TEXAS INSTRUMENTS

The TI Forum

by Ron Albright and Jonathan Zittrain

Am I Crazy Or What?

While sitting about in some waiting room somewhere, I picked up a copy of the September 7, 1987 issue of *Sports Illustrated*. Well, it beats reading *Ladies Home Journal*! Anyway, on page 4, in the "From The Publisher" sidebar, there is a piece on *Sports Illustrated's* David Fisher. It discusses Mr. Fisher's moonlighting for *TenniSTAT*, which keeps numbers on whose shot landed where during which tennis tournament, etc. What the heck does this have to do with the TI 99/4A? Well, seems the piece had a picture of young David sitting courtside and holding a computer. Now, unless my eyes are fading faster than my memory cells, Fisher is holding none other than a beige TI 99/4A. It even has a cartridge snuggled into the port. Could this be? Could this writer, who is described as having "100-words-per-minute typing skills," actually be using a TI? Is this possible? Who knows. But that sure looks like a TI to me. Check it out and let me know if I am dreaming.

According to the Northwest Iowa Computer User's Group newsletter, Trio + Software (P.O. Box 115, Lipscomb, IA 50148) has produced a couple of companion graphics packages for use with TI Artist and a second disk-based package of fonts. They have several other software packages along the graphics and music lines as well. Write for details. The Bunyard Group (P.O. Box 53171, Lubbock, TX 79453) has a hardware manual for the TI available. It describes (again, according to information I have seen) "console design, TMS instruction set, interfacing pitfalls, PEB schematics, XB module description and schematics" and other topics. $19.95 is the price. I have heard nothing but good things about the new "LGMA 99 Fortran" package from LGMA Products (Box 210 Rt. 4, Apple-Butter Road, Coopersburg, PA 18036). It comes with a full-screen editor, optimized compiler, linker, debugger, example programs and a second disk with an object module library of 78 functions and subroutines, along with an extensive manual. Cost is $49.95. I have seen the author (Al Beard) answer inquiries on CompuServe's TI Forum, so there is good support available as well.

All you programmer-types might want to check out this addition. Do yo realize how many languages we have available for the TI? Amazing! Did you know that there is a software program that can transfer text (a.k.a. ASCII) files between IBM formatted disks and TI formatted disks? "PC-Transfer" from Genial Computerware (P.O. Box 183, Grafton, MA 01519; (617) 839-4134) will, reportedly, format a disk to IBM specs on your double-sided, double-density drives and then write files back and forth between IBM and TI format. Note: There has been some confusion on this issue. Only ASCII files can be transferred. This is not an IBM-emulator. It just facilitates using text-based information between the two system types. IBM programs won't run on the TI or vice versa. But you could, at least theoretically, "LIST" a TI BASIC program to disk (it would then be an ASCII file—i.e. Display/Variable 80—and not a PROGRAM file) and transfer that to the IBM. With some minor converting, you might just be able to get some BASIC programs running in this fashion. Just a thought. Understand the difference? Got it? Anyway, if you need this type of file transferring, at $25, "PC-Transfer" is a steal. The famous HUG (Houston Users Group) BBS has a new number and a new Sysop. The new number is (713) 781-4844 and the system is now run by Henri Schlereth. Henri takes over for Bill Knecht who built upon the foundations laid by the legendary Stephen Foster to make the HUG TIBBS one of the foremost TI communications systems around.

Troubleshooting

John Wilforth, writing in the West Penn 99ers newsletter, had this tip:

"I've come across a perhaps not novel, but a very easy way to troubleshoot and repair the video processor's 16K of memory. The symptoms usually fall into these general categories:

—Bits dropping on the screen (characters incorrect)

—Programs doing strange things (syntax errors, etc.)

—Colors wrong

—Screen appears "very dark" or shifted

—Combinations of the above

"Well, if you are what you might consider of medium hardware technical ability [ed. do any hardware modifications at your own risk], I've got an "almost sure fire fix."

"The TI uses 4116 dynamic RAM chips, arranged so that each bit of a byte are located in a different RAM chip (i.e. a byte with "FF" would have a "1" on in the same address in each chip. This makes it very difficult to troubleshoot to the chip, because all chips are involved in any one byte of information.

"The first attempt to fix a "memory problem" was to remove one chip at a time, install a chip socket, and put in a new RAM chip, and test the computer. I did this, wouldn't you know, eight times. It was the last chip in the

continued on page 202

TEXAS INSTRUMENTS

TI Forum continued from page 195

console. There had to be a better way.

"I believe I have found it. The answer was obvious. Why not piggy-back the chips with a good one, trying one at a time, until the defective chip is found? The second console proved this theory correct. I put a 4116 chip directly over the first, and turned on the console. I repeated this until either (1) the symptom changed, or (2) the problem was corrected. The reason I specify a changing of the symptoms as an identifier is because there have been two instances where there was more than one chip at fault, and because the chip may exhibit other symptoms when piggy-backed.

"Now replace the chip with a new one, by either using desoldering tools or cutting the leads to the old chip and installing a new chip in its place."

There you have John's tip. Happy (shudder) hacking. At your own risk, of course. For those of you who do this sort of work, I marvel at your dexterity and your courage.

Quick Reminders

Remember the "TI XPO 88" convention in Las Vegas. February 27 and 28, Palace Station Hotel. Contact Southern Nevada Users Group for details—(702) 647-1062. Also, the Third Annual T.I.C.O.F.F. is right around the corner. March 26 to be exact. Call (201) 241-8902 or 241-4550 for details.

The 99/8 Challenge

Some months back, the Forum published a series of benchmarks run on the Myarc 9640 computer, contributed by Chris Bobbitt. The results prompted the following letter from Bill Addington in Plano, Texas.

"I recently read the benchmark tests you published in the September issue of *Computer Shopper*. I was interested in how my 99/8 would compare in these tests. The following were my results:

Benchmark	Mode	Time (secs)
1	5	12
2	1	41
2	2	41
2	3	37
2	4	37
3	1	63
3	2	63
3	3	63
3	4	63
4		20
5		39

"The 99/8 supports more graphics modes than the other systems reviewed. These modes are:

1 Pattern
2 Text (40 columns)
3 Split-screen: Text top ⅓ of screen
4 Split-screen: Text bottom ⅓ of screen
5 High-resolution
6 Multicolor

"The disk drive is a double-sided, double-density (320K) Hexbus drive. This drive is connected to another stand-alone DS/DD drive. These drives are compatible with both the 99/8 and the CC40 compact computer.

"As you can see, the 99/8 is faster in pure graphics. Text display is not as efficient as the Geneve, but faster than any 99/4A option. Disk access is slower, but this is comparison to a 6 ms. drive.

continued on page 349

TANDY

CoCo Column continued from page 196

Pin 39—TSC—Three State Control—This one is kind of tough to explain in layman's terms, but the voltage state of this pin, along with others, allow other devices and processors to share the bus for data transmission.

Pin 40—HALT—This pin, when low, will cause the 6809 to stop working after the end of it's current cycle. An example of when this pin is used, is during some disk drive activity. Your drive's controller will halt the 6809 while it takes care of some activity.

Hopefully, this has given you some idea of what your 6809 chip does, and in next month's article we'll take a look at what happens to the 6809 chip and registers as it carries out a few instructions. In future articles we'll also take a look at how the 6809 communicates with other chips like the VDG (Video Display Generator) and PIA (Peripheral Interface Adapter) chips.

Multi-Vue Finally Arrives

Back in the fall of 1987 we received the first wind of a new graphics environment for Level 2 OS9, called Multi-Vue. It took the Tandy Corporation and the development company, Microware, nearly 14 extra months to deliver the goods. Although some CoCo users are upset at the delay, it's seems that Multi-Vue is worth the wait at $49.95.

First impressions are good with this product. It gives the Color Computer 3 with OS9 Level 2, the look and feel of a Macintosh graphic environment. Although a new user of Multi-Vue may not immediately see the results from this system (other than it's utilities), the benefits will be seen as software applications add the option of using Multi-Vue to enhance it. Icons made specifically for a software line is one change that developers can begin right now. In time, on information services like CompuServe and Delphi, we'll begin seeing icons appearing by users of previously released software.

Programmers developing in the C language will find the documentation of the CGFX graphics library a definite benefit. It will also allow them to implement the Multi-Vue environment, and additional system calls available through the WINDINT device driver.

It had been speculated prior to release that the Multi-Vue system would lack the Clipboard function. And although to an extent, this is true, it will be up to developers of software to activate the Clipboard. An associated portion of the developer's program will have to activate the Clipboard and deal with it's data. But out of the box, this function is not used.

The manual provided with Multi-Vue contains the complete CGFX library documentation, syntax necessary for the C language programmer to complete OS9 Level 2 graphic's commands. Additionally, an extensive look at the AIF (Application Information Files) format and system disk set up is impressive. C and assembly language programmers will find additional information on system calls which may now be made with the addition of the device driver WINDINT included with the Multi-Vue system.

Figure 2 and Figure 3, on page 196, give you an example of what your Multi-Vue screens will look like. Here is a brief explanation of each menu selection of Multi-Vue:

Box With Dot—This quit's Multi-Vue. If within an application, choosing this selection will terminate the process.

HourGlass—Contains 9 desktop functions included with Multi-Vue;

Calc—A resizable calculator

Clock—A resizable Analog clock

Calender—A month long calender, allows data entry for dates

Control—A control panel for resetting screen colors

Printer—Sets up printer configuration

Port—Sets up configuration of communication ports

Help—As it's name implies, gives you help with Multi-Vue functions

Shell—Allows you to call up a resizable shell window

Clipboard—Not functional

*Disk—Contains 8 functions associated with disk activity;

Open—Use this to use a command you wish to use

List—Same as the LIST command for viewing ASCII files

Copy—Same as the COPY command for copying files

Stat—Gives you details of the file that you activated

Print—Sends the activated file to printer

Rename—Same as the RENAME command for renaming files

Delete—Same as the DEL command for deleting files

Quit—Quits the disk functions

*View—Allows you to change between a 40 and 80 column screen

Multi-Vue does have a design flaw, which could put a damper on it's use. Icons used by the system require an AIF file to tell the system what icon to associate with certain disk files, modules, directories, etc. When Multi-Vue sees a file on your disk with a 'dot' and three letter extension (ie..."**.xxx**"), it expects to have an associated AIF file which will depict the icon to be used with the extension. It seems quite often that if the associated AIF information cannot be found, your system will hang and the window becomes unoperational. I would hope to see icons for standard extensions (.TXT, .BIN, .AR, .VEF, and others) to be posted on electronic services like CompuServe, Delphi, and GEnie. And hope that Tandy/Microware look into this problem.

Multi-Vue, at it's $49.95 price tag, is a bargain for what it will do and for the additional programming abilities given to programmers. If Multi-Vue becomes an accepted environment I do believe we'll see some extremely exciting software being released in the future!

Letters From Readers

The (Color Computer) 3 certainly deserves the recognition. I use a monochrome monitor on my system and had quite a bit of difficulty getting a clear screen. I also think I should inform your readers that the 3 is compatible with Tandy's monochrome VM-3 (monitor), as Tandy doesn't. It offers better resolution than the CM8 and makes the system much more affordable.....My Apple IIe buddy turned green after I read him parts of your November article.

Fred Wiezorek
Clermont, Georgia

A. I agree with you! The Color Computer 3 truly is a unique computer. Although monochrome can't take the place of color, those wishing a less expensive setup will find the VM-3 a viable alternative in choosing a monitor. So your buddy turned green, eh? Maybe he got hold of a green apple!

D.R.

I would like to upgrade my CoCo 2 with 64K to Extended Color Basic but the local Radio Shack refuses to sell me one and has no information on support. I also have an IBM compatible 5 1/4 inch drive and a CoCo I controller, but no one will tell me anything.....Can I modify the controller? Can I still buy the ECB chip?

Richard Davis
Buxton, North Carolina

A. Well first of all Richard, an Extended Color Basic chip IS available. Some franchised stores do not carry them, but many third party vendors like Spectrum Projects in Howarc Beach, NY do carry them. You can also order them directly by calling Tandy National Parts in Fort Worth, Texas. Support and installation will be pretty much up to you.

In most cases, drives compatible with an IBM are compatible with the Color Computer's controller. The trick here is to set the internal jumpers properly inside the disk drive which reflect the drive number, and to have a terminating resistor in the last drive in chain. Your ribbon cable and connector will also have to be configured for the drive and have the proper pin placed correctly.

As always, I invite you to send your comments, suggestions, and questions. Send your correspondence to CoCo Column, The *Computer*, 5211 S. Washington, Titusville, Florida 32780.

More Tandy on page

TI Forum
continued from page 202

I believe that a performance increase is still true when comparing to standard equipment.

"This 99/8 has Pascal and Extended BASIC on ROM. From your description of the current Geneve, I can say that the 99/8 is much more reliable and compatible. It reads my double-sided Corcomp disks and runs all 99/4A programs that do not require a specific memory address. Extended BASIC does not split memory into sections like the 99/4A. When started a display announces 62457 bytes free for BASIC. Logo I works, Logo II crashes when it does a garbage collection. The workspace for Logo I does not increase."

Thanks, Bill, for the informative comparison, as well as the nostalgic look at what "might have been" for the TI consumer. I still wonder why the 99/8 was never released. By all reports, it was a marvelous machine.

One-Liner Of The Month

From the S.N.U.G. (Southern Nevada Users Group) newsletter comes this cutey from John Martin, the group's president. John describes how to type in this one-liner:

"When typing in this program...you type the first 5 lines first. When the cursor stops, press ENTER. Now, press FCTN 8. This will bring the line up for editing again. Now move the cursor back to the end of the fifth line. You will find that you can now type past the end of it. Just type the rest of the program and press ENTER. Now, save it to disk and RUN it. You will be prompted with "DSK." Just type in the number of the drive you want to catalog. When the program has finished cataloging your disk, it will run itself again and prompt you again. If you want to catalog another disk, just enter the drive number and away you go again. To end the program, just press FCTN 4. If you don't want the program to recycle endlessly, just replace the word RUN in the 5th screen line with the word END."

Here is John's one-liner:

```
1 IF F THEN INPUT #1:A$,A,J,K :: IF J THEN PRINT A$;TAB(12);J;TAB (18);SEG$(B$,ABS(A*2) + 1,1);K;TAB (27);A<0 :: GOT 0 1 ELSE RUN ELSE B$ = "AVDFDV IFIVPG" :: INPUT "DSK":F :: OPEN #1:"DSK"& STR$(F)&".",INTERNAL,RELATIVE, INPUT :: GOT 01 ! BY JOHN M
```

Winding Down

I am going to take a break from the freebies this month. I have been innundated the past few months and the old photocopier is dying and needs a rest. I may have to start asking a copying fee, but I will think about that for the next month or so. Drop me a line and let me know your thoughts. The giveaway will continue. Depending on what I have available, the winner will receive either the month's newsletters or software—whatever is available.

JZ's Part

Programmer and TI guru Warren Agee has announced the formation of a new TI-based software development company, Olympys Technologies. "This company is a coalition of top-notch programmers whose main concern is to deliver power and sophistication to applications software for the TI-99/4A and Geneve 9640 computers," said Agee.

At a special conference on CompuServe's TI Forum Agee also introduced J. Peter Hoddie of the Boston Computer Society's TI Users Group as part of the new venture. "We have inked an exclusive agreement with Genial Computerware [of which Hoddie is a part] to handle the marketing of our products," Agee said.

Agee also indicated that talents such as Hoddie will be assisting him in the actual development of new software. Olympys' first product, which, according to Agee, is "80% complete," will be a sophisticated database manager called FirstBase. Planned program features include "full-fledged querying" (allowing operators such as AND and OR to create multiple field key searches), batch-updating (performing mathematical calculations or changes on a group of records), and macros ("scripts" that help to automate much of the database usage).

The program is written mostly in c99, and runs on a TI-99/4A with 32K Memory Expansion and at least one disk drive. The amount of records the program can manage per datafile is limited only by the size of the disk media; theoretically, said Agee, there can be up to 32767 records per file, 3000 bytes per record, and 720 bytes per field. Agee estimated a "typical" database file (five field of 100 bytes each) to be able to contain a maximum 235 records on a single-sided, single-density disk drive. A version of FirstBase compatible with the Myarc 9640 computer (and the new Myarc hard disk controller card) will be released concurrently with the 99/4A version. The program will not function properly with the old-style Myarc hard disk controller.

Agee projected the retail price of FirstBase to be approximately $50, available from Genial Computerware directly or any of its dealers.

The formation of Olympys Technologies represents a shift in both attitude and tactics for one of the TI community's most known and talented programmers. Agee has previously marketed his programs through Asgard Software of Rockville, Maryland.

"I had certain ideas of how my material should be handled...I just wanted to do it on my own," said Agee. With his new company and relationship with Genial, Agee said he will have final word on documentation, packaging, and distribution of his programs—something that he did not have with Asgard. "I still intend to support those programs of mine already available through Asgard," he added, including the fixing of any major bugs. Agee's earlier efforts are many, from clipart collections to Total Filer, a database program that was the seed for FirstBase.

Olympys is actually very much a one-man show, with Agee acting as the primary programmer and developer, occasionally drawing help from associates. "I can use all the help I can get," said Agee. "Especially for the TI-99/4A, a working knowledge of

continued on page 356

TI Forum
continued from page 349

assembly language is required, and I don't have that." Hence Agee's association with Hoddie, who is acknowledged to be among the foremost of the assembly programmers for the TI-99/4A.

While most of the code Agee writes is in the c99 programming language, Hoddie has helped to streamline some bulk parts of Agee's programs, contributing speedier and more efficient assembly routines where possible. "He essentially wrote everything," said Hoddie. "I'm saving room, increasing power."

"Just recently Peter saved me 316 bytes," said Agee. "That's a lot when there were previously no bytes left."

Barry Traver, founder of Genial Computerware, will be helping with beta testing and the writing of documentation for FirstBase.

What other products are on Olympys' horizon? Agee remains unwilling to promise much beyond FirstBase. "Databases fascinate me...they can be so many things to so many different people...I definitely want to work on a relational database," Agee said. Among other features, a relational database allows several files to be open at once and related in useful ways, a feature which even FirstBase is not immediately planned to support.

Olympys' financial goals seem modest. "I would like to make at least $4,160 in one year at this," said Agee. "That's roughly what I would have made at a part-time job twenty hours a week at four dollars an hour," he chuckled.

Agee promised *Computer Shopper* a copy of FirstBase for review as soon as it is available.

Genial Computerware's address for FirstBase inquiries is P.O. Box 183, Grafton, MA 01519.

Ask Dr. John
continued from page 178

reassemble it, turn on your computer and test it. If it is still not moving smoothly, you may have a loose/intermittent wire in the mouse cable or defective sense circuitry. In either case, replacement parts are almost as expensive as a new mouse so I'd suggest getting a new mouse.

Q. I have been looking around for a mouse to use with my computer and can't decide between an optical mouse and a mechanical mouse. What are the pros and cons of each?

DD

A. The controversy between mechanical and optical mice has been going on for a few years now. Proponents of mechanical mice claim that they get better movement sensing with the mechanical mouse. Advocates of optical mice claim the same thing. So much of this issue has to do with emotion that it may be impossible for me to give you a definitive answer. However, let's take a look at the basic technology behind each type and then, armed with this information, you may be able to make a better decision on your own.

Mechanical mice contain a ball that rolls around with the movement of the mouse. Sensing circuitry detects the direction and distance that the mouse moves. Optial mice have internal no moving parts. Instead, they have a couple of optical detectors that track the movement of the mouse across a special mouse pad that is imprinted with a crosshatch of lines.

The primary disadvantage of mechanical mice is that they tend to get dirty as they move around. As a result, their rollers tend to get clogged with dirt and the mice malfunction.

Optical mice also collect dust and dirt, but because they are normally used only on their own mouse pads they tend to stay cleaner. Also, when they get dirty, they are usually a lot easier to clean since the dirt has only a small opening in which to collect.

On the other hand, optical mice are restricted to moving only on their pads. Mechanical mice will work fine on almost any smooth, flat surface.

Both types are generally quite reliable if they are kept clean and they are not abused (dropped on hard surfaces, dropped into liquids, and so on). In actual operation, they are very similar. The only real difference you may detect is the "feel" of the mouse. Optical mice tend to be lighter. Therefore some folks think they are smoother in operation. On the other hand, others like the weight of a mechanical mouse, feeling that it makes for surer movements.

Still not sure what to get? Go out and try some mice. The only real advantage of a mouse is in how it helps you do your work. Since most mice today are reliable, the most important factor in choosing one is feel. And the only way you'll know which one feels best to you is to use them.

Got a question about computers? Send your questions on any computer related topic to: "Ask Dr. John," P.O. Box 20102, Castro Valley, CA 94546. ●

TI-ECHO Announced

The following information comes by way of the Siouxland 99'ers:

What is TI-ECHO??

It is simply a topic-oriented bulletin board message base that is echoed around the country. A message entered into any one of several bulletin boards in the US (and hopefully Australia soon) will be sent to every other bulletin board participating in the TI-ECHO. If you are lucky enough to have a participating bulletin board local to you, or one that you can get onto using PC Pursuit, you will be able to send and receive messages to other TI people around the country. All at no cost to you!

TI-ECHO is originated from Dakota

continued on page 357

Cheryl's Mailbag

by Cheryl Peterson

Once again, I have a sack full of mail to share. Some questions, some answers, some comments.

Jeffrey Beard wonders about a GEM- or GEOS-like system to run under CP/M.

Dear Ms. Peterson:

Even though I don't own a Commodore, I appreciate and enjoy your column. It's nice to know someone still takes CP/M seriously.

I want to pose a question you may have inside information about. Is anyone working on or has anyone developed a GEM or GEOS-like system for 64K CP/M machines? I have a 1984 Kaypro 2X with limited graphics capabilities, according to the literature. I have been using it since January 1985. My wife now wants to use the machine more. She seems a bit bewildered by the A0>prompt. She likes the GEM environment she has seen running on the Atari 520.

Is there anything comparable in the old Morrow, Televideo, Xerox 820, Kaypro, etc. market? If not, why not?

Jeffrey Beard
Bowling Green, KY

Boy....if you had a C128, I could just say run out and get a copy of GEOS 128. As far as I know there isn't anything available for the Kaypro machines that will do what you want. I can certainly understand your wife's uncertainty with CP/M. I too was scared to death of it when I first started. With a husband to push me and a need to justify owning the computer, I didn't have any choice but to learn. Your wife is hardly in the same position.

As far as the why and why not, with the rapid advancements being made in computer technology I'm afraid the commercial developers are more interested in coming out with new products for "those other machines." IBM and MSDOS look much more profitable to developers since so many IBM PCs and clones have been sold. I'm afraid that leaves us CP/M users to lament our fate and struggle on.

Does the Kaypro machine not have

continued on page 358

TI Forum
continued from page 356

InfoNet OPUS run by Rory Binkerd and uses the FidoNet mail network to pass the messages around. At this writing, there are 4 participating bulletin boards.

1. Dakota InfoNet OPUS, 605-336-3578 Sysop Rory Binkerd. 2. NC Central, 919-852-8460 (accessible through PC Pursuit) Sysop Amnon Nissan. 3. Oregon OPUS, 503-692-7024 (PC Pursuit) Sysop Rich Hill. 4. Compulink, 805-494-3350 Sysop Eric Daymo.

These bulletin boards are not run on TI computers, but all have sections devoted to the TI and/or Myarc 9640. N.C. Central has around 235 megabytes of hard drive space with some healthy download areas for TIs. New bulletin boards are joining the echo every day and we should have at least 1 participating bulletin board in every PCP city before the end of January. There will also be others like Dakota InfoNet and Compulink that use PCP to send and retrieve the messages. If you call any of your local BBS's that are hooked into the FidoNet network, ask the Sysop if he would care to join TI-ECHO. If he is available via PC Pursuit, Dakota InfoNet will send the echo to him and also pick up any msgs that have been left in the TI-ECHO msg area.

Anyone wishing more information on how the FidoNet network and EchoMail work, may leave messages on the participating bulletin boards. There is a ton of info out there for anyone wishing to write the necessary software to allow TIs and 9640s to be able to process mail. ●

More TI page 442

Tour de Forth: Part 2

by Glenn Davis

The Future of Forth

(Editor's Note: Part 1 of Tour De Forth, which appeared in our December 1987 issue, was inadvertantly followed by Part 3 in our January 1988 issue. Part 2 is appearing this month. We apologize for any inconvenience to our readers.)

Why all of the hub-bub about Forth-83 when the TI has three Forths (TI, Wycove, Super4th) available for it already? A couple of reasons come to mind: 1) MSDOS computers use Forth-83 almost exclusively; 2) all new books describe Forth-83; 3) It's a better dialect. Given that MSDOS computers are so numerous, a great deal of software is written for them, and that includes Forth. Several years ago Henry Laxen and Michael Perry wrote a public domain Forth called F83 for CP/M and MSDOS computers. F83 is widely used now and has become a semi-standard by itself—*Forth Dimensions* publishes a number of articles that are F83-specific.

Last month I promised to tell intermediate Forth users what Forth-83 can do for you. Quite a bit. First, the DO-LOOP structure has been changed. 0 0 DO ... LOOP will now execute 65536 times, whereas the old DO-LOOP would execute just once (often neither is suitable for the application at hand). However, unlike the old DO-LOOP, which used strictly signed notation (crossing the boundary between 7FFF and 8000 hex [32767 and −32768

continued on page 443

Note And Point continued from page 441

over $200,000 worth of claims, and about $20,000 in assets at the closing. If you lost money to Whitehouse, you can expect to hear from the lawyers with a "cents on the dollar...someday" offer.

Alan Reeve made his mark with some nice PD and later Commercial handlers for the now cheap Atari CX85 keypad. Alan also has some other products, and recently announced a complete development system for the 8-bit Atari computers, still under development in his Chicago based company. Diamond will be a complete programming environment similar in operation to GEM on the ST. Although this has been tried by a few programmers before (RE: GOS, the Zobian Rat, etc.), no equivalent to the C-64 GEOS has yet really arrived. Diamond may be the first. Designed to fully use the expanded memory Atari (either the 130XE or any upgraded machine), it will include windows and icons, alert boxes and mouse support addressable from any language or application software. Included are to be Diamond Write, Diamond Draw, Diamond Publish, and a plethora of desk accessories. Release date is hazy...something we Atarians are used to by now.

Optimized Systems Software (OSS) has a new Toolkit for BASIC XL, adding all the power of BASIC XE except the Extended Program Area. The new extensions file for BASIC XL will include those amazing add-ons like Sort-up and Sortdown, to name a few. By the way, OSS also says no runtime package for BASIC XE will be released, so don't expect too many "extended" PD gems to surface. Other OSS news: The MAC65, considered by many to be the finest macro-assembler system for the Atari ever made, went out of production in 1987. Reports were that the contract with the author expired, and that OSS didn't think there was enough potential sales left for the machine-language programming cartridge to merit the relicensing costs demanded by the MAC65 programmers. OSS had a fire sale to get rid of what they had left, and its over for MAC65. Let's hope that it resurfaces through some other outlet...as several other companies have made interesting new languages that utilize the cart. I may have more about one of them in an upcoming column...so look out for MAC65 carts used or marked down at dealers... while you still can.

Barcode equipment for the 8-bit Atari has long been asked for, and is here from Xenia Research in Washington. For $179, you get the barcode reader and software that lets it imitate keystrokes to the K: handler in any 800, XL, or XE. Also included is the barprint software, point-of-sale and other application software, and complete documentation for adapting your own software in just about any language or DOS. It plugs into joystick port 1. It may sound a little expensive, but barcode wands for any machine are regularly over $100 without software. This might be the ideal thing for cheap automation of small business counter jobs. Contact Xenia, Box 4675, Federal Way, WA 98003; (206) 927-7018.

National Review magazine, July 3, 1987: "Soviet fad for personal computers is wearing off. Problems: the machines are lousy, and the Soviets forgot to make software. Exception: Chess champ Gary Kasparov donated 26 Atari 130s to the Kompyuter youth club." Can anybody figure out how to follow up and see if the Iron Curtain will pass PD Software?

That's it for this month...next month (if I'm here!) I plan a summary of self-help hardware hints. Keep the faith, 8-bit users, and let us know you are out there! ●

Tour de Forth
continued from page 442

decimal] would introduce weird side-effects) the new DO-LOOP will handle full loops of 64K which is useful for addressing memory. An F83 addition to this is ?DO which will skip the entire loop if the initial value and limit are equal, more like TI and MSDOS BASICs would. In fact, I use ?DO more often than DO for that reason! LEAVE was also changed, so it has an immediate effect instead of leaving the loop at the next occurence of LOOP. That is, on executing LEAVE, the next instruction that will be executed is the one following the innermost LOOP. The "old" LEAVE would still execute everything between it and the LOOP. Some pre-83 software has to be rewritten to take this into account, but all Forth-83 compatible software (such as that published in *Forth Dimensions*) will operate properly. The immediate LEAVE has the advantage of not executing the words between LEAVE and LOOP, something that had to be "programmed around" in FIG-Forth so execution of those words was acceptable.

The "true" flag was also changed from 1 to −1, which is what TI BASIC did in the first place. −1 is also more useful because it represents internally (two's complement) as FFFF hex so ANDing a number and a flag will return a zero when the flag is false and the same number when true. The old true flag would sometimes give false negatives. 4 1 AND (number 4 and old true flag) is false because of their bit-patterns, and that may not be what we want!

The "New" Math

Even a simple arithmetic operation like integer division changed in Forth-83. For positive numbers, everything is as it always has been. For negative numbers the quotient is "floored" instead of being "rounded toward zero." Both are equally valid representations, but people are used to dealing with rounded toward zero and not floored. The change is for the better in instances where the old division was discontinuous around zero (an extra zero appeared in a series when you went from positive quotients to negative quotients). See Figure 1 for an example of how the "new" division works:

Figure 1

Dividend	Divisor	Remainder	Quotient
10	7	3	1
-10	7	4	-2
10	-7	-4	-2
-10	-7	-3	1

That is, the quotient multiplied by the divisor added to the remainder is the orginal number. Having negative remainers is probably the wierdest thing about it. In several instances, this is quite useful.

Additional non-forth-83 words in the compatibility set include FALSE and TRUE, which are constants that leave those values. ON and OFF set variables to true and false, respectively. D2/ is standard while D2* isn't but is just as useful. NIP drops the second item on the stack. TUCK copies the top item under second (see the listing for the stack effect). ?NEGATE and ?DNEGATE negate single and double length numbers if the first element on the stack is negative. ' (tick) now returns the code field address (CFA) instead of the parameter field address (PFA) as FIG-Forth does. Since this tick is "s ate-independent" a new word ['] ("bracket-tick") has been added to tick a word in a definition. ASCII and CONTROL are old friends that compile character literals.

SKIP and SCAN are very useful as text processing words, as they take a string address, count and a character to either skip or search for. Unlike ENCLOSE (the FIG-Forth scanning primitive) these words only use the character count as an absolute delimiter. ENCLOSE had to have a null (ASCII 0) at the end of the string to guarantee failure in the search (whice is why the FIG EXPECT appends two nulls to the end of the string. Thre first null is used as a "word" to stop execution of the interpreter and the second delimits that word.) " completes a string, and the run-time code returns and address and count which is suitable for use by TYPE or CMOVE. LIT" is similar but complies the string and the run-time code returns the counted string address.

BETWEEN and WITHIN are similar too. BETWEEN checks that a given value is between two limit numbers and WITHIN checks that the values are correct for a DO-LOOP. They're not essential words, but often are very useful.

Another group of words that are quite useful are the "defered compilation" words. Since Forth compiles things in

continued on page 444

Tour de Forth
continued from page 443

a single pass (much like Pascal) forward references are not allowed. However, forward references are quite useful and sometimes very necessary. So Laxen and Perry included the DEFER/IS words. They are used as follows: upon deciding on a name for the given funtion, say TABLE you code DEFER TABLE. After the actual run-time code is written (say it is call (TABLE) pronounced "paren-table") you merely need to execute ' (TABLE) IS TABLE and now TABLE will function as (TABLE) does. Of course, this isn't a permanent change. You are certainly allowed to change what TABLE executes in the same manner.

Now these words need to be compiled. Provided you entered last months' set of screens, you have only a few more to enter. When you've finished, be sure you're no longer in the editor (to exit from the editor, press FCTN-9). Type the same number as you began editing on followed by a space and the word LOAD, as in 20 LOAD. When the system responds "ok"—which takes awhile: be patient, go make a sandwich or something—you're ready to hit the books. If Forth responds <some word> ? then type WHERE and the editor will show you where the error is. Fix your typos, press FCTN-9, and type FORGET 21 and start LOADing all over again.

If you don't wish to type all of the code in, in Figure 2, or are afraid of insurmountable errors, I'm willing to put it on the disk for you (TI Forth only). Send me an initialized disk in a mailer with return postage and an address label with your name and address on it to *Computer Shopper* and they'll forward it to me. Please specify if you want the Forth compatibility code installed on the disk so it boots with it on, or if you want it in a text file to be loaded by one of the various Forth source text->screen converters. So BEGIN NEXT-TIME UNTIL... INNER LIMTS. ●

Forth Screens

```
SCR #310
  0 \ Forth-83 string compilation extensions          00Oct83map
  1 BASE->R
  2 : ,"    ( -- | parses string up to a " )
  3    ASCII " PARSE TUCK  HERE PLACE 1+  =CELLS  ALLOT ;
  4 : "     ( -- )    ?COMP  COMPILE (")    ," ;  IMMEDIATE
  5 : ."    ( -- )    ?COMP  COMPILE (.")   ," ;  IMMEDIATE
  6 : LIT"  ( -- )    ?COMP  COMPILE (LIT")  ," ; IMMEDIATE
  7 : ?ERROR   ( addr cnt ? -- )   IF  >R >R SPI  BLK @
  8    IF  >IN @  BLK @  THEN   R> R>  SPACE TYPE SPACE  QUIT
  9    ELSE  2DROP  THEN  ;
 10 : (ABORT")   ( ?)  R@ COUNT ROT ?ERROR  R> COUNT + =CELLS >R ;
 11 : ABORT"   ( ? -- | display message and abort if flag is true )
 12    COMPILE (ABORT") ,"  ;  IMMEDIATE
 13 : ABORT   ( -- )  TRUE  ABORT" " ;
 14
 15 R->BASE -->
SCR #311
  0 ( Forth-83 branch construct extensions          27Mar86 GED
  1  Under NO circumstances should you modify these CODE words.
  2  Some of the words address other words in the set, so even
  3  rearranging them will give "unpredictable results." )
  4 BASE->R  HEX
  5 CODE BRANCH  ( -- | unconditional branch )  C35D , 045F ,
  6 CODE ?BRANCH ( ? -- | conditional branch )  C079 , 13F6 ,
  7    05CD , 045F ,
  8 CODE (LOOP)  ( -- | iterative loop )   0201 , 1 , A781 ,
  9    19EA , 022E , 6 , 05CD , 045F ,
 10 CODE (+LOOP)    ( n -- | iterative loop with increment )
 11    C079 , 10F2 , 0704 , 0116 , 0913 , 045F ,
 12 CODE (DO)   ( lim start -- )   C079 , C0B9 , 064E , C7BD ,
 13    0222 , 8000 , 064E , C782 , 6042 , 064E , C781 , 045F ,
 14 CODE (?DO)   ( lim start -- )   C079 , C0B9 , 8081 , 16ED ,
 15    C35D , 045F ,                  R->BASE -->
SCR #312
  0 \ Forth-83 branch construct extensions          27Mar86 GED
  1 BASE->R HEX
  2 CODE (LEAVE)   ( -- )   022E , 4 , C37E , 045F ,
  3 CODE (?LEAVE)   ( ? -- )  C079 , 16F3 , 045F ,
  4 CODE I   ( -- index )   C05E , A06E , 2 , 0649 , C641 , 045F ,
  5 CODE J   ( -- index, outer loop )   C06E , 6 , A06E ,
  6    8 , 0649 , C641 , 045F , 0704 , 0116 , 0913 , 045F ,
  7 : DO      ?COMP  COMPILE (DO)   HERE  0 , 3 ;  IMMEDIATE
  8 : ?DO     ?COMP  COMPILE (?DO)  HERE  0 , 3 ;  IMMEDIATE
  9 : LOOP    ?COMP  3 ?PAIRS  COMPILE (LOOP)  DUP 2+
 10    , HERE SWAP ! ;  IMMEDIATE
 11 : +LOOP   ?COMP  3 ?PAIRS  COMPILE (+LOOP)  DUP 2+
 12    , HERE SWAP ! ;  IMMEDIATE
 13 : LEAVE   ?COMP  COMPILE (LEAVE) ;  IMMEDIATE
 14 : ?LEAVE  ?COMP  COMPILE (?LEAVE) ;  IMMEDIATE
 15 R->BASE -->
SCR #313
  0 \ Forth-83 floored division words                 04Oct83HHL
  1 : UM*      ( u1 u2 -- ud )   U* ;
  2 : UM/MOD   ( ud u -- urem uquot )   U/ ;
  3 : *D   ( n1 n2 -- d | mixed multiplication )
  4    2DUP  XOR  >R  ABS  SWAP  ABS  UM*  R> ?DNEGATE ;
  5 : M/MOD   ( d n -- rem quot | floored division )
  6    ?DUP  IF  DUP >R  2DUP XOR >R >R  DABS  R@ ABS  UM/MOD
  7          SWAP R> ?NEGATE
  8    SWAP R>  0< IF  NEGATE OVER IF  1- R@ ROT - SWAP THEN THEN
  9    R>  DROP  THEN ;
 10 : MU/MOD  ( d n -- rem dquot | mixed divide )
 11    >R 0  R@  UM/MOD  R> SWAP  >R  UM/MOD  R> ;
 12 : M/   ( d n -- n )   M/MOD  NIP ;
 13 : M*/  ( d n u -- d | WWB 040801 )   2DUP XOR SWAP ABS >R  SWAP
 14    ABS >R  OVER XOR -ROT DABS SWAP R@ UM* ROT R>  UM* . ROT 0
 15    D+ R@   UM/MOD -ROT R>  UM/MOD NIP SWAP ROT ?DNEGATE ;  -->
SCR #314
  0 \ Forth 83 numeric words                          04Oct83HHL
  1 : M+   ( d n -- d )   S>D  D+ ;
  2 : /MOD   ( n1 n2 -- rem quot )   >R  S>D  R> M/MOD ;
  3 : /      ( n1 n2 -- quot )   /MOD NIP ;
  4 : MOD    ( n1 n2 -- rem  )   /MOD DROP ;
  5 : */MOD  ( n1 n2 n3 -- rem quot )   >R *D  R>  M/MOD ;
  6 : */     ( n1 n2 n3 -- quot )   */MOD  NIP ;
  7 VARIABLE SPAN
  8 : EXPECT   ( addr cnt -- )   2DUP EXPECT  DUP >R  0 SCAN
  9    R> SWAP -  SPAN !  DROP ;   ( 08Jan86 GED )
 10 : KEY?   ( -- 0=no key | 7b=keyval )  ?KEY ;
 11 : BETWEEN  ( n min max -- ? )  >R  OVER > SWAP R> > OR  NOT ;
 12 : WITHIN   ( n low high+1 -- ? )   1- BETWEEN ;
 13 : CONVERT   ( ud1 addr1 -- ud2 addr2 )   (NUMBER) ;
 14 : RECURSE   [COMPILE] MYSELF ;  IMMEDIATE
 15 -->
SCR #315
  0 \ Forth-83 extentions to TI Forth
  1 BASE->R DECIMAL
  2 : QUERY   ( -- )  TIB 80 EXPECT  0 >IN !  0 BLK ! ;   HEX
  3 : FIND   ( addr1 -- addr2 n )   DUP  CONTEXT  @ @  (FIND)
  4    IF  DROP  DUP CFA  SWAP  NFA C@  040 AND
  5     IF  1  ELSE  TRUE  THEN   ROT  DROP
  6    ELSE  FALSE  ENDIF ;  \ 08Jan86 GED
  7 : >BODY    ( cfa -- pfa )   2+ ;
  8 : >NAME    ( cfa -- nfa )   1- -1 TRAVERSE ;
  9 : >LINK    ( cfa -- lfa )   >NAME 2- ;
 10 : BODY>    ( pfa -- cfa )   2- ;
 11 : NAME>    ( nfa -- cfa )   1 TRAVERSE 1+ ;
 12 : LINK>    ( lfa -- cfa )   2+ NAME> ;
 13 : N>LINK   ( nfa -- lfa )   2- ;
 14 : L>NAME   ( lfa -- nfa )   2+ ;
 15 R->BASE -->

SCR #316
  0 \ Forth-83 extentions to TI Forth
  1 BASE->R DECIMAL
  2 : SAVE-BUFFERS   LIMIT FIRST ?DO  I @  0<  IF  I 2+
  3    I @ 32767  AND  TUCK  0 R/W  I !  THEN  1028 +LOOP ;
  4 : THRU   ( scr# scr# -- )  1+ SWAP  ?DO I LOAD LOOP ;
  5 : FREE   ( --- | display bytes unused)   SP@ HERE - . ;
  6 : CLS    ( clear screen )   16 SYSTEM ;
  7 : PAGE   ( home cursor, clear screen)   0. GOTOXY   CLS ;
  8 : BINARY   2 BASE ! ;
  9 : CRASH   ( -- )   TRUE ABORT" Undefined Execution Vector." ;
 10 : DEFER   CREATE  ['] CRASH ,  DOES>  @ EXECUTE ;
 11 : (IS)   ( cfa -- )   R> DUP 2+ >R  @ >BODY !  ;
 12 : IS   ( cfa -- )   STATE @  IF  COMPILE (IS)
 13    ELSE  ' >BODY !  THEN ; IMMEDIATE
 14 : FORTH     [COMPILE] FORTH ;
 15 : FORTH-83    FORTH  DEFINITIONS ;  R->BASE
```