written in Turbo Assembler 2.0

For more information, contact Swartz, 1401 E. 18th St., Georgetown, TX 78626. Expect a wait of 4-6 weeks or more. He can also be reached through Internet at swartze@ralph.txswu.edu.



MICRO-REVIEWS

Who's Behind the Mexican UFOs?, DIGISYNT

By CHARLES GOOD

I'm really excited about both products I am reviewing this month. If you want to excite me with your 99/4A product, send it to me for a MicroReview to P.O. Box 647, Venedocia OH 45894.

WHO'S BEHIND THE MEXICAN UFO'S by Bodenmiller Computers

Reviews of game software are, to a large extent, subjective. A favorite game of one individual may not be at all interesting to someone else. With this in mind let me state, "Mexican UFOs is the most enjoyable most fun adventure game I have ever played on my 99/4A." At least one of my kids (eight-year-old Meaghan) agrees.

The game pushes the basic 99/4A disk system to the absolute limits of its graphic capabilities in the Extended BASIC environment. UFOs reminds me of many of the graphic adventures my kids play on our IBM clone, particularly the "Where in the ... is Carmen Sandiego" series. In such PC games you see a full screen color picture. A text window opens up within the picture to provide the player with information and accept player input. At appropriate intervals an animated person or character moves across the screen.

Mexican UFOs is written in Extended BASIC (it is completely LISTable) and uses the advanced graphics capabilities of T.M.L. (The Missing Link). The results are comparable graphically to many of the adventure games my kids play on our clone. You need to own T.M.L. in order to make UFO'S work. I purchased T.M.L. several years ago and never found much use for it. Existing T.M.L. applications have not, so far, interested me. Now that UFOs is available I am glad I own T.M.L. Before you play for the first time, you have to install T.M.L. onto the first UFO disk. Once installed you can then put your T.M.L. disk away because UFO'S will then always boot like a regular Extended BASIC program as LOAD from DSK1.

You are a United States secret agent assigned to find out about UFOs shaped like Mexican sombreros appearing all over the world. Usually when one of these UFOs appears, some expensive electronic equipment or military documents get stolen from a nearby location. You get on an airplane and fly to various cities all over the world (Athens, New York, Paris, Cairo, etc.). At each location you usually arrive just in time to catch sight of one of the UFO's. Then you go snooping around looking for clues.

An airplane sprite takes you to each location. When you arrive, a TI-Artist picture is displayed. Then a text window opens within the picture. Any of several sprites may float across the screen at any time. On a 40-column 99/4A system this simultaneous combination of a TIA picture, text window and sprites is possible only with T.M.L. enhanced Extended BASIC. Most of the time you just read the text, pressing "any key" to read the next text segment. Sometimes the text window gives you a choice of 2 or 3 ways to proceed, each of which will yield different results.

If you like adventure games with complex puzzles and scenarios, then UFOs is not for you because it is *easy* to solve. When given a selection of choices, you always eventually proceed to the next part of the game. My eight-year-old had the thing finished in 20 minutes. Playing the game again may lead to the end point via a different route, but any random selection of choices will eventually lead you to the White House to be congratulated by the President. The game's measure of success is how few turns it takes you to solve the mystery. I like such easily solvable adventures. I am a real bimbo when it comes to adventure games and am intimidated by their complexity. After all, I have a PhD so I am supposed to be smart! And yet I must confess that I have not been able to finish any of the Scott Adams adventures without first looking up lots of stuff in the cheat book. I appreciate an adventure game that is both entertaining and logically simple enough that I and my children can solve it.

UFOs is full of original colorful artwork. The title screen and accompanying music are stunning. I sometimes let this title music play for several minutes before restarting a previously saved game or starting another game to try and beat my previous "fewest turns" score. This original music uses all the three-sound channel plus one-noise channel tricks possible in the XB environment.

UFOs comes on two DSSD disks and is designed to play out of DSK1. It is a very large game, with successive segments automatically loading themselves into memory at the appropriate times. You can't play this game with only single sided drives. You can combine both disks onto one DSDD disk which must be named MEXICO2. If you put all the files on a RAMdisk you have to name the RAMdisk MEXICO2 and you either have to make the RAMdisk think it is DSK1. or you have to modify lots of references to "DSK1." in the XB game code to some other drive number.

UFOs is commercial software available only from Ramcharged Computers for $12.95. T.M.L. is available from several dealers, including Ramcharged, for $24.95. Ramcharged will give you a little off these prices if you purchase both products at the same time. Their shipping charge is $4 per order (buy one or both and pay the $4) which you can save if you make your purchase from Ramcharged at the May 14 Lima MUG Conference. Phone Ramcharged at 800-669-1214 or write the company at P.O. Box 81532, Cleveland OH 44181.

DIGISYNT by Stefano Bonomi

Every now and then something new and truly remarkable comes along for our favorite computer, an "I had no idea you could do that with the TI" type of product.

(See Page 25)

MICRO-REVIEWS —

(Continued from Page 24)

The last time I ran into one of these was several years ago when I found CS1*FINDEX, which allows one using only a console and cassette player to load automatically from a menu stored on a cassette tape any TI BASIC or XB software stored anywhere on the tape. Almost all of us now have disk systems, so few 99ers use CS1*FINDEX any more. But don't throw away that cassette cable and tape recorder yet! Dust off that cable, and cassette program recorder, hook them back up to your TI console and let the fun begin.

DIGISYNT is simple and fun to use, it is potentially very useful, and it is very low in cost. It is an all software one-channel sound digitizer that uses the TI's cassette port for sound input. Other than a cassette recorder, TI cassette cable and 99/4A console, all you need is a SSSD disk system with 32K. The software sucks any sound out of the tape recorder into the TI's memory expansion. These digitized sound segments can immediately be played back (without the tape recorder) through the monitor speaker, and can be saved to a disk file. Disk files are in DV80 format and full of control characters (the ASCII 0-31 characters you get with TI-Writer when you invoke "special character mode" by typing CTRL/U). These sound segments can later be loaded back into DIGISYNT for playback. But more importantly, they also can be programmed into any TI Basic or Extended Basic application.

What can you digitize? 1-Anything that comes on a cassette tape including music and voice. Just press "Play" on the recorder, and when a sound segment you want to record comes along enter "1" (Digitize) from the DIGISYNT main menu. The flashing cursor disappears indicating that sound input is being digitized. When the sound buffer is full the flashing cursor reappears. 2-Your own voice or any other sounds in real time, directly through the cassette recorder's microphone or through an external microphone attached to the cassette recorder. To do this you need a cassette recorder (such as the TI Program Recorder) whose earphone jack is not muted when you are recording. Disconnect from the recorder the red "mic" jack of the TI cassette cable and insert a tape on which you can record. Press "Record" on the cassette recorder. When you are ready to speak, enter "1" from the DIGISYNT main menu and begin talking into the cassette recorder's microphone. Your voice will be digitized directly by the computer and simultaneously recorded onto the tape. You have to use a tape because you can't press "Record" without a tape in the recorder. Please note that you do not need a speech synthesizer to digitize and play back voice or music. DIGISYNT is apparently the technology used in creating the "speech synthesizerless speech" found in the Goblin game I reviewed a few months ago and in the game Perfect Push released several years ago.

The digitized sound is not high quality. This is the same one-channel sound made by some PC games (eg. "McGee" and "Kate's Farm") which produce speech through the PC speaker rather than through a sound blaster type of multi channel sound board. Music synthesized by DIGISYNT sounds as if it were coming from a fuzzy AM radio broadcast, but melodies are recognizable. Speech is understandable if you speak clearly. It helps to experiment with the volume and tone controls of the tape recorder.

The duration of a recorded sound segment depends on the size of the sound buffer, the complexity of the sound, and the "delay" you select. You can modify the size of the memory expansion digitizing buffer from within DIGISYNT, allowing you to store sound in both high and low memory areas if you want. There is no provision for using other memory outside of the standard 32K memory expansion area (but you might be able to bank switch using an Asgard AMS card). The default is B000-E000 which can be adjusted in order not to interfere with the execution of TI BASIC or XB programs you might want to run with digitized music and voice. Silent time does not occupy much buffer space. Using the default buffer size, I can record my voice counting off 30 seconds before the buffer is full. I only get about 15 seconds of recorded music in the same size buffer. Such "default buffer size" sound files occupy about 50 sectors of disk space. You can increase the digitizing time duration by selecting "3" from DIGISYNT's main menu. You can then increase the delay value (decrease the program's scan speed). The result is similar to reducing the speed of an audiotape during a recording session. You get more time to record, but the sound quality is poorer. You can also record a sound at normal speed and then increase the delay before you play the sound back. This results in deep slow voices and slows down the tempo and lowers the octave of recorded music.

Utilities included allow you to CALL LOAD and then CALL LINK to digitized sound from within a BASIC program. You need the EA module to do this with TI BASIC software. You can repeatedly play a loaded sound segment with your choice of volume, and you can later load in one at a time other sound segments and play them, all from within a running TIB or XB program. The possibilities here seem endless. How about a verbal message every time an ON ERROR is activated such as, "You forgot to put the data disk in the drive." You can add music and voice to your custom applications. Have the computer say, "Charlie's checking account" when you activate your check balancing program. Verbal game instructions would be both useful and cute.

The DIGISYNT disk comes with a well written sample XB game called "Submarine" (shoot torpedoes at passing surface ships) illustrating the use of digitized sounds and speech in a game format. You have to listen carefully to the words because they have an Italian accent. You hear a very distinct "Game over" at the end of the game. The disk also comes with a variety of sample sound files you can load and play, including sounds of various animals and some disco music.

This is unique software. Only DIGISYNT allows you to create your own digitized sound segments on an all 99/4A system and incorporate them into your own software. DIGIPORT (Horizon) and SOUND F/X (Texaments) allow you to play digitized sound files created on other types of computers and downloaded from

(See Page 26)

MICRO-REVIEWS —

(Continued from Page 25)

bulletin boards, but you can't create your own sound files on a TI system and you can't easily add the downloaded sounds to software. If all you want to do is play short sound segments just to hear them, why not use the cassette deck of a stereo music system? MIDIMASTER 99 (Cecure) lets you create your own very high resolution music, but only if you have an expensive MIDI keyboard. I don't think you can easily add the resulting MIDI files to TI software. Three of my children (ages 8, 13, and 15) and I spent a couple of hours together yesterday evening playing with DIGISYNT. We quickly filled a DSDD disk with sound files of our voices and favorite recorded music. It was fun!

DIGISYNT is fairware. The author asks users to "send what it is worth to you." I think it is worth a lot. I will send it to anyone in the United States and Canada who sends me a $1 bill to pay for the SSSD disk and return postage. Elsewhere in the world send a disk and paid return mailer directly to the author: Stefano Bonomi, Via Sacchi 21, 37124 Verona, Italy.

Putting four (or more) Quest RAMdisks in a P-Box

By BOB CARMANY

In the January issue of MICROpendium, I wrote a letter about the hardware and software project of putting four (or more) 512K Quest RAMdisks in a single P-box. Some of you may have missed the letter, so I'll start at the beginning.

The Quest RAMdisk is an Australian product first introduced by the Hunter Valley Users Group about five years ago. It is user-expandable from 128 sectors (32K) to 2,048 sectors (512K). Two PAL chips with CRU addresses of >1000, >1400 and >1600 control the CRU addressing. The user was limited to three Quests in a single P-box at those fixed addresses. That was never a problem, unless you had a Dijit 80-column card that was also addressed at a fixed CRU address of >1400. If you had one of those (or some other similarly addressed device) you were effectively limited to two possible Quest cards.

That wasn't my problem. In my case, it was simple greed! Over the years, I had acquired three Quests — two purchases and one as a gift. At Christmas time, a fourth arrived from one of my Aussie "mates" as a present. Since one of the original three was a prototype, I had envisioned replacing it with the newest arrival and using the prototype as a backup. Then, I heard of another Quest card doming on the market. It was too much simply to pass up, but the problem was, under current conditions, I could only use three of them at a time. So here I was, with the prospect of five Quest cards, only able to use three of them. An interesting dilemma!

Fortunately, I had also acquired all the requisite materials I needed to alter the Quest environment. I had copies of the PAL code and the notes to go with them. Some years ago, Ron Kleinschafer had sent along his commented source code for the Quest manager program. With all of this material in hand, I embarked on the "Quest Project" — putting more than three Quests in a P-box.

An inspection of the PAL0 output revealed that the CRU addresses were controlled by the outputs from PAL0 pins 16, 17 and 18. They set the addresses at >1000, >1400 and >1600 respectively. A jumper block selected the actual CRU address on the card. It stood to reason that if those outputs were altered, this would alter the CRU addresses. The CRU bits would have to be changed and a new PAL0 burned. My technical manuals indicated that CRU bits A0-A2 were the same for all addresses and just CRU bits A3-A7 needed changing The abbreviated table shows the difference.

CRUAddress	A3	A4	A5	A6	A7	Use
1000-10FE	1	0	0	0	0	Unassigned
1400-14FE	1	0	1	0	0	Unassigned
1600-16FE	1	0	1	1	0	Unassigned
			New Addresses			
1800-18FE	1	1	0	0	0	Thermal Printer
1C00-1CFE	1	1	1	0	0	Video Controller
1E00-1EFE	1	1	1	1	0	Unassigned

Why were these particular new addresses chosen? In going through the commented source code, I never found it clear just how the multiple Quest cards were polled and checked. I reasoned that it might be best to use the same CRU address intervals used in the original in the new set of addresses. I also wanted to avoid potential conflicts with other devices in common use — like my HV99 eprommer at >1900.

Don Walden of Cecure Electronics got the task of rewriting the Boolean equations to reflect the changes in the CRU bits and, since he had the facilities, burning a couple of prototype chips. He had the chips back to me in a less than a week, and I didn't have to take out a second mortgage on the house, either!

In the meantime, I had been going over the commented source code and making wholesale changes. *Wrong!* As I analyzed the code more closely, I found that there were minimal changes to be made. I had to change the list of possible Quest addresses to reflect the new ones and change the limits for possible disk drive numbers as well. Then, there were the cosmetic changes of on-screen messages and other details. I did add a few lines of ASCII letters for display purposes (e.g., ASCF TEXT 'F') as well. I think the number of substantive changes came to about a dozen lines of the code.

The source code was finished the same day that the prototype chips came. It was time to test everything. I was relatively certain that the code was correct, but you never know for sure until you try it out. The new chips worked great! The Quest jumper block selected the correct CRU address and everything appeared to be working fine. Fine, that is, until I tried to load the DSR into the Quest card. I kept getting an 'INVALID DSR VERSION' mes-

(See Page 27)

USER NOTES

PRE-SETTER uses software to change printer settings

The following program, called PRE-SETTER, is designed to be used with Epson-compatible printers. The program lets the user select a variety of printer options, ranging from typestyles to page length. Several commands may be selected at the same time. It requires Extended BASIC.

Users with non-Epson compatible may be able to adapt the program to their printers by modifying printer command codes in lines 390-550.

```
100 CALL CLEAR :: DIM SET$(2
0):: OPEN #1:"PIO" !005
110 DATA -,X,X,-,-,X,-,X,-,X
,-,X,-,X,-,-,-,-,- !152
120 FOR X=1 TO 19 :: READ SE
T$(X):: NEXT X !223
130 DATA "CONDENSED ON","CON
DENSED OFF" !203
140 DATA "11"" PAGE LENGTH",
8.5"" PAGE LENGTH" !114
150 DATA "DOUBLE-STRIKE ON"
,"DOUBLE-STRIKE OFF" !066
160 DATA "EMPHASIZED ON","E
MPHASIZED OFF" !156
170 DATA "ULTRA-DENSITY ON"
,"ULTRA-DENSITY OFF" !120
180 DATA "ITALICS ON","ITAL
ICS OFF" !212
190 DATA "ENLARGED ON","ENL
ARGED OFF" !072
200 DATA "DISABLE PAPER OUT"
 !015
210 DATA "RESET TO DEFAULTS"
 !041
220 DATA "SKIP PERFORATION"
!010
230 DATA "EXIT TO BASIC" !23
0
240 DATA "HELP" !135
250 DISPLAY AT(2,1)ERASE ALL
:"--------------------------
--":"          E P S O N
    " !003
260 DISPLAY AT(5,1):"     P
R E - S E T T E R    ":"----
------------------------" !2
41
270 DISPLAY AT(11,1):"   Thi
s program allows you  ":"  t
o set any Epson (or com- ":"
  patible) printer to turn
":"  on various features." !
047
280 DISPLAY AT(21,1):" NOTE:
 The printer must be  ":"
turned on, and in the     ":"
      ONLINE state.
" !000
290 FOR TIME=1 TO 1000 :: NE
XT TIME !248
300 FLAG=0 :: DISPLAY AT(1,2
)ERASE ALL:"*** EPSON MODE S
ET UP ***" !065
310 DISPLAY AT(3,7):"SET  AC
TION" !031
320 RESTORE 130 !223
330 FOR X=1 TO 9 :: READ C$
:: DISPLAY AT(X+3,2):X;"----
---";C$ :: NEXT X !104
340 FOR X=10 TO 19 :: READ C
$ :: DISPLAY AT(X+3,1):X;"--
-----";C$ :: NEXT X !202
350 FOR X=1 TO 19 :: CALL HC
HAR(X+3,10,ASC(SET$(X)),1)::
 NEXT X !033
360 DISPLAY AT(24,1):"ENTER
CHOICE OF MODES" !003
370 ACCEPT AT(24,23):ANS ::
IF ANS>19 OR ANS<1 THEN 370
!199
380 ON ANS GOSUB 390,400,410
,420,430,440,450,460,470,480
,490,500,510,520,530,540,550
,560,570 :: IF FLAG=1 THEN 3
00 ELSE 350 !101
390 PRINT #1:CHR$(15):: SET$
(ANS)="X" :: SET$(ANS+1)="-"
 :: RETURN !211
400 PRINT #1:CHR$(18):: SET$
(ANS)="X" :: SET$(ANS-1)="-"
 :: RETURN !215
410 PRINT #1:CHR$(27)&"C"&CH
R$(0)&CHR$(11):: SET$(ANS)="
X" :: SET$(ANS+1)="-" :: RET
URN !180
420 PRINT #1:CHR$(27)&"C"&CH
R$(51):: SET$(ANS)="X" :: SE
T$(ANS-1)="-" :: RETURN !197
430 PRINT #1:CHR$(27)&"G" ::
 SET$(ANS)="X" :: SET$(ANS+1
)="-" :: RETURN !157
440 PRINT #1:CHR$(27)&"H" ::
 SET$(ANS)="X" :: SET$(ANS-1
)="-" :: RETURN !159
450 PRINT #1:CHR$(27)&"E" ::
 SET$(ANS)="X" :: SET$(ANS+1
)="-" :: RETURN !155
```

(See Page 28)

FOUR OR MORE QUEST RAMDISKS IN A P-BOX —

(Continued from Page 26)

sage. It had me stumped for a day or so. Finally, I looked at the DSR version in the DSR code itself with a sector editor. The Quest manager version had changed from AA07 to AA08 since I got the original source. I'm not sure if I made the change when I assembled the lot a couple of years back or if Ron Kleinschafer did it-somewhere along the line. Anyway, it is an easy disk patch. Just to be on the safe side, I changed it in the source and reassembled the lot. That minor little "bump in the road" was the only problem I have come across.

I tried and abandoned another idea I had for allowing each Quest to be used as a single 512K RAMdisk. It worked, but I never could get any of the disk manager programs to give a consistently accurate sector count when trying to read the 2,048-sector Quest drive. I settled for the 1,600-sector limit and partitioned each of my Quests into dual drives. The only problem is finding enough material to fill up 8,192 sectors of RAMdisk. I think I can manage, though!

Incidentally, Don Walden will produce a replacement PAL0 for a reasonable price (he has the altered code) and if you want a modified version of the Quest manager program, write me at the following address: Robert M. Carmany, 1504 Larson St., Greensboro, NC 27407. Please include a self-addressed, paid mailer and disk and a $5 copying fee. The source code is *not* included.

USER NOTES

(Continued from Page 27)

```
460 PRINT #1:CHR$(27)&"F" ::
 SET$(ANS)="X" :: SET$(ANS-1
)="-" :: RETURN !157
470 PRINT #1:CHR$(27)&"G"&CH
R$(27)&"E" :: SET$(ANS)="X"
:: SET$(ANS+1)="-" :: RETURN
 !144
480 PRINT #1:CHR$(27)&"H"&CH
R$(27)&"F" :: SET$(ANS)="X"
:: SET$(ANS-1)="-" :: RETURN
 !147
490 PRINT #1:CHR$(27)&"4" ::
 SET$(ANS)="X" :: SET$(ANS+1
)="-" :: RETURN !138
500 PRINT #1:CHR$(27)&"5" ::
 SET$(ANS)="X" :: SET$(ANS-1
)="-" :: RETURN !140
510 PRINT #1:CHR$(27)&"W"&CH
R$(1):: SET$(ANS)="X" :: SET
$(ANS+1)="-" :: RETURN !162
520 PRINT #1:CHR$(27)&"W"&CH
R$(0):: SET$(ANS)="X" :: SET
$(ANS-1)="-" :: RETURN !162
530 PRINT #1:CHR$(27)&"8" ::
 SET$(ANS)="X" :: RETURN !06
3
540 PRINT #1:CHR$(27)&"@" ::
 RESTORE 110 :: FOR X=1 TO 1
9 :: READ SET$(X):: NEXT X :
: RETURN !054
550 PRINT #1:CHR$(27)&"N"&CH
R$(12):: SET$(ANS)="X" :: RE
TURN !125
560 CALL CLEAR :: END !222
570 CALL CLEAR !209
580 PRINT "    MOST OF THE P
RINTER":"   COMMANDS CAN BE
MIXED." !220
590 PRINT :"EXAMPLE:" !102
600 PRINT :"YOU CAN ENTER 5
FOR DOUBLE":"STRIKE AND 13 F
OR DOUBLE" !181
610 PRINT "WIDTH TO GET A DO
UBLE-STRIKE":"DOUBLE-WIDTH P
RINTOUT. EACH" !000
620 PRINT "NUMBER MUST BE EN
TERED":"SEPARATELY." !044
630 PRINT :"EXCEPTIONS:" !09
5
640 PRINT :"YOU CAN'T MIX CO
MPRESSED":"MODE WITH: DOUBLE
 STRIKE," !006
650 PRINT "EMPHASIZED, OR UL
TRA-DENSITY" !075
660 PRINT : :"press any key
to continue" !108
670 CALL KEY(0,K,S):: IF S=0
 THEN 670 ELSE CALL CLEAR ::
 FLAG=1 :: RETURN !148
```

SYSTEM OF THE MONTH

Gilbertson's system offers lots of storage and memory support

This month's System of the Month belongs to Rich Gilbertson, known to TIers for his RGKXB software.

His equipment list includes 3 Teac 3.5-inch drives, 1 Teach 5.25-inch drive, a Super Space II 32K cartridge, S.O.B. board, Toshiba 321 24-pin printer, Horizon mouse, Hayes 2400 baud modem, Commodore 1084S monitor, TIM card with 192K 9958 VDP memory, GRAMulator with 104K GRAM and 16K RAM, AMS 128K card, PGRAM with 160K GRAM and 16K RAM, Horizon 4000 1.5-megabyte RAMdisk, CorComp disk controller, CorComp RS232 card and a Western Horizon Technologies SCSI card. The console has been modified with four switches that give Gilbertson direct control of CPU interrupt, Idle CPU and CPU reset. A fourth switch controls the turbo crystal he installed.

Gilbertson also has an extensive collection of books, including TI Extended BASIC source code ROM/GROM, GPL Disk Interface manual, TI Training Manual (99/4A and 994X), 990/2 Debugger manual, user manual, GPL manual and Fortran compiler manual. He also has the GPL E/A cartridge source code.

Telling TI-Base where to load from

The following was written by Bill Gaskill of Grand Junction, Colorado:

This tip is useful for telling TI-Base where to load from in those cases where it can't find a disk drive number. When this happens, TIB will prompt you to press the number key for the drive you wish to read, and it will then load. The problem is, you have to do this every time you load the TI-Base program. This type of situation is especially prevalent when you have a Horizon RAMdisk configured as multiple drives. Here's the way to fix the program.

(See Page 29)

USER NOTES

(Continued from Page 28)

TI-Base comes with a program named @PATH which is the first file on the 3.02 program disk. @PATH was initially designed to support hard disk pathing, but it also works perfectly well for those HRD drives like 5-9 and A-Z TI-Base also comes with several "Loaders" on the program disk, namely LOAD, TIBASE, TIBASEB, TIBASEP and TIBASEW.

Normally, the TIBASEP (E/A 5 loader) would be used to load TI-Base from a RAM disk, but to cure the "constant prompting for a drive number" problem, use TIBASEW.

Set your HRD Menu selection to the location of TI-Base as follows:
DSK5.TIBASEW

This assumes you have TI-Base on DSK5. Next, make sure TIBASEW is on DSK5 with the rest of the TI-Base files, and then run the @PATH program. It can be run from the HRD menu option 3 (Run a program). You will first be asked for the current location of TIBASEW. Type in DSK5 and then press Enter. The @PATH program makes the changes to TIBASEW and then exits on its own. Select your TI-BASE option from the HRD Menu and TI-BASE will appear on your screen, with no prompt for a disk drive number!

Two-tone that TI keyboard

The following was written by David Hetkerthin of the Lima (Ohio) User Group. It appeared in the group's newsletter, Bits, Bytes and Pixels.

As you probably already know, there are two colors of TI keyboards — black and beige. But what you may not know is that there are at least two types of posts used to attach the actual keys to the keyboards. There are hollow square or "O" posts, which are the most common, and the solid or "+" posts.

If you have or know of someone else who has a keyboard of the opposite color and the same post type, then you both have the opportunity of having a two-tone keyboards. Two-toning can make programming easier and is especially useful when playing games such as Amazing.

Suggested keys to swap and make a contrasting color are: Enter, FCTN, Alpha-Lock, $4, +=, J, K, I, M, E, X, S and D.

The only tool needed is a large paperclip, straightened out and bent into a "J" or fish hook shape. Keep the hook small. Insert the hook between the keys and lift gently, first one side of the key, then the other, until the key cap pops off.

As with any hardware modification you undertake, you do so at your own risk.

One-line graph paper printer

This Extended BASIC one-liner by Sam Carey of Portland, Oregon, will print a sheet of graph paper with 8,611 squares approximately .1-inch by .1-inch in size.

If you have a serial printer, you can change the PIO.CR.LF to something else (RS232.BA= etc.).

The program is written for an Epson-compatible printer. If your printer isn't Epson-compatible, you can change the

(See Page 30)

USER NOTES

(Continued from Page 29)

PRINT #1:CHR$(27)&"1" to whatever the code is to change your printer's line spacing to 7/72-inch (.1-inch would be preferred if available).

The program assumes that the printer will print the "|" (Fctn-A) character as an unbroken straight line. If not, it will still work, it just won't look as nice.

Here is the program listing. When you get to the end of the fifth line and it will not accept any more characters, just press Enter, and Fctn-8 (Redo) and go on to the end of the line and continue entering the program.

```
1 OPEN #1:"PIO.CR.LF" :: PRI
NT #1:CHR$(27)&"1" :: C$=CHR
$(13):: FOR C=1 TO 110 :: PR
INT #1:RPT$("|",80)&C$&RPT$(
"_",80)&C$&CHR$(10):: NEXT C
:: PRINT #1:CHR$(12) :: CLOS
E #1 :: END
```

Goof-proof programs

This note by John Bull was printed in the K-Town 99er, newsletter of the K-Town 99/4A Users Group of Knoxville, Tennessee.

The most annoying thing that can happen when you are using an Extended BASIC program is to hit a key and get an error message. You have to start over again and lose whatever it is that you have already done. If a wrong key can be hit, then sooner or later it will be hit. In programming you do the best you can to keep such things from happening.

The simplest way to make a single key input like "Yes or No" or "DSK#" is with the CALL KEY command. One key press and it is done — quickly and easily — also easy to protect against wrong key presses. For instance:

```
100 DISPLAY AT(R,C):"Press d
isk drive number"
110 CALL KEY(3,K,S):: IF S=
0 THEN 110
120 N$=CHR$(K) :: IF S=0 THE
N 110
120 N$=CHR$(K) :: A=POS("123
4ABCD" ,N$,1) :: IF A=0 THE
N  110
```

This will accept for the drive number, N$, only one of the values in the string "1234ABCD" and no other. For the "Yes or No" choice the string would be"YN".

If the input needs to be more than one character, then the ACCEPT AT command works better and the VALIDATE option can protect against almost all wrong key presses. For instance:

```
100 DISPLAY AT(R,C):"Enter t
he hour"
110 ACCEPT AT(R,C+16)VALIDAT
E (DIGIT):H :: IF H>12 THEN 1
10
```

This will give values of the hour H only from 0 to 12. However, if you should just happen to press Enter before you have pressed a number, then you don't get an ERROR message, but you do get a WARNING and the screen scrolls up. To avoid this unhappy possibility you need one more line:

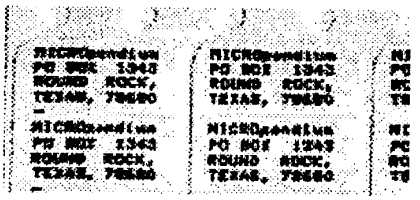
```
90 ON WARNING NEXT
```

This doesn't seem logical, but it does work. If you press Enter prematurely, the program returns to line 110 for another try without any rude message. Now this program segment is goof proof — I think.

Tiny labels from TI-Writer

This comes from Bill Cook of Kane, Pennsylvania. He writes:

I suspect many computer users are already making their own return address labels like the ones shown below, but I thought you might like to share it with your readers for the benefit of those who haven't pickup up on it yet.



I use a sheet of eight labels inserted into the printer sideways. Create a word-processor file (I use Funnelweb) for four side-by-side labels. If using the standard 15/16-inch labels, the tabs are:0, 17, 34 and 51. The "-" (dash) at the beginning and end of the blank line between each set of labels enables perfect cutting with an X-acto knife and straight-edge.

Print out on the first labels, then turn the strip upside down and print out gain on the remaining four.

The following instructions show how to initialize an Epson-compatible printer for the "mini" (compressed superscript) printing. To initialize your print for the small print, do the following on one line of your document:

```
CTRL U, FCTN R, CTRL U,
S,CTRL U, SHIFT 2, FCTN U,
FCTN R, CTRL U, A, CTRL U,
SHIFT F, CTRL U
```

A simple command halts Call Waiting for modem users

This item, by Glenn Bernasek, appeared in TI-CHIPS, the newsletter of the Cleveland, Ohio TI user group.

I was on the Free-Net (in Cleveland) and got blown off. My son-in-law was telecommunicating with a customer, and he got blown off! Why? We've both got Call Waiting service on our phones. Whe. a call comes in while we're on-line, Call Waiting emits a tone that the computers

(See Page 31)

Rocky Mountain newsletter editor dies

Robert P. Caine, a longtime member of the Rocky Mountain 99ers in Denver, Colorado, died Feb. 10 after a long illness.

He was born in 1920 in Logan, Utah.

He had been an associate editor of Tic Toc, the group's newsletter, for four years, and had produced most of its cover design. The group's copy machine used to print the newsletter was kept in his basement.

Burial was in Logan, Utah.

USER NOTES CLASSIFIEDS

(Continued from Page 30)

cannot understand or interpret. This causes our modems to lose their signal.

I am unable to verify the following information, which I found on CompuServe regarding Call Waiting because I do not have the service myself. However, I offer the following by Mel Myhre to those who have modems and Call Waiting. The following paragraph is according to Myhre:

Preface the number of "1170,". The "1170" tells the phone company to turn off the Call Waiting and anyone who calls the number will get a busy signal. The comma is a pause for any Hayes-compatible so as to give the phone company time to react and stabilize the line. Try 1170 manually and you will hear approximately three tone bursts and then a normal dial tone after a slight pause. Call Waiting is restored as soon as the present connection/call is terminated by the phone company computer, so it must be done for every number dialed. The auto-dial phone files I use are all prefaced by 1170. I travel extensively and it has always for me in the U.S.

And now for my endorsement. I tried the manual "1170," and got the three tone bursts followed by a dial tone. So far, so good. I then fired up Telco and called up auto dialer. I retyped the Free-Net number to read "1170,368-3888."

I got on-line with Free-Net and, while the administration menu was waiting for my selection, I dialed the phone number of my daughter's phone (she has her own line). Well, what do you know? I got a busy signal and my connection was undisturbed. All worked as advertised. Just don't forget to include the comma.

MICROpendium pays $10 for items sent in by readers that appear in this column. Send them to MICROpendium User Notes, P.O. Box 1343, Round Rock, TX 78680.

Attend a TI Fair this year

Policy

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

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

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

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

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

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

FOR SALE FOR SALE

Myarc 9640 complete with hard disks and other cards in original TI box with update Myarc disk drive, expanded keyboard, etc. All documents, instructions, albums, numerous software are all included. I have changed to a laptop. Call or write Perry, 535 Sanctuary Drive, Longboat Key, FL 34228, 813-383-1795. 11/2

Try a MICROpendium classified to sell or buy TI/Geneve products

Buy/Sell Used Hardware and Software

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

Myarc HFDC .....Offer
GRAMKracker (80K) 95.00
Super Space II 25.00

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

The ONLY monthly devoted to the TI99/4A

Subscription Fees

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

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

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

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

Address Changes

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

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

Check/MO ❑ 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.

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

Name _____

Address _____

City _____

State _____ ZIP _____

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

Disks, Etc.

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

_____

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

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

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

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

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

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

❑ **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