

# **ENHANCEMENTS TO BASIC**



## ENHANCEMENTS TO BASIC INSTRUCTIONS FOR PREPARATION AND USE

FOR ALL ATARI WITH A MINIMUM OF 48K RAM AND REVISION A, B, OR C BASIC

PLEASE BE SURE TO READ ALL THE INSTRUCTIONS BEFORE USING ENHANCEMENTS TO BASIC. IMPORTANT HINTS AND PROGRAMMING INFORMATION ARE SCATTERED THROUGHOUT THE MANUAL, AND YOU MAY MISS SOMETHING IMPORTANT IF YOU FAIL TO READ THE ENTIRE MANUAL THOROUGHLY.

### COPYRIGHT CAUTION

THIS PROGRAM IS PROTECTED UNDER THE COPYRIGHT LAWS OF THE UNITED STATES OF AMERICA. EACH PROGRAM HAS A UNIQUE SERIAL NUMBER WHICH IS REGISTERED TO YOU, THE PURCHASER. YOU ARE GRANTED THE RIGHT TO MAKE COPIES AS NECESSARY FOR YOUR PERSONAL USE. COPIES FOR ANY OTHER PURPOSE ARE A VIOLATION OF THE U.S. COPYRIGHT LAWS. PLEASE USE CARE IN NOT ALLOWING OTHERS TO MAKE COPIES OF YOUR PROGRAM SINCE THIS MIGHT POSSIBLY LEAD TO COPIES WHICH ARE REGISTERED TO YOU BEING ILLEGALLY DISTRIBUTED.

WE FEEL THAT WE HAVE PROVIDED YOU WITH A QUALITY PROGRAM AT A VERY REASONABLE PRICE. IF, FOR ANY REASON, YOU DO NOT AGREE, PLEASE CONTACT US. WE VALUE YOUR BUSINESS AND ARE WILLING TO TRY TO RESOLVE ANY PROBLEMS THAT MAY ARISE IN ORDER TO KEEP YOU AS A SATISFIED CUSTOMER.

### INTRODUCTION

As all loyal ATARI owners know, ATARI BASIC is one of the most convenient programming languages available for the ATARI home computer. The error checking at the time a program line is entered has saved many programmers hours of program debugging. It is, in short, a very user friendly and easy to use programming language.

ATARI BASIC, however, does have some drawbacks and limitations. It lacks commands to access the common DOS functions. It cannot renumber program lines. It cannot delete blocks of program lines. It has no trace function to aid in debugging. Worst of all, when you are programming with ATARI BASIC, it sometimes locks up causing all your programming efforts to be lost!

ENHANCEMENTS TO BASIC will help to eliminate some of the drawbacks and limitations of ATARI BASIC. It will give you immediate mode commands to access many DOS functions. It will enable you to trace your program flow for debugging and analyzing complicated programs. It has a renumber function, a block delete function, automatic line numbering, a protect function to

save files in an executable but unreadable format, and much more. But best of all, ENHANCEMENTS TO BASIC will help to eliminate the editing lockup tendency which has been a problem for many years on the 800 and has recently shown up in a slightly different form on the 800XL. In short, ENHANCEMENTS TO BASIC makes ATARI BASIC into an excellent, user friendly, programming language.

## PREPARING YOUR COPY OF ENHANCEMENTS TO BASIC

SEVERAL SECTIONS OF STARTUP INSTRUCTIONS ARE PRESENTED HERE. BE SURE TO FIND THE INSTRUCTION SET WHICH IS PERTINENT TO THE TYPE OF DOS YOU ARE USING.

IF YOU ARE PROFICIENT IN USING YOUR DOS YOU MAY FIND THE GENERALIZED INSTRUCTIONS AT THE END OF THIS SECTION TO BE SUFFICIENT FOR YOUR USE.

\*\*\*\*\*

## STARTUP INSTRUCTIONS FOR ATARI DOS 2.0S AND ATARI DOS 2.5

1. Turn on your disk drive and monitor or television.
2. When the busy light on the disk drive goes out, open the door and insert a copy of the DOS you wish to use with ENHANCEMENTS TO BASIC.
3. Close the door on your disk drive.
4. (FOR 800 USERS ONLY) Open the door on your computer, insert the ATARI BASIC cartridge, and close the door.
5. Turn on your computer.
6. When the READY prompt appears, type PRINT PEEK(43234) [RETURN] and write down the number displayed. This number must be 96, 162, or 234. If you get any other value, recheck this step.
7. Type DOS [RETURN].
8. When the busy light on the disk drive goes out remove your DOS master diskette and insert a blank diskette.
9. Type I [RETURN]. (DOS 2.5 users - type P [RETURN] if you wish to use single density format.)
10. Type I [RETURN]. (Be sure to use the number 1 and not the letter I.)
11. Type Y [RETURN].
12. When the busy light on the disk drive goes out type H [RETURN].
13. Type I [RETURN]. (Be sure to use the number 1 and not the letter I.)
14. Type Y [RETURN].
15. When the busy light on the disk drive goes out remove the diskette just formatted and insert your ENHANCEMENTS TO BASIC master disk.
16. Type O [RETURN]. (Be sure to use letter O not number 0.)
17. If the number you wrote down earlier was 162 answer the prompt with BASICA [RETURN]. Otherwise answer the prompt with BASICRC [RETURN].

18. Press [RETURN] again.
19. When the busy light on the disk drive goes out remove the ENHANCEMENTS TO BASIC master diskette and insert the diskette you just formatted.
20. Press [RETURN].
21. When the busy light on the disk drive goes out type L [RETURN].
22. Answer the prompt with BASIC\* [RETURN].
23. When the busy light on the disk drive goes out you should see the ENHANCEMENTS TO BASIC title prompt. If you do not, an error has occurred and you must start over at step one.
24. Turn off your computer. (If you are using an ATARI 800 remove the BASIC cartridge.)
25. Turn your computer back on. The ENHANCEMENTS TO BASIC program should load and run.
26. To run ENHANCEMENTS TO BASIC in the future, you need only boot this working diskette. You may, if you wish, make backup copies of this diskette for your personal use.
27. Under DOS 2.05 the !DDF command is not available.
28. If you use the !DOS command to call the DOS menu you will not be able to return to BASIC in the normal manner. You must return to BASIC by typing M [RETURN] and answering the 'RUN AT WHAT ADDRESS' prompt with 9333 [RETURN]. If you have a MEM.SAV file on your diskette you will return to BASIC with your program intact. DO NOT PRESS [SYSTEM RESET] WHILE IN DOS. TO DO SO WILL USUALLY CAUSE A SYSTEM LOCKUP!
29. You can create a MEM.SAV file on the diskette in drive 1 from ENHANCED BASIC by entering the command !RUNAT \$1746.
30. Since most DOS commands are available from ENHANCED BASIC, the best choice would be to avoid DOS calls completely while programming in BASIC. If you need to use DOS, save your program first, then call DOS. This will prevent accidental loss of programs or data.

\*\*\*\*\*

### STARTUP INSTRUCTIONS FOR ATARI DOS 3

1. Turn on your disk drive and monitor or television.
2. When the busy light on the disk drive goes out, open the door and insert your master copy of ATARI DOS 3.
3. Close the door on your disk drive.
4. (FOR 800 USERS ONLY) Open the door on your computer, insert the ATARI BASIC cartridge, and close the door.
5. Turn on your computer.
6. When the READY prompt appears type DOS [RETURN].
7. Type T.
8. Type PRINT PEEK (43234) [RETURN].

- 9. Write down the number which is displayed. This value must be 96, 162, or 234. If you get any other value recheck this step.
- 10. Type DOS [RETURN].
- 11. Type I. (Be sure to use the letter I and not the number 1.)
- 12. Type 1 [RETURN]. (Be sure to use the number 1 and not the letter I.)
- 13. Type 1 [RETURN]. (Be sure to use the number 1 and not the letter I.)
- 14. Remove the diskette from drive 1 and insert a blank diskette.
- 15. Type Y [RETURN].
- 16. Type N [RETURN] [RETURN].
- 17. When the busy light on the disk drive goes out remove the diskette just formatted and insert your DOS 3 master diskette.
- 18. Press [ESCAPE] and wait for the busy light on the drive to go out.
- 19. Type A and wait for the busy light on the disk drive to go out.
- 20. Type 1 [RETURN]. (Be sure to use the number 1 and not the letter I.)
- 21. If the number you wrote down earlier was 162 answer the prompt with BASICA [RETURN]. Otherwise answer with BASICBC [RETURN].
- 22. Type 1 [RETURN]. (Be sure to use the number 1 and not the letter I.)
- 23. Answer the prompt with the same response as in step 21.
- 24. Remove your DOS 3 master diskette and insert your ENHANCEMENTS TO BASIC master diskette.
- 25. Press [RETURN].
- 26. When the busy light on the disk drive goes out remove your ENHANCEMENTS TO BASIC master diskette and insert the diskette you just formatted.
- 27. Press [RETURN].
28. When the busy light on the disk drive goes out remove the diskette and insert your DOS 3 master diskette.
29. Type N [RETURN].
30. When the busy light on the disk drive goes out remove your DOS 3 master diskette and insert the diskette you removed in step 28.
31. Type L
32. Type BASIC\* [RETURN].
33. Type Y [RETURN].
34. Type Y [RETURN].
35. When the busy light on the disk drive goes out you should see the ENHANCEMENTS TO BASIC title prompt. If you do not, an error has occurred and you must start over at step one.
36. Turn off your computer. (If you are using an ATARI 800 remove the BASIC cartridge.)
37. Turn your computer back on. ENHANCEMENTS TO BASIC should load and run.
38. To run ENHANCEMENTS TO BASIC in the future you need only boot this working diskette. You may, if you wish, make backup copies of this diskette for your personal use.
39. If you wish to be able to call DOS 3 from ENHANCED BASIC with the !DOS command you must copy the KCP.SYS and KCPOVER.SYS files from

your DOS 3 master diskette to your ENHANCEMENTS TO BASIC working diskette. See your DOS 3 manual for instructions on how to do this. CAUTION - IF YOU CALL DOS 3 FROM ENHANCED BASIC YOU MUST RETURN TO BASIC BY USING THE G (GO TO ADDRESS) OPTION AND ANSWERING THE PROMPT WITH 9333. WITH THIS METHOD ANY BASIC PROGRAM IS CLEARED FROM MEMORY. SINCE MOST DOS 3 FUNCTIONS ARE AVAILABLE FROM ENHANCED BASIC, WE RECOMMEND THAT YOU DO NOT USE THE !DOS COMMAND WHEN USING DOS 3.

\*\*\*\*\*

### STARTUP INSTRUCTIONS FOR OSS DOS XL

1. Turn on your disk drive and monitor or television.
2. When the busy light on the disk drive goes out, open the door and insert your single density master copy of OSS DOS XL. If you do not have a master copy of DOS XL, you must make certain that the files DOS.SYS, INIT.COM, COPY.COM, and MENU.COM are on the DOS XL diskette you insert.
3. Close the door on your disk drive.
4. (FOR 800 USERS ONLY) Open the door on your computer, insert the ATARI BASIC cartridge, and close the door.
5. Turn on your computer.
6. If you used a master copy of DOS XL you should now see the DOS XL menu. If you see the READY prompt instead, type DOS [RETURN].
7. Type T.
8. Type PRINT PEEK (43234) [RETURN].
9. Write down the number which is displayed. This value must be 96, 162, or 234. If you get any other value recheck this step.
10. Type DOS [RETURN].
11. Type I. (Be sure to use the letter I and not the number 1.)
12. When the busy light on the disk drive goes out type 2 [RETURN].
13. Type 1 [RETURN]. (Be sure to use the number 1 and not the letter 1.)
14. Remove the diskette from drive 1 and insert a blank diskette.
15. Type Y [RETURN].
16. When the busy light on the disk drive goes out press [RETURN].
17. Type 4 [RETURN].
18. Remove the diskette just formatted and insert your DOS XL master diskette into drive 1.
19. Type C.
20. If the number you wrote down earlier was 162, answer the FROM FILE: and TO FILE: prompts with BASICA [RETURN]. Otherwise answer with BASICBC [RETURN].
21. Type Y [RETURN].
22. When the busy light on the disk drive goes out remove your DOS XL master diskette and insert your ENHANCEMENTS TO BASIC master disk.
23. Press [RETURN].

24. Type Y [RETURN].
25. When the busy light on the disk drive goes out remove your ENHANCEMENTS TO BASIC master diskette and insert the diskette you previously formatted.
26. Press [RETURN].
27. When the busy light on the disk drive goes out press [RETURN] twice.
28. You should now see the DOS XL menu and should have the diskette you previously formatted in drive 1.
29. Type L.
30. Answer the prompt with BASIC\* [RETURN].
31. When the busy light on the disk drive goes out you should see the ENHANCEMENTS TO BASIC title prompt. If you do not, an error has occurred and you must start over at step one.
32. Turn off your computer. (If you are using an ATARI 800 remove the BASIC cartridge.)
33. Turn your computer back on. ENHANCEMENTS TO BASIC should load and run.
34. To run ENHANCEMENTS TO BASIC in the future you need only boot this working diskette. You may, if you wish, make backup copies of this diskette for your personal use.
35. If you use the !DOS command to access DOS XL from ENHANCED BASIC you will not be able to return to BASIC in the normal way. You must type RUN 9333 at the D1: prompt or 9333 in response to the G menu option (Go to address). These methods will erase any BASIC program in memory. If you are certain you have not done anything while in DOS XL to damage your BASIC program, you may return to BASIC with the [SYSTEM RESET] key. However, if your BASIC program is damaged a system lockup will result. As a general rule, intrinsic DOS XL commands will not damage a BASIC program, but extrinsic DOS XL commands almost always will. Also, if the DOS XL MENU.COM file is loaded BEFORE you begin programming, it will be available without damage to BASIC.
36. When using DOS XL with ENHANCEMENTS TO BASIC the MENU.COM file will not load automatically. If you wish to load it you must type DOS [RETURN] and then type MENU at the D1: prompt. Remember that the first time MENU.COM is loaded it is an extrinsic command and you must not return to BASIC with the [SYSTEM RESET] key or a system lockup will result.
37. If you are using DOS XL on an 800XL computer you may use the DOSXL.XL file to increase user memory. See your DOS XL manual for additional information.
38. If you wish to use ENHANCEMENTS TO BASIC in double density, simply use the DOS XL SDCOPY command to copy your ENHANCEMENTS TO BASIC AUTORUN.SYS file to a double density diskette.
39. Since most DOS XL commands are available from ENHANCED BASIC, we recommend that you avoid DOS calls entirely while programming in

BASIC. Save your program to disk first, then call DOS XL. This could prevent an accident which may cause you to lose programs or valuable data.

40. The !SDF and !DDF commands are not available with DOS XL. You must use the appropriate DOS XL commands instead.

\*\*\*\*\*

## STARTUP INSTRUCTIONS FOR SPARTADOS

1. Turn on your disk drive and monitor or television.
2. When the busy light on the disk drive goes out, open the door and insert your master copy of SPARTADOS.
3. Close the door on your disk drive.
4. (FOR 800 USERS ONLY) Open the door on your computer, insert the ATARI BASIC cartridge, and close the door.
5. Turn on your computer.
6. Follow the SPARTADOS routine to set the time and date.
7. When you see the D1: prompt type CAR [RETURN].
8. Type PRINT PEEK (43234) [RETURN] and write down the number which is displayed. This number must be 96, 162, or 234. If you get any other value, recheck this step.
9. Type DOS [RETURN].
10. Type FORMAT [RETURN].
11. When the busy light on the disk drive goes out type Y.
12. Press [RETURN]
13. Type 1. (Be sure to use the number 1 and not the letter I.)
14. Type 1. (Be sure to use the number 1 and not the letter I.)
15. Type 1. (Be sure to use the number 1 and not the letter I.)
16. Type EBASIC [RETURN].
17. Type N.
18. Remove your SPARTADOS master and insert a blank disk into drive 1.
19. Press [RETURN].
20. When the busy light on the disk drive goes out press [ESCAPE].
21. Remove the diskette just formatted and insert your SPARTADOS master diskette into drive 1.
22. Type SPCOPY [RETURN].
23. When the busy light on the disk drive goes out remove your SPARTADOS master diskette and insert your ENHANCEMENTS TO BASIC master diskette into drive 1.
24. Press [START].
25. If the number you previously wrote down was 162 press the space bar. If the number was 96 or 234 press [SELECT] and then press the space bar.
26. Press [START] twice.
27. When the busy light on the disk drive goes out remove your

ENHANCEMENTS TO BASIC master diskette and insert the diskette you previously formatted into drive 1.

28. Press [START] and wait for the busy light on the disk drive to go out.
29. Press [OPTION] then type `RENAME BASIC* BASIC.COM [RETURN]`.
30. When the busy light on the disk drive goes out type `BASIC [RETURN]`.
31. When the busy light on the disk drive goes out you should see the ENHANCEMENTS TO BASIC title prompt. If you do not, an error has occurred and you must start over at step one.
32. Press [SYSTEM RESET] and type `DOS [RETURN]`.
33. Type `ERASE BASIC.COM [RETURN]`.
34. When the busy light on the disk drive goes out type `RENAME AUTORUN.SYS BASIC.COM [RETURN]`.
35. When the busy light on the disk drive goes out type `COPY E: D:STARTUP.BAT [RETURN]`.
36. Type `BASIC [RETURN]`.
37. Hold down the [CTRL] key and press the 3 key.
38. When the busy light on the disk drive goes out turn off your computer. (If you are using an ATARI 800 remove the BASIC cartridge at this time.)
39. Turn your computer back on.
40. The ENHANCEMENTS TO BASIC program should load and run.
41. If you call SPARTADOS from ENHANCED BASIC with the !DOS command, you can return to BASIC with your program intact as long as you do not use any of the commands which are classified as external.
42. With the exception of the DIR command, DOS commands are not available from ENHANCED BASIC when using SPARTADOS.

\*\*\*\*\*

THE FOLLOWING ARE GENERAL INSTRUCTIONS WHICH SHOULD BE USED ONLY IF YOUR PARTICULAR DOS WAS NOT PREVIOUSLY COVERED OR IF YOU HAVE A SLIGHTLY DIFFERENT REVISION OF A DOS WHICH HAS BEEN COVERED WITH SPECIFIC INSTRUCTIONS. SINCE THESE INSTRUCTIONS ARE NOT SPECIFIC TO ANY PARTICULAR DOS, YOU WILL HAVE TO RELY ON YOUR DOS USER'S MANUAL FOR ADDITIONAL INFORMATION.

1. Turn on your disk drive and monitor or television.
2. When the busy light goes out, open the door and insert a copy of the DOS you wish to use with ENHANCEMENTS TO BASIC.
3. Close the door on your disk drive.
4. (FOR 800 USERS ONLY) Open the door on your computer, insert the ATARI BASIC cartridge, and close the door.
5. Turn on your computer.
6. When the ready prompt appears, type `PEEK(43234) [RETURN]`.
7. Write down the number which appears on your screen.
8. Type `DOS [RETURN]`
9. Using the instructions appropriate to the DOS you are using, you

must now format a blank diskette and write DOS to it. If your DOS and disk drive have double density capability, you may format in double density to conserve disk space.

10. Again using the instructions appropriate to your DOS, you must copy a file from the ENHANCEMENTS TO BASIC master diskette to the diskette you just formatted. If the number you previously wrote down was 162 you must copy the BASICA file to your formatted diskette. If the number was 96 or 234 you must copy the BASICBC file to your formatted diskette.

The files of the ENHANCEMENTS TO BASIC master diskette are in ATARI DOS 2.0S format and can be read directly by ATARI DOS 2.0S, ATARI DOS 2.5, and OSS DOS XL. If you are using another DOS you may have to use a conversion program which is supplied with your DOS in order to read the files on your master diskette and save them to your working diskette. In this case, be sure to follow the instructions for your particular conversion program carefully.

11. Once you have copied the proper file onto your working diskette, you must load and run it. Remember, this is a binary file and must be loaded with a binary load command. If you must load and run the file with separate commands, the run address must be entered as hexadecimal 76E0.
12. When this file is run it will create another file named AUTORUN.SYS on your working diskette. This file is your ENHANCEMENTS TO BASIC working copy. If your DOS supports an AUTORUN.SYS file, all you need do is boot your working copy and ENHANCEMENTS TO BASIC will load and run automatically. As an alternative, if AUTORUN.SYS is not supported, you may be able to rename the file to BASIC.COM and create a startup batch file to load and run the BASIC.COM file.
13. Depending on the type of DOS you are using, you may have to use one of a number of methods to return to BASIC from DOS. The possible methods are using the [SYSTEM RESET] key, running at the hexadecimal address 9333, or using the run cartridge option of your DOS. Some of these methods may cause lockups or loss of programs. You must experiment to see which method is appropriate to your DOS. We recommend that you avoid using DOS while programming in ENHANCED BASIC. It is best to save your program first and then call DOS. In this way you will not lose valuable programs or data.

\*\*\*\*\*

XL AND XE USERS -- AFTER YOU HAVE CREATED YOUR WORKING COPY AND ARE READY TO BOOT IT, BE SURE THAT YOU DO NOT HOLD DOWN THE OPTION BUTTON AS YOU TURN ON YOUR COMPUTER. THE ENHANCEMENTS PROGRAM AUTOMATICALLY DISABLES THE BUILT-IN BASIC ON XL AND XE COMPUTERS AND WILL NOT FUNCTION PROPERLY IF THE OPTION KEY IS HELD DOWN DURING THE BOOT PROCESS.

### WHAT YOU SHOULD EXPECT (PROGRAM LIMITATIONS AND PRECAUTIONS)

It should be mentioned at this point that ENHANCEMENTS TO BASIC is not intended to be a substitute for ATARI BASIC. It is intended to be used as a PROGRAMMING aid. The special features and capabilities of ENHANCEMENTS TO BASIC are really of very little use when running already prepared programs. The real value lies in what ENHANCEMENTS TO BASIC can do for you when you are WRITING new programs or even typing programs from a magazine.

One other precaution which should be mentioned is that while ENHANCEMENTS TO BASIC is safe from destroying itself through BASIC POKE statements, it is not protected from machine language subroutines such as USR calls which change memory contents in the area where it is stored. This area extends from 36864 (\$9000) to 49151 (\$BFFF) in the half enhancement mode and from 32768 (\$8000) to 49151 (\$BFFF) in the full enhancement and trace modes. Therefore, it would be wise to be sure you have saved to disk any program of value you have typed BEFORE you attempt to RUN it. Actually, even the cartridge version of ATARI BASIC can have a program destroyed by a single POKE into an improper memory location. No matter what the language, it is good programming procedure to SAVE your programs before trying to RUN them.

ENHANCEMENTS TO BASIC IS DESIGNED TO WORK WITH EITHER THE REVISION A BASIC CARTRIDGE (THE ONE ORIGINALLY SUPPLIED WITH THE 800 COMPUTERS) OR THE BUILT IN BASIC OF THE XL/XE COMPUTERS. ALTHOUGH WE ARE UNAWARE OF ANY, THERE MAY BE SOME REVISIONS OF BASIC WHICH ARE NOT COMPATIBLE WITH THIS PROGRAM. IF YOU ENCOUNTER A SITUATION WHICH YOU FEEL IS DUE TO THIS TYPE OF INCOMPATIBILITY, PLEASE CONTACT US. WE WILL TRY TO WORK WITH YOU TO PROVIDE YOU WITH A WORKING PROGRAM.

### THE TITLE SCREEN

When you first boot ENHANCEMENTS TO BASIC you will be greeted by the title screen. This screen gives the program name, its unique serial number, a short copyright notice, and the prompts to select the type of enhancement you need.

The title bar at the top of the title screen also indicates the type of computer on which the program was initialized. This will be shown as either 800 or XL/XE. A program initialized on an 800 will not function properly on

XL or an XE. Also, a program initialized on an XL or XE will not function on an 800. A 1200XL WILL BE TREATED AS AN 800 BY THIS PROGRAM. THIS IS NORMAL.

The last number in the title bar indicates the type of DOS the program recognized at boot time according to the following table:

- 0 - NO DOS RECOGNIZED
- 1 - SPARTADOS
- 2 - ATARI DOS 2.05
- 3 - ATARI DOS 3
- 4 - ATARI DOS 2.5
- 5 - OSS DOS XL

If you see a zero here, it means that the DOS you are using was not recognized by the ENHANCEMENTS TO BASIC program. In this case, the program should still function properly but MAY give error messages when trying to access some DOS functions from ENHANCED BASIC.

FULL ENHANCEMENT WITH TRACE is selected with the OPTION button. This gives you access to all enhancements except the HELP screen and enables TRACE. The TRACE function sets up an alternate screen to which program lines are printed as they are being executed. You have commands available to you to switch between screens at any time. This allows you to watch your program's execution for flaws or bugs and then switch to the TRACE screen to see which lines are the culprits. One word of caution: TRACE does some very elaborate maneuvering with memory locations to keep the TRACE screen safe from program graphics commands, player-missile graphics and BASIC POKE statements. The ATARI is such a versatile graphics computer that it is not possible to protect the TRACE function against all the things that the ATARI might do to invade the RAM area where TRACE resides. We would recommend, therefore that TRACE only be activated when you have a need for it, and that you have your program saved to disk before attempting to RUN it with the TRACE function active.

FULL ENHANCEMENT WITHOUT TRACE is selected with the SELECT button. This is the recommended enhancement mode as long as the extra 4K of RAM it requires is not a problem. This mode gives you access to all enhancements except the screen trace. There is, however, a printer trace mode available which can be helpful in debugging programs. It will be explained in more detail in the section on commands.

HALF ENHANCEMENT WITHOUT TRACE is selected with the START button. This mode is recommended only for use with very long programs which require the maximum amount of free memory that is available with ENHANCEMENTS TO BASIC. The

commands available in this mode have been limited to a few of the most useful in order to conserve memory space.

### USING YOUR NEW COMMANDS

ENHANCEMENTS TO BASIC makes over forty new commands available for your programming convenience. A list of the commands, their proper format, and their application follows. Remember, TRON and TROFF are the only additional program mode commands available. All other commands are only used in the immediate mode.

### COMMANDS FOR TRACING PROGRAM FLOW

THE FOLLOWING EIGHT COMMANDS RELATING TO TRACING PROGRAMS TO THE SCREEN ARE ONLY AVAILABLE IN THE FULL ENHANCEMENT WITH TRACE MODE.

#### TRON (IMMEDIATE MODE)

TRON in immediate mode causes the last contents of the TRACE screen to be displayed. This could be used, for instance, to see which instructions were being executed when your program was stopped by an error or the BREAK key. TRON will cause the normal screen editor display to become invisible, but any commands you enter from the keyboard will still be carried out when the RETURN key is pressed.

#### TRON (PROGRAM MODE)

TRON in program mode causes the TRACE screen to be displayed.

#### [SHIFT] [ESCAPE]

Holding down the SHIFT key while pressing ESCAPE as a program is running is identical to TRON in the program mode. It causes the TRACE screen to be displayed.

#### TROFF (IMMEDIATE MODE)

TROFF in immediate mode cancels TRON and causes the normal screen display to return.

## TROFF (PROGRAM MODE)

TROFF in program mode cancels TRON and causes your program's graphics, or display screen to return.

## [CONTROL] [ESCAPE]

Holding down the CONTROL key while pressing the ESCAPE key as a program is running is identical to TROFF in the program mode. It cancels TRON and causes your program's screen display to return.

## [SHIFT] [CONTROL] [ESCAPE]

When the RUN command is given the TRACE function defaults to a condition of not printing data to the hidden TRACE screen. Any TRON or TROFF command changes the default so that data are printed to the TRACE screen. This will cause the program execution to become a great deal slower because the program must wait while each line that is executed is printed to the TRACE screen. By holding down the SHIFT and CONTROL keys while pressing the ESCAPE key you can return to the default condition and speed up program execution. This will, however, prevent the TRACE screen from being updated until the next command that changes the TRACE status is encountered.

If you should need to view a program trace from the very first line, the best way to proceed is to make the first line of the program a TRON command. This will prevent loss of any trace data on the TRACE screen.

## [ESCAPE]

The ESCAPE key has the reverse effect of [SHIFT] [CONTROL] [ESCAPE]. It will cause the trace screen to start being updated but will not change the TRACE status. If TRACE is in effect, it will remain in effect. If TRACE is not in effect, it will remain disabled, but program execution will be slowed down due to the updating of the invisible TRACE screen.

\*\*\*\*\*

THIS CONCLUDES THE COMMANDS AVAILABLE ONLY IN FULL ENHANCEMENT WITH TRACE.

\*\*\*\*\*

### LNUMB (LIST ONLY LINE NUMBERS)

LNUMB will cause only the numbers of program lines to be displayed when doing a program trace or variable search.

### LLINE (LIST ENTIRE LINES)

LLINE will cancel LNUMB and cause the entire line to be displayed when doing a program trace or variable search.

### [SYSTEM RESET]

With certain programs having elaborate graphics and also with certain combinations of immediate and program mode TRON and TROFF commands, you may be unable to return the screen display to normal. In these instances use the [SYSTEM RESET] key to reinitialize the screen editor and the TRACE function. The ENHANCEMENTS TO BASIC program is protected from the [SYSTEM RESET] key and should be as safe from it as the normal ROM version of ATARI BASIC. One word of caution is necessary here - check the specific startup instructions for your DOS at the beginning of this manual for information on using [SYSTEM RESET] while in DOS. In some cases it can cause a system lockup.

### COMMANDS TO ENHANCE PROGRAMMING

THE COMMANDS FROM THIS POINT TO THE END OF THE COMMAND LIST ARE AVAILABLE IN THE FULL ENHANCEMENT WITHOUT TRACE MODE.

### NUM (AUTO LINE NUMBERING)

NUM causes line numbers to be automatically printed at the beginning of each line of programming. The default condition is to number in increments of 10 starting at 10 above the last line number currently in use.

NUM x starts numbering at x above the last line number used. The increment is also x.

NUM x,y starts numbering at line number x with an increment of y.

For example, NUM 100,25 will start numbering at line 100 and increment each line by 25. (100, 125, 150, etc.)

You can even insert numbers between existing program lines. When an existing line number is encountered the auto line numbering is cancelled and the READY prompt is shown. Also, any error in a program line will cancel the auto line numbering and print the error line so that corrections can be made.

To cancel auto line numbering you need only press RETURN on a line with no program data except the line number.

The BREAK key will cause the current line number to be skipped but you will remain in the auto line number mode. This can be used to reserve a block of program lines for inserting data at a later time.

CAUTION--If you are in the auto line numbering mode and attempt to use the cursor control keys to move the cursor away from and back into the current PHYSICAL line, the line number will be duplicated and an error message will result. If you are typing in the second or third physical line of a logical line and notice an error that requires an up or down cursor movement, the best solution is to finish the line, press RETURN twice to enter the line and exit the auto number mode, then correct the line and reenter the auto line number mode using the NUM command.

This problem is due to a bug in the ATARI operating system and is not due to the ENHANCEMENTS TO BASIC program. (Users of the ASSEMBLER EDITOR CARTRIDGE and MICROSOFT BASIC should be aware of this problem since the same operating system bug affects them also.

### REN (RENUMBER PROGRAM LINES)

REN is the default command. It causes all program lines to be renumbered using 10 as the first line number and incrementing each following line by 10.

REN x,y,z causes program lines to be renumbered starting at line x, changing line x to line y and incrementing each line after that by z. For example, REN 200,500,20 would renumber your program starting at line 200. Line 200 would be changed to 500 and each line following would be incremented by 20. (180, 190, 200, 210, 220 would become 180, 190, 500, 520, 540 etc.)

Even if only part of a program is renumbered, references to line numbers in the part which is not renumbered are also corrected to reflect the new line numbers.

Line number references containing variables cannot be renumbered. To make it easier for you to check these references, a list of up to 125 such lines will be displayed when renumbering is completed. In the unlikely event that

you have more than 125 lines with variable line number references in your program, the best solution is to insert REM statements at the beginning of some of the first problem lines in the program. The RENUMBER routine will then ignore these lines and be able to list the additional problem lines at the end of the program. As long as there are more problem lines than the RENUMBER routine is able to list, the message 'PLUS MORE' will be displayed at the end of the line list.

Line numbers which are referenced but do not exist will not be renumbered.

CAUTION--If you delete lines from a program you should try to verify that none of the deleted lines were referenced by a GOSUB, GOTO, etc. command in the remainder of the program. If you find any such references, they should be corrected before using the RENUMBER command. If they are not corrected the renumbering process may recreate the line numbers and cause some difficult to correct program bugs to appear.

NOTE--In lines containing multiple statements, the renumber command will list a line number to be checked for EACH STATEMENT containing a problem line number reference. In renumbering a line such as 10 GOSUB A: GOSUB B: GOSUB C, the number 10 (or the number to which 10 is changed) will appear three times in the list when renumbering is completed.

If renumbering cannot be completed due to memory limitations, the message 'CANNOT RENUMBER' will be displayed and your program will not be changed.

### SOME ADDITIONAL NOTES ON RENUMBERING

The table of problem lines which is displayed at the completion of renumbering also includes those lines which reference nonexistent line numbers. The key word to remember is NONEXISTANT. Consider renumbering the following program with the default command REN.

```
1 GOTO 30
2 REM
3 END
```

When renumbering is completed, the program will read:

```
10 GOTO 30
20 REM
30 END
```

Since line 30 did not exist at the time line 10 was renumbered, line 10 will be displayed as a problem line number. Remember, a problem exists even though

the program now appears correct. Since line 30 was created by the renumbering process, the program will not branch to the location originally intended.

Now consider renumbering this program:

```
1 REM
2 GOTO 10
3 END
```

Line 1 will be renumbered first and will be changed to line 10. When line 2 is renumbered, line 10 already exists (it was line 1) and the renumber command will not detect an error. An error does exist, however, and can cause some very confusing results. For this reason, when you have deleted lines, you must verify that there are no references to missing lines in your program BEFORE you renumber it.

### VER (VERIFY A BASIC PROGRAM)

The most practical way to verify a program before renumbering is to renumber an area above your highest line number. For example, if you issue the command `REN 32767,32767,1` your program will be checked for symbolic references and missing line numbers but no line numbers will be changed. You can then correct any problem line references and issue the renumber command again in whatever increments you choose. The `VER` command does this for you with a minimum amount of typing.

### !DEL (DELETE A BLOCK OF LINES)

`!DEL x` will delete line `x`. The exclamation point (!) before the command must be included. This is a precaution against accidentally deleting some of your program. In fact all commands which may cause program damage if not properly used must have the exclamation point before them.

`!DEL x,y` will delete lines `x` through `y` inclusive.

CAUTION--If the `y` argument is greater than 32767 it will be ignored and only line `x` will be deleted.

### DIR (EXAMINE THE FILE DIRECTORY)

`DIR` will display the file directory for the diskette in drive one.

`DIR x` will do the same for drive `x`.

### RENAME (RENAME A DISKETTE FILE)

RENAME "D#:OLDNAME.EXT,NEWNAME.EXT" is the format of the command to rename a diskette file. You must substitute the number of the drive you wish to access in place of the # symbol (or use D: for drive one). You must also substitute the original filename for OLDNAME.EXT and the new filename for NEWNAME.EXT. If you use the SINGLE command RENAME will only rename the first file it finds that matches your filespec. This makes it easy to recover from duplicate filenames if they should ever occur.

### LOCK (LOCK A DISKETTE FILE)

LOCK "D#:FILENAME.EXT" is the format of the command to lock a diskette file. You must change FILENAME.EXT to the name of the file you wish to lock and change the # symbol to the number of the drive you wish to access (or D: to access drive 1).

### UNLOCK (UNLOCK A DISKETTE FILE)

UNLOCK "D#:FILENAME.EXT" is the format of the command to unlock a diskette file. You must change FILENAME.EXT to the name of the file you wish to unlock and change the # symbol to the number of the drive you wish to access (or D: to access drive 1).

### !ERASE (ERASE A DISKETTE FILE)

!ERASE "D#:FILENAME.EXT" is the format of the command to erase a diskette file. Be careful with this command as it does not ask for verification of your filespec or your desire to erase. If you should accidentally erase a file you wanted to keep, you can use RESTORE to get it back. As with the previous commands of this type, you must substitute the number of the drive you wish to access for the # symbol and change FILENAME.EXT to the name of the file you wish to erase.

### !SDF (FORMAT A DISKETTE IN SINGLE DENSITY)

!SDF "D#:" is the format of the command to format a diskette in the drive specified by the # symbol (or drive 1 if only D: is typed). The !SDF command will format in single density. Be EXTREMELY careful with this command as it does not ask for any verification of your intent to format a diskette or the drive number you wish to access.

### !DDF (FORMAT A DISKETTE IN 1050 DOUBLE DENSITY)

!DDF follows the same format as the !SDF command. It will format a diskette in 1050 double density if your DOS and disk drive support this density.

### !PROT (PROTECT A BASIC PROGRAM)

!PROT will alter your program so that it is unreadable but will still execute properly. Be sure you have saved a copy of your program before using !PROT because editing is impossible on a protected program.

!PROT "D#:FILENAME.EXT" will protect your program and also save it to a diskette under the filename you specify in place of FILENAME.EXT. CAUTION- The file saved by this command will also be in the protected format and may only be RUN and not listed or edited.

### LVAR (LIST THE VARIABLES IN A BASIC PROGRAM)

LVAR will display a list of the variables used in your program. This can aid in finding unused variable names which may have crept into your programs.

### SVAR (SEARCH FOR LINES CONTAINING A SPECIFIED VARIABLE)

SVAR "var" will search a BASIC program for all lines containing the variable specified within the quotation marks. By using various combinations of the DUMP, SCREEN, LLINE, and LNUMB commands, you may list entire lines or only line numbers to either the screen or a printer. SVAR can be a great help in editing a program when you need to find and modify all the lines containing particular variables. If the variable you search for does not exist, the message 'var DOES NOT EXIST' will be displayed. If the variable exists in the variable name table but is not used in your program, the message 'var IS NOT USED IN PROGRAM' will be displayed.

### CVAR (CHANGE A VARIABLE NAME)

CVAR "old,new" will change all references to the variable specified by the old parameter to the one specified by the new parameter. CVAR "AA,BB" will change variable AA to variable BB at all locations in your program. If the variable you specify by the old parameter does not exist, the message 'old DOES NOT EXIST' will be displayed.

CAUTION--It is possible to rename a variable to just about anything that can

be displayed on the screen (except inverse characters) with the CVAR command. BASIC, however, will only recognize certain characters as legal variable names. If you change a variable to something which BASIC will not allow, your program will still RUN, will LIST to the screen properly, and can even be SAVED or LOADED. However, you will not be able to edit lines or ENTER the program again after having LISTed it to a storage device. This could be used as a protection device to keep others from decoding or modifying your programs. If string variables are renamed as numeric, etc. a program will still execute properly but will be nearly impossible to analyze or decode. Just be sure to save an unmodified copy for your own use or you may find that you are unable to understand or modify your own program!

ANOTHER CAUTION--It is possible to remove variables completely from the VNT (Variable Name Table) with the CVAR command. If you enter the command CVAR "old," (note that there is no new variable name) the variable specified by the old parameter will be deleted from the VNT completely. This will cause all the variables from that point on in the VNT to be offset from their proper location. Your program will still execute properly, but it will be confusing to read. The best solution here is to be careful with the CVAR command unless you are intentionally trying to scramble a program as a means of protection.

ONE FINAL CAUTION--Do not try to change a variable to anything containing inverse characters. If you do, each inverse character will be interpreted by BASIC as the final character of a variable name. For the following examples assume that the italic characters signify inverse characters. If you were to give the command CVAR "old,*NEW*" the variable old would be replaced by the three variables N, E, and W. An interesting consequence of this is that it is possible to insert variables at specific points in the VNT. (ATARI BASIC adds all new variables at the end of the VNT.) Assuming again that the italics are really inverse characters, the command CVAR"old,*oldnew*" would retain the variable old in the VNT and add the variable new immediately after old. This could be used to restore an accidentally deleted variable to the VNT as long as the proper location were known.

The foregoing discussion requires an in-depth knowledge of ATARI BASIC to understand thoroughly, but you can do no harm if you have your program saved to a disk so why not experiment and see what happens. You might gain some new knowledge about the internal workings of ATARI BASIC.

### !DOS (ACCESS DOS)

!DOS allows you to access DOS. With ENHANCEMENTS TO BASIC most DOS commands are available from BASIC so it is best to avoid DOS unless absolutely necessary. The normal DOS command is trapped in both immediate and program mode to help keep you from accidentally destroying your program. If you try to access DOS

from the keyboard without the exclamation point before the command, or from a program in the normal way the screen will clear and the message 'DOS INTERCEPTED' will be displayed. If the DOS command was during program execution, the line number of the DOS command will also be displayed.

### DUMP (SEND ANYTHING WHICH GOES TO THE SCREEN EDITOR TO THE PRINTER INSTEAD)

DUMP reroutes all output which would normally go to the screen editor to the printer instead. It can be used to print disk directories, program traces, variable name lists, and program responses to the printer instead of the display screen.

For example, the command sequence DUMP [RETURN] LVAR [RETURN] would print a list of all program variables to the printer instead of the screen. The command sequence DUMP [RETURN] DIR [RETURN] would print a directory of the diskette in drive one to the printer. DUMP remains in effect until it is cancelled by a SCREEN command or by SYSTEM RESET.

### SCREEN (RESTORE SCREEN EDITOR OUTPUT TO THE DISPLAY SCREEN)

SCREEN cancels the DUMP command and routes data back to the display screen.

### LMAR# (SET THE LEFT MARGIN TO #)

LMAR# will set the left margin to the value specified by the # parameter. This can be of value if your display screen suffers from overscan such that some characters are not visible on the left side of the screen. In this case you could change the default margin of two to five or six so that the entire line would be visible to you. The left and right margins may be set to values from 1 to 40. SYSTEM RESET restores the default values.

### RMAR# (SET THE RIGHT MARGIN TO #)

RMAR# will set the right margin to the value specified by the # parameter. The same reasoning as described above for the left margin also applies here.

### !RUNAT# (RUN AT ADDRESS #)

!RUNAT# is the equivalent of DOS option M (RUN AT ADDRESS). !RUNAT# will jump to a machine language subroutine at the address specified by the # parameter. If the routine ends in an RTS instruction, control will be returned to BASIC.

### !POKE x,y (POKE A VALUE INTO THE PROTECTED MEMORY AREA)

!POKE x,y will poke the value y into memory location x without regard to whether location x is a protected area of memory. THIS COMMAND SHOULD BE USED WITH GREAT CAUTION AND ONLY IF YOU ARE FAMILIAR WITH BASIC AND ENHANCED BASIC SOURCE CODE. IMPROPER USE OF THE !POKE COMMAND CAN RESULT IN A SYSTEM LOCKUP AND THE LOSS OF ANY PROGRAM IN MEMORY.

The normal POKE command is trapped in both immediate and program mode for any memory locations which are reserved by ENHANCEMENTS TO BASIC. If you attempt to POKE into protected memory, the screen will clear and the message 'POKE INTERCEPTED' will be displayed. If the POKE was during program execution the line number of the POKE command will also be displayed.

### LKB (LOCK THE KEYBOARD)

LKB will lock the keyboard to prevent unauthorized or curious people from causing damage to your programs or data when you must be away from the computer for a moment or two. To unlock the keyboard the proper three digit code must be entered. The default code to unlock the keyboard is to press the numbers one, two and three in sequence while holding down both the SHIFT and CONTROL keys.

### KEY"abc" (SET A NEW CODE TO UNLOCK THE KEYBOARD)

KEY"abc" will change the default unlocking code to the three characters represented by the abc within the quotation marks. You may use any keystrokes which will produce a character on the display screen as part of your unlocking code. We recommend that if you change the code, you should use a code you will not easily forget. The keyboard lock is pickproof! If you forget your code you will have to turn off the computer and start over.

### KDEL# (SET THE KEY DELAY TO #)

KDEL# will set the delay before a key begins to repeat to the value specified by the # parameter. The default value is 48. Values lower than ten usually result in typing errors because the keys will repeat in normal typing. A value of zero will cause the auto repeat to be disabled. SYSTEM RESET will restore the default value of 48.

### KREP# (SET THE AUTO KEY REPEAT RATE TO #) (XL/XE ONLY)

KREP# will set the auto repeat rate to the value specified by the # parameter. The default value is six. Values lower than three are generally unusable. The default value is restored by SYSTEM RESET.

### NOCLICK (TURN OFF THE KEYCLICK) (XL/XE ONLY)

NOCLICK will turn off the keyboard click without turning down the volume on the T.V. or monitor.

### CLICK (TURN ON THE KEYBOARD CLICK) (XL/XE ONLY)

CLICK cancels NOCLICK and turns the keyboard click back on.

### HELP (DISPLAY THE HELP SCREEN)

HELP will display two pages of information on commands and proper format. In depth detail is not provided. This is, instead, a listing of proper format for most commands and a short description of some.

### HEX (OUTPUT NUMERIC VALUES IN HEXADECIMAL)

HEX will cause all integers between 0 and 65535 other than those in REMark and DATA statements to be displayed in hexadecimal. The listing will revert back to decimal for negative numbers, numbers greater than 65535, and fractions. After all, how many of us can understand a hexadecimal fraction? The listing of a hexadecimal number follows the standard convention of placing a \$ (dollar sign) before the number to indicate that it is hexadecimal.

### DEC (OUTPUT NUMERIC VALUES IN DECIMAL)

DEC cancels HEX and returns program listing of numeric values to normal.

### DEF# "string" (DEFINE KEY# AS string)

The keys 4, 5, 6, 7, 8, 9, and 0, when used in conjunction with the CONTROL or CONTROL and SHIFT keys can be used to print strings of characters to the screen. The default values of these keys are as follows:

CONTROL 4 = LOAD "D:  
CONTROL 5 = SAVE "D:  
CONTROL 6 = LIST  
CONTROL 7 = RUN  
CONTROL 8 = POKE  
CONTROL 9 = PEEK (  
CONTROL 0 = DATA

SHIFT CONTROL 4 = PLOT  
SHIFT CONTROL 5 = DRAWTO  
SHIFT CONTROL 6 = LOCATE  
SHIFT CONTROL 7 = POSITION  
SHIFT CONTROL 8 = COLOR  
SHIFT CONTROL 9 = SETCOLOR  
SHIFT CONTROL 0 = GRAPHICS

In addition, the keys from 4 to 9 when used in conjunction with SHIFT and CONTROL can be redefined as strings of up to 30 characters. The 0 key, in the same manner, can be redefined to represent up to 62 characters.

DEF#"string" will redefine the key specified by the # parameter to represent the characters specified by the string parameter. You may use almost any character which will display on the screen within the string parameter. If you use escape sequences or cursor control codes within the string parameter, they will display rather than execute when you use a redefinable key. The ability to redefine keys can be very useful when you must repeat the same data (especially printer control codes using unfamiliar graphics characters) many times within the same program. As an example, if you type DEF4"FIRST BYTE", our name will be printed to the screen each time you press the 4 key while holding down the SHIFT and CONTROL keys.

DEF#" will clear the key specified by the # parameter so that nothing is printed to the screen when that key is pressed in conjunction with SHIFT and CONTROL.

### PTRON (ENABLE PRINTER TRACE)

PTRON will enable a program trace to the printer and not the display screen. With the printer trace TRON and TROFF in program mode function as described previously under the screen trace. TRON and TROFF are not available in the immediate mode with the printer trace. Also, [CONTROL] [ESCAPE] AND [SHIFT] [ESCAPE] serve the same purpose as with the screen trace.

It is recommended that PTRON not be used except when it is really necessary, and that you have your program SAVED before using the PTRON command.

## PTROFF (DISABLE PRINTER TRACE)

PTROFF disables the printer trace and returns operation to normal. This is the default condition.

## !HALF (FREE MEMORY TO SAVE A PROGRAM)

It is possible with large programs to encounter a situation in which you are unable to save a program to disk because of an out of memory error. If this happens in the FULL ENHANCEMENT or TRACE mode, you can type !HALF [RETURN] to free enough memory to allow you to save your program. You must save your program to disk immediately after using the !HALF command. You should then reboot the system and enter the HALF ENHANCEMENTS mode. If the same problem occurs in the HALF ENHANCEMENT mode, try typing CLR [RETURN] and then pressing [SYSTEM RESET]. If you are still unable to save your program, you must then delete at least one line of the program in order to free enough memory to enable you to save it.

## !COLD (COLDSTART THE COMPUTER)

Typing !COLD is equivalent to turning the power switch off and then on. It should only be used to leave ENHANCED BASIC and boot a new disk. Using the !COLD command will eliminate the power surges associated with switching the power off and on.

## RESTORING DELETED FILES

If you are using ATARI DOS 2.05 you can access a program on your ENHANCEMENTS TO BASIC master diskette which will restore accidentally deleted files as long as they have not been written over with new data. The program is named RESTORE and must be loaded with the L (BINARY LOAD) option of the DOS menu. Once the file is loaded it is self explanatory. You need only follow the prompts to accomplish your desired result. If a file has been damaged and cannot be restored, an ERROR 164 will be returned. Remember, the RESTORE program only works with ATARI DOS 2.05.

## COMMANDS AVAILABLE WITH HALF ENHANCEMENT

THE FOLLOWING COMMANDS ARE THE ONLY ONES AVAILABLE IN THE HALF ENHANCEMENT MODE. SEE THE PREVIOUS LISTING FOR A DESCRIPTION OF THEIR USE.

RENAME	NUM
REN	!DEL
DIR	DDIR
RESTORE	DRENAME
LOCK	UNLOCK
!FORMAT	DUMP
SCREEN	!ERASE

## OTHER PROGRAM CHARACTERISTICS

ENHANCEMENTS TO BASIC has several other features to make your programming safer and easier. The most important is that the documented causes of system lockups have been corrected. This should mean that you will not be subjected to accidental lockups when you are trying to edit that thousand line program you just finished and have not yet saved! Of course we cannot guarantee that there is NO POSSIBILITY of a lockup, but we HAVE fixed all the known causes. We have also fixed the 16 byte memory expansion bug of REVISION B BASIC.

IF YOU ARE A SKEPTIC -- And don't believe that the editing lockup problem on the ATARI 800 can be fixed, or perhaps don't believe that there is a lockup problem on the ATARI 800XL, you should try the following programs.

To demonstrate the editing lockup problem on the ATARI 800 enter the following program. (Lines 20 and 30 each have 107 asterisks and line 40 has 18 asterisks. Your count must be exact on these lines!!)

BE SURE TO TYPE NEW [RETURN] BEFORE ENTERING THIS PROGRAM.

```

10 REM
20 REM *****
*****
*****
30 REM *****
*****
*****
40 REM *****

```

Now type LIST [RETURN] to list the program. THIS STEP IS IMPORTANT!!

Next delete line 10 - just type 10 and press [RETURN].

If you are using standard ATARI BASIC your computer should now be in a lockup condition. With ENHANCED BASIC, of course, it should be functioning properly.

To demonstrate the editing lockup problem on the 800XL enter the following

program. (Lines 10 and 20 each have 107 asterisks and line 30 has 17 asterisks. Your count must be exact on these lines!!)

BE SURE TO TYPE NEW [RETURN] BEFORE ENTERING THIS PROGRAM.

```
10 REM *****  
*****  
*****  
20 REM *****  
*****  
*****  
30 REM *****
```

Now type LIST [RETURN] to list the program. THIS STEP IS IMPORTANT!!

Next type A=1 and press [RETURN]. As before, with standard ATARI BASIC your computer should be in a lockup condition. However, with ENHANCED BASIC, it should function properly.

ENHANCEMENTS TO BASIC also allows you to enter numeric values in either decimal or hexadecimal. Hexadecimal numbers in the range \$00 to \$FFFF are accepted as valid. To enter a number in hexadecimal you need only prefix it with a \$ (dollar sign).

If you enter values in hexadecimal, they will be converted internally to the standard ATARI BASIC format so that your programs will still be compatible with other ATARIS which may not be using ENHANCED BASIC. The one exception to this rule is that numbers in DATA statements are not converted and should be entered in decimal to maintain compatibility with other ATARIS.

The ability to accept and print out numbers in decimal or hexadecimal also makes an unusual type of calculator available to you. If you type a line in immediate mode such as PRINT \$EA+\$A6E the computer will do the math for you and print out the answer in decimal or hexadecimal depending on which mode you are currently in. You may do any type of math the ATARI is capable of, entering your numbers in decimal, hexadecimal, or any combination of both. Your results can be either in decimal or hexadecimal. To obtain a hexadecimal result the number must be a positive integer between 0 and 65535. Any number which does not meet these requirements will automatically revert to decimal.

Error messages in ENHANCED BASIC are presented as English phrases in addition to the error number. Running certain programs may cause only the error number to be listed, but this will cause no damage to your program or ENHANCED BASIC.

If you are an OMNIMON user, you will find that going to OMNIMON does not damage ENHANCED BASIC so long as you return to basic with the SYSTEM RESET

key and have not changed any critical memory values while in OMNIMON.

You should not use the DOS menu from ENHANCED BASIC to load any binary load programs which are keyboard interactive. The Special keyboard handler used by ENHANCED BASIC will still be in effect and can cause some very strange and obscure problems to occur.

In order to make room for the TRON and TROFF commands in the program mode, CSAVE and CLOAD are now only supported in the immediate mode. If you try to use CSAVE or CLOAD in the program mode they will be treated as invalid commands and an error will result.

You should remember to remove any program lines containing TRON or TROFF commands before saving the final version of any program you are writing. This will ensure that your programs will be compatible with the standard ATARI BASIC and also with ENHANCED BASIC.

The READY prompt for ENHANCEMENTS TO BASIC has been modified to indicate the type of enhancement you are currently using. The letter T indicates trace. The letter F indicates full, and the letter H indicates half.

When using the TRACE function, the [SHIFT], [CONTROL], and [ESCAPE] keys will override the program mode TRON and TROFF commands. For instance, if you have inserted TRON and/or TROFF commands into your program and then use any of the [SHIFT], [CONTROL], [ESCAPE] combinations to start or stop a trace, the program mode commands will be masked and will not function. If this should happen, you can press the space bar to give the program mode commands priority again.

It is possible (but unlikely) for an immediate mode command which encounters an error to give an improper or nonexistent line number as the location of the error. If this should happen, type LIST 0 [RETURN] to reset the BASIC line pointers and try your immediate mode line again.

ENHANCEMENTS TO BASIC does not use any of page six memory and does not alter the MEMLO pointer. These areas are free for your programming use.

REMEMBER--Some programs may not function properly in the FULL ENHANCEMENTS WITH TRACE mode. Do not use this mode unless you have a specific need for it. If you do use it, be sure you have saved your program to disk before attempting to RUN it.

We wish to emphasize one final time that ENHANCEMENTS TO BASIC is designed as a programming aid and not as a substitute for ATARI BASIC. It can be a great help to you in writing and editing programs, and even examining the operation of programs others have written. It offers no real advantage simply running

game or utility programs. Used in the proper manner, we believe you will find that ENHANCEMENTS TO BASIC is one of the most useful programs and one of the best values you will ever find for the ATARI computer.

### OUR GUARANTEE

WE GUARANTEE THAT YOU WILL RECEIVE A WORKABLE PROGRAM THAT YOU WILL BE PLEASED WITH. IF YOU ARE NOT PLEASED, LET US KNOW WITHIN THIRTY DAYS OF THE INVOICE DATE AND WE WILL MAKE ARRANGEMENTS FOR A FULL REFUND OF YOUR PURCHASE PRICE. WE CAN TAKE NO RESPONSIBILITY FOR ANY LOSS OR DAMAGE BEYOND THE PURCHASE PRICE OF THIS PROGRAM. IT IS YOUR RESPONSIBILITY TO ENSURE THAT YOU ARE PROTECTED AGAINST ANY DATA LOSS BY HAVING BACKUP COPIES OF IMPORTANT DATA AND PROGRAMS.

FIRST BYTE reserves the right to make changes and improvements to ENHANCEMENTS TO BASIC without incurring any responsibility to make the same changes and improvements to programs previously sold.

ATARI is a trademark of ATARI, INCORPORATED

DOS XL is a trademark of OPTIMIZED SYSTEMS SOFTWARE, INCORPORATED

SPARTADOS is a trademark of ICD, INCORPORATED

The ENHANCEMENTS TO BASIC program and manual are copyrighted by:

FIRST BYTE  
P. O. BOX 32  
RICES LANDING, PENNSYLVANIA  
412-627-3596