



is faced with an apparently insoluble dilemma.

One answer is not to use BASIC at all, freeing areas like the BASIC Program Text Area. But it's usually convenient to write at least part of a program in BASIC, using machine code subroutines only where BASIC is too slow — animating screen displays, for instance. If BASIC and machine code programs are to coexist in RAM, then we must steal some space and designate it for machine code. We can move the boundaries of the BASIC Program Text Area, or we can find temporarily unused sections of it or of the operating system RAM.

Moving BASIC's boundaries is very easy on the BBC Micro, since these addresses are held in the system variables PAGE, TOP, LOMEM and HIMEM. PAGE, for example, points to the start of the BASIC Program Text Area — which is usually at address \$1200. If we execute the instruction:

```
PAGE=PAGE+500
```

then the operating system will store BASIC

programs 500 bytes higher in memory, leaving a 500-byte area free for our machine code programs. The same effect on the other machines is achieved by POKEing higher addresses into the system pointers (see the memory maps on page 58). Alternatively, we could reserve space by lowering the address of the top of the BASIC Program Text Area. On the Spectrum, the command CLEAR followed by an address does exactly that. The only constraint on the amount of space stolen from BASIC in this way is that there will be enough space left for our BASIC program.

Small blocks of free space can be found in the operating system RAM, a typical example of which is the cassette buffer of the Commodore 64. This consists of 192 bytes of RAM (from \$033C to \$03FB) and is used by the operating system only when the cassette drive is employed. Many programmers find that this space alone meets all their machine code needs.

Even smaller blocks of space can be found within BASIC programs — REM lines, for example.

CPU Internal Organisation

The Z80 has a more complicated structure than the 6502. Registers are used as work space RAM in the CPU's operations, or for specific tasks such as controlling the stack, holding the address of the next program instruction, showing the effects of the last CPU operation, or holding addresses.

Both CPUs have an eight-bit accumulator connected to the ALU where arithmetic and logic operations are performed, but the Z80 also supports some 16-bit arithmetic in its paired registers (known as BC, DE, and HL).

