

**CONTENTS:**

1. Introduction	2
2. Fitting	3
3. XTEND Commands	4

## **INTRODUCTION**

XTEND is a machine code utility package supplied in an eprom. It adds several extensive commands to the BBC microcomputers repertoire and enhances others. The utility is designed to extend BBC Basic and will not work if Basic is not present.

As an option, the name and address of the owner can be blown into the chip, so as to appear on the screen each time the power is switched on or the break key is pressed. As a consequence, the eprom is not marked, ensuring any casual thief will have difficulty identifying the chip to be removed. If your copy does not include the protection option, or if in future you want the address changing because of a house move, it can be exchanged by completing and returning the pro-forma at the end of this booklet, together with the original eprom.

All XTEND commands must be prefixed with an asterisk \*. Abbreviations may be used (minimum first two characters plus a full stop) but care must be taken if other Roms are present using similar commands. You are recommended to give XTEND a low priority (see fitting instructions) enabling commands from other Roms to be accessed first. XTEND commands with duplicated names may be accessed by preceding the command name with X. In the minimal abbreviated form X may constitute one of the required characters. The X prefix is optional and will only be required if another Rom is fitted which uses an identical call.

Commands may be entered in upper-case or lower-case.

XTEND is a comprehensive Basic Debugging tool offering many facilities to the serious programmer. If used properly these extensions will have a marked effect on the way Basic programs are written. For example, creating multi-statement lines at the coding stage is no longer necessary due to XTENDs ability to completely compact a program. XTENDs dual screen step facility makes single line statements a positive advantage in what was once the tedious business of debugging graphics programs. It is now possible to single step through a graphics program at your own speed and toggle at any time to a mode 7 screen to examine code or dump variables. You can return to your graphics program at any time with no ill effects and continue stepping through.

### **FITTING INSTRUCTIONS**

XTEND must be fitted into one of the spare Paged Rom sockets situated towards the front right of the machine. (As seen from the front.)

NB Ensure that the computer is disconnected from the mains - not just switched off - before attempting to remove the cover to fit any Rom.

Having removed the cover and keyboard, the five Rom Sockets become visible towards the front-right of the circuit board. One socket will be occupied by the Operating System (OS) - normally the left-most socket and another by the Basic chip - often to be found next to the Basic Rom. If a disc interface is fitted the DFS eprom will take up a third. The Basic Rom should first be moved to the right-most socket. This slot has first precedence and whatever occupies it will be initialised on power-up. XTEND should be placed in the now empty socket next to the OS Rom (ie. Left-most but one.), giving it lowest priority. Because of its ability to recognise all commands with an optional X prefix, any other Roms with identical commands will be accessed first and can still be used.

The Rom should be inserted with pin 0 pointing towards the back of the machine. Pin 0 is at the end marked by an indentation.

Once installed and the keyboard and cover securely replaced, switch on the machine. If the security system is included, your name and address should appear on the screen.

All XTEND commands are now available. They may be listed by typing:

\*HELP XTEND                      (abb. = \*H.XT.)

**XTEND COMMANDS.**

In this section each XTEND command is explained in full. Normal command names are given, together with their minimal abbreviation. It should be remembered that all commands may be prefixed with X to prevent conflict with identical commands from other Roms.

eg.                                \*FIND                        =            \*XFIND or (\*FI. = \*XF.)

Commands which require arguments are shown with the argument parameters enclosed within < >. Optional argument parameters are enclosed within parentheses ( ). Multiple parameters should be separated by spaces.

XTEND may be used in direct Command mode or statements may be included within Basic programs.

Printers should be initialised in the normal way ie. CTRL-B before instigating the XTEND command.

**\*FIND**                    (**\*FI.**)

**FUNCTION**            : To find, count or change all occurrences of user specified string.

**PARAMETERS**        : None - two string inputs required.

Upon entering \*FIND the screen will clear to the FIND/CHANGE/COUNT menu. Initially you are requested to input the string to be found. This can be any character combination constituting a normal string, number, variable or keyword (the FIND string). The user is next asked to enter the character combination to be substituted (the SUBSTITUTION string). The SUBSTITUTION string may be longer or shorter than the FIND string.

If <RETURN> alone is pressed instead of entering a SUBSTITUTION string, the FIND string will be copied into the SUBSTITUTION string. No changes are obviously required in this instance and XTEND will merely go on to count matching occurrences.

The menu will now ask whether you want a GLOBAL search to be performed (ie. Every occurrence to be exchanged and/or counted) or whether a SELECTIVE search is preferred (the complete line containing each occurrence displayed in a window at the bottom of the screen, with the first character of the FIND string highlighted).

If only a count is required then GLOBAL should be selected. If exchanges are sought then GLOBAL should be used with great care. The user has no control over substitutions and every occurrence will be found and exchanged. It is easy, particularly with short FIND strings, to overlook embedded occurrences within normal strings and variables.

If SELECTIVE is stipulated the user must confirm each particular exchange when found and displayed at the bottom of the screen.

XTEND will terminate with a display of the number of occurrences of the FIND string found and the number of exchanges performed.

Pressing the <SPACE BAR> upon completion will maintain \*FIND whilst pressing <RETURN> or <ESCAPE> will return you to Basic.

\*KEYS            (\*KE. )

FUNCTION        : To display and allow editing of the contents of user definable function keys.

PARAMETERS     : (Key No.)

\*KEYS without parameters will result in the contents of all defined function keys being displayed on the screen. If a key number is given (range 0-10 or 0-15 if cursor keys are also defined) as an argument then only the contents of that key will be shown.

Displayed keys may be edited with normal cursor control and copy keys (provided they are still functional of course) and new key definitions created.

Basic is maintained throughout this operation.

**\*LJ**

**FUNCTION** : To left justify all Basic statements when listed.

**PARAMETERS** : None.

\*LJ is an extension of the Basic LISTO command. Once initialised, future occurrence of LIST have the effect of splitting all multi-statement lines, displaying each statement separately, justified down the left side of the screen. To send output to a printer the normal CTRL-B instruction should be issued before LIST.

It is advisable to turn off \*LJ with the command \*OFF before running a program.

**\*STEP**                    (**\*ST.**)

**FUNCTION**     :    To step through a program, line by line, at your own pace.    Line numbers are shown discreetly in the top left corner.

**PARAMETERS** :    No formal parameters but the secondary command **\*STEP D** can be used to initiate a dual screen.

**\*STEP** is a utility designed to enhance the resident TRACE routine of the BBC micro.    The normal TRACE routine is too quick to be of much value, to say nothing of the fact that it prints line numbers all over the screen.    **\*STEP** controls the resident TRACE routine, printing the current line number discreetly in the top left hand corner whilst allowing you to step through a program at your own pace by pressing the <SPACE BAR>.

The full specification of TRACE is maintained (see User Guide page 367) and if used on its own, will be no different to normal.    To change to XTENDs enhanced version simply enter the command **\*STEP <RETURN>** before initiating TRACE.

Providing **\*STEP** has been entered, XTENDs TRACE can also be initiated at any time by entering <CTRL>@.    (ie holding down the <CTRL> key whilst at the same time pressing the @ key.)    The same direct command will turn TRACE off.    <CTRL>@ takes the place of the TRACE ON instruction and can be executed at any point within a program run.

**\*STEP D** adds a new dimension to debugging graphics programs in particular but can be equally useful with others.    The D stands for dual screen and the commands can be used instead of the standard **\*STEP** command.    Provided you have 1k of memory between the top of your variable workspace and HIMEM, regardless of the mode currently being used, a MODE 7 screen can be created and the **\*VAR** routine entered. This can occur at any point within a program run by pressing the <TAB> key whilst a TRACE is occurring.

After listing your program or/and interrogating your variables you may return to your previous screen which has been preserved undamaged, to continue your program run.    It should be noted that XTEND uses page &A00 as work-space during this operation.

If 1k of memory is not available to create the dual screen then XTEND will issue the message <No dual screen at line XXX> and the program will stop.    Normal interrogation of variables can now take place or switch to **\*VAR** for selective dumping.    Whatever you do in this instance, or course, your screen will not be preserved. If halfway through a graphics routine you will have to start again.



\*CONDENSE

\*CO.

FUNCTION : To reduce the size of a Basic program.

PARAMETERS : None.

Selection of \*CONDENSE results in a menu of five options being displayed on the screen. You are advised to compact your program only when it is finished. Considerable space can be saved, particularly on discs, but future editing may be more difficult after passing programs through this cruncher.

0. Exit

This option will return you to Basic.

1. Remove REMs

All REM statements not pointed to by GOTO, GOSUB or ON statements will be removed. The process is not selective. You can not choose to keep some and remove others. If particular REM statements require to be preserved, I suggest you insert a dummy GOTO statement pointing to the appropriate line before instituting option 1. It should be removed upon completion.

2. Remove Spaces.

All spaces (except those with strings) will be removed from the program. The despacers works on the tokenised program within memory (held in a different form to that seen on the screen). For this reason, care must be taken when editing a line that has been despaced. Floating point numeric variables do not end in a recognisable delimiter, as do string and integer variables (ie % and \$). A space normally marks the end of floating point variables, and this space will be removed by option 2, with no ill effects. If however, you edit the line at a future date, the delimiter spaces must be reinstated or unrecognisable variables will result. IF A B=C is a valid Basic statement. Option 2 will remove the spaces from its tokenised form so that it appears on the screen as IFAB=C. Quite clearly A and B are separate variables and if the line is edited using the <COPY> key the result will be a test on a variable called AB.

3. Remove Comments.

Assembler comments may be removed by this option. All comments will be removed regardless of where within a line they occur.

4. Compact Program

Use with care! Tokenised code will be compacted to maximum multi-statement form. This will result in lines of much greater length than is possible from the keyboard. Future editing may require these lines to be split-up.

Selection of number 4 will result in you being asked which line number you wish to start compacting from and which line number you want to compact to. It is possible therefore, to compact specific areas at a time eg. Procedures. If <RETURN> is pressed in reply to the initial line number prompt, then the whole program will be crunched.

Throughout each \*CONDENSE operation the bottom of the screen will display the current program length and the number of byte freed so far.

\*VAR

\*VA.

FUNCTION : To dump all variables, defined procedure and function locations.

PARAMETERS : None.

Selection of \*VAR will present a menu offering the full display of the following options;

1. A% to Z%

The current value of all resident integer variables.

2. Floating Point

All currently assigned floating point variables, together with their value.

3. Integers

Integer variables other than the resident integer variables that are currently assigned.

4. Strings

Every defined string and string definition.

5. Arrays

All dimensioned arrays will be indicated. The contents of these arrays will not be shown, merely their name, type and size.

6. Definitions

The line numbers and names of function and procedure definitions.

**\*MEM**

\*ME.

FUNCTION : To display all salient memory locations.

PARAMETERS : None.

Selection of \*MEM will display the contents of all important memory locations. These include:

PAGE

TOP

LOMEM

HIMEM

Memory left

Program length

Space Taken by Variables

PARAMETERS :     <File name>.

Care must be taken with programs containing GOTO, GOSUB and ON jumps. If these are present in either program, the appended file (ie program B) should be renumbered so that line numbers do not coincide with the resident program.

\*SINGLE

\*SI.

FUNCTION : To provide simplified entry of keywords.

PARAMETERS : None.

---

:TAB: Q : W : E : R : T : Y : U : I : O : P : @ :
:TAB:SOUND:ERROR:ELSE:REPEAT:THEN:PROC:UNTIL:INPUT:OR:PRINT:CHAIN:
:LIST:ENVELOP:REPORT:END:RENUMBE:TRACE:ENDPROC:PAGE:INKEY:OLD:PLOT:RUN:

---

: A : S : D : F : G : H : J : K : L :
:AND:STEP:DRAW:FOR:GOTO:HIMEM:GOSUB:COLOUR:LOCAL:
: AUTO: STOP: DEF: TO: GET: LOMEM: RETURN: GCOL: LEN :

---

: Z : X : C : V : B : N : M :
:CHR\$:LEFT\$:CLS:VDU:READ:NEXT:MOVE:
:MID\$:RIGHT\$:CLS:VAL:DATA:NEW:MODE:

---

Once initiated, pressing <SHIFT> and the appropriate key gives you the middle row of keywords in the diagram above. Pressing <CTRL> and <SHIFT> together will give you the lower set.

\*OFF will cancel \*SINGLE.

## **XTEND.**

\*OFF

\*OF.

FUNCTION : To turn off XTEND commands with lasting effect.

PARAMETERS : None.

\*STEP, \*JO and \*SINGLE all remain operative until switched off.

\*OFF is the command which switches these routines off.

**\*FIX**

\*FI.

FUNCTION : To make Bad Programs listable.

PARAMETERS : None.

Bad Program is an error message issued by the BBC micro when it encounters a program of dubious nature. There are numerous reasons why a program can become bad and it would be impossible for a routine to cure every possibility.

\*FIX is designed to recalculate line delimiter corruption and to substitute erroneous control characters within program lines for hash characters.

This will not make a bad program RUN of course, but if it is not too badly corrupted it should become listable. You can then go through the program putting back appropriate code.