



byte contains a number, or an address, or a coded character, or an instruction, or whatever you like. In any case it will be data waiting to be interpreted. Secondly, it does give a rather more understandable view of memory, especially those parts of it which do actually contain character data, some of it used by the machine's Operating System, and some of it used by you.

Operating System data includes all the Error and Prompt messages — READY, for example, or

NONSENSE IN BASIC, or START TAPE THEN PRESS RETURN — anything that it is capable of saying to the user has to be ASCII coded and stored in memory. You may never have thought of that, and it's a revealing insight into a computer's limitations as an 'intelligent' machine. Our intelligence is obviously different: we don't memorise messages like that, we simply frame a thought and then generate an appropriate combination of words to express it.

Memory Maps

A memory map is a simple schematic representation of the use to which memory is being put, showing the parameters of specific areas. Some areas of memory are always used for the same purpose. On the Commodore 64, for example, bytes 0 to 1024 are used by the BASIC Operating System as a Work Area. Other areas of memory have varied uses depending upon the program size and state. The boundaries between these areas can be either *fixed* (shown in our diagrams below as solid lines) or *floating* (shown as broken lines). Fixed boundaries never alter, while floating boundaries are for those areas of memory that fluctuate according to need. In the Commodore memory map, the Screen RAM boundaries are fixed (at bytes 1024 to 2048), while the boundaries of the memory area where the BASIC Variables are kept fluctuate according to how many variables are being

used at any given time.

The Mempeek programs on page 59 can be used to locate the current positions of the floating boundaries in the memory of your machine. The Commodore has six floating boundary pointers (also called System Variables). An example is given in the panel below explaining how the contents of a pair of bytes are used to calculate the required memory address. BBC BASIC has four System Variables to determine, and the Spectrum has five.

It must be remembered that a memory map is a static representation of something that is forever changing while the machine is in use. Each of the floating boundaries is subject to change at any time. We give ideas, on the next page, of how you can extend the Mempeek programs to observe variations in the pointer values.

- Commodore System Variables**
- 43,44 Start of BASIC PROGRAM TEXT
 - 45,46 Start of BASIC VARIABLES
 - 47,48 Start of BASIC ARRAYS
 - 49,50 End of BASIC ARRAYS
 - 51,52 Bottom of BASIC STRING STORAGE
 - 55,56 Top of BASIC STRING STORAGE

Example

Use the Mempeek program on page 59 to inspect the contents of these bytes. As an example, your screen display could look like this:

```
43 0 8 11 9
```

The first column is the address of the first byte accessed. The second and third columns display the contents of byte 43 and byte 44. These are the offset and page bytes (see page 36) of the address of the start of the BASIC Text Area. This is calculated as:

$$8 * 256 + 0 = 2048$$

The fourth and fifth columns in the display are the offset and page bytes for the end of the BASIC Text Area. The address is calculated as:

$$9 * 256 + 11 = 2315$$

