

Funtext Plus

A set of programs for creating teletext pages with tools for manipulating graphics and automating some animation effects.

Requirements: Acorn Master 128, or BBC B with sideways RAM. Four free sideways RAM slots plus a writeable RAM chip for a set of utilities.

The files are on FUNANIM.SSD and ROMCODE.SSD at 4-Tonline.uk/teletext.html. The ROMCODE disc has a file of the same name. *SRLOAD this file to a writeable RAM chip in a Master cartridge slot, eg *SRLOAD romcode 8000 0 (even better, create a ROM with the code, if you have an EPROM programmer). Then import the FUNANIM.SSD.

In Mode 7, type ***exec boot** to start. The boot file defines the red key functions, loads some 6502 code and grabs four RAM slots (4 to 7) and loads a BASIC file which runs a teletext page creation program; you should see the word "base" in the bottom line, the command line, with a block-style cursor on the left.

Type **D** and 0 (zero) or any number up to 98, then **return**, to Display a blank page. The cursor will move to the top of the frame and you are now in edit mode (as indicated bottom right).

You can now create a frame containing text and/or graphics, using the appropriate control characters. When the page is finished (or during construction) it should be stored in memory: press Escape to re-enter the command line and type **R** for Replace and the frame number (**return**).

Up to 99 pages - 0 to 98 - may be created in this way, **but they are only in the Beeb's volatile memory until saved to disc or RAM drive or equivalent**. To do this, type **S** in the command line, enter the drive number (**return**) and the number of pages to be saved (**return**) and a file name (**return**). Note: the pages saved will be from page 0 upwards.

If you are saving more than 18 pages they will be saved automatically as two files - <file> and <file2>. The first will be the pages 0 to 17 held in screen memory, while the second consists of the pages held in SRAM. (Your file name should be six characters or less, because the system will automatically add "2" to the second file. More than seven characters in total will produce a "bad name" error, unless you are saving to ADFS).

To load a file from disc, **Escape** to the command line and type **L**, then as prompted enter the drive number and the file name and the page slot to load to. The file will load from the specified page (if you are loading a <file2> into screen memory - pages 0 to 17 - and there isn't enough room for it, you will get a "too big" error message and you will need to load it into the SRAM area - page 18 upward).

You can use * Operating System commands from the command line. Eg. type *drive <number>, then *. for a disc catalogue. You can *load <filename> to load a file directly without using the **L** command, but this does not give the option to load from a specified page and will load from page 0.

This version of the editor does not contain the **Help** page that comes with the earlier version of Funtext, but a shortened version is shown here as it lists most of the principal commands.

EDITING SYSTEM COMMANDS

(Press ESCAPE to enter command mode, than press appropriate key)

B - Step BACK a page in memory
C - set COLOURS
D - DISPLAY a page
E - EDIT (to leave command mode)
I - set INDENT (left-hand margin)
L - LOAD a file of pages from disc
M - MERGE part of one page with another
N - step to NEXT page in memory
R - REPLACE a page in memory
S - SAVE pages to disc
W - switch WORDWRAP on or off
? - LIST pages in memory
< - Recall buffered page
***** - Operating System command

In addition, holding down the Tab key and using the arrow keys will move a line or part of a line to the right of the cursor, up or down, right or left.

Note: the ? index command will show the first 18 pages; pressing **return** repeatedly will list subsequent pages up to 98 (only up to 80 in BeebEm, for some reason).

Additional commands: typing a lowercase **p** in the command line will load a program for creating animated transition effects, in place of the base program. Typing a lowercase **f** loads another program for creating dissolve-type sequences. In either of these programs, typing **b**, **l**, or **s** will return you to the base program. More on these programs below.

Constructing pages

The appearance of a page is governed by control codes; these are normally invisible but each takes up one character space and affects colour, text and graphics. The red function keys can be used to enter many of these codes, which can also be entered from QWERTY keys using **Ctrl** plus another - as shown on the next page. A control code entered in the first column will define the colour of what comes after it on that row, and whether it is text or graphics, until a further control code is entered.

When writing **text**, if you set an indent (**I** in the command line) and switch on word wrap (**W**) then you can type away without overwriting control codes in the first column/s.

Graphics Lettering and illustrations can be created using the graphics facility. Most alphanumeric keys will produce an equivalent graphic shape when preceded by an appropriate control. This program is admittedly not very user-friendly in this respect but knowing which key produces which shape can be a quick way of composing graphics.

A simple example: Ctrl+q followed by several commas will give a red graphic line. The

full character-space blob is produced by the f0 function key. See below for a chart of the available shapes (each a variation on the six-celled space each character occupies).

Alternatively, When constructing graphics, CTRL + @ will turn the cursor into a sixcel matrix; enter the shape you want using the six keys nearest the arrow keys on the Master keyboard - ie those showing ^ \ [_ :] Each key press will toggle on or off one of the six sectors; when you've achieved the shape required move the cursor on and create your next shape - or turn the feature off with CTRL + p. (Note, you need to be in edit mode to start this feature - a "g" will appear in the bottom line when it is active).

CONTROL CODES

TEXT

CTRL + a = red
b = green
c = yellow
d = blue
e = magenta
f = cyan
g = white

GRAPHICS

CTRL + q = 
r = 
s = 
t = 
u = 
v = 
w = 

RED KEY FUNCTIONS

F0



F1

New background

+ SHIFT

Red text

gr = graphics

+ CTRL

Red gr

F2

Separated gr

Green text

Green gr

F3

Contiguous gr

Yellow text

Yellow gr

F4

Hold gr

Blue text

Blue gr

F5

Release gr

Magenta txt

Magenta gr

F6

Black background

Cyan text

Cyan gr

F7

Conceal

White text

White gr

F8

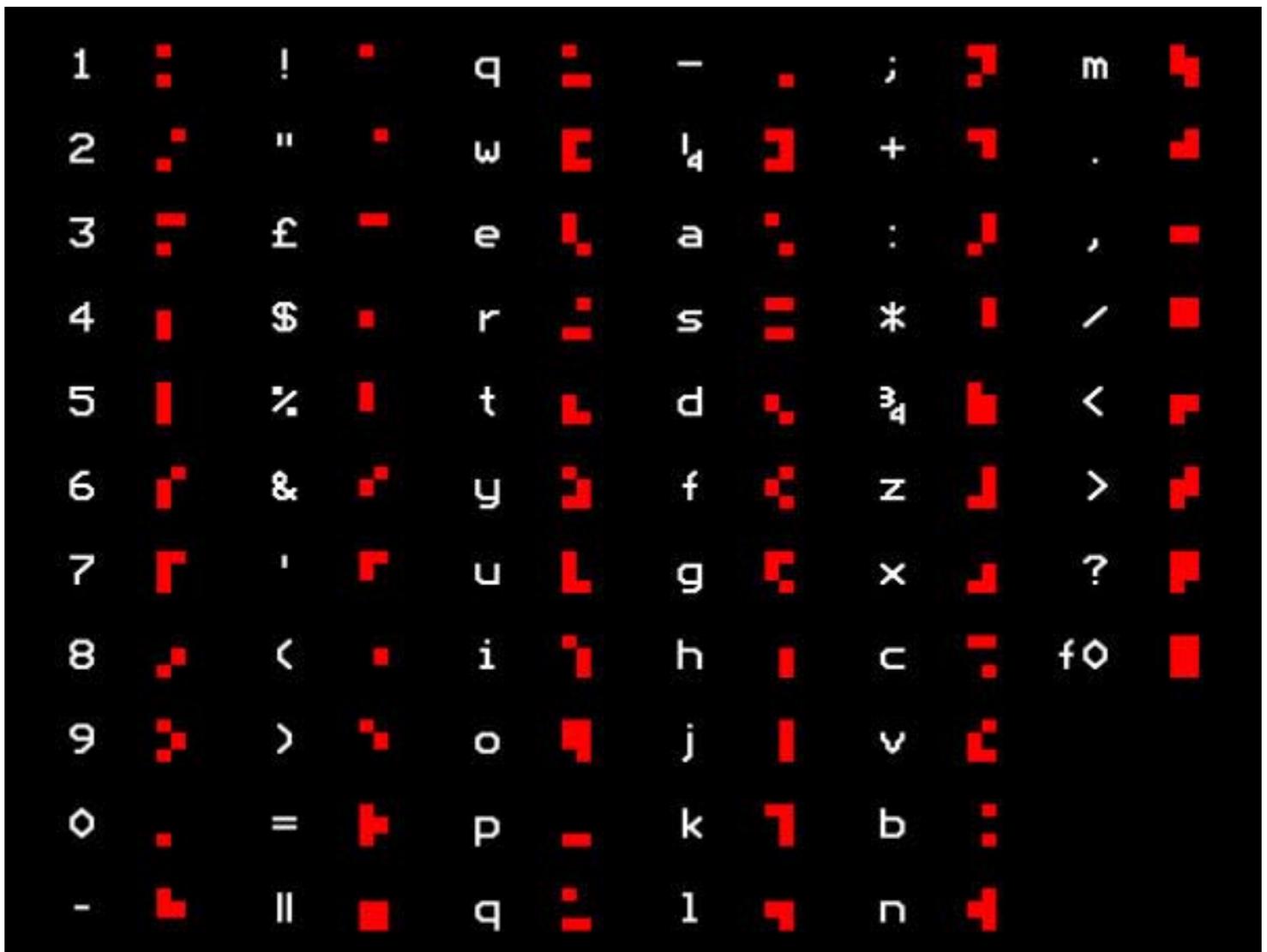
Double height

Flash

F9

Normal height

Steady



Alphanumeric characters and the shapes they produce in graphics mode. Note: the fraction symbols are produced from the Master keyboard in Mode 7 by the { } and \ keys.

Generating animated sequences

In the command line type **p** (lowercase) to load a program for generating transition sequences, such as panning and wipes. "pan" in the bottom line will show that the program is active (the screen will go blank initially, but any pages you have in memory will still be available with the **D** for display command).

Most of the features of the base program are still present but you now have an additional number of commands for creating transitions of different kinds. Nearly all these will act on whatever is in the Page 0 and Page 1 slots. (See 4-Tonline.uk/teletext.html for a video illustrating various effects).

An uppercase **P** in the command line will generate a **panning** sequence, moving Page 0 off left and Page 1 on from the right.

Uppercase **J** will **overlay** Page 1 onto Page 0

^ creates a **push-up** sequence - Page 0 rises off the top of the screen with Page 1 appearing from below

] (right arrow) creates a **wipe down** sequence - Page 1 scrolls down over Page 0

The back-slash \ (or ½) wipes Page 1 across Page 0 **obliquely**

|| creates a **wipe right** sequence - Page 1 is overlaid on Page 0 from the left

@ acts on **Page 0 only** to create a **negative** version of itself, wiping obliquely

? will show the position of control codes on a page, indicated by @ symbols. **D**isplay the page again to recover the image.

These transitions will work best on straightforward block graphics, with control codes in the first column. Control codes in the body of the page can cause unexpected effects, but experiment to determine what "works" and what doesn't.

Before you start, put a timing instruction in the header rows of Page 0 and Page 1. This needs to go as far to the right as possible, in the form of seconds and tenths of a second (for example) **00.1** - one tenth of a second. This is the shortest possible timing and will appear on all the pages created, but individual pages can be easily retimed once the sequence has been created, to make a pause, for example. (The timings are needed for the viewing program - see below).

Except in the case of the @ sequence, Page 0 and Page 1 are not changed by these processes, so to make a sequence flow properly replace Page 1 with Page 0 before saving.

Don't forget to save the resulting pages (the S for save command will automatically return you to the base program, which is required to carry out loading and saving functions).

To view your sequence, if you are in the "base" program or "pan" section, you can scroll through the pages using the **N** (next) or **B** (back) commands, but to view it in (approximately) real time, enter **f** (lowercase) in the command line; this will load another segment of the software - you should see "fader" in the bottom line.

Display page 0, hit Escape to enter the command line and type **V** (uppercase). This will run your sequence according to the timings you have given it in the header row.

The "fader" section has further commands for generating transition sequences such as dissolves, acting on a single page, unlike the "pan" transitions from one scene to another.

Type uppercase **F** in the command line; then enter **U** or **D** (for Up or Down) when prompted, and a "speed" number from 0 to 9.

(If you are displaying Page 0 then **U** will tell the program to create a sequence from Page 0 onwards; whereas if you are showing, say, Page 30 and you enter **D**, the program will operate on diminishing page numbers. The speed function determines the eventual number of pages in the sequence - the lower the number the fewer the pages, and the quicker the effect)

You will now be asked to specify a character which will progressively replace your current image. If you want to blank out your image by degrees you can simply press Return; this will accumulate blank spaces in the image until it is entirely dissolved. Enter any other character, however, and you will end up with a screen full of that character.

When the sequence has been generated the order can be **reversed** by typing **x** and

specifying the range of pages to change (**low number first**). This could turn your dissolve sequence, for example, into its opposite - an image appearing out of a blank screen.

A further command - **+** - operates in a similar way to **F** but will preserve a marked block on the screen by preventing it from being overwritten. Mark a block with the right arrow→ at the start and ← at the end (square brackets] and [on the Master keyboard) then type the **+** command and input the desired parameters.

Unfortunately the arrows will remain on each page generated and would have to be removed manually if necessary.

To recap: the boot file loads the "base" program for general page creation, p in the command line loads the "pan" transitions section, and f loads the "fader" section. Typing b, l, or s takes you back to "base".

Note: when re-loading a sequence you have generated and saved, don't forget it will have been saved as two files - name and name2. (Incidentally, a brief BASIC utility on the SSD called length? can be run outside of Funtext to ascertain the number of pages in a file).

Although the software creates animated sequences such as wipes and dissolves, animating individual objects is more of a slog, requiring frame by frame amendment. But the utilities ROM can be helpful, particularly the ***cut** and ***paste** commands. To show a "walking" movement, for example, create a character and repeat it with legs in appropriate different positions; save each figure using ***cut** (with filenames) then ***paste** them on successive frames in a slightly different position each time.

The ***multi** command enables you to ***paste** the same object in the same position on up to 18 pages at once. If used in conjunction with the panning transition this can give the effect of objects travelling past one that is stationary.

This will only act on pages 0 to 17, however; you would have to paste the object manually on to subsequent pages (although you could get round this to an extent by reversing the order of the sequence - see the **x** command above - so that you have 18 different pages in the 0 to 17 slots, ***multi** pasting the object on to those, and then reversing the order back again - you would now have 36 frames with the object on instead of 18).

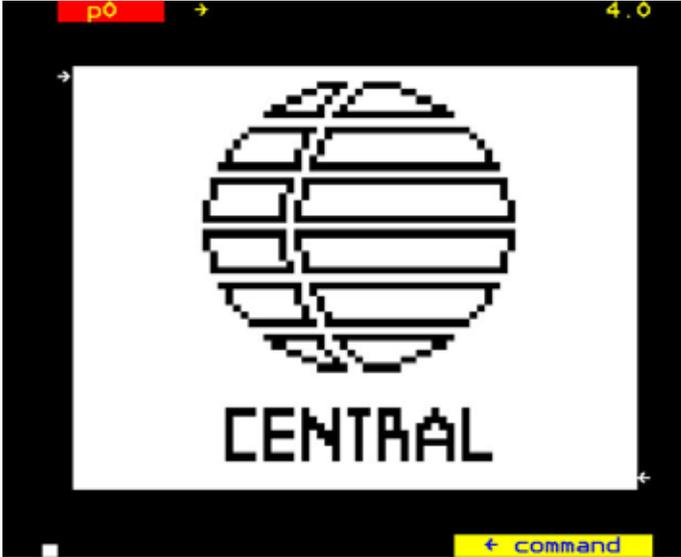
See the following sheet for details of the various utilities on the ROM.

Back in the day, having created an animated sequence for 4-Tel On View, for example, the next step would have been to transfer it from the Beeb into the Intelfax RITE box (Rolling In-vision Teletext Engine) using a short conversion program, ready for insertion into the broadcast signal. Without that option now, you could use the **record video** facility in BeebEm to capture your sequences and run them as video after editing in a suitable program.

(The RITE box timings were more accurate than on the Beeb, which seems to show a sequence more slowly than intended - though even the RITE box was not fast enough when it came to running sequences for the Hands Up! TV series for Channel 4; a video-editing facility was used to speed them up further).

Funtext facilities ROM

The following functions all operate on a marked block using operating system commands. Mark the beginning and end of a block with the arrow symbols (square brackets] [on the Master keyboard) as shown below. Press Escape to enter the command line and then enter the appropriate instruction and press Return.



***rev** reverses a negative image to its positive opposite, or vice versa (pressing Return again will restore the original version).



***up *down *left *right** These commands will move a block one pixel at a time in the required direction. (Especially useful when adjusting the spacing between graphic letters, for example).

***cut** enables a block to be saved to disc - you will be prompted for a file name.

***paste** allows you to insert a saved block at a position specified by a single arrow. You will be asked for the file name.

***turn** will flip a marked block from right to left



***reflect** will give you a mirrored version of your block



***cc** allows you to fill a marked column or columns or part of a column with a specified control code. Use the arrows to mark a block say one column wide, enter the command and then (using only the red function keys) enter the control code you want (eg Ctrl+F1 for red graphics).

***multi** allows the same block to be inserted in the same place on a sequence of up to 18 pages - useful when creating animations, for example. You will be prompted to enter the number of pages to be affected. Note: the pages must exist, even if they are blank - check the index (enter ? on the command line) to make sure. (Display page: N and Replace page: N will ensure there is a page in slot N)

***clear** will erase everything between your markers

***unmark** will remove the markers when you have finished operating on a block. (Bug note: a marker in the bottom right corner of a page may not be removed. This seems to be the same sort of bug that prevents the Premium Bonds computer from ever selecting my numbers)

