Appendix D
*FX calls

*FX calls provide a variety of controls for Operating System functions such as auto-repeat, flash-rate, buffer-flashing, memory allocation etc, etc. The following is a description of all *FX calls available from BASIC.

Call	Description
*FX 0	Prints a message on the screen telling which Operating System you have. Operating Systems are updated from time to time by the manufacturer.
*FX 4	Controls the operation of the four 'arrow' keys and COPY *FX 4,1 disables their editing function, and causes them to generate ASCII codes, just like any other key: COPY 135 'left-arrow' key 136 'right-arrow' key 137 'down-arrow' key 138 'up-arrow' key 139 *FX4,2 allows the five keys to be user-programmable. Their key values become: COPY *KEY11 'left-arrow' key *KEY12 'right-arrow' key *KEY13 'down-arrow' key *KEY14 'up-arrow' key *KEY15 *FX4,0 resets the keys to their normal function of editing. It reverses *FX4,1 and *FX4,2.
*FX9 *FX10	Used to set the flash-rate of flashing colours. *FX9 controls the duration of the first colour, *FX10 the duration of the second.

```
*FX9,25
*FX10,25
```

Will set each colour to say on for equal time of 25 fiftieths (½) of a second. These are in fact the normal values when the machine is switched on.

You could change them to:

```
*FX9,40
*FX10,20
```

which will make the first colour stay on for twice as long as the second; for 4/5 and 2/5 of a second respectively. If one duration is set to 0, the other colour will stay on all the time.

*FX11 Sets the delay, when a key is pressed, before the auto-repeat comes into action.

*FX11,50 will set the delay to 50 hundredths (½) of a second.

*FX11,0 turns off the auto-repeat altogether.

*FX12 Sets the period of auto-repeat.

*FX12,10 sets the auto-repeat to 10 hundredths (1/10) of a second between characters, giving 10 characters per second.

*FX12,0 resets both *FX11 and *FX12 to their normal values. As an example, type in the following:

```
*FX12,1          RETURN
*FX11,1          RETURN
```

and now try typing in anything at all!

*FX13 Disable/enable events.

*FX14 See chapter on Assembly Language.

*FX15 Flushes (empties) certain buffers (short term memories).

*FX15,0 flushes all buffers.
*FX15,1 flushes the currently selected input buffer.

*FX18 Resets all the user-programmable keys to empty.

*FX19 Makes the computer wait for the start of the next screen display frame.

*FX20 *FX20,0 implodes the character definitions. This means that the extra memory set aside for extra character definitions by *FX20,1 to *FX20,6 is returned to the user for storing BASIC programs.

*FX20,1 to *FX20,6 explodes the memory to store groups of 32 extra character definitions. All characters with ASCII codes 32 to 255 may be user defined. As described in chapter 21, ASCII codes 128 to 255 may be defined without using a *FX20 command, but only 32 consecutive characters can be defined. *FX20,1 to *FX20,6 take chunks of memory from the BASIC program storage area to hold specific definitions, and this is shown in the table below.

ASCII code	*FX	Memory allocation
128- 159	*FX20,0	&C00 to &CFF
160- 191	*FX20,1	OSHWM to OSHWM + &1FF
192- 223	*FX20,2	OSHWM + &100 to OSHWM + &1FF
224- 255	*FX20,3	OSHWM + &200 to OSHWM + &2FF
32- 63	*FX20,4	OSHWM + &300 to OSHWM + &3FF
64- 95	*FX20,5	OSHWM + &400 to OSHWM + &4FF
96- 127	*FX20,6	OSHWM + &500 to OSHWM + &5FF

OSHWM stands for Operating System High Water Mark, and means the point where the memory (from &000) occupied by the Operating System ends, and the memory occupied by BASIC programs begins. Turn to chapter 23 for the computer' s memory map. The OSHWM normally sits at &E00, but this will change when a software expansion has been fitted, e.g. a disc filing system.

If you explode the memory allocation in this way, you must remember to reset PAGE higher up the memory. A program stored at &E00 may be lost.

*FX21 Flushes (empties) certain buffers (short term memories).

*FX21,0 flushes the keyboard buffer.

*FX21,4 flushes sound channel 0.

*FX21,5 flushes sound channel 1.

*FX21,6 flushes sound channel 2.

*FX21,7 flushes sound channel 3.

*FX124 Used to reset the flag at memory location &00FF which tells when an ESCAPE has occurred.

*FX125 Sets the above-mentioned flag. Has similar effect to pressing the ESCAPE key.

*FX126 Used when reading characters from an input stream using OSRDCH. *FX126 acknowledges the detection of an ESCAPE.

*FX138 Used to insert a character into the keyboard buffer.

*FX138,0,X will insert CHR$X.

*FX225 Disables all the user-definable keys.

*FX226 Disables FUNC A to FUNC P.

*FX227 Disables FUNC Q onwards.

With a parameter value other than the two given above, *FX255-227 will cause FUNC keys to give ASCII codes. For example, *FX226,224 will cause FUNC A to give 224, FUNC B to give 225, etc. At this setting (*FX226,224), FUNC A onwards will give the standard range of user-defined characters direct from the keyboard. Any number from 2 to 255 may be used as the parameter for *FX226, and determines the base code, which will be given by FUNC A.