

BASIC PROGRAMMERS TOOLKIT

Copyright 1981 by Dick Bartholomew

**Frank Hogg Laboratory, Inc.
130 Midtown Plaza
Syracuse, NY 13205
315-474-7856**

BEDIT

BEDIT - BASIC LINE EDITOR FOR TSC BASIC

The available commands are as follows:

- R - Move the cursor right one position. Single key command
- L - Move the cursor left one position. Single key command
- D - Delete the character over the cursor. Single key command
- I - Insert in front of the character the cursor is positioned under. Enters mode until ^C or CR typed
- O - Overlay new characters over the old character the cursor is under. Enters mode until ^C or CR typed, ^X will abort.
- P - Re-print the entire line. Useful if the original becomes tainted in order to see exactly what the line in the buffer looks like. The line pointer and cursor will be started over at the beginning of the line. Single key command
- C - Change a string to another string or delete the entire string. (Many possibilities, essentially the same as TSC'S EDITOR) (The first character typed after the 'C' will be used as the separator and must be used to separate both strings and optionally the end sequence)
 - (Each string may be up to 32 characters long)
 - (Typing C/OLD-STRING// will delete the entire string)
 - (Typing C/OLD-STRING/NEW-STRING/ terminates the command with the cursor remaining where it previously was.)
 - (Typing C/OLD-STRING/NEW-STRING<cr> terminates the command with the cursor on the beginning of the line)
 - (Typing C,OLD-STRING;NEW-STRING, used the comma instead of slash as a terminator character.)
- ^X - Will in the command mode, abort the edit session, returning to BASIC with no changes being made.
- ESC X - Sends the edited line to BASIC, and exits, returning control to BASIC.

BEDIT

The purpose and intent of this program is to provide the BASIC user with a means of making changes in his or her programs, a line at a time, without the need to re-type the entire line. Anyone who has used TSC BASIC with any regularity knows how frustrating it can be to enter a line and then realize there exists an error in the second last character position of a 128 character line. Of course, the entire line must be re-typed, with the possibility of making another error along the way that will not be detected until the carriage return is hit. Also, having to exit BASIC to use an editor when making many changes can also be a tedious and time-consuming chore.

This package allows the user to enter into the edit mode with a selected line number, make the required changes, and send the line back to BASIC without ever leaving BASIC. The normal method of using the editor is to append it to BASIC and have both programs load and run concurrently. The editor is completely transparent until the correct key sequence calls it into use. The only requirement is for FLEXS' memend location to contain a valid address, which should be the case anyway.

The user should have a terminal capable of at least the following functions:

- HOME CURSOR NON-DESTRUCTIVELY
- HOME CURSOR & CLEAR SCREEN
- MOVE CURSOR RIGHT NON-DESTRUCTIVELY
- MOVE CURSOR LEFT NON-DESTRUCTIVELY
- WRAP LINE AROUND AT END OF LINE TO NEXT LINE
- OPTIONALLY - REVERSE VIDEO

Upon loading BASIC, assuming BEDIT is appended, the editor reads the memend location, determines where it can be executed, reserves some room at the top of memory, changes FLEXS' memend pointer, and re-locates itself. At this time, the EDITOR begins execution and another routine searches thru BASIC to locate the I/O vectors. As it finds each one, it changes the address to point to a routine in the editor package. Upon exiting BASIC, the editor reallocates the memory it saved for itself, restoring memend to its original value.

For the users with the source code, refer to the tables in the beginning of the program if it is desired to change the key codes, port address or editor command codes for their own preferences. The user without the source code can also make these changes by following the instructions listed below.

BEDIT

The editor is originated at HEX 8000 to provide room for BASIC to load. Do a GET BEDIT.BIN to load the editor into memory. Exit FLEX and using some type of display memory command, dump the first 50 HEX bytes starting at HEX 8000. The beginning of the table can be located by being preceded with two (2) bytes of HEX FF. The end of the table terminates also with two (2) bytes of HEX FF.

* THESE CODES ARE XMITTED TO THE TERMINAL

*

	FDB	\$FFFF	HELPFUL TO LOCATE TABLE IN RAM
PORT	FDB	\$E004	CONTROL PORT ADDRESS (STATUS REG SIDE)
CURGHT	FDB	\$1B43	CURSOR RIGHT CONTROLS
CURLFT	FDB	\$1B44	CURSOR LEFT CONTROLS
HOMCLR	FDB	\$1B45	HOME CURSOR & CLEAR SCREEN
HOME	FDB	\$1B48	HOME CURSOR, NO SCREEN ERASE
WRPRND	FDB	\$1B76	ENABLE CURSOR WRAP AROUND AT END OF LINE
EXIT	FDB	\$1B7A	RESET TERMINAL UPON EXIT CODE
REVERS	FDB	\$1B70	REVERSE VIDEO OR LOW-INTENSITY MODE
NORMAL	FDB	\$1B71	EXIT REVERSE VIDEO MODE CODE

*

* THESE CODES ARE TYPED BY THE USER

*

LEADIN	FCB	ESC	LEAD-IN CHAR TO SCREEN
CRIGHT	FCB	'R	CURSOR RIGHT COMMAND FROM KEYBOARD
CLEFT	FCB	'L	CURSOR LEFT COMMAND FROM KEYBOARD
INSERT	FCB	'I	INSERT CHARACTER CODE
DELETE	FCB	'D	DELETE CHARACTER CODE
OVRLAY	FCB	'O	OVERLAY CHARACTERS CODE
CHANGE	FCB	'C	CHANGE STRING CODE
PRINT	FCB	'P	PRINT BUFFER CODE
ABORT	FCB	\$1B	ABORT EDIT CODE
CHKTX	FCB	'X	UPDATE BASIC CODE
TERM	FCB	3	^C TO TERMINATE COMMAND MODES
DONE	FCB	ESC	ESCAPE CODE PART OF COMMAND
BKSPAC	FCB	\$0B	BACKSPACE & DELETE CHAR
LNEWTH	FCB	80	CHARACTER LINE WIDTH OF TERMINAL
EDTMOD	FCB	'E	ESC E TO ENTER EDIT MODE
	FDB	\$FFFF	END OF ALTERABLE RAM

It is suggested that the original codes be retained until the user becomes thoroughly familiar with the program and realizes the consequences of changing these codes. The EDITOR may be used without appending to BASIC with this sequence of commands: GET BASIC.CMD, BEDIT.BIN. No key codes are illegal as set, except for the key sequence of ESC E, to enter the editor.

.BEDIT

These keys may be sent from a BASIC program, but not to a BASIC program, as the editor will be invoked. This single compromise was made due to the fact that the editor is in no way tied into BASIC, and therefore, the EDITOR should work with any version of TSC BASIC including the first TBASIC. Once becoming familiar with the workings and limitations of the EDITOR, it will become a quick and simple matter to make corrections to the program in memory.

A sample session with the EDITOR may be as follows:

LIST

```
10 FOR A=1 TO 5
20 PRINT "NOW IS TEH TIME FRR ALL GOOD MEN TO COME TI THE AID
OF ME"
30 NEXT A
40 END
```

READY

RUN

```
NOW IS TEH TIME FRR ALL GOOD MEN TO COME TI THE AID OF ME
NOW IS TEH TIME FRR ALL GOOD MEN TO COME TI THE AID OF ME
NOW IS TEH TIME FRR ALL GOOD MEN TO COME TI THE AID OF ME
NOW IS TEH TIME FRR ALL GOOD MEN TO COME TI THE AID OF ME
NOW IS TEH TIME FRR ALL GOOD MEN TO COME TI THE AID OF ME
```

READY

(Now to make corrections to line 20, Type ESC E)

LIST 20

(The screen will clear & the cursor will home, then)

```
20 PRINT "NOW IS TEH TIME FRR ALL GOOD MEN TO COME TI THE AID
OF ME"
```

(Each new line listed is assumed to be on the top of the page)

(type C)

```
20 PRINT "NOW IS TEH TIME FRR ALL GOOD MEN TO COME TI THE AID
OF ME"
```

C/TEH/THE/

```
20 PRINT "NOW IS THE TIME FRR ALL GOOD MEN TO COME TI THE AID
```

.BEDIT

OF ME"

(type R until the cursor is positioned under the first R in FRR, and type D) (now type D & ^C to terminate the command)

```
20 PRINT "NOW IS THE TIME FOR ALL GOOD MEN TO COME TI THE AID OF ME"
```

(type R to move the cursor under the I in TI, and type D & ^C)

```
20 PRINT "NOW IS THE TIME FOR ALL GOOD MEN TO COME TO THE AID OF ME"
```

(all errors in the line have now been fixed. Note that the CHANGE command was used for one error and the OVERLAY command for the others. There may often be several ways of correcting the error. It makes no difference which is used.)

Now we will insert some new characters into the line)

```
20 PRINT "NOW IS THE TIME FOR ALL GOOD MEN TO COME TO THE AID OF ME"
```

(type L to move the cursor under the F in FOR, and type I. Then type the new characters)

```
20 PRINT "NOW IS THE TIME &FOR ALL GOOD MEN TO COME TO THE AID OF ME"
20 PRINT "NOW IS THE TIME & FOR ALL GOOD MEN TO COME TO THE AID OF ME"
20 PRINT "NOW IS THE TIME & DFOR ALL GOOD MEN TO COME TO THE AID OF ME"
20 PRINT "NOW IS THE TIME & DAFOR ALL GOOD MEN TO COME TO THE AID OF ME"
20 PRINT "NOW IS THE TIME & DATFOR ALL GOOD MEN TO COME TO THE AID OF ME"
20 PRINT "NOW IS THE TIME & DATEFOR ALL GOOD MEN TO COME TO THE AID OF ME"
20 PRINT "NOW IS THE TIME & DATE FOR ALL GOOD MEN TO COME TO THE AID OF ME"
```

(now type ^C to terminate the command.)

it can be seen that as each character was typed, the remainder of the line moved over one place to the right and the new character appeared. Now to delete characters, move the cursor to the M in MEN, and type D, 4 times)

```
20 PRINT "NOW IS THE TIME & DATE FOR ALL GOOD EN TO COME TO THE AID OF ME"
20 PRINT "NOW IS THE TIME & DATE FOR ALL GOOD N TO COME TO THE AID OF ME"
20 PRINT "NOW IS THE TIME & DATE FOR ALL GOOD TO COME TO THE AID OF ME"
```

BEDIT

A. OF ME"

20 PRINT "NOW IS THE TIME & DATE FOR ALL GOOD TO COME TO THE
AID OF ME"

(now to update the line in BASIC, type ESC X. when the READY prompt appears, the line has been sent back to BASIC and control has been returned to BASIC. Note that if an error exists in the line, BASIC will report it and reject the line just as though the line had been typed in at the keyboard.)

While in the insert or change mode, an error may be corrected by typing the predefined backspace key, normally a ^H (08 HEX). The previous entry will be deleted and the cursor backed up one position. Once the separator character has been typed in the change mode, all preceding data is fixed and cannot be changed. If an error needs to be corrected, a ^X may be typed to terminate the mode and start over.

The above example should give an indication of the capabilities of the EDITOR. There are many other ways the changes could have been made and using all possibilities to gain familiarity with them is recommended.

A few hints on some of the other things that may be done with the EDITOR follows:

Entering the edit mode with an ESC E, LIST 20 will put line number 20 into the buffer to be worked on. Using the CHANGE or OVERLAY command to change the line number will cause a new line to be created in BASIC with the given line number. This function can be utilized to duplicate existing lines.

Entering the edit mode with an ESC E, LIST XX, where XX is a non-existent line, can be used to create new lines. When the EDITOR is entered, the word READY will appear on the screen. Enter the insert mode using 'I, type a line number and the data and then exit the insert mode with a ^C. Then, using the 'D, delete command, delete the word READY and type ESC X. The new line will be sent to BASIC and appear in the program.

If at any time, the cursor position becomes lost due to some problem, simply typing a CR will restart the pointer at the beginning of the line buffer and put the cursor on the beginning of the displayed line, thereby starting fresh.

The command key code is not echoed to the screen. After the command key is typed, all other data until a ^C or <cr> is typed will appear on the screen.

Copyright 1981 by Dick Bartholomew
and Frank Hogg Laboratory, Inc.

BEDIT

If for any reason, the user has lost track of what he is doing, such as what mode he is in or even if he is in a command mode, typing <cr> will terminate whatever is going on, and re-start in the command mode.

Hearing the bell ring during an edit session normally indicates an error was made. Possible error could be the line buffer is full, as a maximum of 128 characters are allowed.

Trying to backspace in the change mode past the number of characters that have been typed in.

Trying to delete more characters than are in the line buffer.

Trying to backspace past the beginning of the line.

Trying to do a change to more than 32 characters at a time.

Trying to change a non-existent string.

Due to the fact that the EDITOR is not tied into BASIC in any way, it was not possible to be able to work on consecutive line numbers without exiting the EDITOR and calling the next line. This is only a slight inconvenience and the possible uses of the EDITOR far outweigh this fact.

All I/O from BASIC now vectors thru the EDITOR. The internal I/O routines use the following TTYSET parameters only.

PAUSE character

Stop display at end of line

Abort list command while paused with a carriage return.