

BATPRO DESCRIPTION

BATPRO

by:

John L Darling



*Still good!
in 2009*

There are four separate BATPRO configurations tailored to match specific PET ROM code configurations. The original PET and SUPERPET are not covered by BATPRO.

VERSION	PET CONFIGURATION	ROM SLOTS
BATPRO(3.0)	40 column upgrade ROMs	\$9000-\$AFFF
BATPRO(3.1)	40 column upgrade ROMs	\$A000-\$BFFF
BATPRO(4.0)	40 column BASIC 4.0 ROMs	\$9000-\$AFFF
BATPRO(4.8)	80 column BASIC 4.0 ROMs	\$9000-\$AFFF

The user needs two Texas Instrument TMS2532 or equivalent EPROMS. The code necessary to burn the EPROMS can be obtained from a Disk supplied to the library.

The rom slots inside the 2001 SERIES PET can be located as follows: The open 24 pin socket to the far right is for \$9000-\$9FFF. The socket to its immediate left is for \$A000-\$AFFF. In the BASIC 3 PET's the next socket to the left is for \$B000-\$BFFF. When inserting the PROMS, be sure that the half-circle notch on the edge of the EPROM points toward the front of the PET.

The rom slots inside the XX32 PET BUSINESS SERIES computers can be located as follows: The furthest open 24 pin socket to the back is for \$9000-\$9FFF. The empty socket closer to the front is for \$A000-\$AFFF. When inserting the PROMS, be sure that the half-circle notch on the edge of the EPROM points toward the power transformer.

For those of you who wish to try BATPRO before burning proms, Ram versions of BATPRO are also included on the library disk. All versions use \$6000-\$7FFF of PET ram memory. Protect top of memory, load your BATPRO version, and activate it. for example:

```
POKE 53,6*16:POKE 52,0:NEW  
LOAD"BATPRO3.0-$67",8  
SYS 6*4096
```

BATPRO DESCRIPTION

BATPRO REVISION TO 1982 VERSION

The following summary describes the changes made to the 1982 version of BATPRO:

1. The following functions were removed from BATPRO:

BATPR

(a) 'BSORT'. A combined version of BSORT and FSORT, located at \$7900, now called FBSORT is provided on the disk containing this BAPRO rom code. A new working paper will be issued describing the additional features contained in 'FBSORT'. It will supercede WP82-3032. Sample BASIC programs are also provided. These sort utilities have also been debugged and modified to work on either BASIC 4.0 or 3.0 PETS.

(b) The 'CURSOR-KEY' repeat option was removed to allow consistent BATPRO configurations. The business keyboard decoding key-values differ from the graphics keyboard. This option is now covered by the 'ALL-KEYS' repeat option.

2. The BATPRO command syntax has been revised to a more consistent format.

3. Additional commands have been added:

- (a) ;EU-Erase screen UP.
- (b) ;ED-Erase screen DOWN.
- (c) ;FL-Find lines longer than a specified length.
- (d) ;FV-Find variable names used in a program.
- (e) ;FR-Find line number references.
- (f) ;GO-SYS to an address specified in DEC, OCT or HEX.
- (g) Continuously view a specified variable as a TRACE option.
- (h) Additional FIND/CHANGE search options.
- (i) ;BU-UNPACK a BASIC program by adding spaces.

(j) The 1st and 2nd cassette buffers are not used! This eliminates previous conflicts with USER programs and BATPRO. There is one exception to this. Trace with variable view uses \$027A thru \$028B in the 1st cassette buffer. If the variable view option of trace is not used this conflict does not exist.

(k) Additional error checks have been made to insure against accidental processing of 'BASIC' commands in the 'TEXT' mode.

BATPRO DESCRIPTION

4. The following bugs in BATPRO were corrected:

(a) .W (WALK) and .Q (QUICK TRACE) 8032 versions did not work and in other versions had minor problems. The keystrokes for BREAK, FAST & slow were modified to be consisteneent with all keyboards.

(b) Find/Change would not work on in all cases.

(c) The old HELP command (now called WHY ;wh) in BASIC 4.0 would not LIST the new BASIC 4.0 disk commands properly. Be aware that BASIC 4.0 disk commands will not list properly on the upgrade rom pets. Tokens do not exist for DOPEN, CATALOG etc in the BASIC 3.0 machines. If your programs need to be compatable between both PET types, DO NOT USE the BASIC 4.0 disk commands!

(d) The printer options had problems with some of the commands. They either did not provide carriage returns in the same manner as on the screen or the IEEE bus was prematurely closed.

(e) The PET stack pointer was not maintained after executing several BATPRO commands. This could result in an erroneous out of memory error if more than 20 BATPRO commands were issued without a CLR.

(f) The SAVE option in the TEXT mode did not report all the error conditions.

ACKNOWLEDGEMENT

A special thanks to Blaine D. Standage who co-authored the PLUG, an earlier version of this program. Without his help, prodding, and constructive criticism, BATPRO could not have been completed.

ATTENTION BATPRO USERS

Batpro is not for sale! This utility is intended for PET users and shall be considered Public Domain. The source code is not available for distribution without permission of the undersigned.

ERRATA SHEET

In spite of all attempts to insure that BATPRO is error free, bugs continue to surface their ugly heads. In the future an ERRATA sheet will be provided if bugs are found. If you find something, please call me.

John L. Darling
JOHN L. DARLING
(714)632-0808
(CAS)232-0808

BATPRO DESCRIPTION

INTRODUCTION

BATPRO (Basic, Assembly, & Text Processor) is a general purpose utility package intended for use with the COMMODORE PET. This utility package is an improved version of PLUG which is described in WP81-3021 and an upgrade of the 1982 BATPRO Version (WP82-3121). It presumes the user has a COMMODORE DUAL DISK DRIVE (8050, 4040 or 2040) and a COMMODORE PRINTER (2023, 2022, 4022 or 8023).

This utility package requires 8k of program space. No storing occurs in this area which makes it suitable for ROM usage. For discussion purposes, it is assumed that BATPRO is located at \$9000 thru \$AFFF, although it could be located elsewhere: ref BATPRO(3.1). It does not require ram storage space in the 2nd cassette buffer area (\$03A8-03FF). There may be a conflict with some programs due to the repeat key being active. ';KO' (REPEAT-KEY OFF) will disable the BATPRO all-keys repeat option. Reload your program and try running it again.

BATPRO contains a comprehensive Disk operating system (DOS) package. It includes many new features not found elsewhere.

XTRAMON, an extended machine language monitor, is included for the machine language user, along with a source code Text Editor and an Object Code Loader. The Loader is compatible with the COMMODORE Assembler. A revised version of that assembler which will run on BASIC 3.0 or 4.0 machines is also included on the companion disk to this working paper.

Screen print and other printer options are also available with this package.

In addition, BATPRO provides extensive editing and debugging capability for the serious PET programmer. A summary of the various options are provided with an index to their description and usage.

BATPRO DESCRIPTION

ACTIVATION

SYS(40960) to activate BATPRO. This will result in a cold start with the BATPRO and COMMODORE power on messages displayed on the screen. The ALL-KEYS repeat option is in effect and the SEARCH MODE is 'OUTSIDE-QUOTES' after activation. Note that if the bytes free are less than 31743 then the top of ram memory (\$34, \$35) was previously protected and will remain protected until manually changed. (eg:POKE 52,0:POKE 53,128:NEW)

To deactivate or kill BATPRO use the ';MK' mode command option and the PET wedge will be restored to its power on condition. Removing PET power also deactivates BATPRO.

There are 49 separate BATPRO command options. They are summarized here by function to give a brief overview of what BATPRO does. An alphabetic summary is provided as the last page in this document.

```
*****  
* MODE COMMANDS *  
*****
```

1....Mode commands are used to force the PET in a configuration to handle a specific job.

M O D E	C O M M A N D S	Page
SYS 40960	ACTIVATE BATPRO	6
;MK	KILL BATPRO	38
;MB	BASIC MODE REQUEST	37
;MT	TEXT EDITOR MODE REQUEST	41
;ML	LOADER FOR OBJECT FILES CREATED BY COMMODORE ASSEMBLER	39
;MM	XTRAMON (EXTENDED MONITOR)	40

BATPRO DESCRIPTION

```
*****
* DISK COMMANDS *
*****
```

2....Disk commands allows the use of the following options when in either the BASIC or TEXT mode.

D I S K	C O M M A N D S	Page
>	DISK COMMAND	13
/	DISK LOAD	15
]	DISK VERIFY	19
↑ (UP ARROW)	DISK LOAD/RUN	15
← (LEFT ARROW)	DISK SAVE	18
\ (BACKSLASH)	DISK APPEND	17

```
*****
* EDIT COMMANDS *
*****
```

3....Edit commands allow the data in memory to be altered. It is recommended that the file in memory be saved prior to using these commands. This will allow you to change your mind without losing the original. Even though FIND does not alter memory it belongs in this category because it is a special case of FIND/CHANGE ;FC.

E D I T C O M M A N D S				Page
Syntax	Function	Basic	Text	
;AU+	AUTO	X	X	20
;BP+	PACK	X		21
;BU+	UNPACK	X		22
;DE+	DELETE	X	X	25
;FI+	FIND	X	X	31
;FC+	CHANGE	X	X	28
;RN+	RENUMBER	X	X	46

```
*****
* SCREEN COMMANDS *
*****
```

4....Addition screen options are available to be used in conjunction with the CURSOR controls.

S C R E E N	C O M M A N D S	Page
;CS	CASE CHANGE-SCREEN ONLY	24
;CP	CASE SELECT-PRINTER & SCREEN	23
;ED	ERASE SCREEN-DOWN	26
;EU	ERASE SCREEN-UP	27

BATPRO DESCRIPTION

 * REPEAT OPTIONS *

5...;The BUSINESS SERIES PET (XX32) has an automatic cursor repeat key, but the 2001 series have no repeat key options. BATPRO has the following selectable options.

REPEAT KEY OPTIONS	Page
;KA REPEAT-ALL KEYS ON	36
;KO REPEAT-ALL KEYS OFF	36

 * PRINTER OPTIONS *

6....The following printer options are available. Note that the ;PO option can be used to cause other BATPRO options to be displayed on the printer instead of the screen.

P R I N T E R O P T I O N S	Page
;PO 'OPEN' PRINTER AS CMD DEV	44
;PC 'CLOSE' PRINTER AS CMD DEV	44
;PL 'LIST' TO PRINTER	44
;PS PRINT SCREEN (immed mode)	44
SYS 40963 PRINT SCREEN (prog. mode)	
;CP CHANGE PRINTER CASE	23

 * TRACE/DEBUG OPTIONS *

7....During the execution of a BASIC program, BATPRO has options available to assist in debugging programs.

BASIC MODE - DEBUG OPTIONS	Page
;WH WHY (was help)	54
;TC+ TRACE CONTINUOUS	49
;TS+ TRACE STEP	50
;TO+ TRACE OFF	50

The trace options can be enabled/disabled during run time within the program. Poke 124,2 To enable 'continous trace' during execution. Poke 124,3 To enable 'step function' during execution. Poke 124,0 To disable 'trace' and variable option during program execution. Poke 124,1 To disable 'trace' but not variable view option during program execution.

BATPRO DESCRIPTION

* VIEW VARIABLES OPTION *

;VA	VIEW ALL VARIABLES		53
;VI	VIEW INTEGERS only	(A1%)	53
;VS	VIEW STRINGS only	(A1%)	53
;VN	VIEW NUMBERS only (flt pt)	(A1)	53

BATPRO DESCRIPTION

* SEARCH/FIND/CHANGE *

8...The SEARCH/FIND/CHANGE commands are more powerful than ever. Take the time to understand how to use them.

SEARCH AND FIND/CHANGE OPTIONS	Page
;FI+ FIND	31
;FC+ CHANGE	28
;SA SEARCH-ALL CHARACTERS	48
;SI SEARCH-INSIDE QUOTES ONLY	48
;SO SEARCH-OUTSIDE QUOTES ONLY	48
;SS SEARCH-START OF A LINE	48
;SE SEARCH-END OF A LINE	48

* SPECIAL OPTIONS *

9....There are several miscellaneous command options available.

SPECIAL OPTIONS	Page
* DISPLAY BATRO COMMAND MENU	55
;	
= + CONVERT DEC,HEX,OCT NUMBERS	12
;GO+ SYS TO HEX,DEC OR OCT ADDR	35
;FL+ Find Lines > (+) char's	32
;FR Find Ref lines and usage	33
;FV Find Ref variables and usage	34
;UN UN-NEW PROGRAM OR TEXT	52

BATPRO DESCRIPTION

BASIC VS TEXT MODE

The normal operating mode of the PET is the BASIC MODE. In this mode, BASIC programs may be loaded, edited or executed. This is the only mode that most utility packages have. The purpose of emphasizing this is that BATPRO has an additional mode of operation where NON-BASIC files may be loaded or edited. These files could be SOURCE CODE files for assembly of machine language programs. They could also be SEQUENTIAL DISK files that normally cannot be viewed except with a BASIC program. With BATPRO these SEQUENTIAL files can be loaded, edited and saved just like a BASIC program when using the BATPRO commands! This mode is referred to as the TEXT Mode. It is this mode that distinguishes BATPRO from most other PET utilities.

The TEXT mode does not convert keywords to tokens (eg GOTO, ON = + etc). Refer to the description of the TEXT Editor Mode for a more complete understanding of the distinction between BASIC and TEXT Modes. Examples are also provided in the TEXT mode section.

;MB forces BASIC Mode.

;MT forces TEXT Mode.

BATPRO DESCRIPTION

HEX, DECIMAL, OCTAL CONVERSION

Convert a positive integer to Hex, Decimal and Octal. The command character is an (=) sign and must be the 1st non-space character on the screen line. Subsequent characters indicate the number system and the number to be converted.

SYNTAX : = N DECIMAL to OCTAL and HEX
 = @N OCTAL to DECIMAL and HEX
 = \$N HEX to DECIMAL and OCTAL

DEFAULTS : None.

RESTRICTIONS : Decimal input range :0 to 63999
 Hex Input range :0 to \$FFFF
 Octal Input range :0 to @177777

SPECIAL CASE : None.

ERROR MESSAGE: 'SYNTAX ERROR' if improper character is specified or if the range is exceeded.

EXAMPLES : = 40960 converts decimal 40960 to hex and octal.
 40960 \$=A000 @=120000

 = @123 converts octal 123 to decimal and hex.
 83 \$=0053 @=000123

 = \$A200 converts hex A200 to decimal and octal.
 41472 \$=A200 @=121000

Performing math functions(+,-,*,/, ,etc) on hex or octal numbers can be done indirectly by using this function. The procedure is to convert the numbers to decimal, perform the math function in decimal and convert the answer back to hex or octal. To subtract \$3FFE from \$CD04...

=\$CD04	=\$3FFE
52484 \$=CD04 @=146404	16382 \$=3FFE @=037776
?52484-16382	=36102
36102	36102 \$=8D06 @=106406

BATPRO DESCRIPTION

DISK SUPPORT

The Pet Disk Operating System is fully supported using a shorthand syntax similar to UNIVERSAL WEDGE or DOS(4.0). The first non-space character on a screen line must be the GREATER THAN '>' symbol. This indicates that the next characters will describe a specific Disk Command.

These shorthand options can only be used in the immediate mode. Under program control the full command syntax must be used as described in the PROGRAM Manuals.

SYNTAX : >Disk Command Syntax

Where the Disk Command can be any of the following:

- (blank)- Read Error Channel
- \$ - Directory
- C - Copy
- D - Duplicate
- I - Initialize
- N - New
- R - Rename
- S - Scratch
- V - Validate

The full syntax is described in the Pet Disk Operating Manuals. Examples are provided on the next page.

DEFAULTS : Read The Error Channel

RESTRICTIONS : None.

SPECIAL CASE : BATPRO causes Directory Commands (>\$0) to force initialization of the specified drive. This was done because the 2040 disk systems do not have automatic initialization.

MESSAGES : The Disk Error Channel is interrogated after each disk command is executed and any message other than '00 OK 00 00' will be displayed.

NOTE : Do not confuse the ';' symbol with the '>' symbol. The '>' symbol is used for Disk Commands. Also, the '@' symbol is not equivalent to '>' as is the case in some DOS packages.

BATPRO DESCRIPTION

DISK SUPPORT EXAMPLES :

1. >\$0:A*
Displays Disk Drive 0 Directory titles beginning with 'A'.
2. >\$1:*=S
Will display only the sequential files on drive 1.
3. >S1:??TABLE
Will SCRATCH any file in drive 1 whose 3RD thru 7TH characters = TABLE. (The number of files actually scratched will be displayed on the screen).
4. >R0:NEW NAME=0:OLD NAME
Will RENAME the file on Drive 0 called 'OLD NAME'
5. >N1:NEW NAME
Short form of 'NEWing' a used Disk. It will SCRATCH all files on Drive 1 and change the disk name. The time to do this is a small fraction of the time required for the full syntax, eg. >N1:NEW NAME,ID. The 2 Disk ID Characters will remain the same. This option will only work on a disk that has already been formatted.
6. >C0:NAME=1:NAME
Will copy a file called 'NAME' on drive 1 to drive 0.
7. >C0:FILENAME-A=0:FILENAME-B,FILENAME-C
Will concatenate two programs together (B and C) and save them under a new name (A).
8. >D1=0
Will duplicate the contents of the disk in drive 0 to the disk in drive 1.
9. >
Will read and display the Disk Error Channel.

BATPRO DESCRIPTION

DISK LOAD

Loads a program or sequential file from the disk into PET memory.

SYNTAX : /Dr:Filename
where 'Dr:' (drive) is optional

DEFAULTS : Searches both drives if drive not specified.

RESTRICTIONS : 1. Only PRG files will Load if in the BASIC Mode.

2. Only SEQ Files will Load if in the TEXT Mode. Line number's are added to SEQ file records during input and are not on the disk.

3. USR and REL Files are not loadable with this command.

SPECIAL CASE : LOAD/RUN : (up arrow) Dr:filename
where 'Dr:' (drive) is optional

In this case the program will execute immediately after load without typing RUN.

MESSAGE : The value of the status variable (ST) will be displayed at the end of the Load Operation along with the current operating mode (BASIC or TEXT). ST=\$40 is normal for a good load operation.

ERROR MESSAGE: If an attempt is made to load a PRG file when in the TEXT mode or to load a SEQ file when in the BASIC mode a 'file mismatch error' will be displayed with the current operating MODE.

NOTES :1. If in the BASIC Mode the Program start and end addresses will be displayed in hex.

2. The BASIC variables DS and DS\$ are not used in BATPRO for DISK status. The variable, ST, is used to maintain commonality between BASIC 3.0 and 4.0 versions.

3. After '/'(LOAD) the memory pointers are fixed based on the program in memory, not the load pointers.

4. After 'up arrow'(LOAD/RUN) the basic pointers are not fixed. The end of basic pointer is the end of load pointer. Notes 3 and 4 apply to programs saved under MONITOR control and do not contain the normal \$0401 start address. Be aware as it could cause you some difficulty. Editing of these programs would result in the same problem.

BATPRO DESCRIPTION

'DISK LOAD' EXAMPLES :

In the BASIC Mode....

1. /GALAXY*

Will LOAD the first program file it finds on either Drive whose Name starts with GALAXY.

2. (UP ARROW)0:GALAXY.W1

Will LOAD program named 'GALAXY.W1' from Drive 0 and then automatically go to RUN after a successful LOAD.

In the TEXT Mode....

1. /DATA.1*

Will LOAD the first sequential file it finds on either Drive whose Name starts with 'DATA.1'.

The first line # will be 1000 and each line number thereafter will be 10 greater than the previous line number. This sequential file can now be listed, and edited the same as a BASIC PROGRAM.

These line numbers will automatically be removed when the file is resaved to the disk under BATPRO control.

BATPRO DESCRIPTION

APPEND DISK FILE

Appends (not merges) a program or sequential file from the disk to a program or sequential file resident in PET memory. A good article on how to merge programs can be found in 'COMPUTE ! OCT 1983' magazine.

SYNTAX : (backslash)Dr:Filename
where 'Dr:' (drive) is optional

DEFAULTS : None

RESTRICTIONS : 1. Only PRG files will Append if in the BASIC Mode. Note that the start address of the PRG to be appended must be \$0401 (A BASIC PROGRAM) or it will not append.

2. Only SEQ Files will Append if in the TEXT Mode. Note that line number's are added during input and are not on the disk as part of the SEQ file.

3. USR and REL Files are not appendable using this command.

SPECIAL CASE : None

MESSAGE : The value of the status variable (ST) will be displayed at the end of the Append Operation along with the current mode of operation (BASIC or TEXT) ST=\$40 is normal.

ERROR MESSAGE: A Disk Error will be displayed with the current mode of operation if the mode differs from the file type.

NOTES : 1. If in the BASIC Mode the Program start and append end addresses will be displayed in hex.

2. Prior to Appending a BASIC program, the user must insure that the line numbers in the program to be appended are greater than the largest line number in the original program. There is no warning or error message if this condition is not met.

3. In the TEXT mode the line #'s are automatically renumbered after the APPEND operation is complete. The 1st line number will be 1000 and each successive line number will be incremented by 10, regardless of the line numbering prior to the APPEND operation.

BATPRO DESCRIPTION

SAVE FILE ON DISK

Saves a PET program or sequential file from memory to the specified Disk Drive.

SYNTAX : (left arrow)@Dr:Filename
where the '@' (replace) symbol is optional

DEFAULTS : Saves from \$0401 to end of Basic. \$28,\$29 is the start of basic pointer(LS,MS) and \$2A,\$2B is the end of basic pointer(LS,MS).

RESTRICTIONS : 1. The contents of memory will be saved on the disk as a PRG file if in the BASIC Mode.

2. The contents of memory will be saved on the disk as a SEQ Files if in the TEXT Mode. Note in the TEXT mode, that the line numbers are stripped prior to SAVE and are not put on the disk.

SPECIAL CASE : None.

MESSAGE : 'SAVING' and MODE of operation (TEXT or BASIC) after the save operation is complete.

ERROR MESSAGE: 1. 'SYNTAX ERROR' if the drive number is not specified.
2. A disk error message describing the error condition.

NOTES : 1. A 'DISK FULL' message may take several minutes before it is displayed. If a HANG CONDITION occurs.....wait a little while before resetting the PET (approx 2 minutes on the 2040).

2. XTRAMON can be used to save machine language programs: Reference Appendix-A for proper syntax.

3. Saving a text file results in the text file in memory to automatically be renumbered.

BATPRO DESCRIPTION

VERIFY DISK FILE

Verifies a PET program or sequential file in memory against the specified Filename on the Disk.

SYNTAX :]Dr:Filename
where 'Dr:' (drive) is optional

DEFAULTS : Both drives are searched if the drive is not specified.

RESTRICTIONS : 1. Only PRG files will be Verified if in the BASIC Mode

2. Only SEQ Files will be Verified if in the TEXT Mode.

Note that line number's in memory are ignored during the verify operation.

3. USR and REL Files are not verifiable with this command.

SPECIAL CASE : None

MESSAGE : 1. The value of the status variable (ST) will be displayed at the end of the Verify Operation. If the verify operation is successful an 'OK' will be displayed.

: 2. If in the BASIC Mode the PET start and end addresses will be displayed in hex.

ERROR MESSAGE: If the verify operation is not successful 'VERIFY ERROR' will be displayed.

NOTE : A hang condition may result if the Disk file to be verified is not the same length as the file in memory. Depressing the STOP key will return control to the operator, and >I will close any open read-file condition on the disk drives.

BATPRO DESCRIPTION

AUTO

Increments line number by the increment value N when editing either a BASIC line or a TEXT Editor line.

SYNTAX : ;AU N

DEFAULTS : N = 0 or blank (Disables Auto)

RESTRICTIONS : NONE

SPECIAL CASE : ;BU, ;BP, ;FL, ;FV, ;FR, ;RN, LOAD, VERIFY or APPEND also disables AUTO.

ERROR MESSAGE: 'SYNTAX ERROR' if N is specified but is not a number. The AUTO mode is disabled if this occurs.

NOTE : The editing of a line is defined as the alteration of the data following a line number. The revised data is not entered until the 'RETURN' key is depressed.

BATPRO DESCRIPTION

BASIC PACK

Packs BASIC program within specified range of line number's by removing any spaces, extra colons, or REM statements found.

SYNTAX : ;BP (Range)
(R1)
(R1-)
(-R2)
(R1-R2)

Where R1 = Start Line number
R2 = End Line number

DEFAULTS : R1 = 0, R2 = 63999

RESTRICTION : Must be in the BASIC Mode.

SPECIAL CASE : None.

ERROR MESSAGE: 1. 'SYNTAX ERROR' if Range is not specified properly.
2. 'OOPS TEXT MODE' if not in the BASIC mode.

NOTE : 1. If a GOTO, GOSUB or other line reference command specifies a line number that is removed by PACK, an undefined reference error will result during execution.

2. After PACK, use ;RN 0 to identify any unfound line numbers which may have been REM lines or lines containing only colons. To repair this type of fault, references to missing lines should be changed to the next higher existing line number.

3. Be patient! For long programs PACK may take a few minutes. A special flashing cursor will indicate that the program is being PACKed.

BATPRO DESCRIPTION

BASIC UNPACK

Unpacks a BASIC program within specified range of line number's by adding spaces within the lines to make the program more readable.

SYNTAX : ;BU (Range)
(R1)
(R1-)
(-R2)
(R1-R2)

Where R1 = Start Line number
R2 = End Line number

DEFAULTS : R1 = 0, R2 = 63999

RESTRICTION : Must be in the BASIC Mode.

SPECIAL CASE : None.

ERROR MESSAGE: 1. 'SYNTAX ERROR' if Range is not specified properly.
2. 'OOPS TEXT MODE' is displayed if in the TEXT mode.

NOTES

1. The general, a single space is installed before and after a BASIC command.

2. Existing multiple spaces will be packed to a single space.

3. A space is not added between math and function operators (+,=,-,*,/,(), ,<,>). However, if the user wishes to have a space adjacent to these operator symbols ;BU will not strip them.

4. Characters following a REM or DATA statement are not altered.

5. UNPACKing a program may result in long lines (> 80 characters). Use the ;FL command to find them. They won't affect program execution but are undesirable because of listing and editing problems.

6. Be patient! Long programs may take a few minutes. A special flashing cursor is provided as an indicator to let you know that the program is being unpacked.

BATPRO DESCRIPTION

CHANGE PRINTER CASE

Switches between upper and lower Case on the screen and displays on the screen the IEEE command to change the case on the PET printer. The 'RET' key must be hit to cause the appropriate action to take place on the printer. This allows the operator be in control and to know what's happening.

SYNTAX : ;CP Toggle screen case and display command to set printer case.

LC=> OPEN 94,4,7:PRINT#94:CLOSE 94

UC=> OPEN 94,4,8:PRINT#94:CLOSE 94

DEFAULTS : None.

RESTRICTION : None.

SPECIAL CASE : 1. During Screen Print(;PS) the screen case will also be reflected on the printout.

2. The PRINTER CASE may not change back to upper case on some early versions of the PET printer. Power the Printer off then on and it will be in the upper case mode.

ERROR MESSAGE: None.

NOTE : 1. If Screen Print (;PS) is to be used the Printer must be in the upper case mode. Screen print will automatically correct for screen case. If you are not sure which case the printer is in then turn off the printer and turn it back on. The printer will now be in step with screen print.

2. Secondary address 7 is used for printer lower case (LC) and secondary address 8 is used for printer upper case (UC).

BATPRO DESCRIPTION

CHANGE SCREEN CASE

Switches between upper (UC) and lower (LC) case on the screen.

SYNTAX : ;CS Toggle screen case only

DEFAULTS : Power on = UC on Graphics keyboard model PETS.

Power on = LC on Business keyboard model PETS.

RESTRICTION : None.

SPECIAL CASE : 1. During Screen Print(;PS) the screen case will also be reflected on the printout.

2. In the PROG mode the screen case may be selected by:
POKE 59468,12 (UC) and POKE 59468,14 (LC).

ERROR MESSAGE: None.

NOTE : Reference ;CP if printer case is also to be changed.

BATPRO DESCRIPTION

DELETE

Deletes specified range of line number's when in either the BASIC Mode or TEXT Mode.

SYNTAX : ;DE (Range)
(R1)
(R1-)
(-R2)
(R1-R2)

Where R1 = Start Line number
and R2 = End Line number

DEFAULTS: : R1 = 0, R2 = 63999

RESTRICTION : At least one range parameter must be specified.

SPECIAL CASE : NONE.

ERROR MESSAGE: 'SYNTAX ERROR' if range is not specified properly.

NOTE : To DELETE all lines use NEW.

BATPRO DESCRIPTION

ERASE SCREEN DOWN

Erases the screen from the line with the ';ED' command to the bottom of screen

SYNTAX : ;ED

DEFAULTS : NONE

RESTRICTION : NONE

SPECIAL CASE : None.

ERROR MESSAGE: NONE

NOTE : On 40 column PETS: If the screen contained a mixture of 40/80 column lines prior to erasing them, those lines will remain 40/80 column lines until they are scrolled up and off the top of screen or the 'CLR' key is pressed.

BATPRO DESCRIPTION

ERASE SCREEN UP

Erases the screen from the line with the ';EU' command to the top of screen.

SYNTAX : ;EU

DEFAULTS : NONE

RESTRICTION : NONE

SPECIAL CASE : NONE

ERROR MESSAGE: NONE

NOTE : For 40 column PETS: If the screen contained a mixture of 40/80 column lines prior to erasing them, that feature will be retained until the 'CLR' key is pressed or these screen lines are scrolled off the top of the screen.

BATPRO DESCRIPTION

FIND/CHANGE

Changes specified Search String to the New String within the range of line number's when in either the BASIC Mode or TEXT Editor Mode. The Search Option commands dictate how the strings will be scanned (;SA ;SE ;SI ;SO ;SS). After a FIND/CHANGE operation the search option used will be displayed as a reminder.

SYNTAX : ;FC/Search String/New String/, (Range)
(R1)
(R1-)
(-R2)
(R1-R2)

Where / = Any character not contained in the 'search string' or the 'new string'. This character is referred to as the delimiter.

R1 = start line number
R2 = end line number

DEFAULTS : R1 = 0, R2 = 63999 (All of memory)

RESTRICTIONS In BASIC Mode: If either S tring contains a REM or DATA statement, the delimiter must not be a Token i.e. +(plus), -(minus), *(times), /(divide), etc. This restriction is due to the rule that all characters following a REM or DATA statement are not tokenized!

SPECIAL CASE : In the BASIC mode, if the selected delimiter is a quote '"' then the characters between quotes will not be tokenized. Also, TWO QUOTE DELIMITERS ARE REQUIRED BETWEEN STRINGS to maintain the quote mode. (i.e. ;FC"chairman""chair person")

In the BASIC mode, if the selected delimiter is a not a quote '"' then the string characters will be tokenized.

In the TEXT mode, none of the characters are tokenized.

ERROR MESSAGE: 'SYNTAX ERROR' if range is not specified properly or restrictions violated.

NOTE : Changing data following a REM token requires some thought because of the way the PET ROM code handles REM statements. Anything following the 1st REM is not tokenized. Consequently changing a REM to another BASIC COMMAND will result in the the changed COMMAND to be entered as ASCII characters instead of tokens. The recommended procedure in this case is to change the REM to a non-token symbol (eg. '@', '[', 'or ']'), then changing this symbol to the desired result later. The same applies to DATA statements.

BATPRO DESCRIPTION

FIND/CHANGE EXAMPLES:

Key the following program is in memory and save it. (All the examples will use this as a starting point).

```
100 FOR I=1 TO 10 : A=A+I :REM A=SUM
110 NEXT I: PRINT "A="A
```

1. Change the variable 'A' to 'SUM' with the search option outside quotes.

```
;FC/A/SUM/
```

```
100 FOR I=1 TO 10 :SUM=A+1 :REM A=SUM
100 FOR I=1 TO 10 :SUM=SUM+1 :REM A=SUM
100 FOR I=1 TO 10 :SUM=SUM+1 :REM SUM=SUM
110 NEXT I: PRINT "A="SUM
SEARCH (OUTSIDE QUOTES)
```

2. Change the variable 'A' to 'SUM' with the search option inside quotes.

```
;FC/A/SUM/
```

```
110 NEXT I: PRINT "SUM="A
SEARCH (INSIDE QUOTES)
```

3. Change the variable 'A' to 'SUM' with the search option of all characters.

```
;FC/A/SUM/
```

```
100 FOR I=1 TO 10 :SUM=A+1 :REM A=SUM
100 FOR I=1 TO 10 :SUM=SUM+1 :REM A=SUM
100 FOR I=1 TO 10 :SUM=SUM+1 :REM SUM=SUM
110 NEXT I: PRINT "SUM ="A
110 NEXT I: PRINT "SUM ="SUM
SEARCH (ALL CHARACTERS)
```

In the above examples a line is displayed more than once because there was more than one item on a line to be changed.

BATPRO DESCRIPTION

FIND/CHANGE Examples (cont)

4. Delete 'A=' after the REM. (all character option)

```
;FC/REM A=/REM/
```

```
?SYNTAX ERROR  
READY.
```

THIS DIDN'T WORK! TRY CHANGING THE DILIMITER TO A NON TOKEN (!).

```
;FC!REM A=!REM!
```

```
100 FOR I=1 TO 10 : A=A+I :REMSUM
```

BATPRO DESCRIPTION

FIND

Finds specified characters between delimiters within the range of line number's when in either the BASIC Mode or the TEXT Editor Mode. The Search Commands dictate how the strings will be scanned. (;SA ;SE ;SI ;SO ;SS)

SYNTAX : ;FI/Srch String/, (Range)
(R1)
(R1-)
(-R2)
(R1-R2)

Where / = Any character not contained in the "srch string". This character is referred to as the delimiter.

R1 = start line number
R2 = end line number

DEFAULTS : R1 = 0, R2 = 63999 (all of memory)

RESTRICTIONS : 1. In BASIC Mode : if the Search String contains a REM or DATA statement, the delimiter must not be a Token (ie: + - * / =).

2. The quote character will not be found even if specified when in the ;SI or ;SO search modes. Quote characters can only be found in search modes ;SA, SS or SE.

SPECIAL CASE : If the selected delimiter is a quote '"' then no characters between quotes will be tokenized.

If the selected delimiter is a not a quote '"' then the search string characters will be tokenized if in the BASIC mode.

ERROR MESSAGE: 'SYNTAX ERROR' if range is not specified properly or restrictions are violated.

BATPRO DESCRIPTION

FIND LONG LINES

Finds LINES in the BASIC mode or TEXT Editor mode whose length are greater than the number specified. This length is based on a LISTed line.

SYNTAX :;FL N

Where N = NUMBER BETWEEN 0 AND 255.

DEFAULTS : N = 79

RESTRICTION : NONE.

SPECIAL CASE : NONE.

ERROR MESSAGE: NONE.

NOTE : This option is useful in determining which lines would exceed a printer line capacity, lines that could not be edited under BASIC, etc.

EXAMPLES:

;FL 0 will list all line numbers and tell you how many lines of code are in your program.

;FL 39 will identify all lines that will scroll on 40 column machines.

;FL will identify all lines that are greater than 79 characters. These lines will list as 3 lines on a 40 column PET and 2 lines on an 80 column PET. It is recommended that these lines be edited to be shorter.

```
1000 A=100*E+ASC(IS$): FOR I= 1 TO 100 STEP .5: IF I = 50 THEN  
PRINT"MESSAGE I  
=50"
```

This line could be modified into 2 lines as follows:

```
1000 A=100*E+ASC(IS$): FOR I= 1 TO 100 STEP .5  
1001 IF I = 50 THEN PRINT"MESSAGE I =50 "
```

BATPRO DESCRIPTION

FIND REFERENCE LINES

Finds all the line number references in a BASIC program and lists them to the screen in numerical order along with the number of times each reference was used. The total number of separate line references is also displayed.

SYNTAX : ;FR

DEFAULTS : NONE

RESTRICTION : BASIC MODE only

SPECIAL CASE : NONE

ERROR MESSAGE: 'OOPS TEXT MODE' if not in the BASIC mode.

NOTES 1 : The number following GOTO, GOSUB ,THEN, ON or RUN is a line reference. This number is what will be displayed.

2 : If the printer option ;PO is used the display will be routed to the printer instead of the screen. Don't forget to use ;PC to close the printer prior to the next operation.

BATPRO DESCRIPTION

FIND VARIABLES

Finds all the variables referenced in a BASIC program and lists them to the screen in alphabetical order. It also indicates how many times each variable was used. The total number of variables used in a program will also be displayed.

SYNTAX : ;FV

DEFAULTS : None.

RESTRICTION : BASIC MODE only.

SPECIAL CASE : None.

ERROR MESSAGE: 'OOPS TEXT MODE' if not in the BASIC mode.

NOTES 1 : If the printer option ;PO is used the list will be displayed on the printer instead of the screen. Don't forget to close the printer (;PC option) prior to the next operation.

2 : Single usage variables probably indicate that a variable was defined but not used (or vice-versa!).

BATPRO DESCRIPTION

AUX SYS COMMAND

Executes SYS to an address specified in decimal, hex or octal for execution of a user machine language (M.L.) program.

SYNTAX : ;GO NNNNN (DEC ADDR)
;GO \$XXXX (HEX ADDR)
;GO @YYYY (OCT ADDR)

DEFAULTS : None

RESTRICTIONS : 1. NNNNN is > 0 but < 64000
2. XXXX is > \$0 but < \$10000
3. YYYY is > @0 but < @200000

SPECIAL CASE : None

ERROR MESSAGE: SYNTAX ERROR if improper range selected. Note: there have been instances where 'SYNTAX ERROR' occurs at exit. It should not hurt anything so don't be concerned.

BATPRO DESCRIPTION

REPEAT KEY OPTIONS

Allows two REPEAT key options.

SYNTAX : ;KA KEYS-ALL All keys will repeat if held down.

: ;KO KEYS-OFF No keys will repeat if held down.

DEFAULTS : All keys will repeat after activation of BATPRO.

SPECIAL CASE : The NO-KEYS option will not disable the repeat function built into the BUSINESS keyboard PET models.

MESSAGE : One of the following messages will be displayed:

1. ALL-KEYS and current MODE
2. NO-KEYS and current MODE

NOTE : If the LOAD or SAVE commands are executed under program control then PET firmware may reset the IRQ vector to the NO-KEYS REPEAT condition. Simply reactivate repeat ';KA' if this happens.

BATPRO DESCRIPTION

COMMAND BASIC MODE

The BASIC Mode (as opposed to the TEXT Mode) will be selected.

SYNTAX : ;MB

DEFAULTS : None (BASIC Mode is the normal operating mode.)

RESTRICTION : None

SPECIAL CASE : None.

MESSAGE : 'BASIC MODE' will be displayed.

NOTE : After Load (/), Verify (]), Save (left arrow) or Append (backslash), the current MODE will be displayed to indicate proper operation.

If you don't remember if the PET is in the BASIC or the TEXT mode use the DIRECTORY ';' command and the current Mode will be displayed.

BATPRO DESCRIPTION

KILL BATPRO

Restores the Pet Wedge CHRGET routine in \$0070 thru \$0087 to the Power-On condition. It also disables the REPEAT key option to the PET power on state. BASIC Memory contents will not be affected.

SYNTAX: : ;MK

DEFAULTS: : None.

RESTRICTIONS : None.

SPECIAL CASE : To reactivate BATPRO, SYS(40960), however, the 'NEW' function is also executed during reactivation so Memory will be lost. Use the UN-NEW (;UN) command if you wish to recover the previous program in memory.

MESSAGE : 'KILLED' will be displayed.

BATPRO DESCRIPTION

ACTIVATE M.L. LOADER

Allows machine language OBJECT FILES created by the COMMODORE ASSEMBLER to be loaded into the PET as machine language programs. The object data file is a standard MOS Technology format.

SYNTAX : ;ML

DEFAULTS : None.

RESTRICTIONS : Filename must be 13 characters or less.

SPECIAL CASE : 4 digit hex offset is provided to allow for temporary relocation of M.L. code. Typically this would be used for burning of proms or comparison testing of memory.

MESSAGES : Prompting messages are provided during execution. Error messages describe any problem that may be encountered during execution of the LOADER.

NOTE : The hex offset has a wrap around feature. Suppose your code starts at \$A000. A hex offset of \$8000 will result in a code start address of \$2000.

A modified version of the Commodore Assembler is included on the disk containing the ROM code. This modification was done by Mike Sosnoski and will operate on either PET 3.0 or 4.0 BASIC. 'ASSEM(34.2)' is the filename on the BATPRO disk.

BATPRQ DESCRIPTION

COMMAND EXTENDED ML MONITOR

Causes control of the PET to be transferred from BASIC or TEXT Editor Mode to the Extended Monitor (XTRAMON).

SYNTAX : ;MM

DEFAULTS : None.

RESTRICTIONS : None.

SPECIAL CASE : .X will return control to BASIC or TEXT Mode.

MESSAGE : The 6502 registers will be displayed at entry. Only the IRQ PC & SP registers are valid at initial entry.

XTRAMON

XTRAMON provides 16 monitor commands. The XTRAMON description is provided in APPENDIX-A.

BATPRO DESCRIPTION

COMMAND TEXT MODE

The Text Mode is used for the creation and editing of ASCII data files, SEQUENTIAL DATA FILES or SOURCE CODE FILES for use with the Commodore Assembler. Refer to the Text Mode Description for more information.

SYNTAX : ;MT

DEFAULTS : None.

RESTRICTION : 1. When editing a file in memory the 1st character in any line must not be a number (0 to 9) or editing of the line will result in an error. If there are no quote characters on that line then simply make the 1st character a '"' and that line can be edited.

2. If upper and lower case characters are to be used within the text editor files the first data character should be a quote or the LIST command will not display the line properly.

SPECIAL CASE : None.

MESSAGE : 'TEXT MODE' will be displayed.

NOTE : 1. After Load (/), Verify (]), Save (left arrow) or Append (backslash), the TEXT MODE will be displayed to indicate proper SEQUENTIAL FILE access.

2. PRG files will not load, append or verify when in the TEXT Editor mode. Conversely, SEQ files will not load, append or verify in the BASIC mode.

3. If the 1st char of a SEQ type file is a quote it will be ignored if the data is retrieved via an INPUT# command. The quote character is only recognized if it is retrieved via a GET# command in a BASIC program.

BATPRO DESCRIPTION

TEXT MODE DESCRIPTION

With BATPRO the PET operates in one of two general modes: BASIC or TEXT.

The TEXT Mode is used to create and edit ASCII or TEXT files. In this mode the PET does not convert KEYWORDS (such as GOTO, REM, etc) to tokens when entering lines of data. As such it acts like a text editor. That is the major difference between the BASIC Mode and the TEXT Mode.

Originally the TEXT Mode was used to create source files for the generation of machine language programs, but examination of this mode revealed some interesting applications.

For example, a common function of the DISK unit is to utilize SEQUENTIAL data files with a BASIC program. Such a file can be created as follows:

```
PROGRAM 1
100 OPEN 3,8,3,"@0:ADDRESS,SEQ,WRITE"
150 READ A$
200 PRINT#3,A$;CHR$(13);
250 IF A$="END"THEN CLOSE 3:END
300 GOTO 150
500 DATA "MIKE FARAD*203 CAP ST*SILICON, CA 93412"
510 DATA "MILLIE HENRY*P.O BOX 1776*COULOMB, WI 45065"
520 DATA "END"
```

This file can be read with another BASIC program as follows:

```
PROGRAM 2
700 OPEN 5,8,4,"ADDRESS,SEQ,READ"
750 INPUT A$:PRINT A$
800 IF A$="END" THEN CLOSE 5:END
850 GOTO 750
```

With BATPRO the address list on the disk can be loaded as if it was a PRG file. First get in the TEXT Mode by ';MT'. Then load the data by /ADDRESS. There are no line numbers on the disk so BATPRO inserts them when reading the data. Thus the data will appear as if it was a BASIC Program and can be listed or edited in a similar manner. For example after loading use the LIST command.

```
1000 MIKE FARAD*203 CAP ST*SILICON, CA 93412
1010 MILLIE HENRY*P.O BOX 1776*COULOMB, WI 45065
1020 END
```

BATPRO DESCRIPTION

TEXT MODE DESCRIPTION (cont)

You can add or revise lines as required. Note only that in this example that 'END' should remain the last line in the file so the original BASIC program can be used to read it later.

To save the file on disk, use the shorthand syntax provided with the BATPRO DOS:

(left arrow)@0:ADDRESS

BATPRO will strip the line numbers and save the data on the disk as a SEQUENTIAL FILE. The same read program can be used as before. In addition, SEQ Disk files can be verified, or appended to other files in the same manner as PRG files are handled in the BASIC mode.

Certain precautions should be taken when creating or editing data in the TEXT Mode:

1. The LIST command can't distinguish between BASIC Mode or TEXT mode. If shifted characters are used in the data files they should be bounded within quotes. If not, the shifted characters will be treated as tokens and won't LIST properly. The LIST operation works fine if the shifted characters are in quotes. Note also that the INPUT# statement will not see leading quote characters!

Thus, it is good practice to put a quote character as the 1st char in a data file line as it avoids many of the problems associated with editing text files!

2. If the 1st character of data on a line is a number the line cannot be revised using normal edit procedures. Another method of getting around this is to insert a pseudo character between the line number and the data number and strip it later using find and change (;FC).

3. If the data contains trailing spaces at the end of the line, normal edit procedures will strip them. A pseudo character can be used here also and removed later. Enclosing the data in quotes will also preserve trailing spaces.

4. Avoid the use of commas or colons in sequential files unless there is a good reason to use them. In BASIC, INPUT#3,A\$ (for example) will ignore all characters after either of those two delimiter characters. If the ':' or ',' is inside quotes they are ok to use because the BASIC interpreter will not see them as terminator characters.

BATPRO DESCRIPTION

COMMAND PRINTER OPTIONS

Allows several printer options. Screen print (;PS) causes the contents of the screen above the cursor to be displayed on the printer.

After execution of the shorthand notation shown below for ;PL, PO, PC, the screen will display the normal IEEE printer commands on the screen for possible revision to your specific needs. The 'RET' key must be hit in order to execute the Printer commands.

SYNTAX : ;PS Print screen image above the command (;PS)
See Note 2.

: ;PL LIST current PET program in memory to Printer
OPEN 94,4:CMD 94,"CLR";:LIST
PRINT#94,"HOME";:CLOSE 94

: ;PO Open a CMD channel to the Printer and set the
paging option.
OPEN 94,4:CMD 94,"CLR";

: ;PC Close a CMD channel to the Printer, force a new
page and clear the paging option.
PRINT#94,"HOME";:CLOSE 94

DEFAULTS : None.

RESTRICTIONS : ;PC will give an error if the file was not previously opened.

SPECIAL CASE : Screen Print (;PS) prints in the Case that the screen currently displays. See NOTE 2.

ERROR MESSAGE: File not open error

NOTES : 1. The ';CP' command (preiously described) can be used to set the PRINTER case if you wish to use lower case printing for options other than Screen Print.

2. The Printer must be in the upper case mode or the opposite case will be printed. This can be forced by toggling Printer power off then on. SYS(40960+3) will also print the screen (during Program Mode). It will print the screen contents above the current cursor position and then return control to BASIC.

BATPRO DESCRIPTION

PRINTER OPTIONS (cont)

3. For the ;PL or ;PO commands, A TITLE may be added by inserting it between the 'CLR' character and the end '"' character. Also, if the "CLR" is removed then the paging mode will not be set and will remain in its previous setting.

4. For the ;PC command ,if the "HOME"; is deleted in the PRINT#94 Statement then page ejection will not occur, but the previous printer paging mode will be retained.

5. In certain instances, the PET will encounter an error and abort all IEEE files instead of properly closing them down. This may result in the PET and the Printer being out of step. The safe thing to do is power everything down and start over.

6. If a command is processed and terminated with a RETURN key, and if the cursor does not return to the 1st position of the line, then the printer was not closed properly. Refer to Note 5 if this occurs.

7. Don't use CLR or RUN after the ;PO command as the Printer will be opened but the PET will abort the opened files as described earlier.

BATPRO DESCRIPTION

RENUMBER

Renumbers specified range of line number's when in either the BASIC Mode or the TEXT Editor Mode. In the BASIC Mode it also fixes program line number references (e.g. GOTO, GOSUB, THEN, RUN, ON, GO TO) to agree with the new line numbers. RENUMBER also identifies any unfound line number references.

SYNTAX : ;RN NI,NS,(Old Range)
(R1)
(R1 -)
(- R2)
(R1 - R2)

Where NI = New Line Increment value
NS = New Start Line number
R1 = Old Start Line number
R2 = Old End Line number

DEFAULTS : NI = 10, NS = 1000, R1 = 0, R2 = 63999

RESTRICTION : NS >= R1
new start line number (NS) >= old start line # (R1)
NS+NI..+NI <= R2
new end line number must be <= old end line # (R2)

If the BASIC program contains a LIST statement the range of line numbers following the LIST will not be renumbered.

SPECIAL CASE : If N=0 then only unfound references will be printed. The program will not be renumbered.

MESSAGES : 'RANGE ERROR' if the range restrictions are violated.

'UNFOUND LINE number's' are displayed. Use FIND to see the lines containing them.

BATPRO DESCRIPTION

RENUMBER (cont)

EXAMPLES :

1. ;RN 5,2000,1000-3000

The above command will renumber a PET program or Text Editor File starting with line 1000. That line will become 2000. The next line will be 2005. This process will continue until line 3000 of the original program is renumbered.

Note that the last renumbered line must be 3000 or less. If this is not the case a 'RANGE' error will be displayed and the program will not be renumbered.

2. ;RN 15,100

This command will renumber the entire program in increments of 15. The first new line number will be 100.

3. ;RN

This command will renumber the entire program in increments of 10. The first new line number will be 1000.

4. ;RN 0

This command will NOT renumber the program. It will only report any unfound line number references.

BATPRO DESCRIPTION

SEARCH OPTIONS

A set-up instruction for FIND and FIND/CHANGE commands. It allows 5 separate search options for these two commands (;FI & ;FC).

Once set, this option will remain until changed by another SEARCH command. After a FIND or FIND/CHANGE operation the SEARCH Mode will be displayed to remind you what has been searched.

A pictorial display is given to describe the SEARCH Mode as shown below: '[' is used to indicate start of 'RVS' and ']' indicates the end of the 'RVS' field.

SYNTAX : ;SA (All Characters) [.".".]
;SI (Inside Quotes) ."["."].
;SO (Outside Quotes) .]"["."].
;SS (Start of Line) [."]."
;SE (End of Line) ."["."].

DEFAULT : ;SO' after activation of BATPRO.

RESTRICTIONS : Reference ;FI (FIND) and ;FC (FIND/CHANGE).

SPECIAL CASE : None.

ERROR MESSAGE: None.

NOTES : 1. SEARCH-ALL (;SA) will cause the entire line to be scanned during ;FI(FIND) or ;FC(FIND/CHANGE).

2. SEARCH-INSIDE QUOTES (;SI) will only search for characters contained between quotes during ;FI or ;FC.

3. SEARCH-OUTSIDE QUOTES (;SO) will only search for characters not contained between quotes during ;FI or ;FC.

4. SEARCH-START OF LINE (;SS) will cause only the first characters in a line to be scanned during ;FI or ;FC.

5. SEARCH-END OF LINE (;SE) will cause only the last characters in a line to be scanned during ;FI or ;FC.

Quote characters can only be found as part of the search string in the following search modes: ;SA ;SS ;SE. They cannot be found in ;SI or ;SO as the quote character defines the limits of the search in these modes.

BATPRO DESCRIPTION

TRACE BASIC PROGRAMS

When executing a basic program it is often desirable to trace the execution for debugging purposes. Three TRACE mode commands allow for this feature: Trace Continuously, Step and Off. With each option a BASIC variable may be selected and it's value will be displayed on the top line of the screen during program execution.

SYNTAX : ;TC (VAR NAME) TRACE CONTINUOUSLY
 ;TS (VAR NAME) STEP
 ;TO (VAR NAME) OFF

where (VAR NAME) is optional

On 40 column screens the trace displays 6 windows on 6 screen lines. On 80 column screens the trace displays two windows on 3 lines.

TRACE CONTINUOUS

Allows viewing of last 5 line numbers executed plus the current line being executed in a BASIC program. The line numbers are displayed on the screen in reverse video. The bottom line number is the most recent line executed.

SYNTAX : ;TC (var name)

DEFAULTS : No variable selected

RESTRICTIONS : 1. Must be in the BASIC Mode
 2. Array variables cannot be set for viewing.

 3. If the VAR NAME option is used the 1st cassette buffer (\$027A -\$028B) is used. This will conflict with any program using this area.

SPECIAL CASE : TRACE OFF ';TO' disables TRACE.

MESSAGE : None, although TRACE window is displayed to indicate that TRACE is in effect.

ERROR MESSAGE: None

NOTE : BASIC execution time is slower than normal by a factor of 10 (approx).

BATPROQ DESCRIPTION

TRACE-STEP

This is a special case of TRACE. STEP allows execution of one BASIC line. The 'shift' key must be pressed and released to execute the next program line. The current line number will be displayed in the bottom TRACE window.

SYNTAX : ;TS (var name)

DEFAULTS : No variable selected

RESTRICTIONS : 1. Must be in the BASIC Mode
2. Array variables cannot be set for viewing.

3. If the VAR NAME option is used the 1st cassette buffer (\$027A -\$028B) is used. This will conflict with any program using this area.

SPECIAL CASE : TRACE OFF ';TO' disables STEP

MESSAGE : None, although the 6 line reverse-video window and the value of any selected variable will be displayed during BASIC execution.

ERROR MESSAGE: None

TRACE-OFF

Disables TRACE (;TC & ;TS). The Reverse field line number windows will no longer be displayed and the PET will resume normal speed.

SYNTAX : ;TO (var name)

DEFAULTS : No variable selected

RESTRICTIONS : Same as TRACE or STEP.

SPECIAL CASE : Reference STEP or TRACE.

ERROR MESSAGE: None

BATPRO DESCRIPTION

TRACE-APPLICATIONS

Trace can be activated and deactivated DURING program execution. Temporarily inserting POKE commands in your program will allow you more flexibility in program debug. Don't forget to remove these commands when you are done!

POKE 124,0: Turns TRACE off. (;TO)
POKE 124,1: Variable Display Only. (;TO) Reference Note 1.
POKE 124,2: Turns TRACE On. (;TC)
POKE 124,3: Turns STEP On. (;TS)

NOTE 1. The VAR NAME must have been previously selected by a TRACE command while in the immediate mode.

VIEWING VARIABLES

The value of the real time clock TI\$ can be displayed continuously on the screen without impacting program speed appreciably by the following command:

```
;TO TI$
```

If a string variable is selected and it contains cursor movements, these cursor movements will be executed as in any print statement!

Any non array variable may be selected. It will be displayed each time a new line number is encountered during execution of the program.

BATPRO DESCRIPTION

UN-NEW PROGRAM MEMORY

If a program or text file was inadvertently erased by a NEW command or a BATPRO activation, it can be restored by the UN-NEW command.

SYNTAX : ;UN Un-new program memory.

DEFAULTS : None.

RESTRICTION : None.

SPECIAL CASE : If there is no program in memory, the contents of memory will not be altered. Only the first 250 characters will be scanned for a valid file.

MESSAGE : ERROR will be displayed if the UN-NEW operation cannot be performed.

NOTE : Un-new can be used in either the BASIC Mode or the TEXT Editor Mode.

BATPRO DESCRIPTION

VIEW VARIABLES

Displays current value of the variables created during execution of a BASIC program. VIEW is normally performed after hitting the STOP key or after a programmed STOP or END statement or aafter an error encountered in the BASIC program.

SYNTAX : ;VA (All variables)
;VI (Only Integers)
;VS (Only Strings)
;VN (Only Real Number Variables)

DEFAULT : NONE

RESTRICTIONS : 1. Dimensioned variables [eg. A\$(N), A(N), A%(N)] and Function variables (DEF FN) are not displayed.

2. VIEW must be done prior to editing a line or executing a NEW or CLR Basic statement and prior to any other BATPRO commands which may interfere with variable data (eg. ;FV ;FR).

SPECIAL CASE : None.

ERROR MESSAGE: None.

NOTE : One method to display dimensioned variables in the Immediate Mode is as follows:

```
FOR I=0 TO 5:PRINT A$(I):NEXT I
```

BATPRO DESCRIPTION

WHY, WHAT, WHO

Displays last BASIC line number executed prior to halting. The last part of the line which was executed will be highlighted. In the case of an ERROR halt the display will highlight the portion of the line where the error was detected.

SYNTAX : ;WH

DEFAULTS : None.

RESTRICTIONS : BASIC MODE ONLY. Must be commanded immediately after halting. Other immediate mode commands may affect the line number pointer and result in no display of a BASIC line.

SPECIAL CASE : None.

ERROR MESSAGE: None.

BATPRO DESCRIPTION

COMMAND MENU

A MENU of the 49 command options, search mode, repeat key option and current operating mode (text or basic) are displayed.

SYNTAX : * or ;

DEFAULTS : None.

RESTRICTIONS : None.

SPECIAL CASE : If improper letter (e.g. *F) is specified or only part of the BATPRO syntax is used, eg. ;F, the Command MENU will also be displayed.

ERROR MESSAGE: None.

```
SEARCH .["."].
=      >      /      .      .      .      ]      *
;AU   ;BP   ;BU   ;CP   ;CS   ;DE   ;ED   ;EU
;FC   ;FI   ;FL   ;FR   ;FV   ;GO   ;KA   ;KO
;MB   ;MK   ;ML   ;MM   ;MT   ;PC   ;PL   ;PO
;PS   ;RN   ;SA   ;SE   ;SI   ;SO   ;SS   ;TC
;TO   ;TS   ;UN   ;VA   ;VI   ;VN   ;VS   ;WH
;
ALL-KEYS [ BASIC MODE ]
```

NOTE : An alphabetic index of all of the 49 commands are shown on the last page of this document.

XTRAMON

XTRAMON is an extension of the existing PET MONITOR. The extended MONITOR command set is summarized as follows:

XTRAMON INSTRUCTION LIST

(-)	=	CBM RESIDENT MONITOR INSTRUCTIONS	
(*)	=	INSTRUCTIONS ADDED BY XTRAMON	
			Page
(*)	A SIMPLE ASSEMBLER	57
(*)	B BREAK SET	57
(*)	C COMPARE MEMORY	57
(*)	D DISASSEMBLER	58
(*)	F FILL MEMORY	58
(-)	G GOTO	58
(*)	H HUNT MEMORY	59
(*)	I INTERPRET MEMORY	59
(-)	L LOAD MEMORY FROM IEEE	60
(-)	M MEMORY DISPLAY	60
(*)	N NEW LOCATOR	61
(*)	Q QUICK TRACE	61
(-)	R REGISTER DISPLAY	62
(-)	S SAVE TO OUTPUT DEVICE	62
(*)	T TRANSFER MEMORY	63
(*)	W WALK CODE	63
(-)	X EXIT TO BASIC	64

In the following paragraphs the operation of XTRAMON is shown by a series of examples.

SIMPLE ASSEMBLER

```
.A 2000 A9 12    LDA #12  
.A 2002 9D 00 80 STA $8000,X  
.A 2005 DEX:(garbage)
```

The user entered 'A 2000 LDA #12' to start assembly at \$2000 with the instruction 'LDA #12'. The ASSEMBLER retyped the first line inserting the two HEX CODE bytes and typed the ';A 2002' PROMPT for the next entry.

To exit from the ASSEMBLER, Type a RETURN after any new line address prompted.

A colon ':' can be used to terminate a line as illustrated in LINE 3 of the EXAMPLE. THE 'garbage' after the ':' will be deleted when the computer retypes the line

If a syntax error occurs in the coding, the line will not be assembled and a '?' prompt will appear.

BREAK SET

```
.B 1000 0004
```

BREAK is used with QUICK TRACE. This example sets a BREAK at location \$1000. After the instruction at \$1000 has been executed 0004 (hex) times during QUICK TRACE, a break will occur.

A BREAK set with a \$0000 or BLANK count will STOP at the first occurrence of the BREAK address.

COMPARE MEMORY

```
.C 1000 2000 5000
```

This example compares memory from \$1000 THRU \$2000 to the contents of memory beginning at \$5000. COMPARE will print the address of all UNEQUAL compares. The list may be produced in ascending or descending address order, depending on how the range was selected.

DISASSEMBLER

```
.D 2000
., 2000 A9 12    LDA #$12
., 2002 9D 00 80 STA $8000,X
., 2005 AA      TAX
```

This example begins DISASSEMBLY of memory contents starting at address \$2000 and will continue until terminated by the 'STOP' key. The display may be slowed by pressing the 'OFF/REV' key (left arrow key on business models).

With the DISASSEMBLY on the screen, the bytes following the address may be changed. When 'RETURN' is pressed the bytes in memory will be changed and the line will be DISASSEMBLED AGAIN to reflect the new byte contents

```
.D 2000 2010
```

This example will DISASSEMBLE the code from \$2000 thru \$2010.

FILL MEMORY

```
.F 1000 -1100 AA
```

This example FILLS every byte in memory from \$1000 thru \$1100 with the hex value \$AA. Only a single fill byte value may be specified, no multiple byte patterns.

GOTO

```
.G
```

In this example the computer will GOTO the address contained in the PC REGISTER and will begin execution at that address.

The contents of all the registers will be replaced with the REGISTER contents currently displayed.

```
.G 1000
```

For this example the computer will GOTO location \$1000 and begin execution there. XR, YR, AR and SR will be as displayed.

HUNT MEMORY

.H C000 D000 'READ

The computer will hunt memory from \$C000 thru \$D000 looking for the ASCII string 'READ' and will print all addresses where the string is found. A maximum of 32 characters may be used in the search string.

.H C000 D000 20 D2 FF

The computer will hunt \$C000 thru \$D000 looking for the byte pattern '20 D2 FF' and will print all addresses where this sequence is found. A maximum of 32 bytes may be specified in the search string.

HUNT can be stopped with the 'STOP' key or slowed with the 'OFF/RVS' key (left arrow key on business models).

INTERPRET MEMORY

.I F000

.' F000 54 4F 4F 20 4D 41 4E 59 (TOO MANY)
 .' F008 20 46 49 4C 45 D3 46 49 (FILESFI)

-- NOTE -- The characters inside the '()' are printed in reverse field.

This example will display memory in both HEX and ASCII from \$F000 to the end of memory or until terminated by the 'STOP' key. The display may be slowed with the 'OFF/RVS' key (left arrow key on business models).

.I F000 F008

This example displays two lines of memory contents (from \$F000 thru \$FO0F) in HEX and ASCII.

The bytes following the addresses may be modified on screen and entered in memory by the 'RETURN' key.

If there is no ASCII equivalent a '.' will be printed in the RVS ASCII field.

LOAD FROM INPUT DEVICE

.L

Loads the next program found on CASSETTE #1.

.L "RAM TEST",02

Loads the program named 'RAM TEST' from CASSETTE #2.

.L "0:RAM TEST",08

Loads the program named 'RAM TEST' from DRIVE #0 of the DISK DRIVE UNIT (Device #8).

CAUTION Executing the LOAD COMMAND breaks the IRQ used by XTRAMON. Exit to BASIC (.X) and re-enter XTRAMON before using the '.G' (GOTO) COMMAND.

MEMORY DISPLAY

.M 0000 000F

```
:: 0000 00 01 02 03 04 05 06 07
:: 0008 08 09 0A 0B 0C 0D 0E 0F
```

This example displays the contents of memory from \$0000 thru \$000F. The display can be terminated by the 'STOP' key or slowed by the 'OFF/RVS' key.

The bytes following the addresses may be modified on screen and entered in memory by the 'RETURN' key.

NEW LOCATOR

.N 5000 57FF 3000 0400 7FFF

.N 57E2 57FF 3000 0400 7FFF W

In this example, the first line adjusts all THREE-BYTE INSTRUCTIONS in the RANGE \$5000 to \$57FF by adding \$00 \$30 to the right two bytes of those instructions. These adjustments apply to only those ORIGINAL address values in the range of \$0400 to \$7FFF.

The second line adjusts two byte '.WORD' values located from \$57E2 TO \$57FF, with the same ORIGINAL address range restrictions (\$0400-\$7FFF).

NEW LOCATOR stops and displays any invalid OP CODE it encounters. This feature can be used by purposefully inserting an invalid instruction byte (\$80 for example) after the last program instruction and before any .BYTE strings or .WORD tables. WORD tables should be the final part of a relocatable program.

QUICK TRACE

.Q

.Q 1000

The first of these two examples begins execution at the address in the PC REGISTER. The program will stop when it satisfies the conditions imposed by the previous setting of the BREAK command, and will enter the WALK mode at that point.

The second example starts the trace at \$1000.

QUICK TRACE executes each instruction individually as with the 'GOTO' command, but uses the 'BREAK' command setting to halt after N executions of a given address. Execution may also be stopped by pressing 'B' key.

REGISTER DISPLAY

```
.R
      PC  IRQ  SR AC XR YR SP
.; 0000 E62E 01 02 03 04 F5
```

This command displays the current REGISTER values. The values may be changed by normal on-screen editing methods previously described. When XTRAMON is activated, this command is automatically executed, but the values of SR, AC, XR and YR are not valid at that time.

The meanings of the register display elements are as follows:

```
PC = Program counter
IRQ = Interrupt Vector in $90,$91
SR = Status Register
```

```
Bit 7 - N = sign bit
Bit 6 - V = overflow indicator
Bit 5 -   = not used
Bit 4 - B = break indicator
Bit 3 - D = decimal mode indicator
Bit 2 - I = IRQ disabled indicator
Bit 1 - Z = Zero or equal indicator
Bit 0 - C = Carry bit
```

```
AC = Accumulator -A Register
XR = X Index Register
YR = Y Index Register
SP = Stack Pointer ($01XX) XX = Reg. value
```

SAVE TO OUTPUT DEVICE

```
.S "0:TEST FILE",08,0800,0C80
```

This example SAVES to DRIVE #0 of the disk unit (device #08), the contents of memory from \$0800 to (but not including) \$0C80 as a PROGRAM FILE and names it 'TEST FILE'.

***CAUTION (1)** This instruction breaks the IRQ used by XTRAMON. Exit to BASIC and re-enter XTRAMON before using the '.G' (GOTO) COMMAND.

***CAUTION (2)** When SAVING to the DISK always specify drive # or your disk will be corrupted!

TRANSFER MEMORY

```
.T 1000 1100 5000
```

This example shows a TRANSFER of the memory contents from \$1000 thru \$1100 into the locations starting at \$5000. (This is actually a 'COPY' as the contents of the source locations are not changed).

WALK CODE

```
.W F239
A1 A1 B8 01 F4 F23A 4C D8 E3 JMP E3D8
```

The format of the display is:

```
SR AC XR YR SP ADDR CODE-BYTES INSTRUCTION
```

This example causes SINGLE STEP EXECUTION and DISASSEMBLY of instructions on screen along with the REGISTER CONTENTS beginning at \$F239. The display shows the contents of the registers AFTER execution of the instruction at \$F239, the current address of \$F23A, and the byte and symbolic versions of the instruction at \$F23A which is ready to be executed.

```
.W
```

This example starts disassembly and display at the address specified by the PC REGISTER.

The speed of execution for the 'WALK' command can be controlled as follows:

```
COMMA FOR SINGLE STEP
RVS   FOR SLOW SPEED STEPPING
      (left arrow on business keyboards)
C     FOR FAST SPEED STEPPING
STOP  EXIT FROM WALK MODE
```

QUICK TRACE with BREAK can be used in conjunction with WALK to rapidly execute a program up to a specified break point and then do a slow and detailed examination of the execution in the area of concern.

NOTE: The repeat key operation is disabled during WALK. Execution of most other XTRAMON commands will re-enable the REPEAT key option.

EXIT TO BASIC

.X

This command causes an EXIT to BASIC 'READY' mode. The stack value saved when XTRAMON was entered is not restored when exiting XTRAMON.

RELOCATION PROCEDURE

In general, a program can be relocated by XTRAMON if it was properly formatted at assembly time. The code to be relocated must be in sections. The sections can be in any order but the addresses of each section must be known.

1. Program Section: The program must be contiguous (ie: without any word tables or byte tables intermixed with the code).
2. Byte Table: The byte table or message bytes must be separate. This code must remain unchanged during relocation.
3. Word Table: The word tables consist of 2 byte address references and must be contiguous.

The relocation procedure consists of:

- (a) Use the .T command in XTRAMON, to transfer ALL the code from its original area to the relocation area.
- (b) Use the .N command to adjust the address of the program code.
- (c) Use the W option of the .N command to adjust the address of the word tables.

XTRAMON (BATPRO version) does not follow this format and cannot be relocated. The following example, however, illustrates the procedure.

RELOCATE EXAMPLE

Assuming that a ML program resides at \$9000 thru \$97FF and it is desired to relocate it to \$6000 thru \$67FF, use the following procedure:

When in the MONITOR Mode key in the following:

```
.T 9000 97FF 6000  
.N 6000 66F5 D000 8000 9800  
.N 67E1 67FF D000 8000 9800 W
```

The first line moves the program at \$9000 thru \$97FF to a new location at \$6000 thru \$67FF. The second line adjusts the PROGRAM machine code address references. The third line adjusts the '.WORD' tables. The byte tables in \$66F6 thru \$67E0 are not altered.

PAUSE FEATURE

During the XTRAMON options that scroll, (.D, .I, .M) a pause feature has been added. Hit the 'B' key and the display will freeze until another key is pressed. If the next key is the stop key the scroll action will be terminated. If the next key is not the stop key the scroll action will resume.