

X	X	BBBB	#20
X	X	B B	
X		BBBB	By
X	X	B B	Jim
X	X	BBBB	Swedlow

[This artical original appeared in the User Group of Orange County, California Rom]

ERROR TRAPPING

We have talked before about making your programs 'user prof'. No matter what the user does, your program should have a defense. A while back I covered one area of vulnerability - when the user inputs something from the keyboard. This month the subject is error trapping.

Say for example. That the program must access a disk file to run. Fred Kluz, your program's user, puts the wrong disk in the drive (or doesn't put any disk in). What happens? Well. Your program opens a disk file and the disk controller goes to the specified drive to look for the file. When it doesn't find it, program execution stops, an error message appears on the screen and any data held is virtually lost.

There is a way around this. Two XB commands can let you decide what happens when an error occurs: ON ERROR nnn and CALL ERR().

The default condition for ON ERROR is ON ERROR STOP. This means that if an error occurs, program execution stops and an error message is displayed. The alternative is ON ERROR nnn, where 'nnn' is a line number. With this, when an error occurs, program execution transfers to the specified line number.

I do not fully understand error trapping. I can use it but I don't understand it. Once you get the hang of error trapping, try intentionally causing an error with TRACE active. You will see that the computer does not exactly go to the error instructions even though it follows them.

Here is an example of how ON ERROR works:

```
200 INPUT "File Name: ":A$
210 ON ERROR 500 ::
OPEN #1."DSK1."IA$
```

Program Continues

```

500 * Error Instructions
510 PRINT "Could not find DSK1.":A$
520 ON ERROR 540 :: CLOSE #1
530 ON ERROR STOP :: RETURN 200
540 RETURN 530

```

Fearless Fred inputs a bogus file name in line 200. We set the error trap in line 210. Our 4A tries to open a file in line 210 but can't find it. Control transfers to line 500.

First we tell Fleckless Fred that the file name was bad. Then we try and close the file. The code may seem odd, but it works. Sometimes, if you don't close the file an error will occur when you re-OPEN it but closing the file will also cause an error. So we put in an ON ERROR before closing the file just in case.

You have three options with RETURN in an ON ERROR routine. RETURN by itself will send you back to the instruction that caused the error. RETURN NEXT will return you to the very next instruction. And RETURN nnn will return you to line number nnn. These RETURN's do not work with GOSUB.

ON ERROR executes like a GOSUB. You could end the error language with a GOTO but you would create a pending RETURN that eats memory just as it does if GOSUB is not followed by RETURN. It could cause a problem if you use GOSUB later in the program!

Why the ON ERROR STOP in line 530, you ask. Well, once an ON ERROR nnn is triggered by an error, error control reverts back to ON ERROR STOP. However, I never know if the CLOSE #1 will cause an error condition and I don't want the ON ERROR 540 to be active after the file is closed, so I override it just to be safe.

Back to program flow. We had an error when opening the file, we told the user, we closed the file and we returned back to asking for a file name. The process starts over. If Fleabit Fred inputs a good file name, our program can continue. Anticipating a problem, we reset the error trap in line 210.

There is another tool you can use after ON ERROR has transferred control to error trapping language. It is CALL ERR(A,B,C,D). Look it up in your XB manual. It can tell you the error type, the line number in which the error occurred and the file number associated with the error if it is an I/O error. This information can be quite valuable in deciding what to do with an error.

A couple words of caution. First, do not add error trapping language to your program until you have completely debugged it. Otherwise, other errors in the program will be very difficult to locate.

Second, your TI executes ON ERROR STOP until you give it other instructions. In our sample program above, an error before line 210 would not trigger the error trapping language in line 500. Also, the ON ERROR 500 remains in effect until an error occurs or you execute another ON ERROR statement. This means that if an error occurs anywhere after line 210, the error message, "Could not find . . ." will appear even if it is not appropriate.

If you have more questions, just as when all else fails, read the manual -- it does give good information about XB.

WEST PENN 99'ERS CLUB INFORMATION

NEXT MEETING DATE: MAY 18 1993
 MEETING LOCATION: PENNS WOODS
 CIVIC ASSOCIATION
 JUST OFF ROUTE 30
 N. HUNTINGDON, PA
 TIME OF MEETING: 7:00 P.M.

LIST OF WEST PENN OFFICERS FOR 1993

PRESIDENT: MICKEY 412-265-5201
 VICE PRESIDENT: NORM 614-264-6442
 TREASURER: LYNN 412-835-4304
 RECORDING SEC: FRANK 412-751-6065
 CORRESPONDING SEC: PAUL 412-478-2754
 LIBRARIAN: BOB 412-863-5672
 NEWSLETTER EDITOR: RALPH 412-379-8762

GENERAL ITINERARY OF THE CLUB'S MEETING

6:45 P.M. DOORS OPEN
 7:00 P.M. GENERAL MEETING
 7:45 P.M. DEMOS AND NEW INFO
 9:45 P.M. ONE ON ONE HELP
 9:45 P.M. SOCIALIZING
 11:00 P.M. DOORS CLOSE

MEETING HIGHLIGHTS FOR THIS MONTH

LATEST T. I. NEWS AND SOFTWARE DISCOUNTS
 REPORT ON THE 1993 LIMA T. I. CONFERENCE
 "NEW SOFTWARE" FROM MS EXPRESS SOFTWARE
 "SPECIAL GUEST SPEAKER" - KEN GILLILAND
 "T. I. CASINO" - DEMO BY KEN GILLILAND
 "AUTO-CASSETTE SYSTEM" - DEMO BY MICKEY

RENEW YOUR MEMBERSHIP DUES!

\$15.00 PER YEAR FOR INDIVIDUAL / FAMILY
 \$10.00 PER YEAR FOR ONLY OUR NEWSLETTER

FREE SOFTWARE CATALOGS

Anyone interested in obtaining a free software catalog from MS Express Software should write to:

MS Express Software
 P.O. Box 498
 Richmond, Ohio 43944

WANTED

(Needing only to Borrow)

BYTE Magazine -- December 1980

If anyone has a copy of this magazine, please contact:

Mickey Cendrowski
 R.D. #1 Box 133
 Russellton, Pa 15076

412-265-5201

EXTRA!

Don't miss this month's meeting, as it promises to be an extra special one.

Highlights include, our report from the 1993 Lima Conference and a "golden opportunity" to meet with Ken Gilliland, from Notung Software.

Yes, folks, you heard me right. Ken will be attending our May 18 meeting.

Don't miss this chance to see some of Ken's latest software being demoed by the man, himself.

Who knows, he might even be persuaded to sell you a product or two, if you're interested.

The President's Two Cents

For those of you who may be wondering what happened to my column last month...what can I say...twas the season of the "EASTER (I mean DISK) BUNNY." Needless to say...I didn't get any "disk organization" done at all last month.

Getting back into the swing of things...it seems I had promised to tell you how I created the data base example that appeared in my last article...

So here goes... (DOWN AND DIRTY STYLE).

- Turn on your computer and disk system.
- Insert the Extended BASIC Module into the module port.
- Insert your TI-Base disk into Drive #1.
- Insert a blank initialized disk into Drive #2.
- Press any key to get the TI menu screen.
- Select "2" for the Extended BASIC Module.
- Your TI-Base disk will now autoload.

Once your TI-Base disk has completed its autoload procedure, you will be asked to enter the date in the following format:

MM/DD/YY

Once you have entered the date, you should see the following information displayed on your monitor or TV screen:

```
>001 *      Welcome to TI-BASE
>002 *      QUIT will terminate TI-BASE
>003 *
>004 *PRINTER EPSON
>005 SET CURSOR 2
>006 DISPLAY STATUS
>-----
>DATDISK = DSK1
>PRGDISK = DSK1
>PRINTER = PIO.CR.LF
>PAGE = 056
>HEADING = ON
>TALK = ON
>SPACES = 01
>RECNUM = ON
>LSPACE = 0256
>CURSOR = 02
>DATE = 00/00/00
>-----
>007 *      FUNCTION (7) for help.
>008 RETURN
>
```

At this point, check to see if your DATDISK is set to DSK2. If it is not, you must type in the following line at the dot prompt:

```
SET DATDISK=DSK2
```

then press the ENTER key.

Once your DATDISK has been set to DSK2, you should type in the following line at the dot prompt:

```
CREATE LIBRARY
```

then press the ENTER key.

At this point you should see the following information displayed on your monitor or TV screen:

arrows to move, enter to advance

```
FIELD  DESCRIPTOR  TYPE  WIDTH  DEC
```

```
1  _____  _____  _____  _____
```

You should now type in the following information:

```
PROGRAM      C      025
SOURCE       C      025
CODES        C      005
MOD          C      003
TYPE         C      004
DC           C      002
DN           C      002
COMMENTS    C      049
```

then press FCTN 8 to execute.

Your data base structure is now ready for input. It's as simple as that!

Let's take a closer look at each of these fields, so that you will understand them better.

```
PROGRAM - this is the name of the program.
SOURCE  - this is the source (or author) of the program.
CODES   - this requires further explanation.
         T stands for a T.I. Original program.
         C stands for a Copyrighted program.
         F stands for a Fairware program.
         P stands for a Public Domain program.
         S stands for a Single File program.
         M stands for a Multiple File program.
         - the next 3 numbers are for the Total Sectors the program requires.
MOD     - this is the Module needed to run the program.
TYPE    - this is the Type of program it is.
DC      - this is the Disk Case that the program is located in.
DN      - this is the Disk Number in that particular disk case.
COMMENTS - this is for any additional comments you may wish to make.
```

All in all, it's a DOWN AND DIRTY system, but it works for me.

Catch ya'll next month, as I continue my personal endeavor of stopping those "DISK BUNNIES" in their tracks.

PAGE PRO CATALOGER

BY NORMAN ROKKE

GIVES YOU A QUICK
HARDCOPY PRINTOUT
OF ALL OF THESE
PAGE PRO FILES

A MUST PROGRAM
FOR ANY
PAGE PRO USER

• LINE FONTS •
• SMALL FONTS •
• LARGE FONTS •
• BORDERS •
• PICTURES •
• PAGES •
• TEXT FILES •

Distributed By

MS EXPRESS SOFTWARE

P. O. Box 498

Richmond, Ohio 43944

ONLY \$14.95

Plus \$1.00

Shipping and Handling

Page Pro Cataloger requires a TI-99/4A console, a monitor or TV screen, a disk drive system (minimum configuration of 2 SS/SD drives), 32K memory expansion, a TI-99/4A compatible printer, and the Extended BASIC Module.

Page Pro Cataloger is a graphics printing utility program that should prove to be quite useful to all Page Pro users, young and old alike.

Now, no more wondering what characters are included in any particular Page Pro Font...or what that font even looks like in the first place.

No more wondering what a Page Pro Border looks like.

No more wondering what a Page Pro Picture looks like.

No more wondering what a Page Pro Page looks like.

No more wondering...period!

With Page Pro Cataloger, you can choose to print out any combination of the various Page Pro file formats, or you can choose to print out all of the Page Pro file formats that are on any particular disk.

In addition, our Page Pro Cataloger program prints out a "File Type Listing" (similar to a modified disk directory) so you'll always know what Page Pro files are on each of your disks.

If that wasn't enough, Page Pro Cataloger prints every page with ample space on each side of the page, making it easy to keep your pages in any full-page notebook. It automatically skips over your page perforations and prints each new category at the start of a new page, thus making it possible for you to catalog all of your disks without ever having to move any of your files.

ORDER YOUR COPY TODAY!

THREE COLUMN TEXT BY JOHN VUKMAN

Last couple of articles that I wrote about, covered two column News Text. This month I shall write three and four column text.

You must, as all other text programs, write your article in Funnelweb or Ti-Writer and save it. I use Ron Prewitt, Column Text III, Program to make the three and four column. This program will permit you to choose your printer - Epson - Gemini or Panasonic. Next will be asked if you want two - three or four column text. You can have Bold or Draft Print. Right justified is available.

You must set your Margins to 0 and 33 for two

column text. Setting for three and four column text is 0 and 27.

The program is very easy to work with. Instructions are easy to follow.

As you can see, this is Bold Print Three Column text.

Now to write about something to fill in this article.

Last month I wrote about Translite Commands. I found that I made a type-0 error when I gave you .TL commands, at the end of the page to try. After the first set of numbers I placed a comma, this should be a colon.

Sorry, but maybe you caught the error. It was printed in our issue #89, page 7.

Here is a little HINT for the month. I prefer to print on the TI with UPPER CASE letters. But as you can see, Lower Case characters are in this text. When I am finished typing and before I save the article, I press FTCN 6 and get to 0001 line and begin proof reading and at the same time I press - CTRL (.) Period - at each UPPER CASE character that I want printed in lower case characters. If you wish you can go from lower to UPPER case by pressing CTRL (;) semi-colon.

FOUR COLUMN TEXT

This will be four column printed text. Ron Prewitt has made a fine program for writing two, three and four column text newsletter. If you are one that does a lot of writing and you want to be different, this is the program to go with.

Just to fill in this article let me tell you how I started with the TI. Back in 1984 my kids gave me a birthday gift. I opened this long box and found it contained a Ti Home Computer. On the outside of the box was a tag which stated \$50 Rebate. (This must of been the time that TI was getting out of the Home Computer business. I can still remember, telling my wife, what am I going to do with this thing? I began reading and typing into this little computer. It got more and more interesting, but each time I used it, I had to retype

everything all over again. So I purchased a TI Recorder and then a few months later I got a Panasonic Printer. At the local High School they had five classes on Computers and this is where I started to write my own programs in Basic. I began purchasing any business program I could get on cassette. I would get into the programs a change them around to do what I needed. I would take from one program and add it to another. It got to the point that I was using my computer and Home made programs to run my sons Business and all on Cassette.

Thru a newspaper article I met Ray Brondner, then we joined West Penn 99ers. I found out about Disk Drives and purchased a PE Box. Frank Zic invited me to his home and showed me his system and gave me many pointers. Then there was

John Wilworth helping me set up printer to the PE Box. Jerry Petrulak and Gary Taylor were great teachers. But Bob McGinity, a young fellow that got, Ray, Al Kozlesky and myself to purchase Horizen Ram Disks. He assembled and installed the units in our PE Box's. At this time I showed him the programs that I had made up. He took them home, and made many improvements, that saved space and much more easier to operate. Ray and I are still in touch with Bob, but I believe the club has lost him. He has sold all his TI equipment and purchased an IBM.

Next came Drives, Al found a place to purchase DSDO Half Height Drives. We purchased six. Two for each of us. Then we added a third Drive. This where I stopped. No more additions, I am comfortable

and very busy with what I have. I do not wish to add a Modem to my collection.

I know that we all miss Jerry, I miss Al Kozesky, when Ray and I had problems with a program, we gave it to Al and he figured out how to operate it.

I purchased all of Al's TI Equipment from his wife. I have set it up at my sons home, Al's complete system. Which is being used for his business and also the grandkids are using it with games and educational disks. I have also set up TI systems at the my daughters home for my two granddaughters

Mickey talks about her bunnies. Look at the bunnies that I have accumulated since that day I said, "WHAT AM I GOING TO DO WITH THIS THING."

ON PRE-SCAN AND USER SUBS

The XB book says that the frist CALL for all subprograms must be within the active pre-scan area. It also states that all SUB and SUBEND statements must be Pre-scanned. In debugging a program. I found that the frist CALL of a user-defined sub does not need to be pre-scanned. Only CALLS of BUILT-IN subs must be pre-scanned.

This program will run without a hitch:

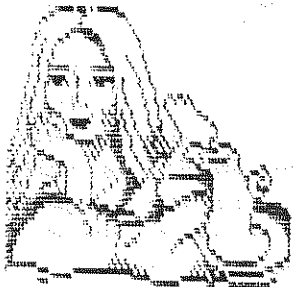
```
100 !JP-
110 CALL TEST :: END
120 !JP+
130 SUB TEST :: PRINT "OK" :: SUBEND
```

END NOTES

This is the last of the XB series for a while. Look for something new under my name next month.

ENJOY!

HAPPY MOTHERS DAY



WEST PENN 99'ERS

% Mike Sealy
RD #1 BOX 184
Toronto, OH 43964-9719

MADE IN U. S. A.

**BULK RATE
U.S. POSTAGE PAID
DONORA, PA 15033
PERMIT NO. 160**

Address Correction Requested

NEXT MEETING
MAY 18, 1993

CALL THE PUG BBS
300/1200/2400
(412) 885 3483
24 HOURS A DAY

F 731231
MICKEY CENDROWSKI
RD1 BOX 133
RUSSELTON PA 15076