

6 Introducing commands and programs

The Electron computer, like any computer, has to be told what to do before it will do anything for you. The only way you can ‘talk’ to it is by typing *commands* to it on the keyboard, and the Electron tells you what you typed by displaying it on the TV screen.

The Electron understands two languages, one called BASIC and the other called *Assembly Language*. As these are written languages only – in other words they are meant to be typed into the computer rather than spoken to it – they each have their own special vocabulary and grammar.

Talking to the computer in Assembly Language is no easy task for the beginner so it is discussed in a separate section towards the end of this book. From now on, we will be helping you to speak to your computer in the language called BASIC. This language consists of a number of words or commands. Some of these are printed on your Electron keyboard in light brown letters.

Note that the Electron displays a > sign followed by the flashing cursor. This is called a ‘prompt’ and means that the computer is waiting for your instructions. Normally the Electron will prompt you when it is waiting for you to type something in. The > prompt means that the Electron is expecting a BASIC command.

Press the **BREAK** key to clear the screen, and type the following line

```
PRINT "HELLO"
```

Then press **RETURN**. As soon as you press the **RETURN** key, the computer obeys the BASIC command **PRINT**, and because the rest of the line was in quotation marks it displays the word **HELLO** on the screen.

Now type this

```
PRINT "I'M"  
PRINT "LEARNING"  
PRINT "BASIC"
```

22 Introducing commands and programs

DON'T FORGET TO PRESS THE **RETURN** KEY AT THE END OF EACH LINE!

As you can see, the Electron reads each command and executes it immediately after you press the **RETURN** key. You can also type in all these commands first and then make the computer execute them when you tell it to do so. This is done by putting a number in front of each instruction.

Now type the following:

```
10 PRINT "HELLO"  
20 PRINT "I'M"  
30 PRINT "LEARNING"  
40 PRINT "BASIC"
```

DON'T FORGET TO PRESS THE **RETURN** KEY AT THE END OF EACH LINE!

The computer hasn't carried out your instructions like it did before, so now type

RUN

Followed by the **RETURN** key.

This time the computer has printed your message on the screen all in one go.

Congratulations, you have just run your first program on the Electron Microcomputer!

So a program is simply a collection of numbered instructions. The line numbering has two purposes: firstly to tell the computer not to execute each line after you have pressed the **RETURN** key, and secondly to help the computer decide in what order it should execute the instructions – after you have typed **RUN** of course.

The actual numbers you type in can be any numbers you like as long as you remember that the computer will execute the program lines in numerical order.

The chapters which now follow serve as an introduction to the Electron BASIC language, and how to use the facilities it offers you. The chapter

on editing your programs will help you speed up the process of typing in programs and making changes to get them working.

Unlike a typewriter, you *don't* need to press the **RETURN** key when what you have typed has filled up the current line on the screen – just carry on typing. What happens is that the computer starts a new line on the screen, and the subsequent characters you type are displayed on the new line. Prove this to yourself by typing a lot of characters.

Remember that pressing the **RETURN** key tells the computer that you have reached the end of the command or program line you have just typed. If a command, the computer then executes it, and if a program line, the computer stores it in its memory.

What is hexadecimal?

Once you get more familiar with the Electron, you may come across things called 'hexadecimal' numbers. Here is a brief explanation of what they are.

Hexadecimal numbers, sometimes called 'hex' numbers for short, have sixteen separate digits, compared to our decimal numbers which only have ten (including the number zero). This is how you count in hexadecimal, with the decimal equivalent underneath.

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12...
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18...

To show you how to carry on counting in hexadecimal:

12	13	14...	19	1A	1B	1C	1D	1E	1F	20...	2F	30...
3F	40	...	90...	9F	A0	A1	A2...	AF	B0...	BF	C0...	
. FF	100	101	...	etc								

To help you and the Electron tell the difference between decimal numbers and hexadecimal numbers, you should always type an '&' sign in front of a hexadecimal number. If you don't, then the computer will assume your number to be a decimal. So we can now say &A0 = 160 (hexadecimal A0 is equal to decimal 160).