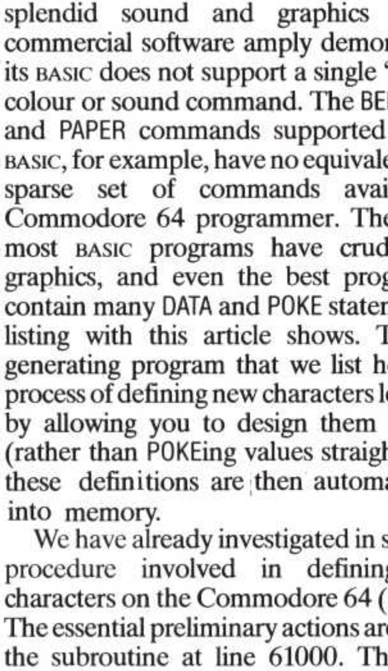
TYPE CAST

The version of BASIC used on the Commodore 64 is not without limitations, although these are seldom a great obstacle to thoughtful and creative programming. We give you a program that enables you to define your own character set.

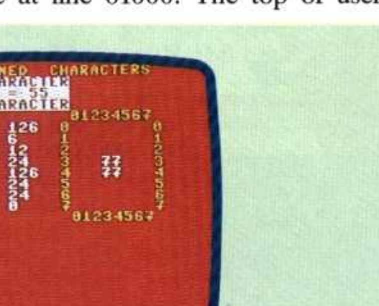
The Commodore 64 is capable of producing splendid sound and graphics — as much commercial software amply demonstrates — but its BASIC does not support a single 'purpose-built' colour or sound command. The BEEP, DRAW, INK and PAPER commands supported by Spectrum BASIC, for example, have no equivalents among the sparse set of commands available to the Commodore 64 programmer. The result is that most BASIC programs have crude sound and graphics, and even the best programs tend to contain many DATA and POKE statements — as the listing with this article shows. The character-generating program that we list here makes the process of defining new characters less demanding by allowing you to design them on the screen (rather than POKEing values straight into RAM); these definitions are then automatically POKEd into memory.

We have already investigated in some detail the procedure involved in defining your own characters on the Commodore 64 (see page 232). The essential preliminary actions are performed in the subroutine at line 61000. The top of user memory is lowered from location 40959 to 14335. The entire two Kbyte upper case character set that the Commodore has resident in ROM (from address 53248 onwards) is then copied into RAM (14336 onwards) where it can be accessed and manipulated using PEEK and POKE statements. Finally, the VIC (video interface chip) is switched to address the relocated character set.



USER-DEFINED CHARACTERS
CHARACTER
7 = 55
CHARACTER
76543210 01234567
126
6
12
24
126
24
24
0
76543210 01234567
USER-DEFINED CHARACTERS

Sans Serif
In the left-hand, or edit, window is the re-defined version of character number 55 — the figure seven. The serif has been erased, and the continental cross-bar added. In the right-hand, or text, window are four copies of the seven. The changes are visible in these sevens and in those around the windows, but the seven in the status box retains its original definition



USER-DEFINED CHARACTERS
CHARACTER
% = 37
CHARACTER
76543210 01234567
156
12
148
255
249
232
222
224
76543210 01234567
USER-DEFINED CHARACTERS

Claws
Sprites are just multiple user-defined characters. Nine characters have been re-defined, and placed together in the text window. The per cent symbol in its re-defined form is the middle rightmost section of the crab

IAN McKINNELL

Once the character set has been relocated in RAM, two 'windows' are displayed on the screen by the initialisation routine, and control passes to the input routine at line 2500. This routine scans the keyboard and maintains a flashing cursor in the left-hand or 'edit' window. The character currently being redefined is displayed (suitably magnified) in this window, with the values of its eight defining bytes next to it.

The unshifted function keys (f1, f3, f5, f7) control cursor movement inside this window. The cell under the cursor (corresponding to a bit in one of the eight definition bytes) can be toggled on or off with the shifted function key f2. When this happens, the eight definition values are updated, and all occurrences of the character elsewhere on the screen can be seen to change immediately.

Pressing the shifted function key f4 allows you to replace the character in the edit window with another character. Characters are described by their POKE (or screen code) values as listed in Appendix F of the User Manual. These values are not the same as the CHR$ codes (although there is a correspondence), but are more convenient to use here since the character definitions are arranged in memory in the order of these codes.

The shifted function key f6 allows you to write a 'character-sized' copy of the character being edited into the right-hand (or 'text') window, at the position corresponding to that of the edit cursor. If the cursor is in the top left corner of the edit window, for example, and 'A' is the character being edited, then an 'A' will be written in the top left corner of the text window when f6 is pressed.

Finally, pressing the exclamation mark key STOPs the program; CONT will restart it. When you 'quit' or exit from the program, you can type NEW and then LOAD another program without disturbing your re-defined character set. However, there are problems involved with this. First of all, the top of user memory has been lowered so that there are only 12 Kbytes available for the new program. Secondly, switching off the machine destroys the new set. Both problems will be discussed in a future article. In the meantime you should note the definitions of your newly-defined characters and use them, if necessary, as shown in the sample program on page 233.