

**sinclair**

# **ZX Spectrum 128**

Derby++ User Guide

# Getting started

After powering on the computer, use the cursors and Enter to select 128 BASIC from the boot menu (don't hold down Enter, unless you want to start the Tape Loader). The copyright message is displayed. Press Enter again to start the editor.



There are a number of differences from 48 BASIC, the UK version of 128 BASIC, and even the original Spanish version of 128 BASIC on which this version is based. For example, there is no SPECTRUM command (because you can select 48 BASIC from the boot menu). You should take time to familiarize yourself with the unique features of the fastest version of 128 BASIC.

There are two differences that are not immediately apparent. The first is that almost all of the bugs in 48 BASIC, and all the bugs in 128 BASIC are fixed. The second is that variables are persistent, that is they are not CLEARED when you RUN a program. This is because it would be very annoying to lose all the work you had done in the full screen simple string editor before you had saved it. Simple

strings (not DIMensioned) are saved when you SAVE your program.

The other menu options have the following effects:

- Loader - Open the esxDOS NMI browser (when using divide, divMMC and so on). When you autoload programs from the NMI browser they run using the menu ROM (similar to USR0 mode). If this causes compatibility problems, try loading from 48 BASIC or 128 BASIC instead. Loads the auto file on a TNFS file system (when using Spectranet).
- Tape Loader - Hold Enter when selecting 128 BASIC to get this option. You can use this after mounting a tape in the NMI browser to run programs that require a 128 BASIC environment (for example, *splATTR*).
- Diagnostics - Run the Spectrum +2 test program (RS232 test requires a loopback adapter).
- 48 BASIC - Selects ROM 1 (the original 1982 version of BASIC) and lock paging. Use this for programs that trigger 128 paging (for example *Jet Pac*), or perform a ROM checksum (for example *Travel with Trashman*).
- TR-DOS - Enters the TR-DOS command line (requires BetaDisk interface or esxDOS TR-DOS emulation). Commands are typed out in full. The RETURN command is ignored; press the reset button to return to the boot menu. Programs run using the menu ROM (similar to USR0 mode). If this causes compatibility problems, try loading from 48 BASIC or 128 BASIC

instead. You can enter the TR-DOS command line from BASIC by entering PRINT USR 15616.

If you have a 32K ROM machine, you should use ROM2/ROM1. Under esxDOS you will boot in USR0 mode. You can lock paging from the Dr Slump NMI browser. You can switch to 128 BASIC with the .128 command (or OUT 32765,16). In 128 BASIC you will have to enter symbols the same way you do in 48 BASIC and some symbols may clash with the extended editing commands.

Note: *HiSoft BASIC* will not work because it requires the UK Spectrum 128 editor ROM to run. However, programs compiled with it will run as normal.

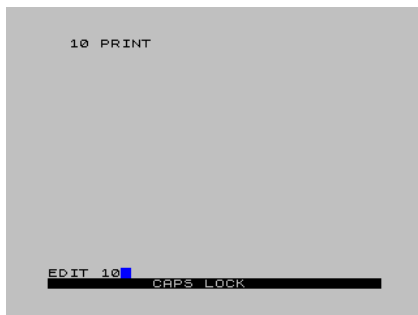
---

## Using the editor

---

The editor comprises three areas:

- Upper screen
- Lower screen
- Status line



When the editor starts, the cursor is in the Lower screen, above the Status line. You can switch the cursor between the Upper and Lower screens by pressing Edit. The Upper screen displays your

program listing. You can edit your program directly from this area. From the Lower screen you can enter *direct* commands (DELETE, EDIT, RENUM, WIDTH), use the running calculator, or enter program lines. The Status line displays the mode (EXTEND or GRAPH) and if CAPS LOCK is enabled. You can change the editor colours using the standard commands (BORDER, INK and PAPER).

---

## Using the keyboard

---

The keyboard functions differently from 48 BASIC. Commands are typed out in full, and EXTEND mode is not needed, except to enter embedded control codes or use the extended editing functions. To produce a symbol, hold Symbol Shift and

press the key the symbol appears on. The exception to this is the copyright symbol (©) which is produced by pressing Symbols Shift+I. When CAPS LOCK is enabled, pressing Shift produces lowercase letters.

---

## Using the extended editing functions

---

The extra editing functions originally provided on the numeric keypad, are remapped to the main keyboard:

SCREEN = Edit, DEL = Symbol Shift+E  
PG UP = Extend D, PG DN = Extend F  
HOME = Extend Z, END = Extend X

## Entering control codes

You can use EXTEND mode to directly enter control codes, just like in 48 BASIC. But in 128 BASIC you will see the hidden codes represented by a GG. Don't enter any characters between the two Gs or you will produce an error.

Extend n - INK (0 to 7)

Extend Shift+n - PAPER (0 to 7)

Extend 8 - BRIGHT 0

Extend 9 - BRIGHT 1

Extend Shift+8 - FLASH 0

Extend Shift+9 - FLASH 1

True Vid - INVERSE 0

Inv Vid - INVERSE 1

## Use reserved words as variables

In 48 BASIC you can use reserved words as variables. But if you try to do this in 128 BASIC, the variable is converted to a token. However, the tokenizer has two modes of operation. If CAPS LOCK is enabled when you press Enter, only words in all CAPS are tokenized. For example, LET Let = 1 would be accepted. If CAPS LOCK is not enabled, everything is tokenized. Variables are case insensitive, so you can enter them as lowercase or mixed case.

## Entering direct commands

Direct commands cannot be used in your programs and can only be entered from the Lower screen.

DELETE *f*, *l* - Delete lines *f* to *l*. To delete a single line *n* enter DELETE *n*, *n*.

EDIT *n* - Edit line number *n*.

EDIT *s*\$ - Edit the simple string *s*\$ in the full screen editor.

RENUM [*l* [, *s*]] - Renumber the program starting at line *l* in steps of *s*.

WIDTH *c* - Set the column width of the RS232 device to *c* columns.

## Using TurboSound AY

128 BASIC supports up to six channels of audio using the TurboSound AY interface. You can use the PLAY command as normal, but if a second AY chip is available, channels 4 to 6 will make use of it. For example, PLAY "c","g","e","C","G","E". If only one AY chip is available, these channels will be ignored. However, all eight channels will still be sent to a MIDI device if it is connected.

## Using ULaplus

ULaplus provides up to 64 colours on screen at once using four CLUTs (colour look up tables). Each CLUT is made up of eight INK colours and eight PAPER colours. A CLUT is selected by combining the value of the BRIGHT and FLASH bits of the attribute to give four possible values. However, you cannot combine colours from different CLUTs in a single attribute cell. Derby++ simplifies this by providing a default palette that enables you to choose any INK or PAPER from a 16 color palette without having to think about CLUTs.

To enable ULaplus enter PALETTE 64,1 (you can switch it off again with PALETTE 64,0). The default palette uses the same numbers for colours as the standard Spectrum palette (0 is still black). You set INK and PAPER as normal. However, BRIGHT now only applies to INK. To do the equivalent for PAPER, use FLASH. Whereas the standard Spectrum palette has 15 colours, in the default ULaplus palette, bright black is in fact dark grey.

Note: If editing you want to redefine the default palette to continue to function as a 16-color palette, you must set palette entries in each CLUT (four times for each colour).

## Using the running calculator

128 BASIC provides a running calculator that you can use from the Lower screen.

You can use the full range of mathematical functions and variables from BASIC. Enter your calculation and press Enter. The results are displayed in the Upper screen. Because you can also enter program lines from the Lower screen, if you want to enter a positive number on its own you must prefix it with a plus sign (+). To clear the calculator memory, enter \*0.

## Using the full screen simple string editor

From the Lower screen, enter EDIT \$\$, where \$\$ is a simple (non-DIMensioned) string you want to edit. The Status line is similar to the one in 128 BASIC. It displays the mode (EXTEND or GRAPH) and status of CAPS LOCK (shown by whether the status line is in uppercase or lowercase). But it also shows the cursor mode (INSERT, INDENT or REPLACE) and if *wrap* is on or off.



Press Symbol Shift+Q to change the cursor mode:

- INSERT - As you type, the content is inserted at the cursor position.
- INDENT - Works like insert mode, but when you press Enter, the next line is indented by five characters.
- REPLACE - As you type, the content replaces content after the cursor position.

Press Symbol Shift+W to change the wrap mode:

- ON - If a word breaks across a line, it is moved to the next line.
- OFF - Words are allowed to break across lines.

The extra editing functions originally provided on the numeric keypad, are remapped to the main keyboard:

EXIT = Edit, DEL = Symbol Shift+E  
PG UP = Extend D, PG DN = Extend F  
HOME = Extend C, END = Extend V

These editing functions perform a move, unless Symbol Shift is held, in which case they perform a delete:

< WRD = Extend A, > WRD = Extend S  
< LINE = Extend Z, > LINE = Extend X

## Using esxDOS

Derby++ enables you to use esxDOS with 128 BASIC. The editor is completely rewritten to prevent crashes caused by the host hardware paging memory when executing instructions in certain memory areas.

After you start the editor, if the first command you want to enter is an esxDOS command, you must first type REM and press Enter. Any command will do, but this is necessary to flush the error stack that esxDOS uses to interpret commands.

You must enter esxDOS commands in lowercase, but ensure CAPS LOCK is enabled before you press Enter. This is because the 128 BASIC tokenizer converts uppercase characters to tokens before esxDOS evaluates the line. For example .MORE would be converted to .M OR E **which isn't a valid command.** You can enter esxDOS commands that **don't contain tokens, such as .LS** in uppercase, but it is better to get into the habit of always entering them in lowercase.

If you want to use an esxDOS command as the first command in a BASIC line, you must prefix it with a colon. Otherwise the running calculator will treat the command as an undeclared variable. For example, 10 : .ls.