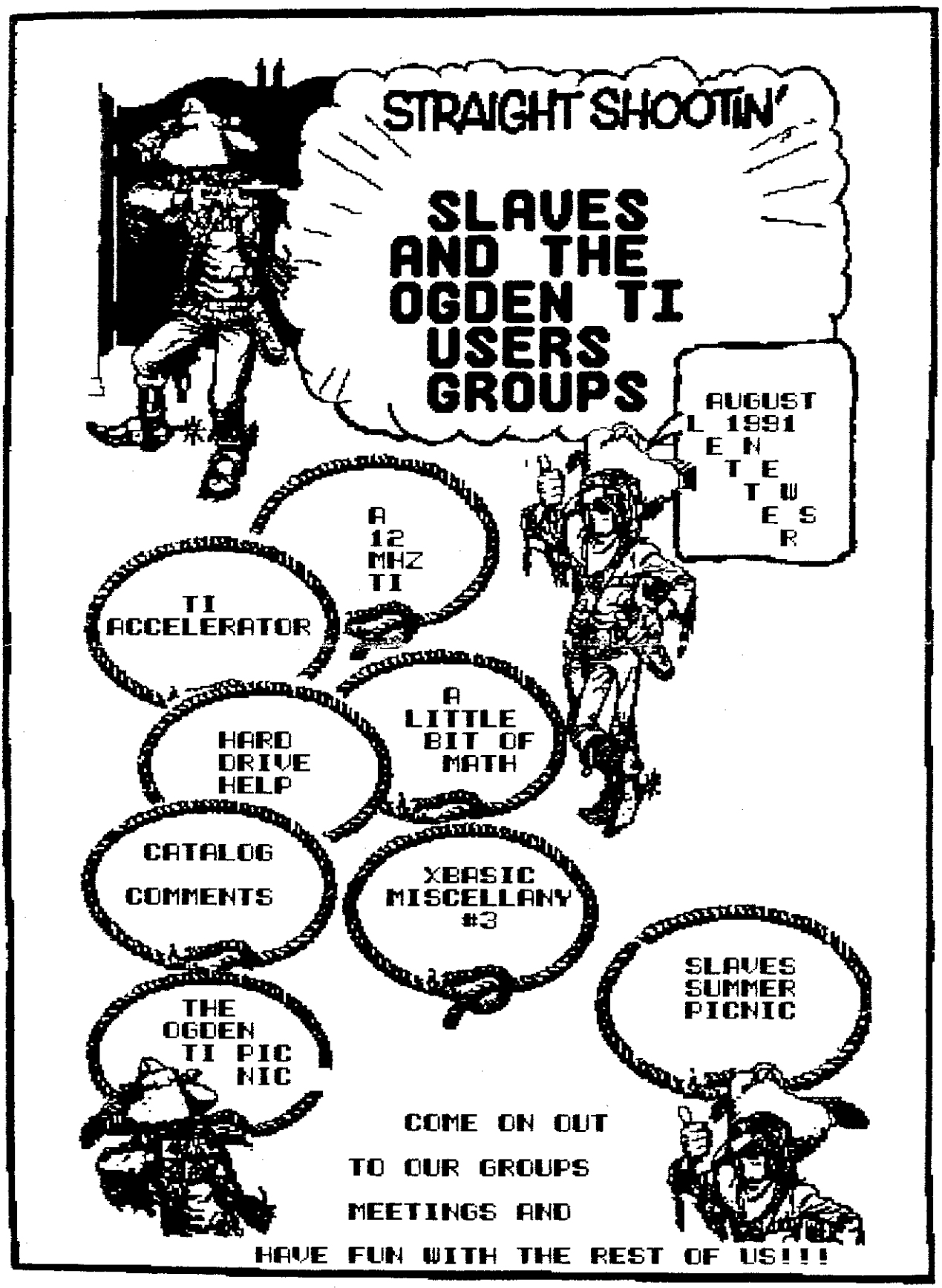
STRAIGHT SHOOTIN'
SLAVES AND THE OGDEN TI USERS GROUPS
AUGUST 1991
NEWSLETTER
A 12 MHZ TI ACCELERATOR
A LITTLE BIT OF MATH
HARD DRIVE HELP
CATALOG COMMENTS
XBASIC MISCELLANY #3
THE OGDEN TI PIC NIC
SLAVES SUMMER PICNIC
COME ON OUT TO OUR GROUPS MEETINGS AND
HAVE FUN WITH THE REST OF US!!!

# A 12 MHZ TI!!

If speed is what you've dreamed about, then mark July 1991 on your calendar. This month marks the scheduled debut of Don O'Neil's accelerator for the TI. This extraordinary device mounts directly into the console, requires no soldering and turns your TI coupe into a Ferrari. All for $250.

The device will be available through Bud Mills Services and Opa. And it looks like just the beginning of a number of peripheral devices that combined will turn the TI into an entirely new computer. The next thing up is a new interface card between th TI console and the PEB, which is expected to debut about the time of the Chicago TI Faire. In addition to connecting the two pieces of equipment, this card has a 16-bit bus with room for up to 8 megabytes of RAMDDDRAM and 16-bit slot for future expansion, which O'Neil says will most likely be used for a 9978-based video card.

And then, there is also a potential for development of emulators for MDOS( MSDOS( CP/M and Apple II. Gary Bowser of OPA is working on a Z80 emulator that may be used to port software from Sega and Sega Genesis systems.

What it looks like is that the accelerated TI will be giving the Geneve a run for its money.

## 99000 in-console accelerator promises big speed gain for TI

The following is an edited transcript of a conference held on Delphi with Don O'Neil, developer of the new 99000 Accelerator for the TI. The device plugs directly into the TI console.

Let me give you all a little background.

I announced, at the Lima Fair, my 99000accelerator for the 99/4A. This new device will install in your console and give you the following features:

%Up to 10 times performance over your current 4A;

%A 129K EEPROM for developing new console ROM's;

%Macrostore; %And a port to gain access to the 16-bit bus for no-wait state operation on future peripherals.

The card installs over the existing 9900 in the console, after three pins are removed from the 9900. It is a clip on design, easy to install. It is 100 percent compatible with 9900 code and the TI operating system.

Gary Bowser of OPA is antisipating taking all of the Macrostore source code he has and stuffing it in the EEPROM, as well as cleaning up the console ROMs and building in a full disk utility program.

The accelerator itself does not have any more additional RAM on board, but with the use of a Horizon RAMdisk with RAMBO, you can have up to 4 megabytes (on both an accelerated, and non-accelerated 4A). My second project also had RAMBO style memory built into it, but let's cover the accelerator first.

I have kept this project secret until we felt it was ready to be released. It will be distributed through both Bud Mills Services and OPA.

About what percentage of our current software will be able to take advantage of this extra speed? And how many might not be able to run at that speed at all?

All software could gain something from the faster speeds, especially programs like Multiplan. However, games will not functionif they require loops executed in the CPU for timing. We will not build in any means of slowing it down. If you want to play games, get another console. The speed increase at this point is at least to that of the Geneve, until we do hard core tests on PC Boards, we won't know how much faster we can go.

If we were able to push it to the top, it would be about 4 times faster than the Geneve. If you compare the code running on the 99000 to the code running on the builtin RAM in the 9995, (it would run about twice as fast as the Geneve.) But once you go outside that RAM, things start slowing down, and at times the approch one-quarter the speed of the 99000.

But games should run, right? Just too fast to play?

It depends. If the author used the jmp command and incremented a counter just to waste time for colision and such, it won't work properly, because the instructions will execute faster, but the sprites would not since they timed off the VDP. But games that use the VDP for timing will function fine.

Had you considered using the 99110?\ It has some interesting parallel processing capabilities.

The 99105 and 99110 differ in one way, Macrostore. The 99110 has built in Macrostore for floating point math, and the 99105 does not. Gary Bowser happens to have the source for the 99110's Macrostore stuff (and is trying to get the 99120's) and it will be put in the EEPROM on the card. Also, the 99110 is no longer available from TI, only from surplus vendors. With the EEPROM the 99105 will function exactly as the 99110.

When was the 99120 released?\ The 99120 was never released, only prototyped in the lab, it was a 99105 with complete PASCAL Kernal written in Macrostore.

Tell us about the peripheral port.\ The peripheral port built into the accelerator has only one peripheral it will plug into, and that is a new Flex Cable interface card for the P-BOX. This P-BOX interface will function exactly as the existing one but will also have some new features that can be added as options whenever you want. The first new thing is the attachment to the console, a sleek 1.5 inches protrudes and a round cable goes towards the rear to the P-box. On the P-box card, there are two sets of buses: the 8-bit standard TI bus that just drives the P-box (like the old flex cable), and a new 16-bit bus that stays local to the card.

On that 16-bit bus are, 32K of 0 wait RAM, MBP Clock option, 4 8K DSRs (static RAM battery backed like the Horizon RAMdisk), 68881 co-processor option, and 8 1 megabyte IBM-style SIMM (single In-line Memory Module) slots for RAMBO DRAM (up to 8 megabytes on board), and up to 32 megabytes off-board.

There is also a 16-bit "processor direct slot" for future expansion. This will most likely be used for a 9938 video card. All of the devices on the card that are 16-bit run at a maximum of 6 MHZ, and the P-box at the standard 3 MHZ. On the accelerator card there are two small clips that get clipped to the 9901 and GROMs.

Gary is working on a few

programs that will utilize the 9938 and 9958 better, one of which is a Z80 emulator. This will allow us to port ove Colecovision, Sega, and Sega Genesis games and programs to the TI with the 9938/58 and accelerator. Once I finish with the basics of the 9938, I will probably go into that. I am anticipating that the new XB, which will have a compiler for it, (will) be a (good) route to go, since BASIC is widely known.

What sor of speed increase can you expect with the new flex card alone?

With the 0 wait 32K, it is about a double, plus with the 68881, it could triple. Of course the Macrostore ROM on the accelerator will allow access to the 68881 through GPL using the same GPL commands. The ROM will check for the 68881 and access it if it is available.

It seems that the 9978 is the ideal choice for the 99105 unless you would like to use a 34020.

The 34020 at this point is a little pricey, but the 34010 is not. I have ideas for a graphics card that would use the 34010, but I don't know. That is too far down the road.

What's the possibility of emulating other computers? Mac, Amiga? CP/M?

Gary already has the CP/M one done. I would like to see a MDOS (Geneve) emulator, as well as a MSDOS (PC) emulator, also an Apple II emulator. They are all possible, but someone has to do it.

The performance of the upgrade will allow reasonable speed in the emulation?

The CP/M emulator currently runs on a 1-1 speed basis with equivalent clock speeds on the Geneve and the CP/M machine. With the 99000 it will be 2-4 times faster.

What drive type does the CP/M emulate? Osborne single density?

I don't know, I do know it is compatible with the Morning Star CP/M card that was made for the TI.

Do you have a timetable for all these projects?

Accelerator in July, P-box card by the Chicago Fair. The rest is unknown at this point.

As for space constraints in the console, do TIM (TI Image Maker by OPA) and accelerator cohabitate well?

Yes they fit fine, also, since the TIM and the accelerator drain LESS power than their predecessors, there are no fears of damaging anything.

Is there a new XB for this now? (Back to the accelerator)

Just the one that is in the works now in Oregon (Rich GKXB).

What's the time frame for the MDOS emulator?

I have to say probably around Chicago when the P-box card will be available, we may package it with the card or something.

When can I get an accelerator and I/O card and what are the payment plans?

The payment plans are the same as what Bud Mills has now, cash, check, credit card, COD. When the accelerator is shipping we will then take orders, no sooner. We anticipate that by the end of the month.

The P-box card will be available the same way.

Will the card be in a clamshell or bare?

Bare. It will have the standard screw clamp like on the existing one, so it will be attached to the P-box.

Will the 32K in-console offered by Bud conflict with the 32K in the new flex card?

Yes, the 32K in-console modification interferes with

the physical attachment to the 9900. I suggest you use a different console.

Does the accelerator support speech? With or without a RAVE card?

Yes, the accelerator will support any sidecar device, you just have to get a longer jumper cable from the extended 16-bit bus to the card edge connector.

Here are addresses and phone numbers for Bud Mills Services and Oasis Pensive Abucators: Bud Mills Services, 166 Dartmouth Dr., Toledo, OH 43614; 419-385-5946: OPA, 432 Jarvis St., Suite 502, Toronto, Ontario, Canada, M4Y 2H3; 416-960-0925.

## HELP FOR HARD TIMES WITH YOUR HARD DRIVE

This comes from John L. Teague, of Chandler, Texas. He is amember of the Tyler TI99/4A User Group and the Dallas TI Home Computer User Group. He writes:

Many TI users, as I, have been using the Myarc HFDC and the hard drive for some time, with BOOT on either the Horizon or P-GRAM serving as autoload menu. There are times when it becomes necessary to operate with the hard drive and no Horizon or P-GRAM, and no readily available menu for running the programs on the hard drive.

To allow for such occasional eventualities, I have been operating with BOOT, BOOU and BOOTLOAD in the root directory. This allows one to access BOOT from XB with RUN "WDS1.BOOTLOAD" Since BOOT saved the edited menu configurations back to DSKn, I had been doing the editing on a floppy and transferring the file to WDS1 with MDM%, a real inconvenience; hence a session with SECTOR 140 in searching for the string "Save to DSK" in BOOT and BOOU took me to the third sector from the end in BOOU. I changed DSK to WDS here, and in the second sector from the end, I changed DSK.BOOT on your hard drive.

Robert Smith, of Carthage, Texas, suggested creating a one-line program, RUN"WDS1.BOOTLOAD", and saving it as "LOAD" on a floppy. This disk is placed in drive 1, and when XB is called for from the TI main screen, the short program runs, and places your WDS1 BOOT menu on screen in short order.

These two innovations will allow the use of a hard drive with menu and (almost) autobooting, without the Horizon or P-GRAM.

## CATALOG COMMENTS FROM COMPUSERVE

Have you ever wanted to put comments on a catalog listing, especilly when you are passing it on to a friend? One way to do it is to write the comments with pencil, but a neater way is to use TI-WRITER. Here's how:

1) When you go into DM1000 to catalog the disk. First change the output device by pressing FCTN 3 when the main menu screen appears, and change the "pio" default to "DSKm.filename". You don't want printer codes and you don't want to save this permanently, so press N for both these questions.

2) Go through the normal steps to catalog your disk, i.e. MAIN MENU:OPTION 1; once the catalog is displayed, then FCTN 7 to get to the printer, it will print to the designated disk, giving you a DV80 file under the name you gave it in 1) above.

3) You can now access this file with TI-WRITER to add your comments and for printing out. It is

# a little BIT of math
by: Dan Eicher

*editor's note: A BIT is made from the words: Binary digIT. Binary is the 1s & 0s representation of Hexadecimal numbers . You will see the "greater than" sign used in TI documents to denote Hexidecimal numbers. ie: >0A The digits 1 thru 9 are the same in Hexidecimal or decimal numbers .*

*A decimal number from 0 to 15 is encoded into 1 byte, (4 bits) in this form:*

*The DECIMAL value of bits in a byte is 8421*
*so the bit value of Hex 0 or >0 = 0000 ( no bit on at all) = 00 dec.*
*and >1 = 0001 (1 bit only) = 01 "*
*" >2 = 0010 (2 bit only) = 02 "*
*" >3 = 0011 (1 bit & 2 bit added ) = 03 "*
*" >4 = 0100 (4 bit only) = 04 "*
*" >5 = 0101 (4 bit & 1 bit added ) = 05 "*
*" >6 = 0110 (4 bit & 2 bit added ) = 06 "*
*" >7 = 0111 (4, 2, & 1 bit added ) = 07 "*
*" >8 = 1000 (8 bit only) = 08 "*
*" >9 = 1001 8 bit & 1 bit added ) = 09 "*
*" >A = 1010 (8 bit & 2 bit added ) = 10 "*
*" >B = 1011 (8, 2, & 1 bit added ) = 11 "*
*" >C = 1100 (8 bit & 4 bit added ) = 12 "*
*" >D = 1101 (8, 4, & 1 bit added ) = 13 "*
*" >E = 1110 (8, 4, #& 2 bit added ) = 14 "*
*" >F = 1111 (8, 4, 2,& 1 bit added ) = 15 dec.*

To really understand how computers work the first key element that must understood is Boolean Alegebra. An easier way of thinking of this is as TRUTH tables.

| And Table | Or Table | Exclusive or(XOR) | Not Or(NOR) | Not ) |
|---|---|---|---|---|
| 1 and 1 = 1 | 1 or 1 = 1 | 0 XOR 0 = 0 | 0 NOR 0 = 1 | 0 NA |
| 1 and 0 = 0 | 0 or 1 = 1 | 1 XOR 0 = 1 | 1 NOR 0 = 0 | 1 NA |
| 0 and 1 = 0 | 1 or 0 = 1 | 0 XOR 1 = 1 | 0 NOR 1 = 0 | 0 NA |
| 0 and 0 = 0 | 0 or 0 = 0 | 1 XOR 1 = 0 | 1 NOR 1 = 0 | 1 NA |

Examples:

| | | | | |
|---|---|---|---|---|
| 00111010<br>01011100 | 00111010<br>01011100 | 00111010<br>01011100 | 00111010<br>01011100 | 0011<br>0101 |
| 00011000 | 01111110 | 01100110 | 10000000 | 1110 |

For the sake of terminology, lets say a 1 is true or positive and 0 is negative of false. This is how the chips in our computers determine what is going on ... a negative voltage level means a negative bit and a positive voltage level means a positive bit.

Remember 8 bits make a byte, 2 bytes make a word. The ability of a single assembly instruction to operate on 16 bits (a word) at a time is what makes our machine a true 16 bit computer. This seperates it from earlier technology like the 6502 or the Z-80, these cpu's moved data around the computer 8 bits (a byte) at a time.

The last two logical operators Nor and Nand are used far more commonly in electrical engineering than in programming.

At this point you are probable wondering "Ya.. So whats the big deal, how is this going to help me write better programs?"

Well one of the major uses for this type of instructions are in setting "flags". A flag is an indicater to your program of a condition that can be true or false. Some possable uses would be to save the configuration of the machine your program is running on in a single byte instead of wasting memory.

In one byte you could hold all the following information:

(bit) 1 If = 1 then color monitor, else adjust colors for black and white.
(bit) 2 If = 1 then disk system, else assume cassette and do not try to save high score.
(bit) 3 If = 1 then system has a printer, give option to output to printer else prepare all output for screen only.
(bit) 4 If = 1 then speech synthesizer is attached,use speech

If think you get the idea how, then the use of flags can save an enormous amount memory if used properly. This is especally important in a machine like ours. In a day when most computers are counting memory by the Megabyte (thats thousands of Kilobytes which is, in turn thousands of bytes).

1 Meg = 1000K = 10,000 Bytes

One last bit of arcane knowledge that you should at least have a passing knowledge of is how to get the "Twos Compilment" of a number.

Basically the twos compliment of a number is how a computer distingushes between a positive and a negative number, the computer must to do this internally before subtracting two numbers, here is how it is done.

*(editors note: Computers do NOT subtract, but add the COMPLIMENT of the subtrahend to find the difference.)*

Lets say you have the number 53 thats 00110101 in binary (>35 = 53 dec)
and you want to subtract 26 thats 00011010 in binary.(>1A = 26 dec)

First the computer must compliment 26 and heres how its done:

first you take the number 26 ========> 00011010 (1A hex = 26 dec)
change all 0 to 1 and all 1 to 0======> 11100101 (E5 hex)
add 1 (trust me here)================> 1
the complimented number is ==========> 11100110 (E6 hex)

NOW to "subtract" the computers:

Take 53==========================> 00110101 (35 hex = 53 dec)
ADD 26's compliment==============> 11100110 (E6 hex)
The answer is ====================> 100011011 (1B hex = 27 dec)

Since this is BYTE arithmetic, the 9th digit goes in the bit bucket leaving only 00011011. Thats 27 decimal, which is the result if one subtracts 26 from 53.

On July 27th 1991 the Ogden TI-99/4A Users Group sponsored a picnic for group members and their families. The picnic was held at Washington Mobile Estates, Mobile Home Park, in Ogden, Utah. The picnic was held from 3 p.m. to 9 p.m. The members brought their own meat to cook and a covered dish or salad and or desert. Before dinner, some members enjoyed using the swimming pool, while others enjoyed each others company or hung out around the pool table. The grill was ready about 5:30 and the membership all sat down around six o'clock to chow down on all the good food. After dinner we had a drawing for several door prizes. Later that evening we played bingo and enjoyed each others families. Everyone helped clean up afterwards! This was our groups first annual picnic and was a great success. Picnic committee members were Joe Harris, Loraine and Ken Wadman, and David Mischler.

**(continued from page 5)**

**suggested that you turn off Word-Wrap (ctl 0), which will display a hollow cursor. If the comments take more than one line, you will have to insert line (FCTN 8).**

**By using this method, you can make the listing more informative to the person getting it.**

**Copied from the SCCG Computer Voice.**

## XBASIC MISCELLANY #3

By Earl Raguse

Last month I promised a list of the token codes. Got the following from the computer, It told me what they where. You can see the listed program, that got it. I tend to believe things when the computer itself gives me this type info. I find it lies very little, and most errors are operator errors. I just did NUM 100 to get automatic line numbers, starting with 100. I typed line 100, as you see it, I will explain that later, then I just went down through the alphabet etc, entering the letter, followed by space, then CTRL and the letter, and Enter. As I did it, all I saw was space for the token but I listed it, WYSIWIG, What You See I What I Got.

```
100 ! SAVE DSK1.TOKENCODES
110 ! Key Function
120 ! A ELSE
130 ! B ::
140 ! C !
150 ! D IF
160 ! E GO
170 ! F GOTO
180 ! G GOSUB
190 ! H RETURN
200 ! I DEF
210 ! J DIM
220 ! K END
230 ! L FOR
240 ! M LET
250 ! N BREAK
260 ! O UNBREAK
270 ! P TRACE
280 ! Q UNTRACE
290 ! R INPUT
300 ! S DATA
310 ! T RESTORE
320 ! U RANDOMIZE
330 ! V NEXT
340 ! W READ
350 ! X STOP
360 ! Y DELETE
370 ! Z REM
380 ! 0 THEN
390 ! 1 TO
400 ! 2 STEP
410 ! 3 ,
420 ! 4 ;
430 ! 5 :
440 ! 6 )
450 ! 7 (
460 ! 8 OPTION
470 ! 9 OPEN
480 ! . ON
490 ! ; PRINT
500 ! = CALL
510 ! / AND
```

Now then, line 100. A very useful thing. I always put the name of a program, or file on the first line, so I will know what it is later. I am rather forgetful and sometimes after I have loaded a program, especially somebody elses, I do forget what it was called. Then after I make changes, and I always do, I don't know the name to SAVE it to. I have saved one on top of another and lost it because I used the wrong name. This is a very simple fix. Note that the first thing is the comment marker (!) that is because SAVE is illegal in a program, so I hide it thusly.

```
100 ! SAVE DSK1.TOKENCODES
```

It is also a labor saver, as well as a brain saver. When I am ready to SAVE, I just type 100, then FCTN X, Enter, FCTN 8 (Redo), then space 4 times, then Enter again, and it is saved to

DSK1.TOKENCODES

just like that, no pain, no strain. I know of very few people that make use of that. I notice, that only the very experienced programmers use it. Now I don't claim you will be an experienced programmer if you use it, but it helps.

Then I LISTed it to "DSK2.TOKENCODES", Yes, Virginia, you can LIST to disk, and get a DV80 file. If you LIST to the same disk as your program is on, you had better use a different name, or poof! goes your program. Actually, no large problem, however, you can always get it back with TEXTLOADER, a program that lets you make runnable XB programs out of DV80 files. This remarkable program was written by Curtis Allan Provance, Paragon Computing. He is with the New Hampshire 99ers. Its fareware, and if it is not in our library, just ask me.

I have discovered that statements like 5550 ON ERROR 6000, in a subprogram, are global, even though 6000 is also inside of the subprogram. A subsequent error outside of the subprogram will goto line 6000 inside the subprogram, very disconcerting.

I find the best way to overcome this is to do an ON ERROR (Outside Line # ) just before exiting the subprogram. The XB manual clearly says that you must not transfer control out of a subprogram with an ON ERROR statement, but fails to tell you that an ON ERROR statement within the subprogram is effective outside the subprogram, and will transfer you back into the subprogram, albeit usually the wrong place.

Transferring to inside a subprogram with ON ERROR is strictly forbidden by the XB Manual in the ON ERROR command explanation. They tell you that will result in a syntax error, but how do you differentiate that from the syntax error that put you there. If you wind up inside of a subprogram, get suspicious.

submitted by
David Mischler
Member OTIUG

## THE GREAT SALT LAKE

From far away, it's a beautiful sight, THE GREAT SALT LAKE, Light shimmering off its surface in spectacular splendor, tiny flecks against a mass of blue.

Facts you need to know: Drive time:about one and half hours from Ogden to the Great Salt Lake State Park Beach and site of Saltair, the former lakeside resort now in despair. Directions - follow I-80 past the Salt Lake International Airport going west toward Reno. Take exit 104. Visitor's center is about 100 yards west. Public beach is about two miles east.

What to do: Wade in the lake's shallow waters. Go boating. Buy souvenirs. Go on a 20 minute speedboat tour.

Cost: Access to the beach costs $3.00 per car or $1 per person for passengers in buses. Entrance to the area surrounding Saltair is free. Twenty-minute speedboat tours, sponsored by a privately owned business, Saltair Resort, cost $4.00 for adults,$3.50 for children.

Hours: Visitor's center near Saltair, a small trailer with a pictorial history of the lake as well as a few souvenirs for sale, is open from 9 a.m. to 6 p.m. Monday through Friday and from 9 a.m. to 5 p.m. Saturday and Sunday. Public beach gates usually open at 9 a.m. every morning but time varies depending on the lake level.

Where to EAT: The only refreshments available at the lake are sold at the ice cream stands near the gift shop. Also, there is a pop machine next to the State Parks and Recreation building along the marina.

What's for Sale: Saltair Gifts sells postcards,crystals,salt souvenirs,mugs,maps,T-shirts, sea gull jewerly, hats,salt water taffy,Utah honey and booklets describing the lake's history, among other items. Call 801-250-9660.

Facilities: Portable restrooms are set up near the visitor's center and at the beach. There are no shower facilities available, only a few open hoses for swimmers.

Information: For sailing information, call Saltair Yacht Sales at 801-250-8454 or the State Division of Parks and Recreation at 801-533-4000. For information about speedboat tours sponsored by Saltair Resort, call 801-250-9660. For information at the visitor's center, call 801-533-4083.

Facts that are fun to know:

Here's why the Great Salt Lake remains salty: Often called "America's Inland Sea", the lake has no outlet. That means minerals from tributaries stay in the lake. It is estimated that more than 2 million tons of minerals are added to the lake each year. Also, the continual process of evaporation reduces the amount of water, thereby increasing the percentage of salinity.

Until recently, nothing could live in the salt solution other than blue-green algae and brine shrimp.(The adult brine shrimp are netted, frozen and sold for tropical aquarium fish.) However, in spring of 1986, because of high water and low salinity, a small fresh-water fish-the Rain Water Killie- was found thriving in great numbers. Never before in recorded history have fish been known to live in the Great Salt Lake.

Only a few species of insects survive in and around the lake. The most notorious is the brine fly, which migratory birds feed on. When the lake level is low, this tiny fly(about the size of a small fruit fly) exists in overwhelming abundance during hot summer months. Brine flies do not bite, but they are annoying.

Four thousand tons of salt is harvested every year from the Great Salt Lake. Industries also glean magnesium, used in structural metal;potash,used as fertilizer; and lithium,used as an alloy in steel, from the lake.

Since its discovery, the lake has been surrounded by myth. Early settlers claimed there was a treacherous whirlpool in the center of the lake with no escape.Witnesses reported a horrible monster crawling out of the briny depths. There was talk at one time about a school of whales and sub-terranean outlet to the Pacific Ocean.

Sources:The U.S. Department of the Interior;the Salt Lake Convention and Visitors Bureau;and the Great Salt Lake,edited by J. Wallace Gwynn and published by the Utah Department of Natural Resources.

Those of you TI USERS who may be in the area on vacation, or on business may contact the SLAVES TI USERS GROUP in Salt Lake City, at 801-262-4045 or the Ogden TI-99/4A USERS GROUP at 801-782-1004 for additional information or help with your stay in Utah.

# TI SLAVES AND OGDEN TI USERS GROUPS OFFICERS

| TI SLAVES | | OGDEN TI USERS GROUP | |
|---|---|---|---|
| PRESIDENT | BOB BEAUDOIN 262-6045 | HAROLD BINGHAM | 394-6382 |
| VICE PRES | WARREN YOUNG 278-1052 | DAVE DEHEER | 394-6815 |
| SEC/TREAS | JOE MASARONE 966-3694 | HELEN HILBURN | 773-0622 |
| LIBRARIAN | RENN CRUMP 966-7144 | ED ISLER | 825-9158 |
| ASST.LIB. | JIM FISHER 571-0804 | TONY LEAVITT | 723-3597 |
| NEWSLETTER EDITOR FOR BOTH GROUPS | | MEL BRAGG | 393-9605 |

# AUGUST 1991 NEWSLETTER

## TI SLAVES

OUR NEXT MEETING IS AUGUST 17 1991 AT 9:00 am. WE MEET IN THE DISABLED AMERICAN VETERANS HALL AT 273 E. 800 S. PLEASE BE THERE PROMPTLY.!!

Just a reminder we are having a Club picnic so come Join the fun.

## OGDEN TI USERS GROUP

OUR NEXT MEETING IS AUGUST 3 AT 9:00am. AND(AUG 10th BASIC CLASS 9:00am)AND AUG 20th 7:00 pm AND(JULY 24th 9:00am BASIC CLASS) WE MEET AT THE OGDEN MUNICIPAL AIRPORT IN THE FIRST BUILDING JUST EAST OF THE NEW TOWER.

Slaves & Otiug
1396 Lincoln APT B
Ogden, Utah 84404