Chapter Nine
Postscript

Nobody who has read the rest of this book will have missed the fact that I am an unashamed admirer of the BBC Micro. As far as I am concerned it is at the top of its league and rightly so. The factor that makes it stand out from the other currently available home computers is its overall design concept.

Even the BBC Micro Model A has a very impressive specification for a microcomputer. When expanded to the Model B it is one of the most capable and sophisticated machines on the market. However, the expansion doesn't stop at the Model B. A quick look inside to see all the unfilled spaces on the printed circuit board soon confirms this. The BBC Micro is not just a brilliantly designed microcomputer, it is the start of a complete system. There is provision on the printed circuit board for a disc interface, a speech synthesizer, an Econet interface and a removable ROM Pack.

With all this *on-board* expansion it is easy to miss the overall unity behind the design of the BBC Micro system. For example, the standard Model A and B machine, just like most small micros, uses a cassette recorder to provide a backing store. However, the BBC Micro uses the tape recorder as only one of a number of *filing systems* whereas other micros use completely different ways of dealing with different forms of backing store. When you add a new piece of equipment, such as a disc or an Econet interface, you not only have to install the necessary chips, you also have to provide new filing system software in the form of additional paged ROMs. You can select one of the possible filing systems by using MOS commands such as *TAPE, *DISC or even *NET for the Econet filing system.

There are other extras that can be stored with the BBC Micro apart from the on-board circuits. The mysterious *tube* has already been mentioned in Chapter One as a way of connecting another CPU to the system. The two add-on processors available at the time of writing are

remarkable in themselves. A 3 MHz 6502 processor board will be capable of running programs even faster than the 2 MHz 6502 in the BBC machine itself - and the BBC machine already has a reputation for being a very fast machine. The Z80 card can run CP/M programs with a much larger amount of user RAM than is normally available in a CP/M system because the BBC Micro takes care of all the disc and other I/O handling. If these two extra processors weren't enough, the promised 16-bit processor should bring large computer performance to the microcomputer world.

You may feel that using the BBC Micro as nothing more than a graphics VDU to the second processor is a waste of hardware. This is far from the truth. Most of the cost of a micro lies in the video display, the keyboard and all the other peripherals. The microprocessor itself and the RAM are rapidly becoming very cheap commodities and using the BBC Micro as an 'environment' in which other microprocessors can work is a very sensible arrangement. Acorn have been clever enough to find a way of interfacing these second processors over a fast and well-defined interface - the tube.

For technical and laboratory situations, it is worth knowing that the 1 MHz bus can be used to connect the original Acorn range of 'Eurocard' modules. This is particularly useful when a wide range of extra digital and analog interfaces are required. It is also particularly easy to build your own cards to work from the 1 MHz bus. This is more than can be said for the tube which needs a special Acorn-designed ULA. With its speed, graphics and internal interfaces, the BBC Micro is an ideal scientific machine (and any machine that is a good scientific machine is a good business machine!)

This book has tended to stress the BBC Micro as an isolated computer. The reason for this has been the need to keep to a reasonable length! Even the on-board expansion - for example, the disc system and speech synthesiser - has been neglected to make room for a discussion of standard Model A/B features. The problem about starting to discuss the BBC Micro's expansion capabilities is that they are vast. Each of the pieces of equipment mentioned in the previous few paragraphs deserves at least one chapter to itself and that's just the start of the list of possible extras. To it we already need to add the Prestel interface and the Teletext add-on. In some ways it is premature to write about the BBC Micro's expansion modules - at the time of writing not all of them are available in the sense of being on the market, although I have seen all of them (apart from the 16-bit second processor) in pre-

production form. More importantly, it would be misleading for me to select which are the vital interfaces for inclusion in this final chapter. The interfaces that *you* will choose to expand your system will depend entirely on the purpose you want your BBC Micro to serve. Not everybody will want a disc drive or even a printer. Certainly, lots of BBC Micro systems will be complete without an Econet interface. Prestel and Teletext are currently billed as being important to every home in the future, but whether this prediction comes true remains to be seen. The 'home information' age has, however, arrived with the advent of the BBC Micro!

The only real way to appreciate the full implications of the BBC Micro's role as the heart of an expandable system is to study a few of the add-ons and see how they fit into the existing machine. For myself, this is certainly what I intend to do - and having done so will be in a proper position to write, not just a chapter, but a complete book about The BBC Micro System.

Index

A to D convertor, 93-107
ACIA, 9
adding commands, 48
addressing modes, 124-152
ADVAL, 99
analog, 98
arrays, 37
assembly language, 115

BASIC format, 31
BREAK key, 15

CALL, 124, 134-5
cassette, 7
character generator, 56
character table, 65
chords, 80
colour, 58
conditional assembley, 141
CPU, 2

digital, 97
disc interface, 153

Econet, 153
ENVELOPE, 83
ESCAPE key, 47
Eurocard modules, 154
events, 52

fire buttons, 100
functions, 23, 27, 37

garbage collection, 40

handshake lines, 104, 107
heap, 32
heap dump program, 38
hexadecimal, 29-30
HIMEM, 31

indirection, 29
instruction set, 130
interfaces, 6
interfacing, 96
integer variable, 35
interrupts, 49

joystick, 101-102

keyboard, 13

label, 123, 136
language work area, 31
local, 25, 42
LOCAL, 26, 42
LOMEM, 32

macros, 141
memory map, 19-20, 31
mode 7, 74
MOS, 43
MOS indirection, 45

NEW, 33

OLD, 33
OSBYTE, 45
OSWORD, 45

PAGE, 32
page zero, 31
paging, 3
palette, 69
parameters, 25
PEEKing the screen, 64
Prestel interface, 154
procedure, 23, 26-7, 37

real variable, 36
recursion, 28

registers, 128
resident integer variables, 40
RS243, 7

screen dump, 142
screen memory map, 55, 59-64
scrolling, 72
second processor, 154
serial control register, 12
serialiser, 69
shift register, 112
soft machine, 43
SOUND, 77
sound effects, 82
sound generator, 76, 89
sound queue, 79
speech synthesiser, 153
stack, 32
string indirection, 30
string variable, 37
structured programming, 22
subroutine, 23

Teletext add-ons, 154
Teletext graphics, 74
timers, 109
tokens, 33
TOP, 32
transducer, 97, 101, 103
Tube, 17, 154
tunes, 79
two-pass assembly, 133

user port connections, 104
USR, 134

variable format, 34
variable storage, 34
VIA, 13, 104
video, 4, 49, 54
video generator, 4, 67
video processor, 4, 67
voltage divider, 102

1 MHz bus, 15, 154
6502, 2, 115
?, 29
!, 30
$, 30
&, 30
~, 30
*FX, 46