24 User-programmable keys

In the same way that FUNC and the alphabetic characters give BASIC keywords at a single stroke, you can program the keys marked 1 to 9 to give any string you choose.

For example,

```
*KEY1 "*CAT"
```

will cause FUNC 1 to print ***CAT** on the screen.

Control characters may be placed in the string, by typing the | character. For example, CTRL M is |M, which performs the same function as the RETURN key. For a list of control characters, see Appendix A. So,

```
*KEY1 "*CAT |M"
```

will cause FUNC 1 to print ***CAT** on the screen, and to set the command in operation. Therefore, a whole program, or a series of commands can be stored in one key.

A useful routine is to have one key which returns the computer to MODE 6 and lists the program in paged mode:

```
*KEY0 "MODE 6 |M |N LIST |M"
```

Here is a key definition containing a small BASIC program:

```
*KEY3 "10 REPEAT |M 20 PRINT CHR$(RND(95)+3
1) |M  30 UNTIL VPOS = 24 |M RUN |M"
```

For the more advanced

The BREAK key can be programmed also. It takes the value 10. The following program cannot be stopped either by ESCAPE and BREAK:

```
10 ON ERROR GOTO 30
20 *KEY10 "OLD |M RUN |M"
30 PRINT "YOU CAN'T STOP ME!"
40 REPEAT UNTIL FALSE
```

ON ERROR is described in chapter 27. Actually, this program can be halted by pressing **CTRL** **BREAK**.

CTRL **BREAK** is called a ‘hard reset’. It resets everything very nearly the way it was when the machine was first switched on. When you try it you’ll hear a beep and you’ll see that the ● reappears after the message at the top of the screen.

BREAK on its own is called a ‘soft reset’. It is roughly equivalent to pressing **ESCAPE** and entering the commands **NEW** and **MODE 6**.

The soft reset does not clear the ***KEY** definitions for example.

The five screen editing keys can also be re-defined, just like **BREAK**, after the issue of a ***FX** command.

Keys can also be loaded with the contents of BASIC variables. The instruction which does this is OSCLI, which stands for operating system command line interpreter. It can be used with any operating system call from BASIC (distinguishable by a preceding asterisk), for example, ***KEY**, ***SAVE**, ***LOAD**, and so on. Each BASIC variable assigned to the **KEY** definition must be converted into a string, and the asterisk omitted, as follows:

```
OSCLI "KEY" + STR$X + "LIST |M"
```

This will put "**LIST|M"** into KEY X, where X is a BASIC variable.

There is no limit to the number of BASIC variables which may be used in an OSCLI assignment, provided that they are all either string variables, or are turned into strings using **STR$**.