Pre-Scan IT!

By J. Peter Hoddie

Introduction:

The Extended BASIC language provides many enhancements over regular TI BASIC which allow a competent programmer to write professional quality programs for all applications. Among these enhancements are the ON ERROR and ON BREAK commands, DISPLAY AT and ACCEPT AT, commands for chaining programs together, and the pre-scan feature as described in the TI Extended BASIC manual addendum.

The pre-scan feature, or more accurately being able to turn it off within a program that you know is correctly coded, lets you write programs that begin running very quickly, thus hiding the one thing about BASIC which makes it a poor language, it's speed of execution. A programmer that uses the pre-scan feature properly can create programs that are seemingly as fast as cartridges or assembly programs. It will only be apparant to the user in sorting and graphics display the actual speed. Making programs run quickly is very important to whether a program succeeds or fails, gets the job done or doesn't.

So why isn't the pre-scan feature used more? One major reason is that it is very poorly documented. The other is that it is very time-consuming to implement. The pre-scan features requires many important pre-conditions be met before you can execute the program without causing errors. In order to use the pre-scan feature you have to make an extraordinary effort to document your program as you write it. You have to list all variables, all calls, all DEFs, the OPTION BASE, and the first DATA statement at the very beginning of the program before turning off the pre-scan. Additionally, you have to make sure every variable that is used as a dimensioned array is actually defined as such in the program. If you start a program from scratch with the assumption that you are going to use the pre-scan to your advantage, then it isn't as difficult. However, most people that write programs don't do this because (a) they don't know how, (b) they actually want the error checking that leaving the Pre-Scan in entails (thus catching the programs bugs), or (c) they are lazy. There was also a large library of Extended BASIC programs available BEFORE the pre-scan feature was widely known about.

This program, Pre-Scan It!, is designed to do all the work of installing the pre-acan feature for you.

Description:

What is Pre-Scan It! and what does it do? Pre-Scan It!, in short, re-writes your program in pre-scan format, and inserts the appropriate coding to turn on and off the Pre-Scan feature when neccessary. Additionally, Pre-Scan It! will rewrite your program to save program memory, thus allowing you to make more modifications to your program at a later date. Also, because Pre-Scan It! does not change any of your variable or subroutine names, it can be used when you are writing a program with relative ease.

Finally, Pre-Scan It! will work with programs written in ANY Extended BASIC. By using special disk files that describe for Pre-Scan It! the special features of each version of Extended BASIC, this utility will function with every Extended BASIC that supports the pre-scan feature. Three files are included with the initial package for use with Myarc's Extended BASIC II, Mechatronic's Extended BASIC II+, and Miller's Graphics Extended BASIC modifications. Pre-Scan It! automatically supports TI Extended BASIC as the default version. More files will be created for other versions as neccessary and distributed free of charge.

Pre-Scan It! is the most advanced programming utilities since Smash (from Oak Tree Systems), and XB Detective (from Utilitee Software). When used in conjunction with either or both of these programs, you have a complete programming environment geared towards quickly producing programs that execute quickly.

Using Pre-Scan It!:

Before you can use Pre-Scan It! you must first prepare the program that you wish to have pre-scanned. This manual assumes that you have some knowledge of the Extended BASIC programming environment. If you don't, you may wish to express specific questions to a friend, or Asgard Software.

In order to prepare a program for Pre-Scan It!, first, enter your Extended BASIC's programming environment. Next, load the program that you wish to pre-scan. It is a good idea to RESequence the program by 100s so that there is adaquate room at the beginning of the program for any lines that Pre-Scan It! (hereafter PSI) will add. Then, save the program in MERGE format with a command in the following format:

```
SAVE DSKn.XXXXXXXXXX,MERGE
```

Where XXXXXXXXXX represents a filename different from that of the original program. After saving the program in MERGE format, insert the PSI disk and type the following to load and execute Pre-Scan It!:

```
"RUN DSK1.LOAD"
```

Or, if your program was previously saved in MERGE format, again insert the Pre-Scan It! disk in disk drive one, and the program will automatically load and execute when you select your Extended BASIC from the master menu. The load program will ask you to select the version of Pre-Scan It! that you wish to use. Select the one that best reflects the hardware configuration of your TI-99/4A system.

Once PSI starts, you will first be asked to enter the filename of the MERGE format file you wish to have pre-scanned. This is the file you saved above. Next, PSI will ask you to enter the name of the file you wish the changes that will be added to the program written to in MERGE format. You will MERGE this file back into your Extended BASIC program and re-save it to disk as the pre-scanned version.

After typing the filenames and <ENTER>, you must input the file name of the special descriptor file for the version of Extended BASIC that the program in MERGE format was written in. There are three files on the PSI disk for the following BASICs: MYARC for use with Myarc's Extended BASIC II, MECHA for use with Mechatronic's Extended BASIC II+, and MG for Miller's Graphics Extended BASIC modifications. Files may be made for other versions of Extended BASIC by modifying one of these files with TI-Writer to include any additional CALLs found in these new versions of Extended BASIC..

Simply enter the appropriate filename at this prompt, or press <ENTER> if the program you are having pre-scanned was created for TI Extended BASIC. PSI will next ask you a series of questions about specific modifications that will save memory space. If you are still working on the program, you may wish to answer "No" (N) to all these prompts. However, if you don't want to make any further changes to the program, then you'll definitely want to say "Yes" (Y), because the smaller a program is the faster it runs.

PSI will next ask if you wish to replace up to 5 numbers in the program with the variables "@", "[", "]", "_", and "\". The reason for this is because it takes only 1/3 as much space to store this type of variable as opposed to the number itself. A considerable amount of memory can be saved thus if every instance of the number "1" in a program is replaced with a "@". PSI allows you to enter the value that you wish to be replaced. PSI will also ask you if you wish to remove all REM comments in the program and text and replace them with exclamation points, thus saving a lot of program space.

The final option is a feature of PSI which will permit you to save a considerable amount of time when using this program. If you do not specify that PSI should replace constant numbers with variables, then usually the amount of pre-scanning lines to be added is minimal, therefore the resulting output file can simply be merged into the old program and the new program re-saved to disk. However, if you wish to replace program constants, then the output file created will be nearly as long as the input file, therefore you might as well include the parts of the program where no changes are made, along with the parts that are, in the output file. By answering "Y" to the query for a "Complete MERGE file", the resulting output file will be the complete program, which can simply be merged into memory as is and saved to disk in program format.

After answering these prompts, PSI will begin to process your program. If you re-sequenced the program you are processing prior to running it through PSI, you will most likely receive no error messages from PSI. If you did not resequence the program, and there is no space at the top of the program for the extra lines that turning off the pre-scan requires, then you will receive a warning from the program to resequence your program and re-run PSI to create a valid file.

Getting the most from Pre-Scan It!:

PSI has a number of obvious functions. You can use it to make existing public domain programs faster, use it when you are developing your own Extended BASIC programs to make them run quickly, or use it on those offending commercial programs of yours that do not turn the pre-scan off, and hence take enough time to run for you to take a pretty long coffee break. For the sake of your waistline and blood pressure, there are some things you may want to remember about processing commercial programs.

Before you can process any commercial program, it must first be saved in MERGE format. This may require you to save several programs this way, since many commercial (and not a few public domain) programs are broken into parts that are chained together. You will have to find out which parts must be processed, and then save each individual program. Many commercial programs will be protected with the Extended BASIC save file option of the same name. These programs may not normally be resaved - in MERGE format or in any other. In order to circumvent this you need the 32K memory expansion.

If you have it first load the program by typing in OLD DSK1. XXXXXXXXXX where the Xs represent the filename. After it has loaded from the disk, enter in the following commands:

```
CALL INIT
CALL LOAD(-32187,0)
```

The file will now be unprotected and can be saved in MERGE format. This information is provided only to aid you in processing your program, and not to promote software piracy.

After you have processed the program you may want to save it on a backup copy of the original program disk, or on the original if the program doesn't permit it. Programs that are processed with PSI are still legally copyrighted to the original manufacturer, and may not be legally exchanged or otherwise sold or distributed. Asgard Software takes no responsibility for any unauthorized copies resulting from the use or misuse of this product.

If you wish to run a compactor program (such as SMASH) on the program as well as PSI, you will want to run it through PSI first, merge the output file into the original program, and then re-save the new program before running it through the compactor.

If you use Pre-Scan It! on a program that is a hybrid of assembly language and Extended BASIC (such as the loader program for Funnelwriter), you MUST always load in the original program before MERGEing in the output file, even if you specified that PSI should make a complete MERGE file. The resulting program will not work if you don't.

If your program contains user defined sub-programs (see the appropriate Extended BASIC manual for more information on these), you must make sure that the program only contains one subprogram per program line. Otherwise, strange results may occure within the program once PSI adds the appropriate pre-scan commands.

Finally, because of the complexity of Pre-Scan It!, it is strongly recommended that you make a backup of your program before running it through PSI.

Disclaimer:

Asgard Software is not responsible for any damage incurred from the use or misuse of the product known as Pre-Scan It!. Asgard Software does not guarrantee that this product will meet the needs or expectations of the user, or in fact be *suited for any particular purpose, or function as described in this document.*

Asgard Software does agree to replace the software free of charge any time within 90 days of purchase if any damage is the result of materials or workmanship. Asgard Software may refuse to service any software that is damaged through unreasonable use, neglect, abuse, unauthorized service, or misuse during this period. Asgard Software is not responsible for any damage aquired through transporting or otherwise shipping the product to the customer.

Asgard Software in no way promotes this software, or the accompanying material for the commission of any crime as a result of it's use. Asgard Software takes no responsibility for any such use of the product. This product is sold expressly with the intention that it will be used solely by and for the benefit of the purchaser and no other, and that it will not be used in order to perform illegal acts.

Finally, Asgard Software provides this product to the user unprotected so that the user may freely exercise his/her right to safeguard his/her investment. This can in no way be construed as a liscense to the unauthorized distribution of this product. Asgard Software can in no way support the users of this product if a substantial number obtained it illegally. Be kind and support your software manufacturers.