A to F for the numbers 10 to 15. Each memory location comprises eight binary digits (bits) and these may be represented by a pair of 'hex' digits.

For example, the binary pattern 01011101 would first be split into two halves: 0101 and 1101. These would be translated into the decimal numbers 5 and 13 — which equate to 5 and D in 'hex'. Thus 01011101 is referred to as 5D when programming in machine code. This is the slowest way to develop programs, but probably produces them with the fastest execution times. The following are some extracts from typical programs:

BASIC 100 INPUT "Enter hours worked"; HOURS 200 PAY = HOURS \* RATE

The first line displays a message on the screen inviting the user to enter the hours worked. It accepts the input, then in the second line, multiplies it by a rate of pay (entered earlier in the program) to give a gross pay figure.

> Assembly Language MVL C, 01 CALL 05

The first instruction above moves the value '1' into a part of the microprocessor's memory called the 'C register'. The second instruction asks the operating system to take control. The operating system then passes control back to your program. Here is a direct translation of the assembler lines above:

> Machine Code 0E01 CD0500

A machine code translation of the two BASIC statements would comprise many commands. It is clearly preferable to allow a compiler or an interpreter to do this work rather than to write in assembly language or machine code.

Some operating system commands are almost cryptic as assembly language programs. Here are some examples from CP/M (Control Program for Microcomputers):

DIR \*. BAS

The above means 'list all the files on the current disk drive whose suffix is BAS'. DIR is an abbreviation of Directory.

Despite this rather odd approach to its commands, CP/M is installed on more than a million machines. It has massive advantages for professional software writers. Programs can be easily transported from one machine to another, provided they are written in such a way that they pass control to CP/M to handle the disks, keyboard, printer, and the screen.

Manufacturers who introduce a new machine can benefit from an existing base of software, and would-be purchasers can buy machines, confident that their software requirements are likely to be met.



## From Key To Screen

(1) Pressing a key on a computer keyboard usually results in a letter appearing immediately on the monitor. The process is more complex than it looks, however. When a key is pressed, a signal is sent via the I/O chip (2) to the CPU. The CPU (3) runs a small program in the ROM that works out which key has been hit, and outs a corresponding code in the memory that is currently being used as storage for the screen image, screen RAM (4).

Mearwhile the Video Circuitry (5) is constantly scanning the screen RAM, and by fetching the bit patterns associated with each letter from a 'charactergenerator' ROM, sends a signal to the television or monitor, which then draws the letter on the screen (6), in the right place, and f selected, in one of the available colours.

All this apparently frantic activity nappens in a few millionths of a second



Timing within the computer is critical. For example, if the CPU reads a byte from memory, it must set the address lines to the byte location. The CPU needs to be sure that the memory chip has had time to put the byte of information onto the data bus. Only then will it accept and "read" the byte into the CPU's internal memory. The clocks in home microcom puters 'tick' as fast as four million times a second