INFORMATION TECHNOLOGY

Gower

The Personal Computer Book

by Robin Bradbeer

Second Edition

THE PERSONAL COMPUTER BOOK

The Personal Computer Book

Second Edition

ROBIN BRADBEER

Gower

First published in Great Britain in 1980
by Input Two Nine.

Second edition published by
Gower Publishing Company Limited,
Gower House, Croft Road,
Aldershot, Hampshire GU11 3HR
Phototypeset by Tradespools Limited, Frome, Somerset

Bradbeer, Robin
The personal computer book. — 2nd ed.
1. Microcomputers
001.64'04 QA76.5

ISBN 0 566 03423 9 (Paperback Edition)
ISBN 0 566 03445 X (Hardback Edition)

Reproduced from copy supplied
Printed and bound in Great Britain
by Billing and Sons Limited and Kemp Hall Bindery
Guildford, London, Oxford, Worcester

CONTENTS

FOREWORD

The response to the first edition of this book showed the need to explain personal computers in everyday language. There is also need to give advice on how to purchase an initial system. Nobody would claim to have all the answers — and I certainly don't. However, my experience over the past few years, as retailer, customer, teacher, journalist and computer club chairman has helped to crystallise some of the questions that need to be answered.

Most of the material in this, and the previous edition, has been used in a series of talks to novice computer hobbyists at the North London Hobby Computer Club. The positive comments from the listeners have helped me eliminate a lot of material that is not necessary, and also made me include things that I initially didn't consider! Since the first edition, letters and conversations with readers have shown areas that need reinforcing. These are now fully covered, I hope! Comparison with the first edition will indicate certain differences. These are basically to make certain sections easier to understand. I am indebted to all those who gave me advice, solicited and unsolicited in this area!

In a book of this sort — where such a wide ranging subject is covered in a few pages — criticisms will always be made about the technical explanations. My apologies in advance to those "experts" who expect a technical discussion to take at least fifty pages! If you have only a page to describe, say a disk drive, then that explanation will obviously be rather superficial. I hope that I have been able to cover the essentials in enough depth to do the machinery justice! At the same time it has been impossible to look at every computer around. In choosing those mentioned, I have looked at machines that are generally available — and therefore represent over 99 percent of the market. Apologies to those who are offended if their favourite machine is not included. Whenever details are quoted it must be obvious that with specifications increasing and prices decreasing changes occur daily. Please consult the latest edition of a magazine, or chat to your local store, before estimating the cost of a system. Prices and specifications are for guidance only!

In writing this book I have had help from many sources. These include Gary Marshall, Mike O'Reilly, Richard Ross-Langley, G. Winnestein, and Mike Fluskey the publisher.

With the BBC and ITV series presenting the ideas of personal computing to a massive audience, my hope is that this book will provide more detailed background reading to supplement information coming from those sources.

R.T.B.

CHAPTER 1

Introduction

Everybody's talking about computers; on TV, in the press, and on the radio. Words like "home computing", "hobby computer" and "personal computer" are mentioned all the time.

Computer technology has now grown so cheap that people like ourselves can now afford to buy our own computer system. We can now have fun with, and make use of, all the properties a computer has to offer.

In fact the home computer is a reality. During the last 3 or 4 years, small, cheap computer systems have begun to appear on the market — many of them as powerful as large computer installations of the 60s. Instead of costing millions of pounds, the price tags now show £100, £500 or perhaps £1,000.

At the end of 1981 there were around 200,000 home computers in various sizes and versions in the UK. The number of people interested is far greater. What has generated this interest?

For the first time in the history of computer technology (which in fact, only goes back about 35 years, counting from the first electronic computer), individuals can now operate and control a computer in their own home.

Previously, large computer systems have been controlled by companies and institutions. Many people have had to co-operate in working with the computer, so it has cost a lot of money to "talk to" the computer. In many cases, this has limited the possibility of using a computer to its fullest ability.

These limitations don't exist any more. We are now in a position to decide **what** we want to do. And **when** we are going to do it.

The words "home computer", can now take on a similar meaning to words like "domestic help", "personal servant" and "private secretary".

What are computers being used for?

We can play games on the computer or we can let the computer control various functions in the home. The computer can keep order in our private financial affairs. It can practise vocabulary lists or maths homework with the children — not to mention the value it has for today's children, allowing them to come into contact with computer technology at an early age (for they are going to live in a computer society).

We can keep a check on books, records and other property by storing information in computer files. The computer can help us to write letters, keep a tab on addresses, etc.

It is possible for businessmen to rationalize their work in bookkeeping, stock accounting, customer files, the dispatching of advertisements, etc. The personal computer does the job cheaply so that human resources can be freed for more creative occupations.

For those who already have a hobby (model railways, philately, amateur radio, stock investments, rose-gardens or whatever) surprisingly, computer technology can often be integrated with the hobby and enrich the experiences it gives.

Real enthusiasts say, "There is no limit, really, to what you can do with a computer — only your own imagination". What must be remembered is that a computer is just a programmable machine. It's an intelligent idiot! It can do, within reason, whatever you tell it to — and then very quickly and, usually, without mistakes.

This sounds too good to be true. Of course there are limits. One's wallet, for instance! Even though today's computer technology is cheap it **can** run away with the money.

There are technical hurdles to be overcome before the computer can work in the applications one has thought out, but the technology is now developing very quickly. The opportunities are being improved daily by new products, concepts and ideas.

This book offers you guidance in the new computer technology. It points out the possibilities — and warns you of the pitfalls!

How is it possible?

What is it, then, that makes this revolutionary development possible?

First and foremost, the answer is semiconductor technology, leading to the integrated circuit. This new technology makes it possible for a little silicon chip (only 5-6 mm square) to contain tens of thousands of transistors.

In the early 60s researchers discovered how to miniaturize even further by bringing several transistors together in a single unit during the process of production and the **integrated circuit** was born. Since then, development has proceeded at breakneck speed. Circuits have become even more complex as production methods become more refined. Today, about 100,000 transistors can be fitted onto a silicon chip measuring 6 mm x 6 mm

The tiny chip of silicon on the girl's finger is a microprocessor which contains the equivalent of 20,000 transistors.

Manufacturers' predictions suggest that in 1985 there will be one million transistors in the same place. With this technique (Very Large Scale Integration, known as VLSI) manufacturers can build up the exact electronic circuit required.

Manufacturing technology, however, is highly advanced. It costs a lot of money to develop each individual configuration of circuits. This is why manufacturers concentrate on integrated circuits that can be sold in large numbers. This makes it possible to reach very low manufacturing costs per integrated circuit. Some simple circuits produced by the million, cost only a penny or two to manufacture. More complex circuits cost more, sometimes as much as a few pounds.

The development that has been of decisive importance in the last few years is the **microprocessor**, which arrived in the early 1970s. In simple terms, the engineers collected all the components of the central parts of a computer and placed them, by means of the VLSI technique, on one small silicon chip. The result was a cheap building component, of universal application, and readily available. Since then, the manufacturers have not been idle and better microprocessors have appeared. Larger memories have also been developed by similar techniques, and more and more functions within the computer have been realized with the VLSI technique.

This has resulted in lower prices, greater reliability, and above all — electronics moving into areas where it had not been practical before.

The microprocessor has made it possible to place computer power where it is needed. In cash registers, cars, sewing machines, lathes, industrial automation systems, calculators or vending machines — not to mention the great importance of the microprocessor in building computers, both for professional and home use.

A simple method for looking at these developments is to consider the dimensions of the equipment. Five or ten years ago, a computer installation filled a fair-sized room. Today the equivalent computer power can be put on a table.

On the consumer side, there have already been advance signs of the approach of the home computer:

The pocket calculator, which at one blow took the wind from the sails of the calculating-machine industry, with its continuous price reduction coupled with ever more advanced technical content.
Anyone who has bought a TV game has in fact taken the first step towards a home computer. The TV games now coming on to the market are often small computers even though they are made with one purpose in mind, to act as TV games. A real home computer has a more universal range of applications — that is its importance.

Given the direction of present developments, the TV set will become the computer centre of the home — an information centre that provides not only TV programmes but also the possibility of storing, processing and communicating with other similar systems. In a few years, the home computer is likely to be as natural a part of the home as the TV set is today.

The first hobby computer

The first hobby computer, or personal computer that deserved the name, came onto the American market in January 1975. It was the small New Mexico company MITS that launched a kit for under $500, and gave it the name Altair, which was the name of a star in the TV series "Star Trek".

Altair turned into a real comet. It set fire to the market — and to everyone who discovered how cheaply they could get hold of their own computer.

Altair also attracted a veritable rain of stars. In only a month or so various other manufacturers of hobby computers — and above all of accessories for hobby computers — had appeared by the side of MITS.

The Altair concept became something of a guideline for many of the other manufacturers of personal computers. Several other computers of basically similar design appeared from manufacturers like Imsai, Processor Technology, Polymorphic and many others. Even more firms in the USA established themselves as manufacturers of accessories for these computers. Unfortunately, their pioneering spirit was not backed up by the financial expertise to cope with the massive demand generated, and most of these early companies are now defunct.

The main point about these early companies was their use of **BASIC** as the main programming language, and the development of a common inter-connection system, the S-100 bus. This allowed different manufacturers to have interchangeable components, thus giving the consumer not only a choice of varied supplies, but also a range of accessories that were relatively similar. It was this common bus structure and language that really got personal computing off the ground, and laid the foundation for the present boom.

Quite soon after the first Altair had seen the light of day it was possible in the USA to buy a computer, a keyboard, an interface to change an ordinary TV set into a data screen, and an interface to make a cassette recorder into a computer memory — for a little over $1,000. Then one could work with this system using the programming language, **BASIC**, with the knowledge that most others were doing the same thing.

This meant a properly working computer system at the same price as a good stereo set or colour TV. And an entirely new computer industry sprang up from nowhere. Hundreds of new producers of computer supplies have established themselves in the USA, with many of these and indigenous companies now operating in the UK.

Computer shops began to appear. Today there are something over 2,000 computer shops in the USA. Their numbers are still growing. In the UK we now have around 750 — up from 5 in 1978!

There are also vast numbers of clubs for computer enthusiasts throughout the world. They have lively meetings and fast-growing membership.

At least a dozen really good magazines have become established in the field of personal computing in the USA, eight in the UK. The leading one, *Byte Magazine*, a monthly that started in 1976, has a circulation that's going up like a rocket and is now a good bit over 180,000, which makes it the world's largest computer magazine of any type.

Trade fairs, conferences and seminar courses are being organized for thousands of participants. The most popular regularly attract 15,000 or more personal computing enthusiasts.

In the UK market predictions indicate that by 1985 around 400,000 units a year will be sold, generating a market of around £500m. This is similar to the radio/TV/hi-fi market — big business indeed.

Who buys personal computers?

What sort of people are prepared to spend £100, £500, £1,000 — often even more — on building up a computer system of their own at home or in their small firm?

We find that many people with technical know-how are among the converts of personal computing — engineers, programmers, do-it-yourselfers.

But then there are the Smiths, Joneses and Robinsons of Britain; quite an army of dentists, car mechanics, young people, pensioners, bus-drivers, lawyers, businessmen, teachers, restaurant owners and landlords are buying their own computer systems.

Why do they do it?

It's probably for one of these three reasons:

1 They have one specific application ready for the computer (quite often in connection with their job).

2 They have a general interest in technology and want to learn more about computers.

3 They feel they should know something about this new technology in general.

We can also divide the "personal computer people" into these four categories:

1 The hobby enthusiast pure and simple, who prefers to put his — or her — own computer together. Mainly because it's cheaper that way, but also for the joy of doing the whole job alone.

2 The person with a technological interest who buys a computer kit. The price is not generally the most important question: the most important thing is that the computer can be used for something that's fun and interesting and that the price is more or less reasonable for what may be a life-long friend.

3 The professional person with his eye on the future who wants to use the computer in his work; the writer who wants to build up a text processing system; the architect who wants to rationalize the calculations for his tenders; the businessman who wants a cheap accounting system; or the parent whose children are using computers at school and wants to either learn himself, or provide a similar facility in the home. For this category of people it is of decisive importance that the computer system can do a good job.

4 Schools, institutions and companies with limited budgets have started to discover the personal computer as an answer to their needs that up to now would have been too costly.

Developments in the next few years

What is going to happen over the next 3 or 4 years? What applications will be possible then?

Games will be developed further and will become more sophisticated. The educational side of computing will certainly expand. A lot of new software will appear, making it possible to sit at the computer and learn various things; language translation, foreign language pronunciation; or even to run things in the house — home security for example.

Although very many programs are already available, the aim will be more sophisticated applications. Word processing and the control of various functions in the home such as radio/TV, telephone, air conditioning and so forth will gain popularity. In the late 1980s the total domestic information and data processing system, something which we will come to find just as natural as today's electric cooker, washing machine and colour TV, will become a reality. Total systems of this type will provide us with a number of services — all of them in the hope that the quality of life for the people of the 1980s will be enhanced. The system will help us to co-ordinate the functions of all the various items of technical equipment in the home. It will assist us in many errands and serve as a memo pad.

The new developments available through commercial media, such as viewdata, or Prestel, will allow us to communicate outside of the home. We can already order goods using the telephone and television. Our personal computer is the key to communicating with the bank or the grocer.

It will entertain us with an inexhaustible supply of intelligent games. It will give us and our children individualized teaching. It will partly replace the postman by functioning as a telephone-based communication system for information transfer to friends and firms. When we are away we can ring up the system and alter various functions in the home by telephone. The home gains an advanced "secretary" who is really good at text editing.

Today's personal computer, is the introduction to all this. The electronics, the hardware, is already there. What remains is to develop the software, to put it all together to form an attractive package — and to convince consumers of the necessity!

CHAPTER 2

The computer — what is it and how does it work?

The computer — confusingly versatile

From the very beginning computers, or computing machines as they used to be called, have had a strange kind of aura, almost as if they possessed supernatural powers.

Well, a computer possesses no such powers.

The computer is just like any other machine, and, just as we are surrounded by masses of different machines we are also surrounded by many different computers. Computers of different capacities, dimensions, appearance — computers with different purposes and names and at different prices. Some are small enough to be held in the palm of the hand, others may need floor space of 100 square metres.

Just like a human being, who in the course of one day can do many entirely different jobs, such as being a filling-station attendant, a parent, an odd-job person, a car driver and a stamp collector, so the computer can change from one specifically distinct job to another.

This makes computers such useful working tools. This very versatility however — together with the fact that computers take so many different forms — probably makes a confusing and obscure impression on anyone starting to approach the computer world. A particular characteristic of a certain computer that you might appreciate and exploit may be quite unknown to another user of the same computer, who may be quite satisfied in his own way.

It is by taking a closer look at the most marked characteristics of computers as a whole that we can gain a sensible overall picture. To use our TV or car, we do not need detailed knowledge of how the channel sector or gearbox works; and neither do we need to scrutinize every little detail of a computer to be able to understand and make use of computer technology.

Take your relationship to a bicycle, for instance: you know what it can be used **for**. You know **how** to use it and you know its limitations.

You can have the same attitude to computers. If you know a little about the general construction of a computer you will also know the range of applications, and how to exploit these possibilities. You will soon learn to see the limitations of the computer and computers do have limitations!

Big computers, or **mainframes**, are found in large firms, institutions, computer centres and government offices, where they are used for a number of different tasks: financial accounts, production planning and follow-up, personnel registers and much else. Since their working tasks are very large and constantly changing, these large computers are characterized by speed, large capacity and flexibility. Often large computers are used in systems where many people need access to information at the same time; this is where the power of a large computer to hold many tasks together really comes into its own.

Somewhat smaller, **minicomputers** are not capable of quite such general use. With a minicomputer it is possible to solve data processing problems more effectively because the system can be built up in a more specialized way.

The user has greater ease of access to his minicomputer installation. Costs do not approach the hundreds of thousands for large computers: one of the more simple minicomputer systems will cost around £5,000 or more. This means that many firms and institutions can afford to invest in minicomputers, which are used for everything from accounting to industrial automation.

Computer power, however, is needed in many different places. The full power of the minicomputer is not always needed. Something cheaper and simpler is often adequate.

So we have the **microcomputer**, the smallest member of the computer family. It is small enough and cheap enough to be built into machines and plant to carry out specific and limited tasks — which are often more complex than one thinks. The heart of a microcomputer is the **microprocessor**. Some special microprocessors today cost no more than £3 to large consumers.

It is possible to extend the microcomputer and equip it with a sufficiently large memory and suitable peripheral equipment. It then approaches the functions and range of applications of a minicomputer. This is true of most of the units sold for personal use. A small system can be extended, and it is therefore possible to have the power of a minicomputer available with a personal microcomputer at its centre.

We all know how to use a washing machine: we put in dirty clothes, the machine washes for a while, and then we take out clean clothes.

It's the same with a computer. We feed in data — figures, symbols or words — the computer processes what has been fed in, and then we are given a result in the form of new figures, symbols or words.

Just as the washing machine has a program to follow during the wash, the computer has instructions on how to treat the data that has been fed in.

Even if the computer can't actually **think** in the same way as man, it may be interesting to make the following comparison with how we function (at least, how we function at times!)

We receive information at a certain moment (by feeling, hearing, seeing, smelling or tasting).

We perceive the information, ie. we memorize it.

We take out other information from another place in our memory.

By means of logical thinking — in this case by comparing the different information — we arrive at a suitable decision.

We act on the basis of the decision (by using some suitable "output organ" and speaking, walking, using our hands, etc).

The computer works in a similar manner. It receives its information via an input unit. The computer stores the data it receives in its memory. Thus far, the analogy with the human brain holds good. The computer memory, however, only retains the data it has been programmed to store. Everything else is "forgotten". This saves storage space and it makes it quicker for the computer to search through what it has in store. (On the other hand, the computer lacks the intuition and imagination of the human brain.)

When the computer needs certain data in its work, the data is brought out of store on command.

Processing of data is often similar to our way of thinking logically. One part of this process is to compare two items of information. It is often necessary to compare the item of information being fed in with one that is stored in the computer memory. The item of information already in the memory may itself have come in as "new information" and been stored. Or it may have been given to the computer as part of the working instructions — the program.

The computer may be programmed to compare the new information with information in the memory already so that it can determine whether the information is satisfactory or not, or to divide the information fed into groups and so forth.

The computer could even make a comparison between incoming data and a certain item of information in the memory to see if they are identical — in which case the computer has instructions to proceed in a certain manner; or it may be instructed, when a new item of information is fed in, to search through — and compare with — a number of different data in the memory.

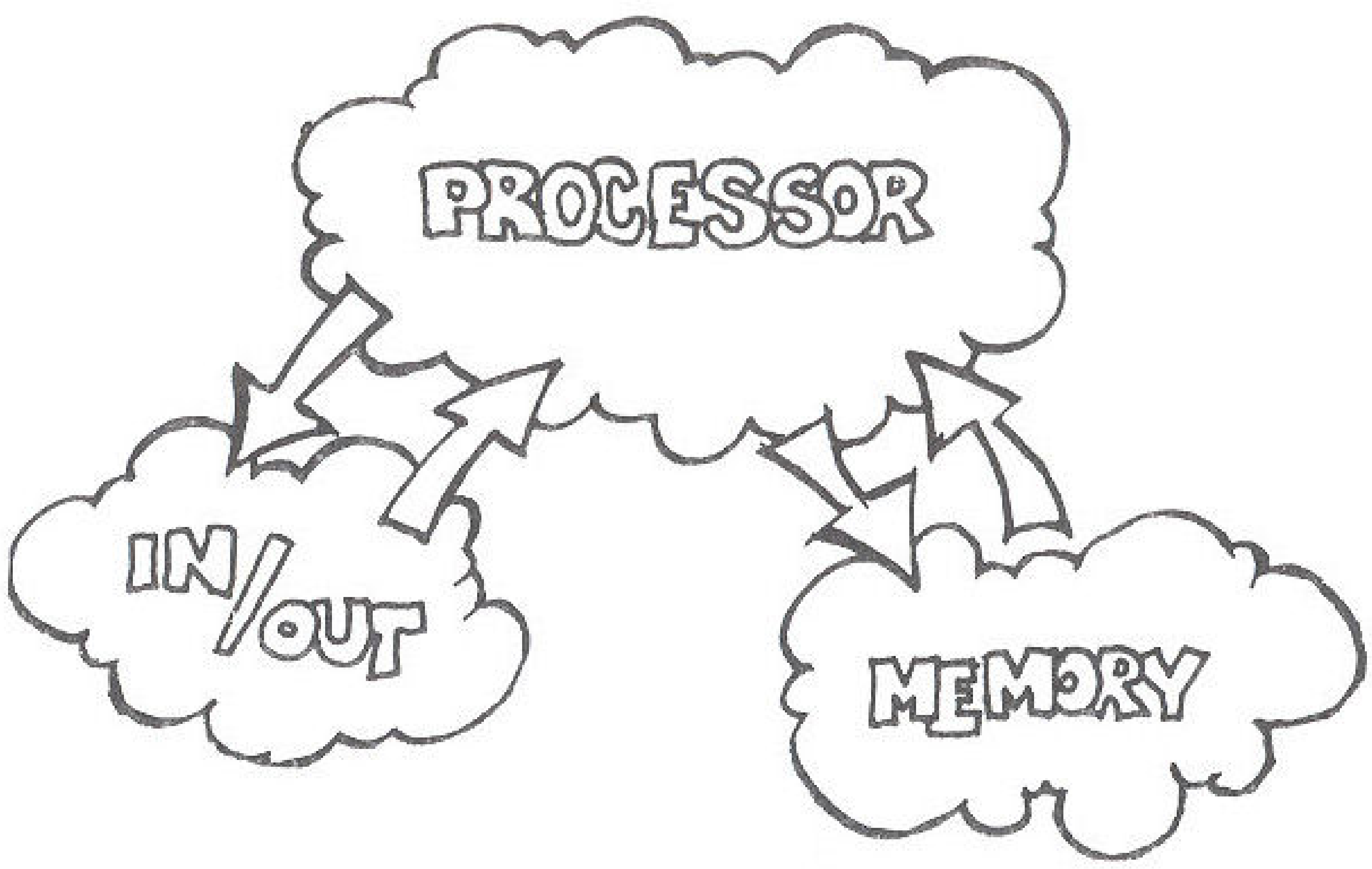
The result — in the form prescribed by the program — is delivered to the output unit of the computer.

How the computer works, in simple terms

Two concepts often used when talking about computers are **hardware** and **software**.

Hardware is everything in the computer system that you can in theory pick up and handle — all physically existing items such as electronic connections, wiring, tape recorders, boxes, keyboards and integrated circuits.

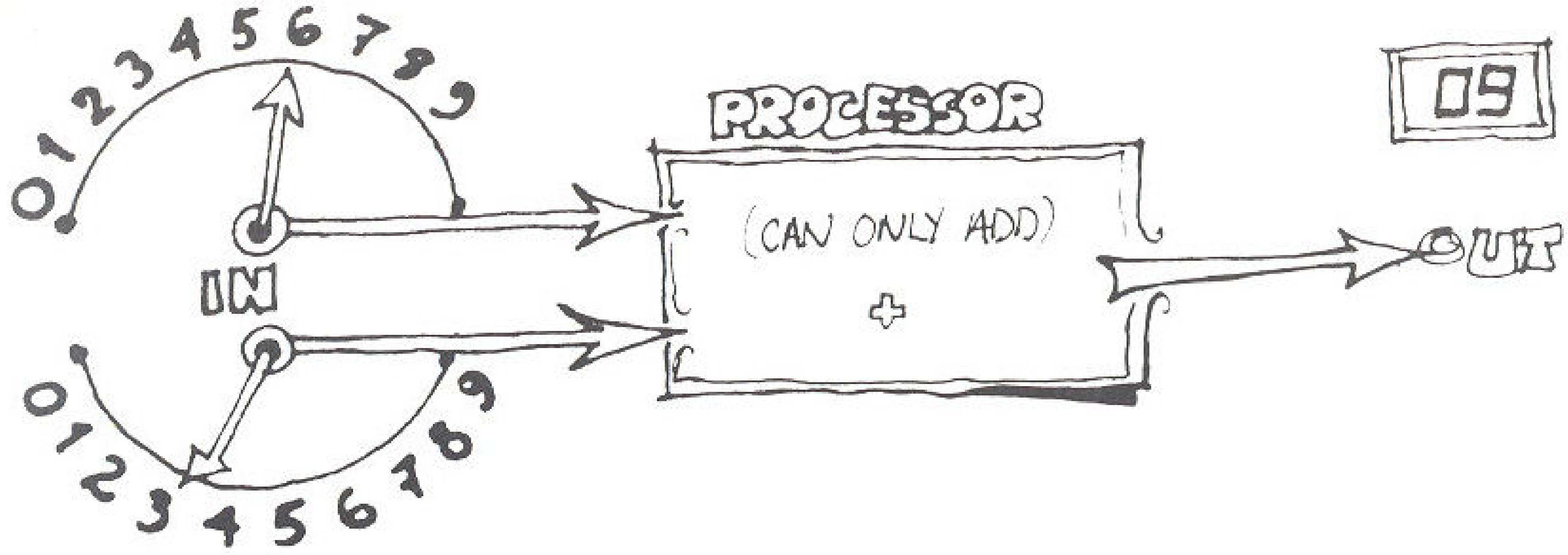
Software is the overall term for programs, communication languages, processing procedures, data and stored information.



From what has been said so far, it can be seen that the hardware of a computer system consists of three basic parts: **a processing part** (the central unit, CPU, processor, controller — there are many words for the same thing), **an input and output part** (peripheral equipment or I/O equipment), and a **memory** or **store** of some kind.

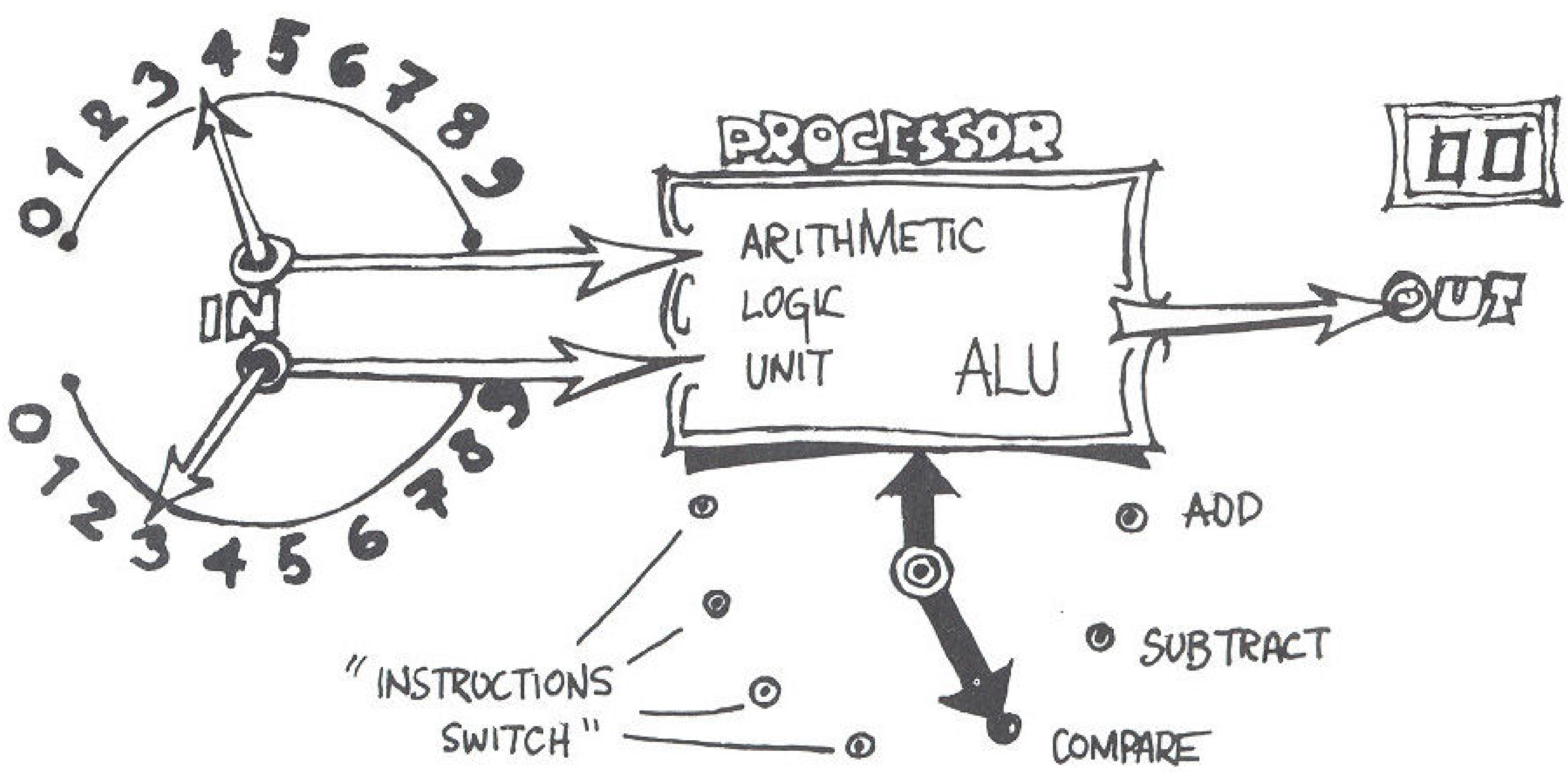
The software — the program — is a list of **instructions** to the central processing unit or CPU.

Let's look at a highly simplified form of computer — an arrangement consisting of a very simple input unit, an equally simple output unit, and a central processing unit.



This certainly is a very limited "computer": it can only work with a few digits (0-9), and can do nothing more than add two digits together. In fact this kind of machine isn't much use to anybody.

Let's extend the possibilities a little ... we can replace the "adding processor" by an arrangement that can not only add but also subtract, compare and carry out other logical operations. We call this an **Arithmetic Logic Unit** or **ALU**.

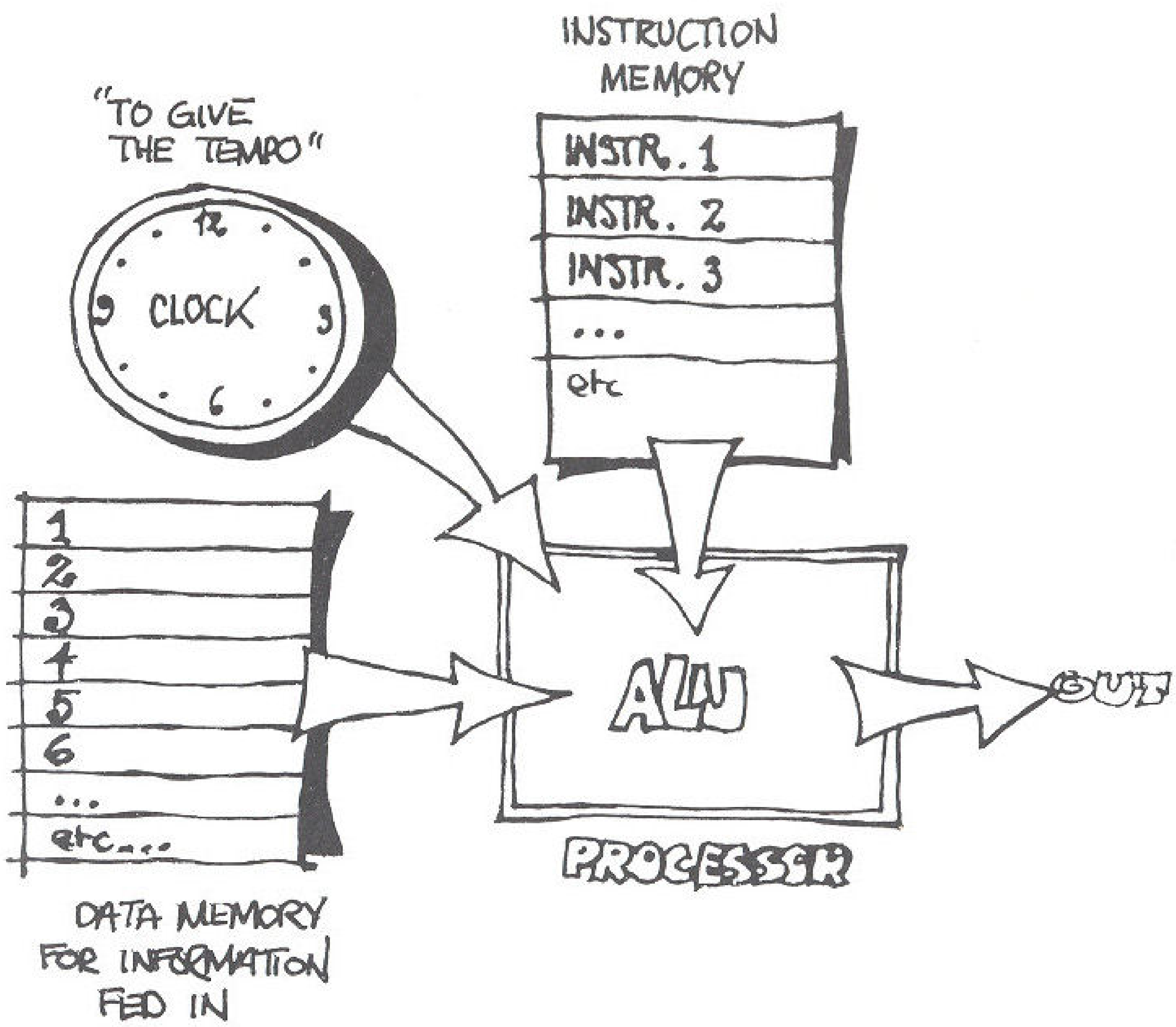


Our extended system has now been given a further switch, an instruction switch. We can use this to decide if the ALU is to add, subtract, etc.

Even though the range of applications in this version is greater, it is still not sufficient for the "computer" to have any practical value. Our simple processor can't even work with more than two numbers at once. If we wanted to add several figures together we'd come to a halt.

So we add a unit where the figures can be stored — a memory. This may be a number of switches, a tape recorder, a disk or electronic devices that can store information.

Now we replace the instruction switch by an instruction memory or store. Here we can feed in a number of instructions intended for the processor.



And we add one more thing: we put in a clock, (a sort of metronome) which will help the computer through the program step by step and instruction by instruction.

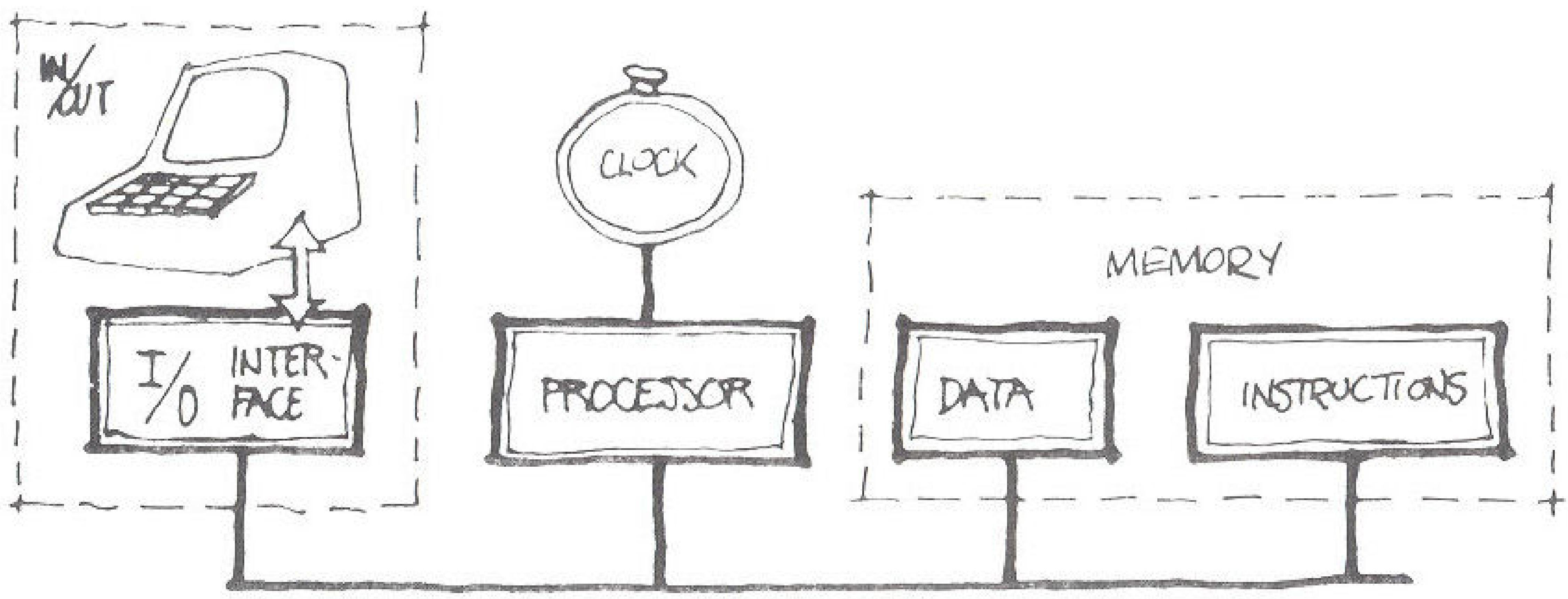
We can now program this system to make the processor carry out the following instructions, for example, one by one:

Instruction 1: Add the first two figures in the store.
Instruction 2: Add the next two figures.

Instruction 3: Compare the two totals.
Instruction 4: Feed out the least total.

There are no essential differences between the basic logical and functional construction of a real computer and this simplified one. In practice, the ALU has the capacity to carry out further operations. There are additional devices to simplify the flow of information and instructions within the computer to facilitate input and output.

The block diagram for a computer looks like this:

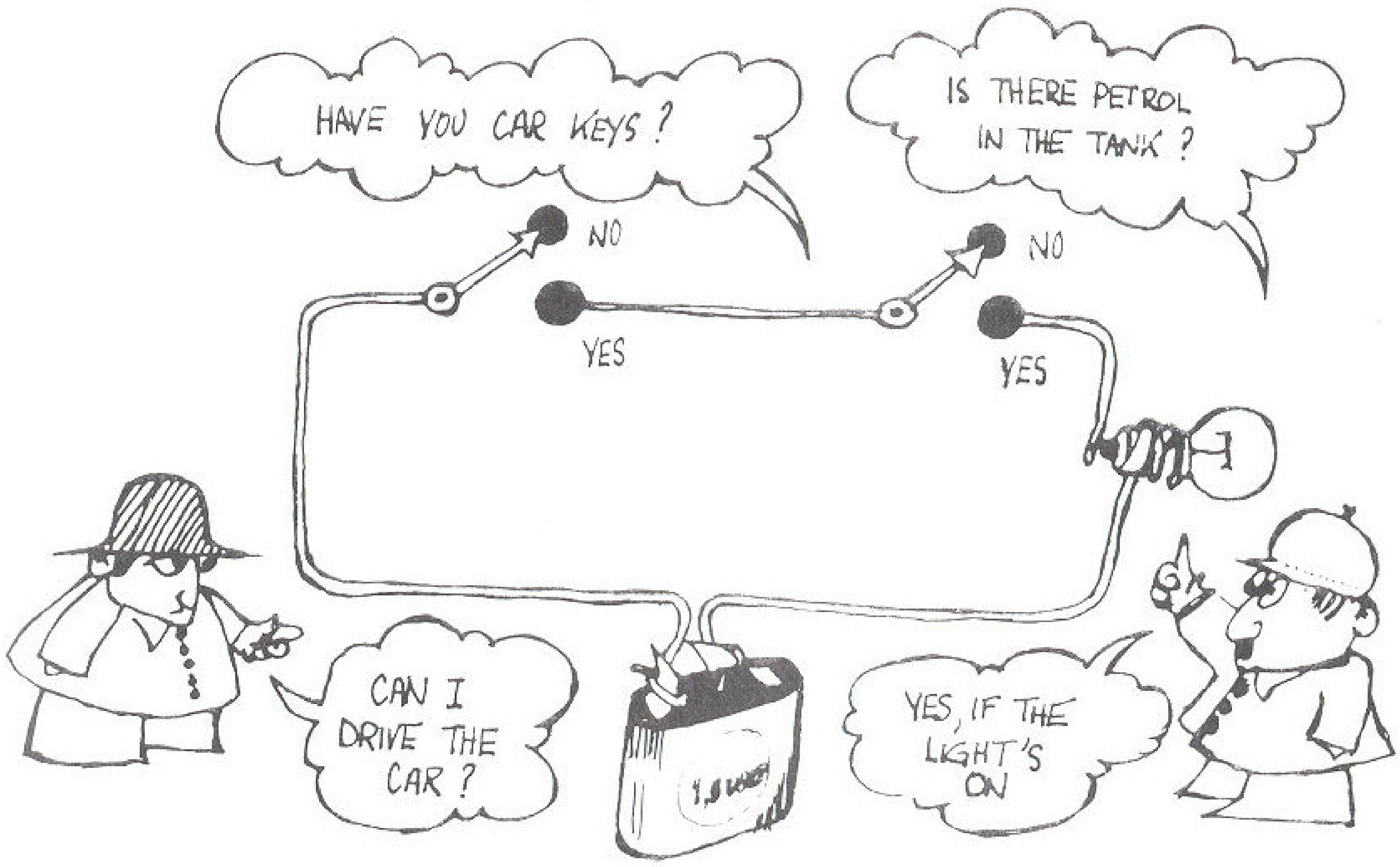


In a microcomputer, each of these blocks could be an integrated circuit. Microprocessors are now beginning to appear with all these blocks enclosed in a single integrated circuit. In minicomputers and larger computers each block is realised in the form of individual circuit boards or modules. Since computers work by electricity there is also a unit for power supply!

A computer is built up of hundreds of thousands of electronic circuits which "think" logically.

Here is an example of a logical electronic circuit — not from a real computer of course — but it will do well enough to explain something on the secret of **how** a computer can calculate, carry out logical operations and so forth.

To drive a car, you need both access to the keys and some petrol in the tank — it won't work with only one of these things. The following illustration is a diagram where all the decisions involved have been broken down into their basic, essential parts.



We can make up a table — a "truth table" — to illustrate the problem and how we can use the above diagram to determine whether it is possible to drive the car.

Keys?	*Petrol?*	*Light?*		*Drive the car*
No	No	Off	=	No
No	Yes	Off	=	No
Yes	No	Off	=	No
Yes	Yes	On	=	Yes

The truth table may also look like this:

A	*B*	*Q*
0	0	0
0	1	0
1	0	0
1	1	1

A and B stand for data in and Q for the result, or data out. Or in our case A is the condition of the keys and B the condition of the petrol. Q is the condition of the light and indicates whether the car can be driven.

The positions of the switches — "on" or "off" — we have called "yes" or "no", "0" or "1". We can also call them "true/false", "right/wrong", "sun/rain", "high/low" etc. This sort of diagram can be adapted to other purposes.

A computer contains many logical circuits making decisions of this type, in various forms, and this arrangement allows the computer to work in series of logical operations.

Binary numbers

Let's plump for the alternatives "0" and "1" — these concepts are used very often in computer work. You have probably already heard how computers only count in zeros and ones. We speak of the **binary number system**. It works to the base 2 (only two digits). The **decimal number system** we normally use works to the base 10 (ten digits).

With the binary number system we can simply express the position of the switches in our circuit. "01" means that we have petrol but no keys, "11" that we have both petrol and keys. From the truth table we see that there are four different alternatives: 00, 01, 10, 11.

In each one of these alternatives there are two pieces or bits of information. In information theory — and in computer work — the concept of **bit** is used for the quantity of information needed to answer a question to which there are two possible alternative answers. "Bit" is an abbreviation of **binary digit**.

These bits are very often collected in groups of eight, eg. 00101101.

This group is called a **byte** (pronounced "bite").

Every computer works with a definite number of bits. Most small, personal computers work with data and instructions in eight bit, or byte lengths. Some simpler microprocessors use only four bits or nibbles! Large computers use twelve, sixteen, twenty four or even thirty two bits as their working length. The working length for a computer is usually called its word length, and this can vary from machine to machine.

A working knowledge of binary arithmetic is a useful, but not necessary skill in using computers. If you are interested in refreshing your memory, or starting from scratch, then look at Appendix A for some elementary binary arithmetic.

Most of the small computers that we shall be looking at use eight bits, as mentioned earlier. One problem, however, is that although these machines think in eight bit lengths, they actually remember where the information is stored in sixteen bit numbers! It becomes useful to know something about working to base 16 or hexadecimal. This is also described in Appendix A.

How does the computer handle binary numbers?

It's this kind of information — 00111101 etc. — which is read into the computer memory and thus forms the computer's working material.

How is the memory or store constructed to be able to handle all these zeroes and ones?

The store is organized in matrix form somewhat like the pigeon holes used for internal mail: a number of small compartments with room for a certain amount of information. Each compartment has its own address.

Each address in the data usually accommodates a whole byte, ie. eight bits. The memory cell consists of a small space where it is possible to store electronic charges: charge = "1", and no-charge = "0".

In order to make it possible in practice to get a sufficiently large number of store addresses to work with, most computers (microcomputers) are designed for the address to be 16 bits long, eg. 10001111 00011011.

With binary numbers of this length one can express 65,536 different addresses (2^{16}).

In the general block diagram for a computer which we drew earlier, there are two types of store, the data store and the instruction store.

As the computer only stores 1's and 0's it is usually possible to mix up the instructions and data, with the instructions taking two bytes and the data, one. It is good practice to "partition" the memory space so that, say, those memory addresses below 0200_{16} are used for data, and those above for instructions. In most computers, however, it is useful to have some instructions, or even programs, stored permanently. So if the machine gets switched off it will perform some basic tasks when it is powered up again. The store used in this case must therefore be relatively permanent and, to protect the information stored, fairly difficult if not impossible to erase. This type of storage is called read only memory or ROM. This means that it is only possible to read ie. fetch data or instructions. They are permanently stored and cannot be changed. Another word for this sort of storage is "non-volatile" — meaning that it is not destroyed when power is taken away.

The other type of storage used in computers is called random access memory — or RAM. This means that one can enter any memory address and store or fetch information from it. It is usually volatile ie. it loses all information stored when power is switched off.

Consequently, all the instructions that keep the computer going; telling the clock when to send out certain pulses; when to move data from the memory to ALU or even how to output answers; these instructions are kept in ROM and are usually called the monitor. In most small computer systems the language used to program is also kept in ROM eg. BASIC. Some peripheral hardware also contains ROM — printers and discs for example — so that the information needed to get the electrical pulses in the right format is available all the time.

In the same block diagram there is also something called an **I/O interface.** This is sometimes called a **port**, or an **I/O port.**

The input unit (I) receives data from the peripheral equipment — in this case a keyboard — and adapts this information so that it can be understood and used by the computer. The output unit of the interface (O) takes care of data that the computer wants to read out.

This data is adapted so that the exterior output unit — in the diagram a display — can understand and use the information.

There are basically two different types of I/O — **serial** and **parallel.**

Serial I/O works lengthwise with data, ie. the bits are transmitted back and forth by byte, one by one — rather like a camel caravan.

Parallel I/O works breadthwise, for instance eight bits at once — rather like a platoon of soldiers on the march. Parallel transmission works considerably faster than serial transmission.

A serial interface contains special circuits which determine the speed of transmission, this is called the **baud rate.** This is a measure of the number of bits transmitted per second.

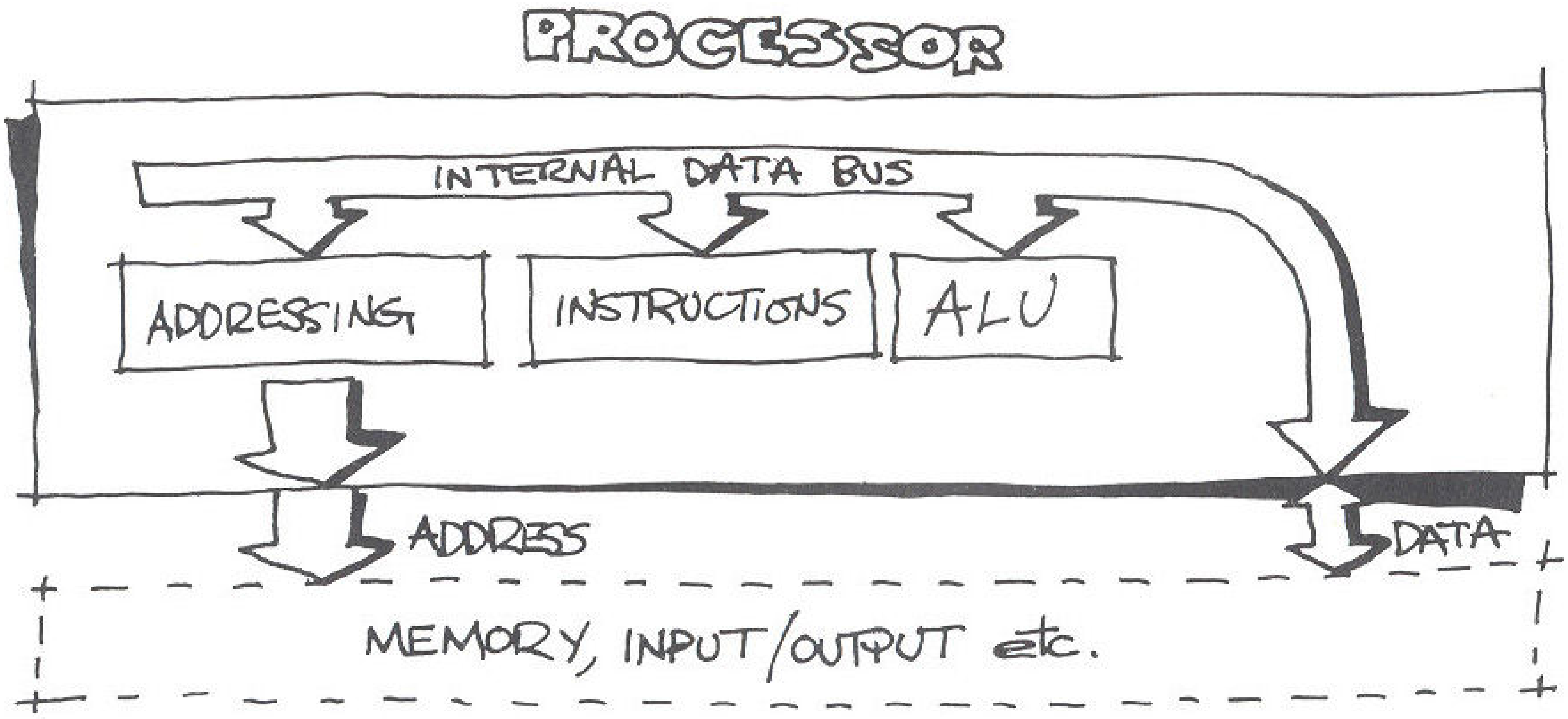
The processor — the CPU — from the inside

Though it doesn't look anything special and doesn't cost very much, the processor is the part which determines what the computer is capable of. Generally the processor in a microcomputer is built as a very complex semiconductor circuit. It contains a large number of logic circuits.

The processor has many important functions to perform in the computer. These include the following:

1 Fetching and "understanding" instructions.
2 Transmitting data to and from the store and the I/O units.
3 Performing arithmetic and logic operations.
4 Controlling other parts of the computer.

Basically, the processor can be divided into four functional sections. The first of these — we may call it the **addressing section** — works with the addresses that are needed at any given moment (for fetching or output of data, ie. reading or printing).



The function of the second section, the **instruction section**, is to translate and "understand" the instructions the computer is to work with.

The third section is an old friend — **ALU**, the arithmetic and logic unit. This is where all the computing and "thinking" goes on.

All these sections in the processor are connected to each other by a complex network called the **internal data bus**, which forms the fourth functional section.

Every one of these sections in the inside of the processor contains one or more **registers**. These are small memory sections used for the temporary storage of instructions, intermittent results and so forth. As a rule these registers are only large enough for one or two bytes (8 or 16 bits).

How the processor carries out the instructions

An instruction is the smallest part of a program that has the power to transmit a complete message. These instructions often resemble commands in normal language. The command "Add B to A", for example, may be written "ADD B to A".

For it to be stored in the computer memory, however, and to be handled in other ways by all the electronic circuits in the computer, the instruction must be written in binary digits: the ADD instruction, for instance could be 10000111. This method of communicating is called **machine language**. The processor understands **only** machine code, ie. 1's and 0's.

Each instruction carried out by the processor represents a small step in the computation or process that is programmed into the computer.

There are a number of steps necessary in carrying out each instruction. These are done one by one, not all in one go.

The instruction cycle

As the program has been stored in the memory, each instruction must first be brought from the memory and placed in one of the registers in the processor. The instruction is translated — ie. the processor finds out the meaning of the binary code — and the operation prescribed by the instruction is carried out.

The total time required for fetching and carrying out an instruction is called the **instruction cycle**. The length varies according to the task performed. For most microprocessors it is in the order of millionths of a second, or a microsecond.

There are thus three steps:
Fetching the instruction from the memory.
Translating the instruction.
Performing the prescribed operation.

The program counter

To make it possible to fetch an instruction from the memory, the address where the instruction is stored must be known. This is kept in a register in the processor which is called the **program counter**. The space available in the program counter exactly fits the length of the address — 16 bits.

The instruction register

When the program counter has selected a byte (the instruction or part of it) in the memory, it is sent to the processor and temporarily placed in a register called the **instruction register**.

From this register, the instruction goes to the instruction decoder where its meaning is translated into signals which carry out the operation prescribed.

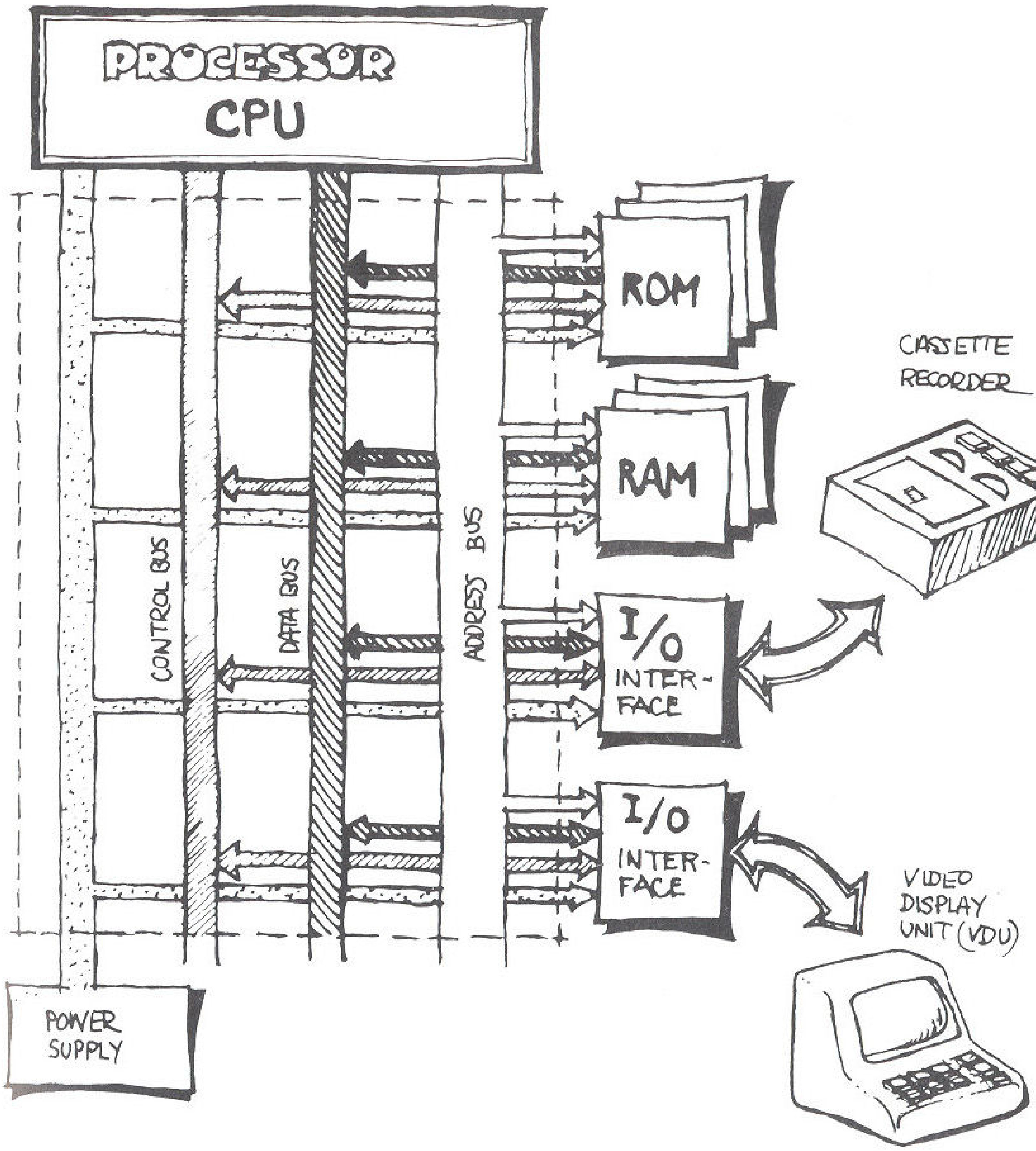
The accumulator

The program counter is one of the registers in the addressing section. The instruction register is in the instruction section. In the arithmetic and logic section we find a register called the **accumulator**. This is a register of eight bits which temporarily stores results while the arithmetic and logical instructions are being carried out.

The interrupt function

One important function in modern microprocessors is called **interrupt**. This function might be described as a "temporary break in the work which the microprocessor is busy on".

An interrupt functions like this: when the processor is working its way through a certain program, an item of peripheral equipment — the keyboard, for instance — may start calling the attention of the processor. The processor makes a note of the point in the program where it is forced to break off, receives data from the keyboard, and then returns to the original task.



The computer's own road network — the bus

The system of electrical circuitry connecting the processor, the memory or store, the I/O interface and the power supply unit is called the **bus** system.

The bus in the computer is divided into four separate "main roads":

the power supply to all the blocks of the computer,
the control bus along which all control signals flow,
the two-way data bus which sends eight bits at a time between

the processor and other blocks, and
the address bus directed from the processor which handles address signals 16 bits in length.

To make a simple, efficient and clear design of the computer possible, the bus is usually constructed in the form of a number of parallel conductors. These are on the printed circuit board on which the computer is constructed.

Sometimes the bus takes the form of a printed circuit "back plane", or "mother board", to which modules containing various parts of the computer can then be attached. The memory, processor, input/output circuitry, can be on separate boards and the user can configure his computer as he wishes.

The most common form of this type of bus is called the S-100 bus, and has the distinction of being not only the first hobby computer bus, but also the first to have its own international standard. The S-100 bus has 100 conductors and details of the pin connections and the international standard are given in Appendix B.

Other buses are the SS-50 (50 lines), SS-44, SS-48 .... the list is endless. In Chapter 5, where actual real computers available over shop counters are discussed, the bus system used will be mentioned in each case.

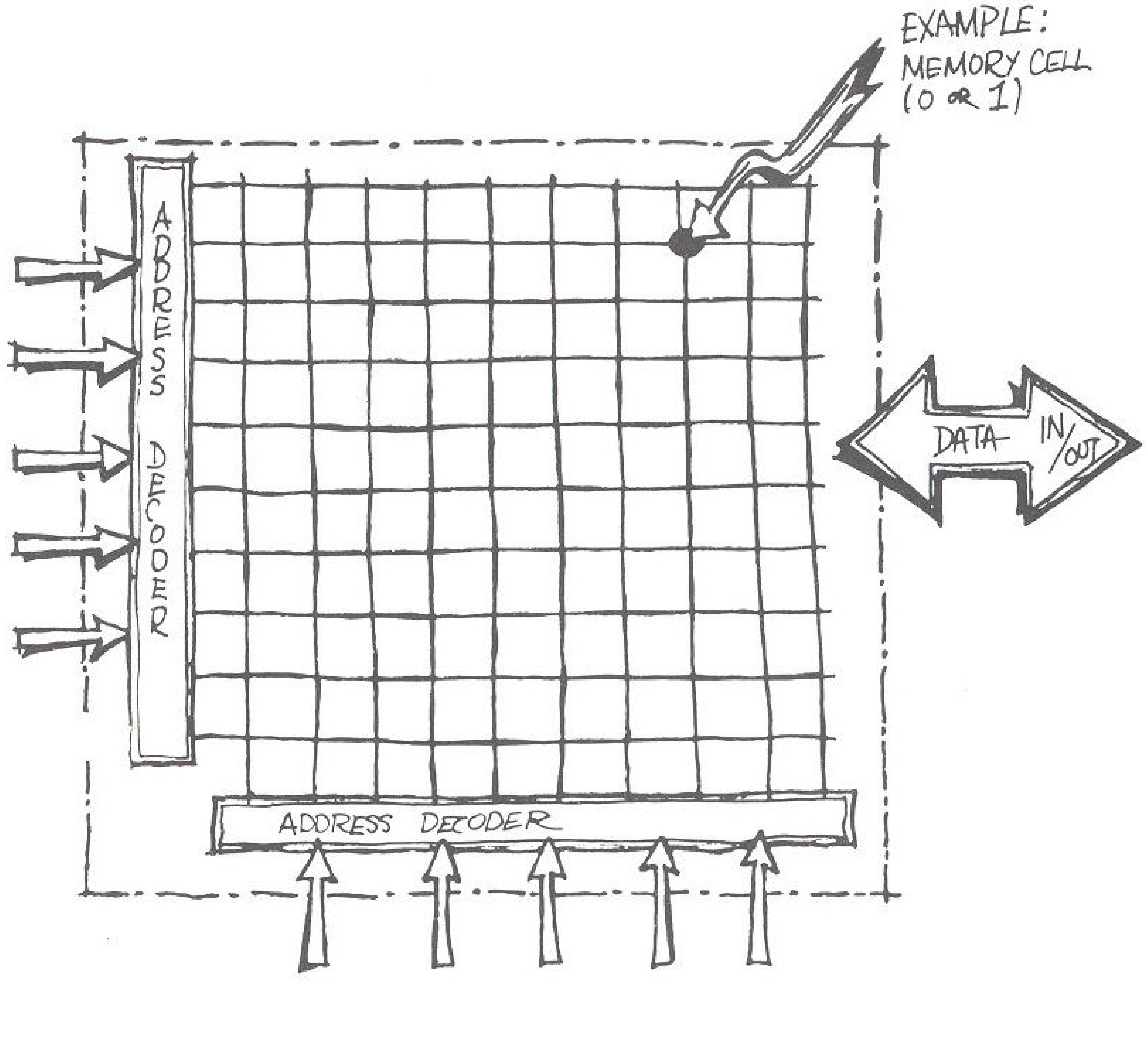
Not all boards, or systems, that claim to follow any one bus standard are interchangeable with boards or systems from other manufacturers who claim the same thing! This unfortunate situation has prompted many manufacturers to "invent" their own buses — which is sad but inevitable. The new international standards may overcome this.

Storage inside the computer

It is essential that the computer has enough memory to carry out its task of storing programs and data. Memory can be internal, or external. We will consider internal memory here, and external memory in a later chapter.

The internal memory is of a cellular nature and is usually controlled directly by the processor. It should be possible to retrieve discrete items of data from a particular address. With external memory, data is usually retrieved in blocks. The internal memory has a physical limitation in size determined by the size of address that the processor can handle, ie. 64K for an 8 bit processor using 16 bit addressing. The limitations for external memory are determined by the method of storage used.

Internal memory is invariably semiconductor storage. They use the same method of construction as microprocessors and similar large scale integrated circuits, and are thus compatible as far as voltage goes.



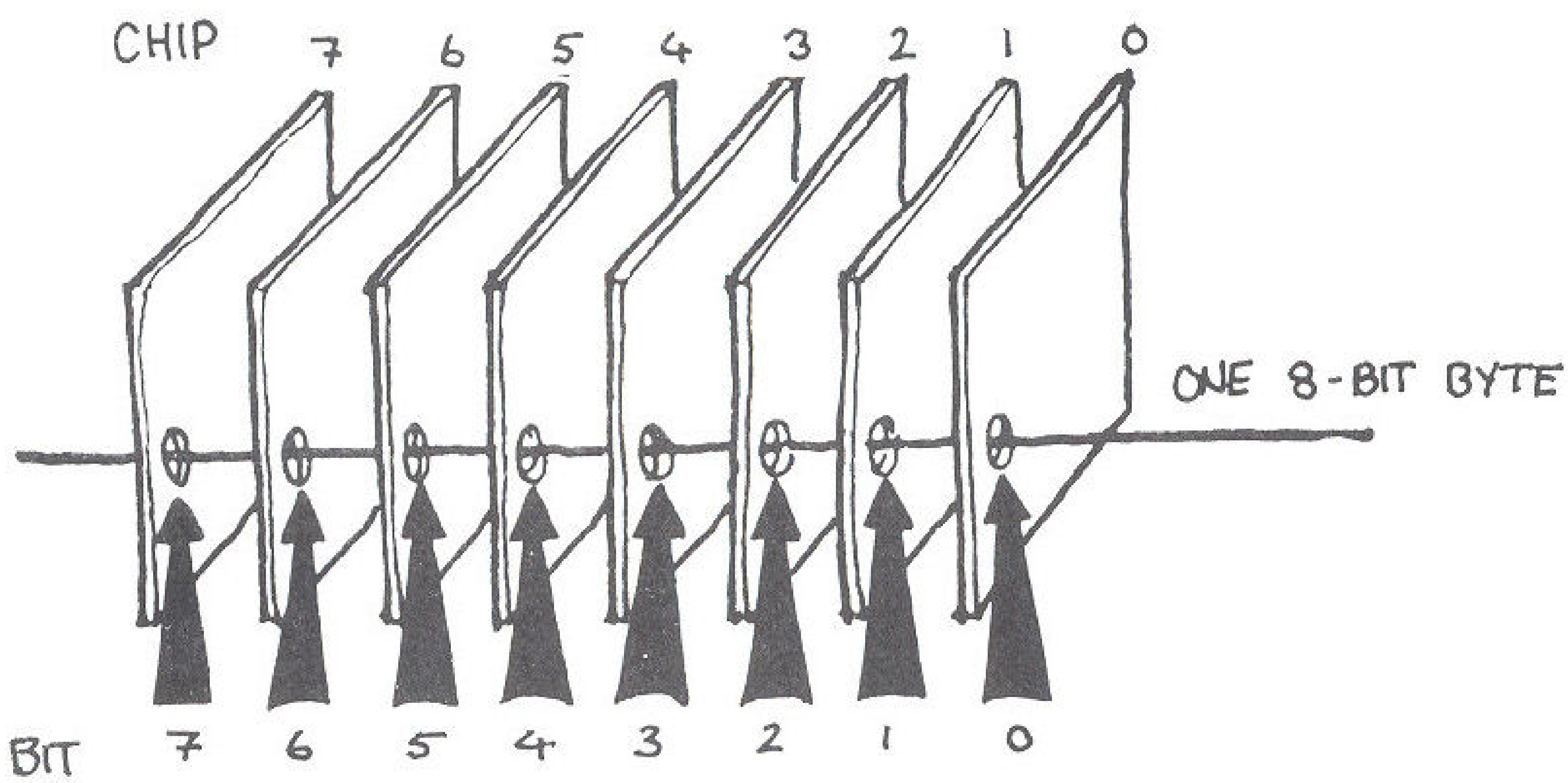
BASIC DIAGRAM OF THE CONTENTS OF A MEMORY CHIP: BY PLACING BINARY SIGNALS AT THE ADDRESS ENTRIES, IT IS POSSIBLE TO SELECT ONE SPEZIFIC MEMORY CELL IN THE MATRIX – EITHER FOR INPUT OR OUTPUT OF DATA (ONLY ONE BIT).

The memory in a normal semiconductor memory is arranged in a matrix or grid.

As mentioned previously semiconductor memories come in two types — Random Access Memory, or RAM, and Read Only Memory or ROM. Let's look at RAM first.

Each memory cell can contain a '1' or '0', in the case of semiconductor memory, indicated by a voltage or no voltage at the output of a transistor circuit.

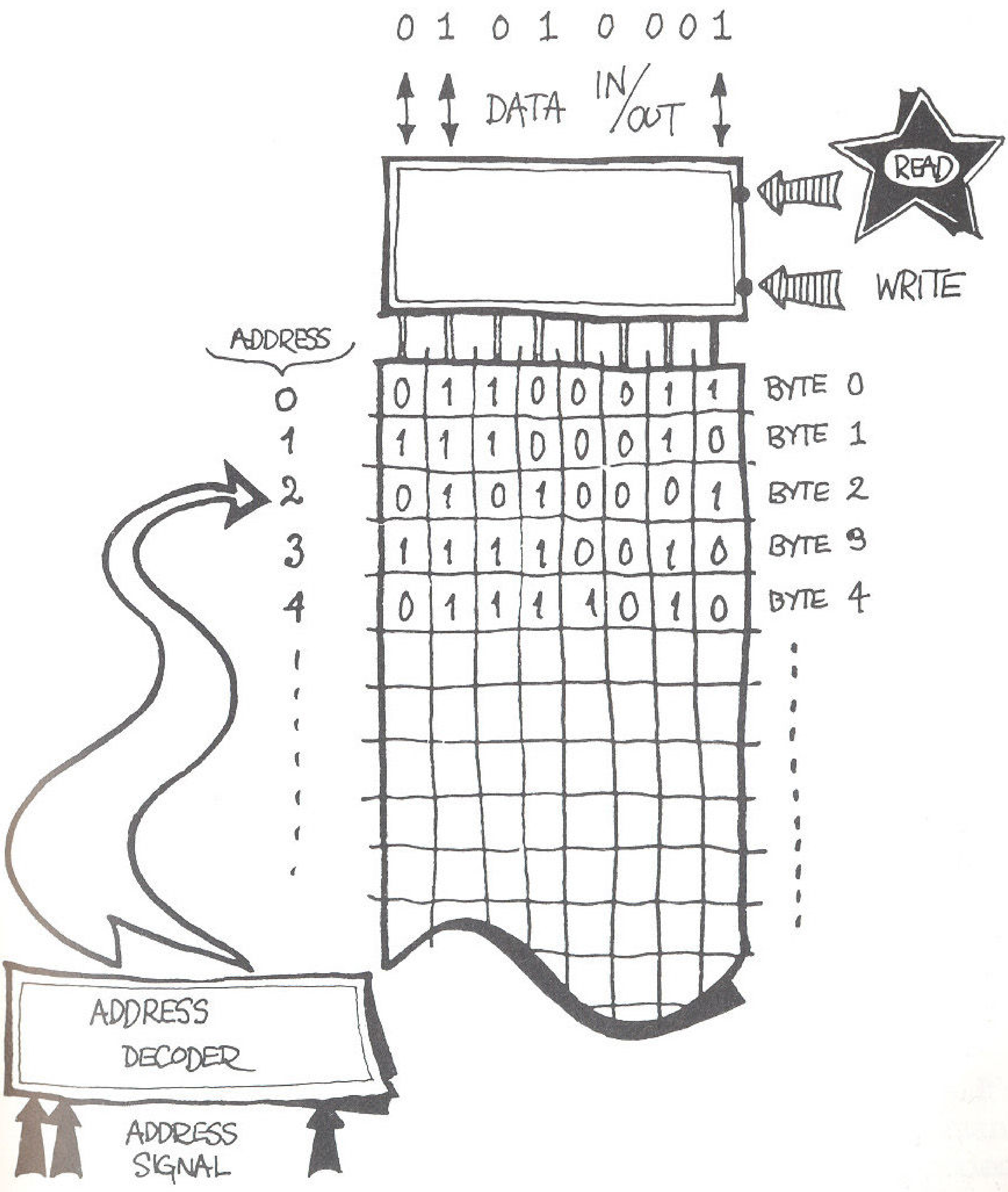
A small memory may contain, say, 32 rows by 32 columns ie. 32 x 32 = 1024 or 1K bits. RAM chips come in many sizes but 1K, 4K, 8K and 16K are the most common.



A MEMORY BLOCK IS COMPOSED OF SEVERAL MEMORY CHIPS.

As most personal computers have an 8 bit, or byte, word length, a number of chips must be interconnected to form a complete block. If 1K chips are used, eight chips would be needed to store 1 Kbytes. For .4 Kbytes, 32 chips are needed. Some smaller memories, usually used in single-board microcomputers may have their memory organised differently. Instead of 1K x 1 bit, they could be organised as 256 words of 4 bits.

There are two types of RAM — static and dynamic. Static RAM will store information until the power supply ceases or the memory contents are changed. Dynamic RAM will do both the above, but needs extra chips to "refresh" the memory constantly, as the transistor cells cannot hold the information for very long. Static RAM is more expensive than dynamic on a chip for chip basis, but there is little to choose when assembled onto a board, as the support chips for dynamic RAM push up the cost. However static RAM needs more power than dynamic RAM and for large memory systems heat dissipation becomes a problem.



With Read Only Memory the above method of splitting up the memory is also needed. The address lines will only cause a memory cell to output data, as it is impossible to write data into a ROM cell.

It is possible to have ROM's which can be changed if the stored information is incorrect, or needs updating. These are called programmable ROM or PROM and come in many different types. Their principle of operating is similar however. A high voltage or charge of energy, in the form of ultraviolet light, erases the contents. It is then possible to place new information into the ROM with a special PROM programmer.

CHAPTER 3

How do I talk to the computer?

The actual mechanics of communicating with your computer are detailed in the next chapter. What we are going to look at here is the "language" that is used in talking to a computer.

We have seen in the previous chapter that a computer works in binary code ie. with 1's and 0's. It is clear that if the person operating the computer had to think in 1's and 0's he would soon go mad! Hex is not so difficult, and some people who absorbed computer science with their mother's milk can do it! However, this is not very often the case. Some other method is needed to overcome the problem.

We could just talk to the computer in normal English. Unfortunately, although computers can be designed to understand hundreds of words of the English language, they can't get the meaning of sentences right. The English language is too full of ambiguities. Think how often two people can be talking to each other for, say, half an hour, and both end up with different ideas on what was being discussed.

Our computer needs clear, unambiguous statements if it is to do the job efficiently.

We have seen that the machine language is difficult to use. It's all right for machine to machine communication but not for man-machine communication. It is possible to group the binary coded words together so that each step in the program can be represented by a word, or mnemonic, describing that operation. This is called **mnemonic machine code**.

The next level of programming is to give the variables names, instead of working with their respective addresses. This is called **assembly language programming**.

Finally, there are the so-called **high-level languages** such as BASIC, FORTRAN and ALGOL. These bear a passing resemblance to everyday language, and the computer translates from these to its machine code.

A working knowledge of a high level language is essential for anyone who wants to take personal computing seriously. Consequently, we will look at the most popular language called BASIC. A brief outline of machine code and assembly programming will also be presented so that some idea of what's happening at the higher level can be appreciated.

It is not necessary for the normal user to bother with anything "below" BASIC. Unfortunately, some people in the computing field like to baffle newcomers with jargon. Anything that can deflate a few egos is always worth it!

Machine language

Machine language is the most primitive programming language from the human point of view, but it is the only language which the computer can really understand. The manufacturer of a particular processor provides the user with a set of instructions. Each instruction relates to the operation required. The instructions may be quite simple: eg. to add the contents of two registers and place them in a third.

A program of this type may look like this:

```
10101101
01000000
00000000
01101101
01000001
00000000
```

10001101
01000010
00000000

It needs a well-trained eye to see what this program is supposed to do: (a simple addition).

The computer must perform considerably more complex operations than this, which makes it extremely inconvenient to program in machine code. It is *time-consuming* and it is *very easy to make mistakes.*

One way of simplifying the machine language is to equip the computer with a small conversion program to translate the binary figures into hexadecimal code. The program example given above will then read like this: AD, 40, 00, 6D, 41, 00, 8D, 42, 00.

This has immediately become somewhat easier to check, but it is still difficult for the untrained person to state that two numbers are being added here.

The structure of the machine language depends to a great extent on the internal design of the processor, and thus varies according to the type of processor. This example applies to a 6502 processor.

The programmer can make life a bit easier for himself by assigning a mnemonic to each hex byte. For example, the instruction "load the contents of memory address xxxx into the accumulator" could be written as LDA XXXX instead of AD XXXX. This type of machine code programming still needs the programmer to know the address location of the data and instructions.

Our program now becomes:

LDA 0040	ie. load accumulator with contents of 0040
ADC 0041	ie. add contents of address 0041 to number in accumulator
STA 0042	ie. store result in 0042

The programmer has to know that the first number is in address 0040, the second in 0041 and that the answer will be found in 0042. An even simpler method is to give the data variable names and let the computer do the rest.

Assembly language

The same mnemonics as previously are used, except the variables are called, for example, NUM 1 and NUM 2.

Our program now looks like this:

```
LDA NUM 1
ADC NUM 2
STA RESULT
```

Before it can be run in the computer, this program must be translated into machine language. When the program is as short as in this example, this work can be done by hand. But it is most often done by the computer. The translation is performed by a program called an *assembler*, usually quite a comprehensive program written in machine code, which in its simplest form consists of a large "table" where the computer looks for the meaning of any given mnemonic in binary code. Each instruction is thus translated on a one-for-one basis into machine language.

The assembly language can be considerably improved by what is known as the macro technique. This means that instead of one-for-one translation there is translation of a small number of assembly instructions into a larger number of machine instructions. This increases the efficiency in programming.

High-level languages

High-level languages are oriented towards the user and his problems rather than to the machine. A high-level language is comparatively easy to learn and relatively simple to read and write.

There are many different types of high-level languages in use. Some of the most common are **ALGOL** (Algorithmic Language, used for scientific problems), **APL** (A Programming Language, originally an IBM language), **BASIC** (Beginner's All-Purpose Symbolic Instruction Code, simple and popular for solving numerical problems), **COBOL** (Common Business Oriented Language), **FORTRAN** (Formula Translator, a popular all-round language), **LISP** (List Processing), **Pascal** (a further development of Algol), **PL/1** (a general combination of Algol, Cobol and

Fortran. Also available for microcomputers, eg. PL/M). *The most popular high-level language for personal computers is definitely BASIC.*

A simple addition in BASIC, for instance, is written on a single line: LET C = A + B (consider the simplicity compared with machine language and assembly language).

Programming in a high-level language is very efficient as far as programming time is concerned. It is normally reckoned to be at least three to five times as quick as assembler programming.

On the other hand, high-level languages make for less efficient use of the computer's speed and storage capacity. Generally, the processor requires $1\frac{1}{2}$ to 3 times as much time to do its work. The storage requirement increases proportionately.

A program written in a high-level language has to be translated into machine code before the computer can understand it. This work is done by a special, quite complicated system program, normally developed by the computer manufacturer. According to how this program works it is called either a **compiler** or an **interpreter**.

The compiler translates what is called the whole BASIC program once and for all into machine code.

The interpreter, on the other hand, translates one instruction at a time into a number of machine instructions. This takes place continuously at the same time as the program is being executed in the computer — the interpreter and the high-level language program are at work at the same time.

Interpreters are very common on the personal computer scene when BASIC is being used. Compared with a compiler, it is more convenient to use, but the drawback is that it is up to ten times slower.

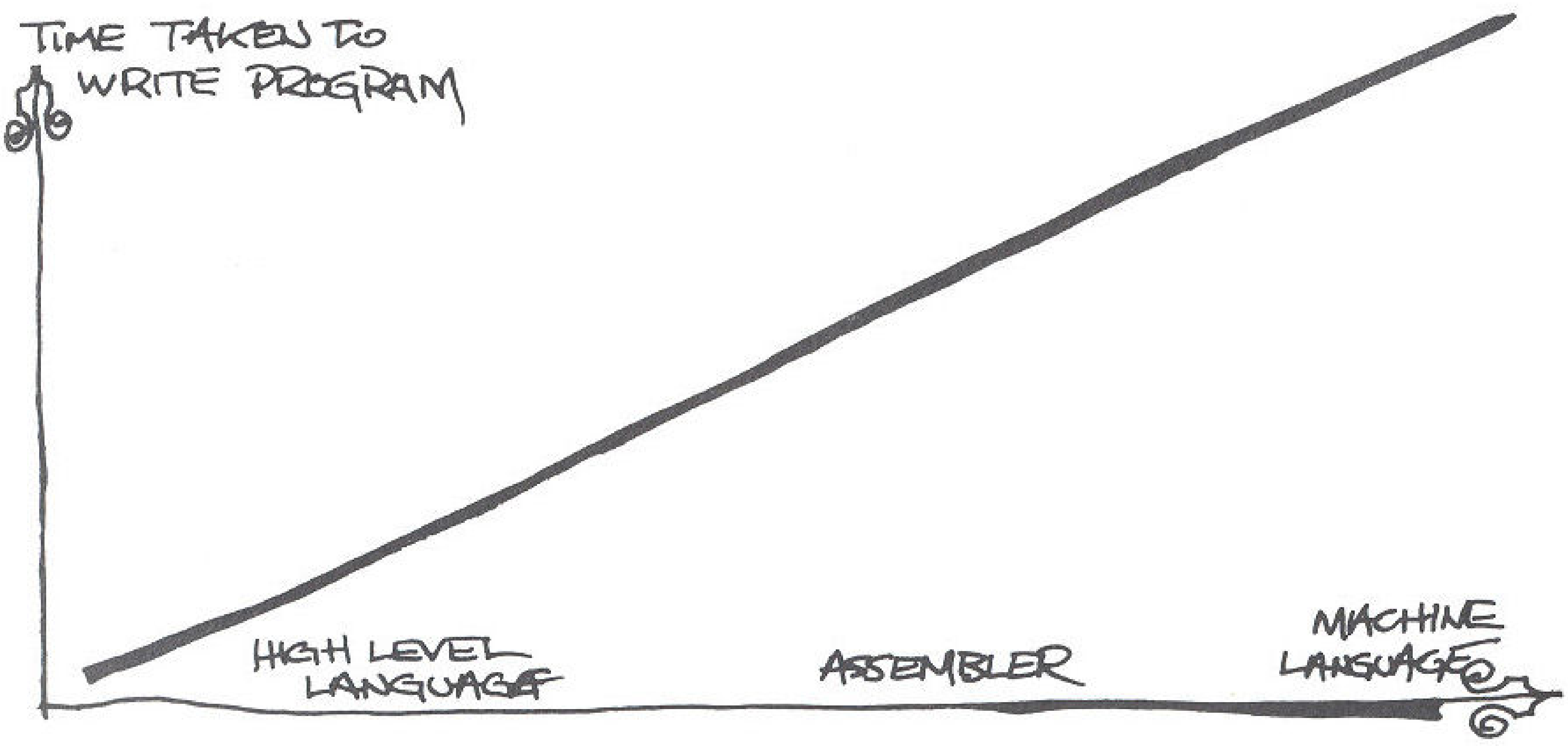
At which level do I begin?

Obviously, the skill of the programmer has to be considered when answering this question. Similarly the task to be solved also needs to be considered. A high level language like BASIC is more efficient in programming time, and is generally considered simpler to use. Other programmers can also understand what has been written which aids "portability". Assembly language and machine code, on the other hand, give the programmer the ability to use the computer to its fullest capacity. Programs are shorter and quicker to run.

The following comments are only guidelines:

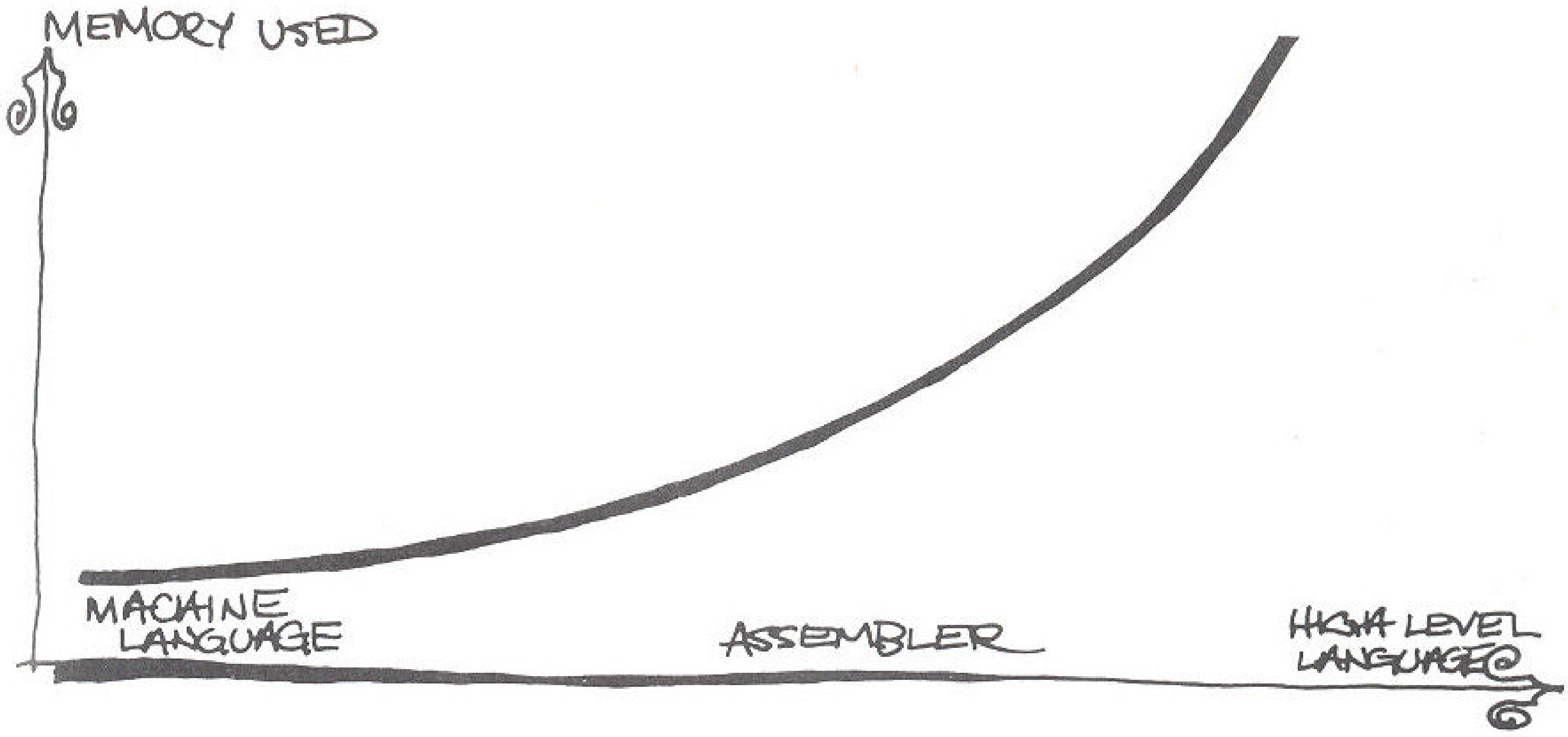
BASIC: Most suitable for interactive applications, where the operator and computer want to "talk" to each other! Simple calculations, games and other general purpose programs — such as training programs — are easier at this level.

Assembly language: More suitable for control of peripherals and input/output routines, real-time applications and programs within the computer system. It also makes maximum use of the small memory capacity of personal computers.



Programming in a high-level language such as BASIC is considerably quicker than in assembly language. Machine coding is very time-consuming

High-level languages are usually the best method for the majority of personal computer owners. Writing most application programs is a quick and easy business. Troubleshooting (debugging) and testing (a not inessential part of the work!) are also relatively simple. A discussion of the relative merits of BASIC and other languages appears later in this chapter.



Programs written in high-level language generally require 50% to 300% greater storage capacity than those written in assembly language or machine code.

One must balance out the programming time and the storage costs from case to case.

BASIC — a convenient language

BASIC means **B**eginner's **A**ll-purpose **S**ymbolic **I**nstruction **C**ode and was originally developed at an American college in the 1960s. It was mainly intended for people who knew nothing about computers but were keen on using them. Since then BASIC has been further developed and become extremely widespread throughout the world. Hundreds of thousands of different computer programs have been written in BASIC. Many of these are available in books, compendiums, etc.

To make the best possible use of your computer it is advisable to learn at least the fundamentals of BASIC. After a few minutes' study it should be possible to write your own simple programs. For reasons of space, this book cannot give a comprehensive description of BASIC, but an introduction will be given here, and if you want to go further there are several good books recommended in the bibliography at the end.

The art of writing a computer program — in whatever language it may be — can be divided up into four blocks.

The first of these is the most difficult: to define the actual problem. The second step in the program is the actual input of data. The next step is the part of the program which looks after the calculation — the processing — of the problem. The last step is the output of the result.

Like most programs used by computers, the system needs to know the order with which the instructions are to be executed. Hence each statement must be preceded by a line number. It is usual to use line numbers in steps of ten, starting with ten. This allows lines to be inserted in between existing lines if the original program needs amending. On most computers the lines do not need to be typed in their final sequence. Any inserting is done by the computer. If a line number is repeated the original statement will be destroyed and replaced by the new one.

Most computers with an interpreter, and this means most of those available for personal use, can be used without having to write a program. In this interactive mode it is possible to make the computer do something directly from the keyboard.

For example if you typed PRINT "FRED" on the keyboard the screen would respond with FRED when the RETURN or ENTER key was depressed. Similarly PRINT 2 + 4 and the computer would respond with 6 after the RETURN command. This interactive nature of a BASIC interpreter is one of the strengths of using this type of language.

To write a simple program using the above example is quite easy. Say that we wanted to get the computer to display FRED BLOGGS on the screen each time a program was run. First we need to tell the computer where to start so a line number is needed, say 10. Our simple program is then;

```
10  PRINT "FRED"
```

If this is typed in and the RUN command given after RETURN FRED should appear on the screen. If RUN is now typed again FRED will appear again after the RETURN command. One line programs are not very useful so let's look at a two line program;

```
10  PRINT "FRED"
20  PRINT "BLOGGS"
```

If this is entered and RUN the screen will show

FRED
BLOGGS

It is possible to run the two words together, and the symbol ";" after the first line will make the computer print the second line immediately after the first. Similarly "," will space the BLOGGS fifteen spaces in from the edge. The first effect is called concatenation and means that expressions run after one another. The second is a form of tabbing, and usually the semicolon will move a word to the next column defined by the numbers 1, 16, 31, 46, etc, ie. at 15 column intervals. 15 is usual, though some machines use 10 spaces. There is another way of printing words — or strings as they're called — one after another.

This method involves allocating a "variable" or symbol — to the collection of letters being considered. The usual convention calls for a dollar sign, $, to indicate a string of letters. So FRED could be given the symbol A$ and BLOGGS, B$. This can be done in the program eg.

```
10  A$ = "FRED"
20  B$ = "BLOGGS"
30  PRINT A$ + B$
```

Notice that line 30 treats the string variables just like the numbers earlier. If this program is entered and RUN the output will show FREDBLOGGS. (Spaces become important, but the rules vary with different implementation of BASIC. Look at the handbook of the computer being used for the convention used.)

So far we have printed out two words, and needed to type RUN each time we needed output. Say we want FRED BLOGGS displayed ten times. It would be possible to repeat line 30 ten times, using line numbers up to 120, but this is tedious. What we need is a form of counter that can keep control of the number of times a process is repeated. This is usually referred to as looping. For example consider a variable I. Let this be given a value 1 to start with and let one be added each time FRED BLOGGS is displayed, then if the computer keeps track of the value of I all we need to do is instruct the computer to stop working when I becomes 10.

```
10  A$ = "FRED"
20  B$ = "BLOGGS"
30  FOR I = 1 TO 10 STEP 1
40  PRINT A$ + B$
50  NEXT I
```

Line 30 sets up the initial value of I, the value when to stop and the value to be added each time the NEXT I command is met. This program will now output FRED BLOGGS ten times before stopping. (To be accurate a line 60 should be added with an END command. The example shown is a typical case of "lazy programming"!)

It is also possible to jump around in the program. If we just wanted to display FRED BLOGGS over and over again with no control on the number printed the program could have been

```
10  A$ = FRED
20  B$ = BLOGGS
30  PRINT A$ + B$
40  GOTO 30
```

Here the program keeps jumping between 40 and 30 so displaying line after line of FRED BLOGGS! The only way to stop it is to use a BREAK command of some sort — or turn the machine off!

There are many other commands in the BASIC language — but the ones shown are the most common, and less likely to be subject to variation. For further programming examples look in the handbook, or read a good book on BASIC as indicated in the Appendix. This is not a book on programming, and the preceding section is just an example to show how easy it is to program in BASIC!

Some versions of BASIC will only handle whole numbers, and are not capable of working with fractional numbers. These are called **integer** BASICS. Some BASIC dialects, usually called Tiny BASICS, are among these. As most BASIC dialects are based on interpreters, the system throws up errors and indicates the line with the error. Some form of error code will also be shown. If a compiled version of BASIC is used then error checking is more complicated as the whole program has to be 100% correct before a successful compilation is completed.

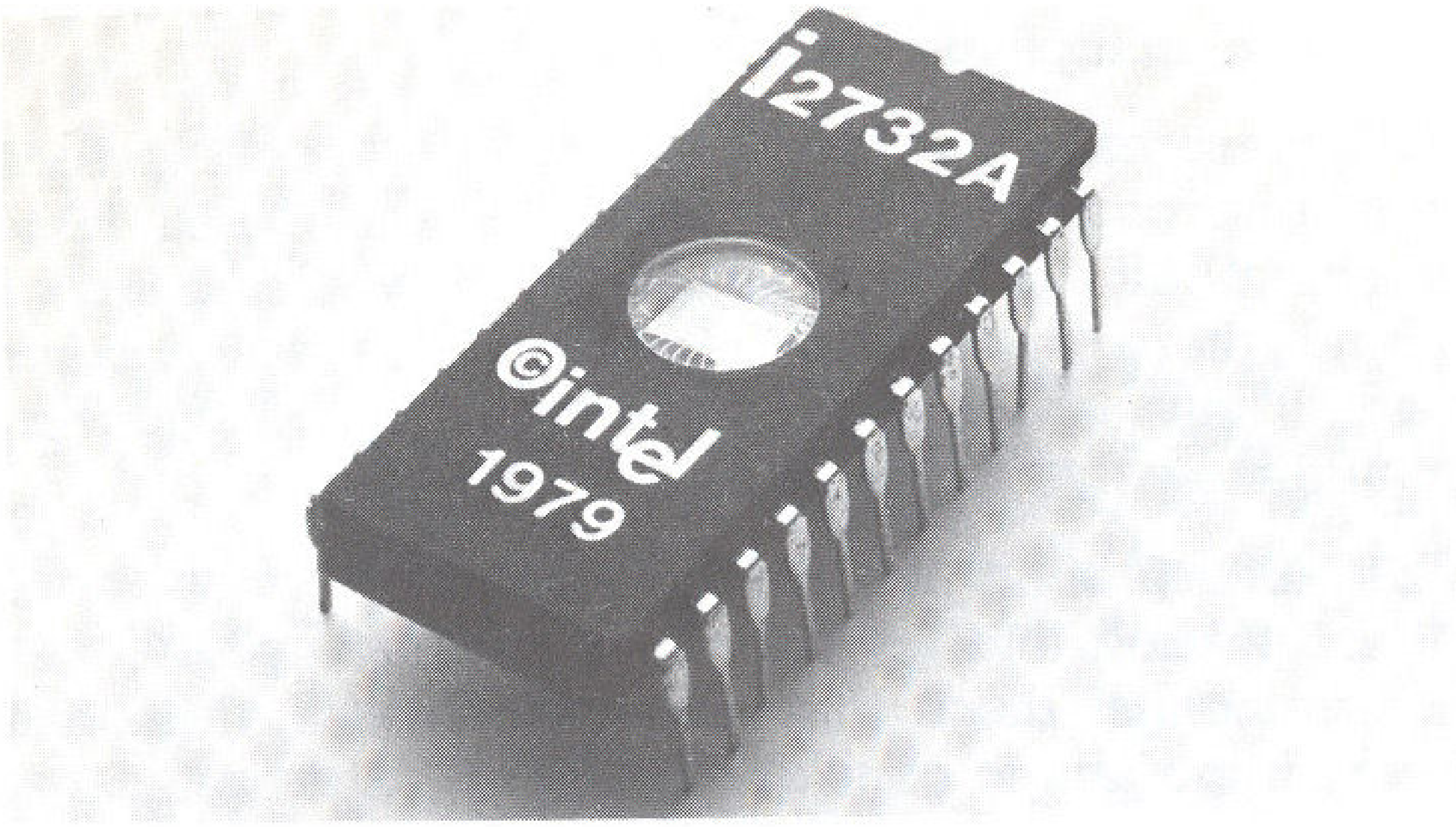
Most BASIC interpreters used by the more popular computers originate from one company in the United States called Microsoft. Microsoft BASIC is now available for a whole range of processors and there is a certain common thread running through them. These are usually resident in ROM and are available immediately the computer is switched on.

Other BASIC interpreters are available on disk or tape and they have to be loaded into the computer before anything can be done. It is amongst these that the most serious variations occur. It is very important to consider the BASIC interpreter that is going to be used.

Firmware

When a computer system is switched on a number of things can happen — a) nothing ie. the system sits there waiting for **you** to do something; b) a symbol appears on the output device prompting some action; or c) a message comes up saying that the system is ready for use. The amount of software support available in the monitor ROM — called firmware — determines this response.

The simplest computer firmware puts the appropriate codes into the microprocessor and sets it up ready for operation. The system does nothing more until the necessary program is entered in machine code. Obviously this is ideal for the hobbyist at this level, but does nothing to recommend a computer to the businessman! If the system monitor is more powerful it may have the capability of "booting" the system and allowing a number of alphanumeric characters to be used. "M" — to examine memory; "G" — to start a program. If a disk or cassette is used however, a machine code program to translate the information into the correct format, and to send the data to the cassette or data interface, is needed. On some simple computers this has to be entered from the keyboard. More sophisticated monitors will allow a "D" or "L" (load) command to automatically "boot" the disk or cassette.



The Intel[R] 2732A 32-kilobit, high-density EPROM

Sometimes the monitor ROM can be a very sophisticated development tool. It may contain an assembler, or even editor, that allows quite complex assembly language routines to be written. The most comprehensive firmware, though, is when a high level interpreter is included. This is usually for BASIC.

If BASIC does not reside in ROM it must be first loaded from disk or cassette before any BASIC programming can be done. If BASIC is going to be the main language used then this is a pain in the neck! Systems that have no high level language in firmware — most S-100 bus systems, Sharp MZ80K etc — can be used for other languages.

There are lots more programs being sold in firmware form than just interpreters or monitors. On some systems plug-in ROM pacs allow quite complicated BASIC or machine code programs to be used. The advantage with firmware is the access time. Whereas a disk or cassette can take between seconds and minutes to load, firmware takes a fraction of a second. Exidy Sorcerer, Texas Instruments TI99/4 and VIC20 are three examples of this approach.

Another form of firmware is available for the Commodore PET for example. These are in the form of plug-in ROMs that sit inside the machine in sockets that have been left vacant for this purpose. BASIC toolkits, word processors and accounting packages are just three applications of this approach. In the PET, as few ROM sockets are available, a small expansion board can be bought that allows up to sixteen ROMs to be used, all at the same memory address! Each one is "dialled" either by software or a hardware switch, depending on the ROM required.

In fact firmware, in the form of plug-in ROMs, is going to do for the personal computer what the golfball did for the typewriter!

Software

Most system software does not reside in ROM form, unfortunately. Languages, other than BASIC, are available, and in many circumstances preferable! These would take up too much memory if in ROM format and the only method is on disk or cassette.

The most obvious software that determines a systems capability, especially when using disks, is the operating system. As most cassette operating systems require more hardware than software we will only consider disk operating systems — or DOS — here!

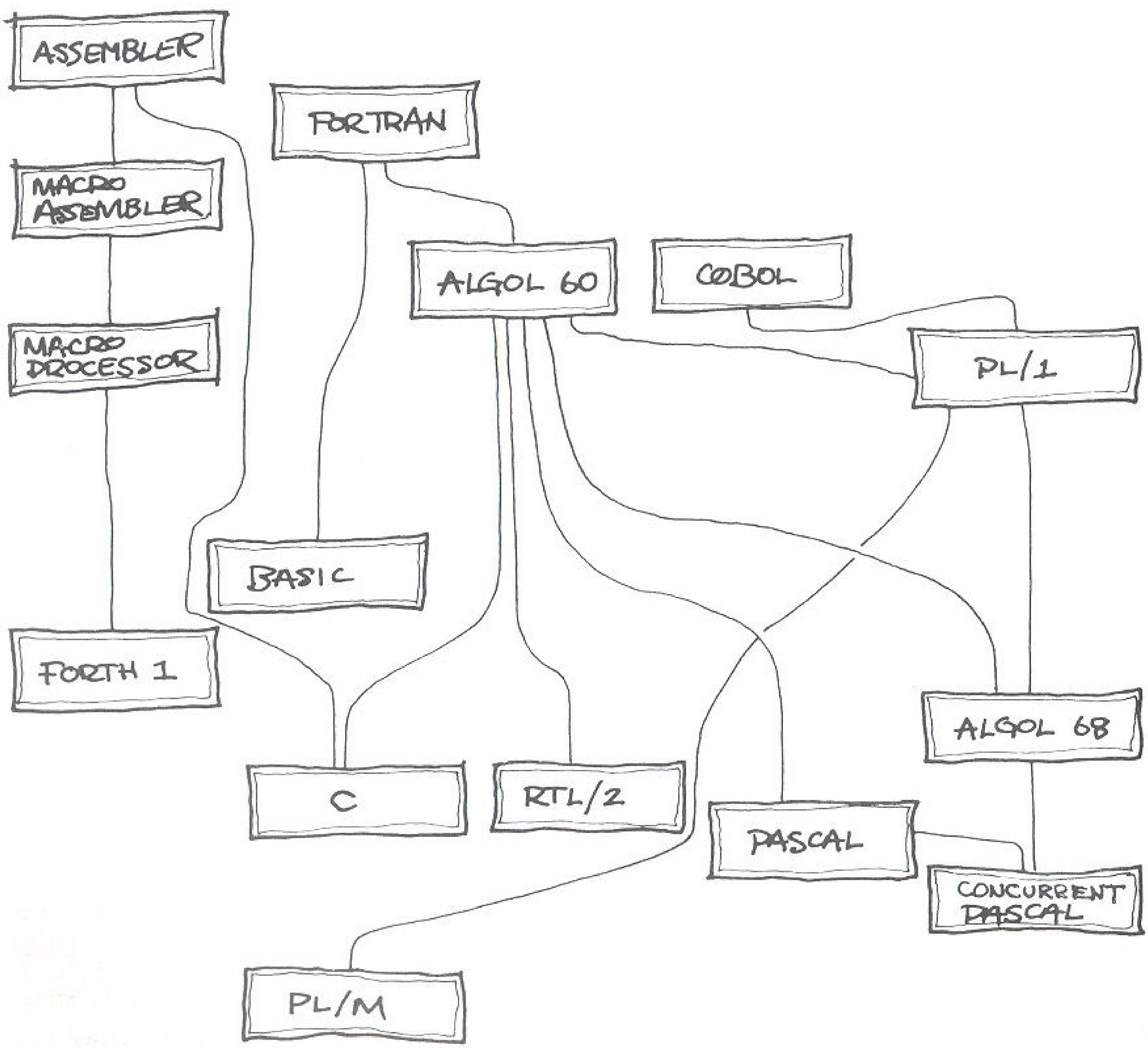
The DOS for any system determines more than anything else its capabilities. It also determines the "portability" of any software developed. Some attempts have been made at standards in this area. On personal computer systems there are four in accepted use.

The standards used by Apple, Tandy and Commodore have become widely accepted, essentially because of the number of machines sold. None has any feature that would recommend itself to outshine the others. Other companies have adopted these standards basically to sell software and/or plug compatible accessories.

The major DOS in use, at the moment, is CP/M developed by Digital Research. It basically lays down a standard for formatting the disk, and encoding the data. In theory it should be portable — and on most Z80 or 8080-based computers using the S-100 bus, it can claim to have some success. Some 6502-based computers like the Apple II or the Commodore PET can accept plug-in units that allow CP/M to be run on them.

It is essential when looking at a disk based system to have a very good idea of the software needed. If the programs available can be adapted to your system very easily, and cheaply, then this may determine the hardware requirements!

A brief summary of the pros and cons of different languages and operating systems is given below.



A family tree of some of the common computer languages as used on small computer systems.

BASIC

BASIC was devised at Dartmouth College in the USA as a high-level language that would be easy to learn and to teach. Its recent rapid increase in popularity has stemmed from the speed with which it can be learnt and from its ready availability on microcomputers. It is the language that is available on the Commodore PET and the Apple. Although there is a standard version of BASIC, so many variations and extensions are currently available, including extensions for text processing or real-time applications, that the standard has little meaning.

Input instructions are interactive, and when executed cause the machine to wait until an input is entered from the keyboard. In BASIC programs, every instruction has a line number. Before executing a program BASIC uses the line numbers to sort the instructions into order.

BASIC provides facilities for handling strings. A variable whose name ends in $ can have a character string assigned to it.

LISP

LISP is a list processing language. The list is a useful representation in a variety of applications. For example, character strings may be regarded as lists of characters, and text as lists of the obstacles to its movement. It can determine whether a move it proposes to make is obstructed by scanning this list.

LISP is a functional language. Every instruction consists of a function and its arguments, and is executed by evaluating the arguments, applying the function to them and returning the resulting function value.

PL/1

The facilities possessed by PL/1 include a combination of those of FORTRAN and COBOL. As a general purpose language it is very complicated, and has not achieved its expected popularity. The slowness of its early implementations was a factor contributing to this.

COBOL

The pre-eminence of COBOL for business data processing stems from the US Government policy that required the provision of a COBOL compiler with any computer bought using their funding. As a commercial language, COBOL emphasises the handling of alphanumeric data and files, so that tasks such as reading and updating file records and automatic form filling can be accomplished.

ALGOL

ALGOL 60 is formally defined in a report dated 1960, and although it is a more rational language than FORTRAN, it has never managed to dent the popularity of the latter to any marked degree.

Input/Output is the one language feature not defined in the ALGOL 60 report, so that it varies from implementation to implementation. All variables must be declared before they are used in ALGOL programs. The sub-program in ALGOL is the procedure. Unlike FORTRAN and BASIC, ALGOL supports recursion, that is, sub-programs may call themselves.

FORTRAN

FORTRAN owes its supremacy over the early high-level scientific programming languages, to its support from IBM. Once established as the language that most scientific programmers knew and in which most scientific software was written, it was naturally difficult to dislodge. Although there are many dialects of FORTRAN, the definitive version is ANSI standard FORTRAN IV.

FORTRAN automatically performs arithmetic operations in the correct order, so that, for instance, multiplications are performed before additions. Brackets can be used to change this order in exactly the same way as in algebraic formulae.

FORTRAN possesses a range of standard functions broadly comparable to that of a scientific calculator.

Input and output are achieved with READ and WRITE instructions. In each READ or WRITE instruction, the key word is followed by a pair of numbers in brackets. The necessity of formats can be aggravating, but it gives the programmer complete control over the layout of his input and output.

Repetition is achieved with a DO loop. This facility gives the automatic repetition of all the instructions between a DO and its matching CONTINUE statement as often as indicated.

FORTRAN supports both functions and subroutines. The function sub-program computes a single value and returns it to the main program. A subroutine can return multiple values besides being executed for its side effects.

The only data structure available in FORTRAN is the array.

The handling of strings and characters in FORTRAN is somewhat limited. However, some dialects permit a solution similar to the one presented in the section on BASIC.

Pascal

Pascal is descended from the ALGOLs. It was designed as a teaching language to demonstrate programming as a systematic discipline. It was also intended that it should be possible to implement the language compactly and efficiently. Pascal appears likely to take over from BASIC as the most popular high-level language for microcomputers.

Pascal provides all the control features necessary for structured programming. Arrays and complex data structures are also supported.

Disk Operating Systems

CP/M

Originally designed to provide a manufacturer independent operating system for 8080 based computers, it has now evolved to work with most S-100 based systems. Basically CP/M is a collection of routines that sit at the top of the computer's memory, and it takes up around 4K of memory. It allows a whole range of machine code routines, disk utilities and high level compilers/interpreters to be run. For example disk-copy, BASIC, Pilot, word processors, assemblers and Pascal can be obtained.

There has been some controversy recently as to the actual "portability" of the DOS; and some problems have arisen when using software developed on a different system.

There are now over 100 suppliers of CP/M software. Unfortunately quite a few S-100 based systems have DOSs similar to CP/M, but not compatible. For example Cromemco's CDOS has a close resemblance, and needs software "translation" to achieve near compatibility.

In practice at least 24K of RAM must be available to run programs using CP/M. In fact the more RAM the better. Similarly a reasonable amount of disk storage is required, and a dual disk system with at least 250K of memory is useful.

Flex

Flex is the "CP/M of the 6800" and was developed by SWTPc to do for the SS-50 bus what CP/M did for the S-100. It includes all the file and disk handling routines required for normal running, and has a utility command set e.g. copy. Its strength lies in the utilities like dynamic file space allocation, automatic "removal" of defective sectors, space compression and expansion on all text files. Again, a minimum RAM of around 20K is required, and a dual drive system holding around 160K is needed.

Which microprocessor is "best"?

This is a question that will inevitably get asked. It is impossible to answer directly! It all depends on what you want to do with it!

When a microprocessor is chosen for a microcomputer system, a number of things are considered by the designers:

The instruction set: Some micros are better, or more efficient, at performing some instructions than others. Unfortunately, there is no one chip better in all aspects than any other.

Speed: In some uses, especially with BASIC interpreters, some micros are much faster than others.

Programming support: How easy is it to program? A micro which is relatively slow but has a wealth of programs to support it will succeed.

Documentation: The designer doesn't want to spend most of his time working out how it works!

The most popular micros are:

8080 — one of the first microprocessors, and the one on which the S-100 bus was first based. Getting a bit long in the tooth now, and not found in too many machines.

Z80 — a development of the 8080. It is based on the 8080 and runs its instruction set — and a lot more besides. Becoming increasingly popular, as it has a fast speed version — the Z80A — that operates with a clock frequency of 4 MHz. It is ideal for the new IEEE standard S-100 bus

6800 — again, like the 8080, getting a bit old fashioned. Didn't really catch on as a processor for computational purposes.

6502 — based on the 6800 but more flexible. Picked up by Apple, Atari and most of the personal computer manufacturers, after Commodore — who designed it — came along with the PET. Incredibly fast when used with the BASIC interpreter from Microsoft Ltd. A 4 MHz version is used in some newer computers — and is *very* fast.

8085 — based on the 8080, but unlike all the preceding chips is not an 8-bit micro. It uses 8-bit addressing and data but has a 16-bit internal architecture — a halfway home to the ...

8086 — which is a true 16-bit processor.

9900 — also a 16-bit chip that hasn't really caught on — and used by Texas Instruments in the 99/4 (they designed the 9900!).

SC/MP — used by hobbyists along with the ...

1802 — and ideal for a single board system.

Z8000 — is an upgraded Z80 with 16-bit architecture but is too new to be in many machines at the moment. Coupled with the new S-100 bus, that can take 16-bit processors, it promises to be the system of the future — along with the ...

68000 — a 16-bit processor that is an upgraded 6800.

CHAPTER 4

What's in the boxes?

We have seen that a basic computer system consists of an input section, a processing section containing the actual processor and memory, and an output section. This chapter will take a look at the various components of a system, and consider, briefly, how they work.

Input devices

The simplest input device to a digital computer is an on-off switch! However, programming and entering data using this method is not everybody's cup of tea! The most common means of entering data and information is via an alphanumeric keyboard. Some basic microcomputers that are only programmable in hexadecimal machine code, use a numerical calculator-type keyboard, but most users want a normal typewriter style keyboard so that high level languages can be used.

Although the keyboard is the most common type of input peripheral, other ideas are beginning to emerge. Speech recognition is foremost amongst these. Although outside the scope of this book a brief decription is included. Finally it is possible to input information and/or data directly through various devices such as temperature sensors, intruder alarms etc.

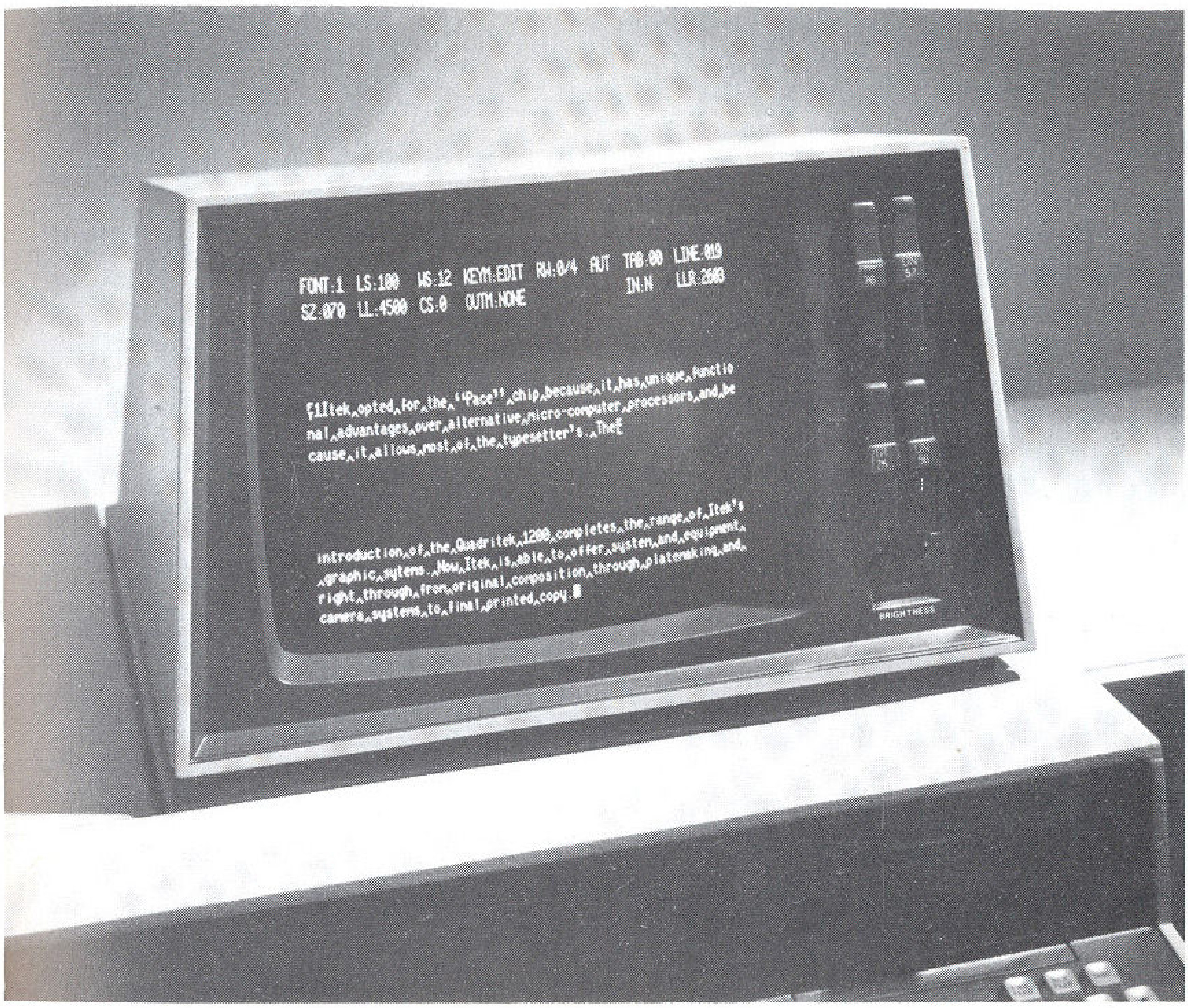
Keyboard-based input

The simplest typewriter style keyboard consists of a push-button switch. When the key is pressed a contact is made and a pulse is produced, and, if only a few keys are needed, then a wire can come from each key and the computer can be programmed to recognise the input.

However, the usual number of keys on a typewriter is around 50. It is impossible for the computer to decode 50 different inputs efficiently. Hence the keys are linked together in a matrix, with the information being presented as row and column data. Most keyboards are more sophisticated than this and have integrated circuits that take the row and column data and present it to the computer in the parallel eight-bit format that can be handled most efficiently. They also have other circuits that detect only one key at a time, and even ignore keys that are being pressed if the previously keyed information has not been acknowledged by the computer. This stops the system from getting clogged up.

The PCD MALTRON Ergonomic Keyboard.

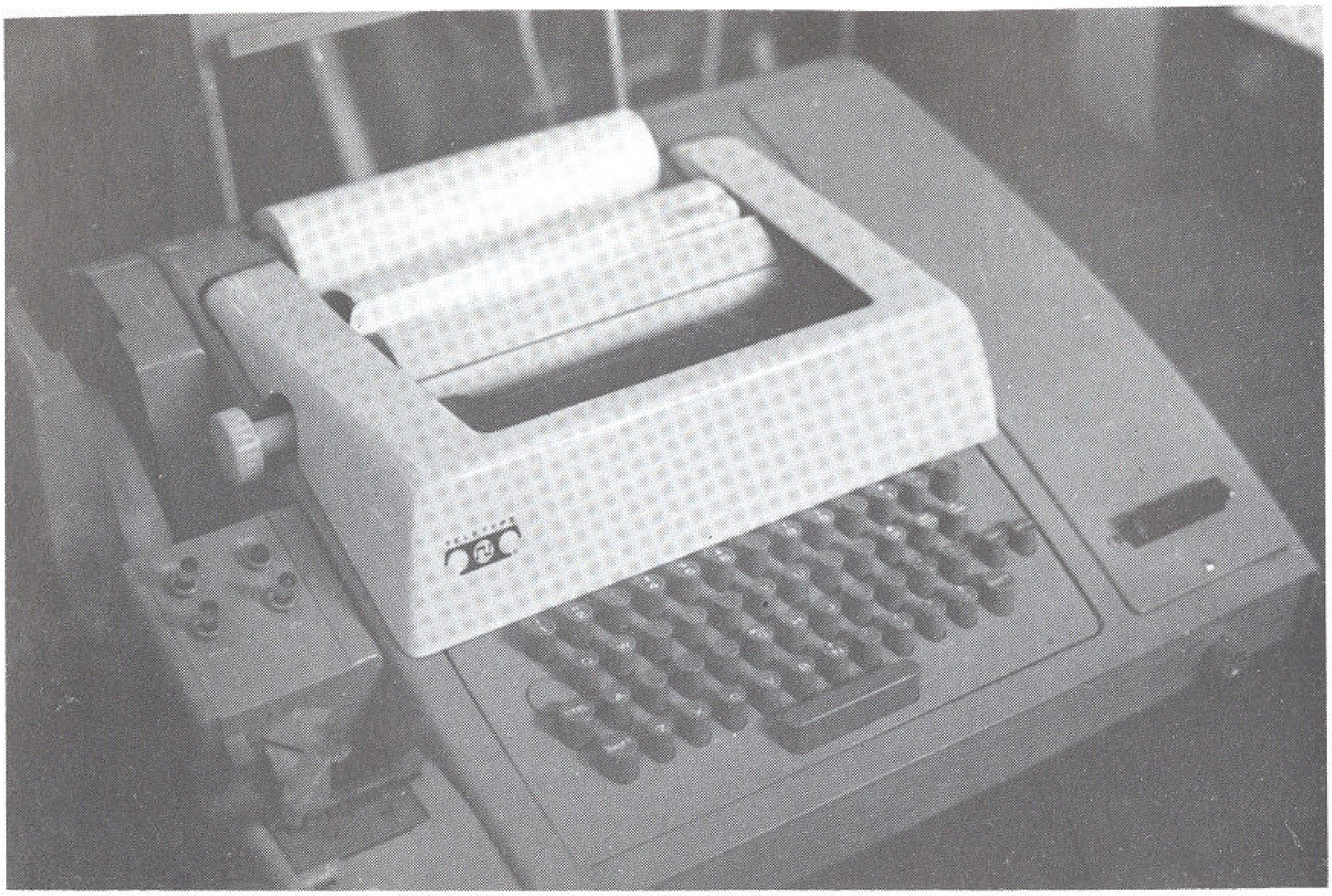
On some systems the keyboard is separate from the main processor, and is just an input unit. On other systems a TV screen is included with the keyboard, and this is called a video display unit, or VDU. The screen not only displays information from the computer, but also shows what is going into the system from the keyboard.



A visual display unit.

Sometimes a printer is attached, and this performs the same function as the screen in a VDU. These are called teletypes, which is the name of the company that first invented them. Although a trade name (and a registered trade mark!) the term teletype, or TTY is now used for any printer based system that can be used to enter data from a keyboard.

Most VDUs and TTYs are designed to be machine independent. This means that they can be used on any computer system. However this involves having an interface that is fairly well accepted as a standard. The most common one for these two pieces of equipment is the RS232C interface standard, and this is discussed later in this chapter.



A Teletype

Speech recognition

Speech recognition is becoming more widely accepted as its capabilities are being extended with increasingly sophisticated advances in technology. Only one or two personal computer systems can handle speech synthesis at the moment, but with the technique appearing in other consumer orientated electronic goods — such as TVs and calculators — it is only a matter of time before it is generally available.

Basically the computer compares the input information with that stored in its memory. When it recognises various parts of a phrase or sound it works out what is being said. Most systems can only handle 20 words at the moment, and this requires a lot of memory. The present systems have to be trained to recognise the operator's voice — and then can only respond to that individual. Obviously this has great benefits as far as security goes — but does not recommend the method for general use!

Direct input

Sometimes it is useful to have input to the computer from various external devices, as well as keyboards. Various interfaces have been designed to deal with these — the most common being the IEEE-488 and RS232C standards. These deal with data in both directions — to and from the computer.

The IEEE-488 interface standard — or GP-IB (General Purpose Interface Bus) is based on a sixteen way parallel transmission of data. It was initially intended as a standard for controlling and interrogating electronic instruments. But Commodore introduced it to the personal computer world and now use it for all their peripherals like disks and printers. A full explanation of the standard as applied to personal computers is given in Appendix B.

Atari^R 830^TM Acoustic Modem.

The other standard — RS232C — is much more common — and is used by the vast majority of teletypes, VDUs and modems (devices for sending data over telephone lines). Unlike the IEEE-488 standard the RS232C standard defines the way that data is transmitted and received in serial format, ie. pulses following one another. Consequently only three lines are really needed to get the system working — data receive, data send and earth. (The standard in fact specifies 25 lines, but 15 are usually used as a maximum.) These other 22 lines include facilities for telling the computer to accept the signals. The rate of data send and receive can also be set automatically. Normally this is done by using switches on the input port of the computer and the output port of the peripheral. It is important when using devices that use the RS232C standard that the rate of data transmission is compatible. The usual rates are 110, 300, and 1200 bits/sec.

When the computer is used to accept or send data to a peripheral, or even another computer, using telephone lines, two methods can be used. An acoustic coupler translates the pulses from the computer into acoustic levels that the normal telephone lines can carry. Alternatively the output can be connected directly to telephone data lines via a box called a modem. Most acoustic couplers have a 25-pin socket that connects directly to the RS232C input/output port of the computer. As prices drop they are becoming more popular and most personal computer manufacturers are offering, or will offer such devices.

Storage media

Methods of storing data inside the computer using semiconductor memory, RAM or ROM, were considered in an earlier chapter. The normal 8-bit microprocessor uses 16-bits to address this memory. Thus the maximum amount of memory addressable directly from the processor is 2^{16} or 64K bytes. With the monitor and/or interpreter taking up to 20K of this in ROM form, and the video needing up to 8K of RAM to store the screen information, sometimes only 36K is available for user programming space. In fact most small systems are limited to 32K or 48K as a maximum.

It is clear that other forms of storage are needed. Unfortunately none of those generally available come anywhere as close to semiconductor memory in terms of speed and ease of use. Two things are required to overcome these problems. Firstly an interface and/or a program that will transfer the data into a format that can be used by another storage medium, and secondly some method of changing the speed of data transmission from the fairly slow rates required by most peripherals to the faster ones used by the computer. This is usually achieved by using a buffer that accepts an amount of data from the peripheral, storing it and then transmitting it to the processor, or RAM, at the appropriate speed.

Cassette storage

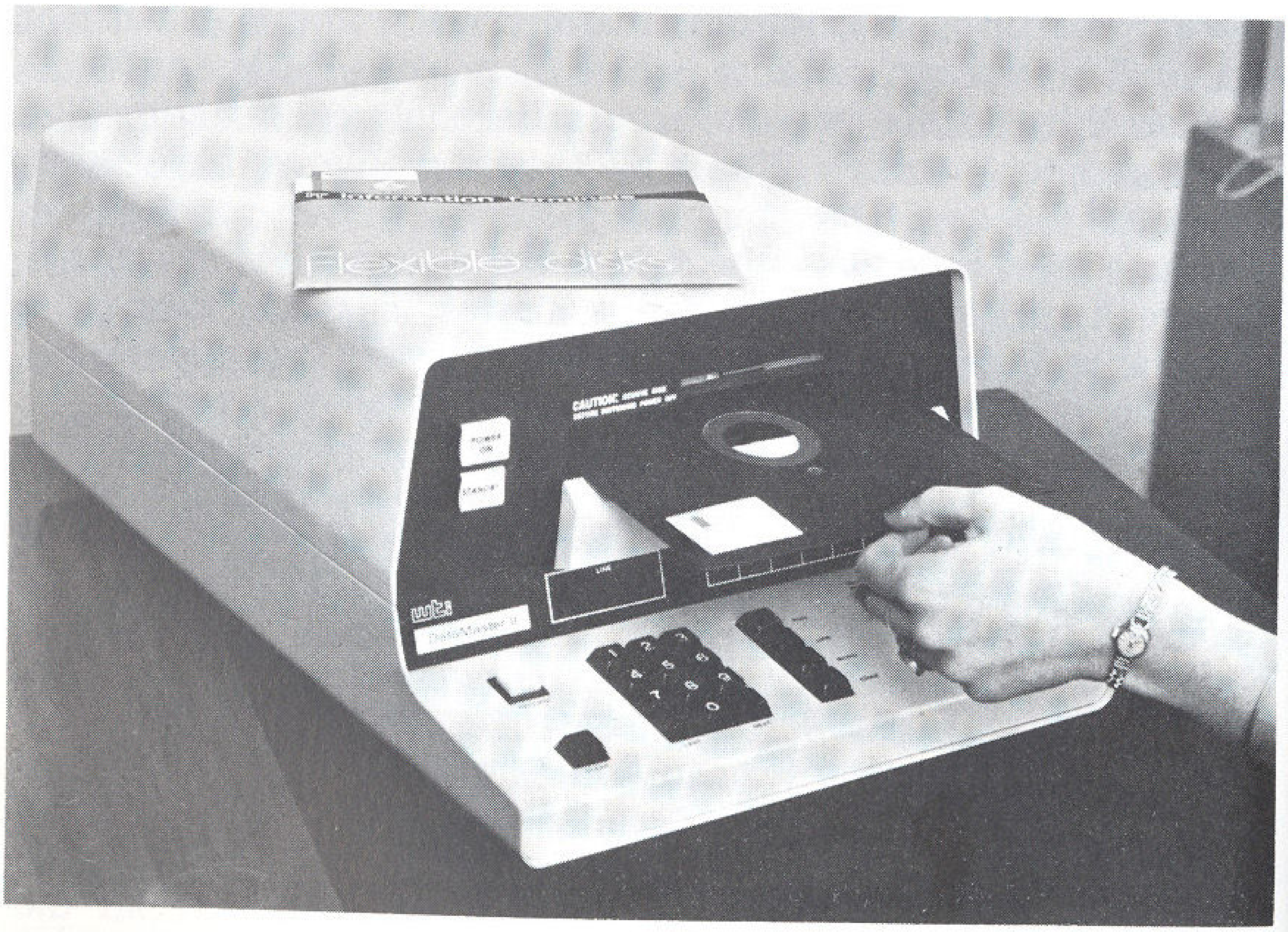
The simplest method of storage is to use a domestic cassette recorder. The interface changes the pulses into a format that can be recorded and played back by the cassette recorder electronics. The only problem is that the rate of transmission is very slow — anywhere between 100 and 1000 bits/second. The electronics must be very tolerant of speed variation as the average cassette recorder is not very good at keeping constant speed! The average cassette is also of dubious quality as far as data recording goes. Our ears are far more tolerant of errors than any computer system! When all these problems have been overcome — and the state of technology has reached an adequate level by now — the only major drawback is the speed. For small programs, up to say 4 Kbytes, cassette storage is acceptable. But when it takes up to 10 minutes to load a program, and then still have the possibility of errors, things are nowhere near ideal. Cassette storage is cheap though. Most homes now have a cassette recorder, and if not they only cost a few pounds to buy. Cassettes themselves are also fairly cheap. Hundreds of kilobytes can be stored on an average C60 cassette costing under £1.

The other main problem with cassette storage is the lack of any standard recording format. An early attempt at producing a standard was made in Kansas City. The result is what is now known as the Kansas City standard. This is the only accepted standard for personal computers, and is by far the most common, although used by a small minority. Another one is the KIM standard, and this is used by similar single board computers such as the AIM and SYM and UK101.

Even with cassettes recorded to a single standard, programs may still not be transportable from one computer to another. The BASIC interpreter may be different; certainly the memory map will be and this will affect any part of the program in machine code. There is even the problem of different head alignment on various tape recorders. Even the same tapes may not be played back on a different recorder to that which recorded it.

Disk storage

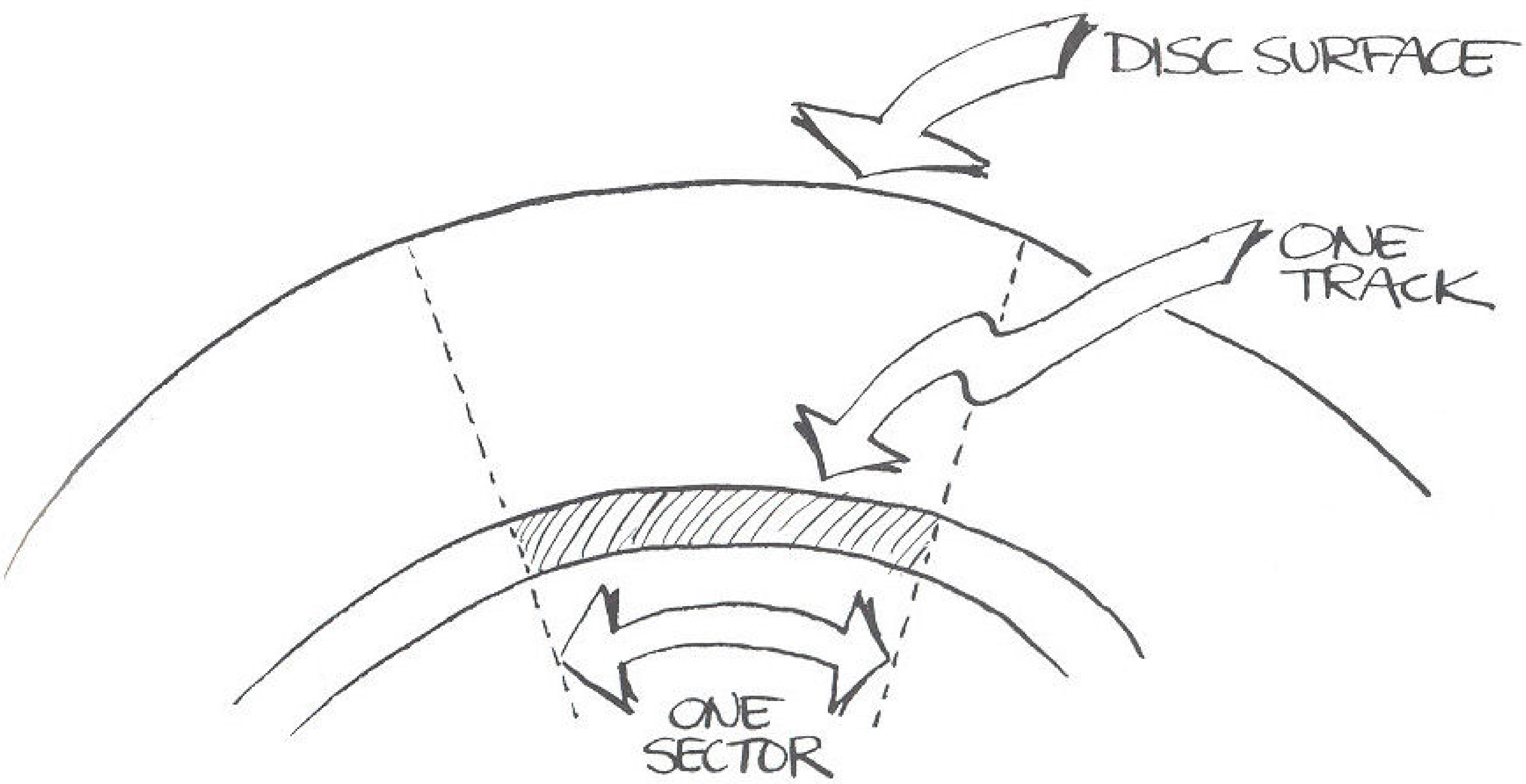
The generally accepted method of overcoming most of the problems of cassette storage is with disk drives. These are not cheap though — and start at around 10 times the price of an average cassette recorder. The disks themselves start around £2 each.



Flexible Disk (Courtesy of Datamaster).

Unlike records that have information stored in one groove and are accessed serially, magnetic disks are divided into a number of discrete tracks. Each track is then divided into sectors. This can be done by the computer, under software control, or by a separate microprocessor in the disk unit.

There are two basically different ways for the read/write head to find the right place in the different disk sectors — "hard sectoring" and "soft sectoring". A hard-sectored disk has a number of holes punched inside around the centre: each of these holes is a sign for the beginning of a new sector. The soft-sectored disk only has one hole indicating the beginning of each track — the length of each sector is determined by programming (hence the term soft sectoring). These two sectoring methods are not compatible — in other words, a hard sectored disk cannot be used in equipment for soft-sectored disks.



Diagrammatic representation of a floppy disk, showing sectors, segments and tracks.

Disks come in various diameters. The most common are $5\frac{1}{4}''$ and $8''$ diameter floppy disks. The name is indicative of the material used for their construction. It is a flexible plastic with a film of magnetic oxide on each side. Because it is easily damaged the disk is enclosed in a thin card case. A small slit is provided for the record/playback head to come into close proximity to the disk. The exposed sections of the disk must not be touched by greasy fingers, as some of the information stored might get corrupted. Obviously extremes of temperature must be considered, and under no circumstances must any magnetic field come near the disk.

Disk drive

This is a unit consisting basically of a holder for the disk, a motor driving the disk via a spindle at 300-360 revolutions a minute, and a stepping motor driving a radially movable mechanism holding the magnetic head for reading and writing. There is also an electronic controller (as a rule in the form of a separate circuit board which is placed in the computer bus). On command, it is the controller which operates the magnetic head of the unit, moving it to the desired track on the disk to read or write the data. The controller can serve several floppy disk units at once.

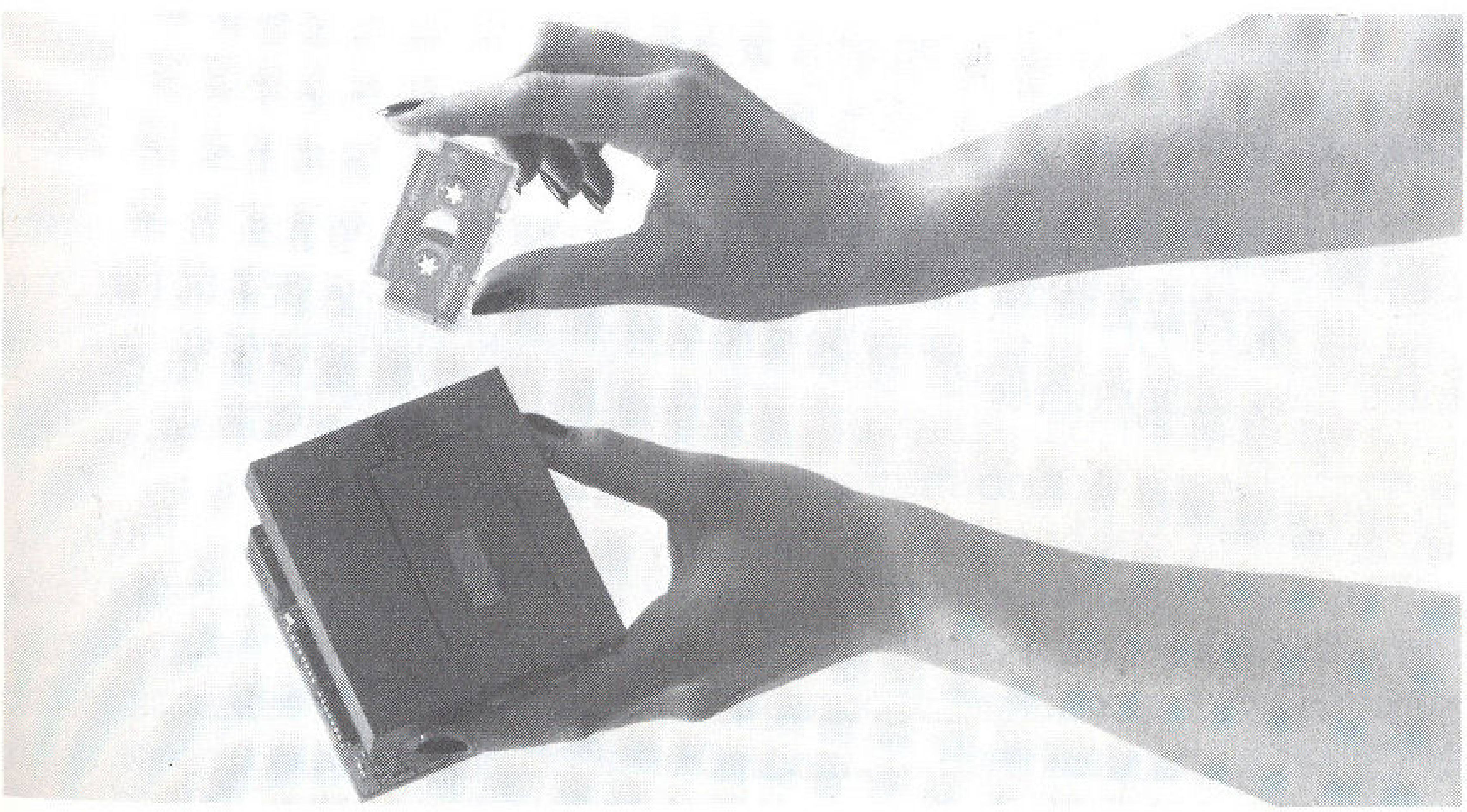
The normal $5\frac{1}{4}''$ disk has 35 tracks with 16 sections — although this can vary. According to the method of formatting the information, anything between 75 and 360 Kbytes can be recorded on each side. Some drives have two heads, one for each side, and on these dual-sided drives up to 700 Kbytes can be stored. It is important to know whether the disk drive is single or double sided, single or double density formatted. The disk manufacturers make different disks for each type. Needless to say it is not possible to exchange a disk between different types of disk drive, however portable it may be between drives of a similar type.

The normal 8″ disk has 77 tracks and 26 sections and stores between 256 Kbytes and 1.2 MBytes. 8″ disks are usually formatted along similar lines to those used by mainframe computers like IBM and DEC.

A disk that is coming onto the market and will certainly be important for small business applications in the future, is the 8″ hard disk, or mini-Winnie! The mini-Winnie is a smaller version of an IBM mainframe disk called a Winchester unit and can store up to 26 Mbytes. The main difference between hard and floppy disks is the speed of revolution — around 10x greater for hard disks compared to floppy disks. The record/playback head also sits very close to the disk — somewhere around 1 millionth of a metre away. As this is somewhat smaller than the diameter of human hair, or dust particle, hard disks are enclosed in an air-tight environment usually filled with inert gas under pressure. Speeds of transmission are far greater than floppy disks and approach 2.5 Mbytes/second. Unfortunately mini-Winnies are still rather expensive and only a few personal computers can interface with them.

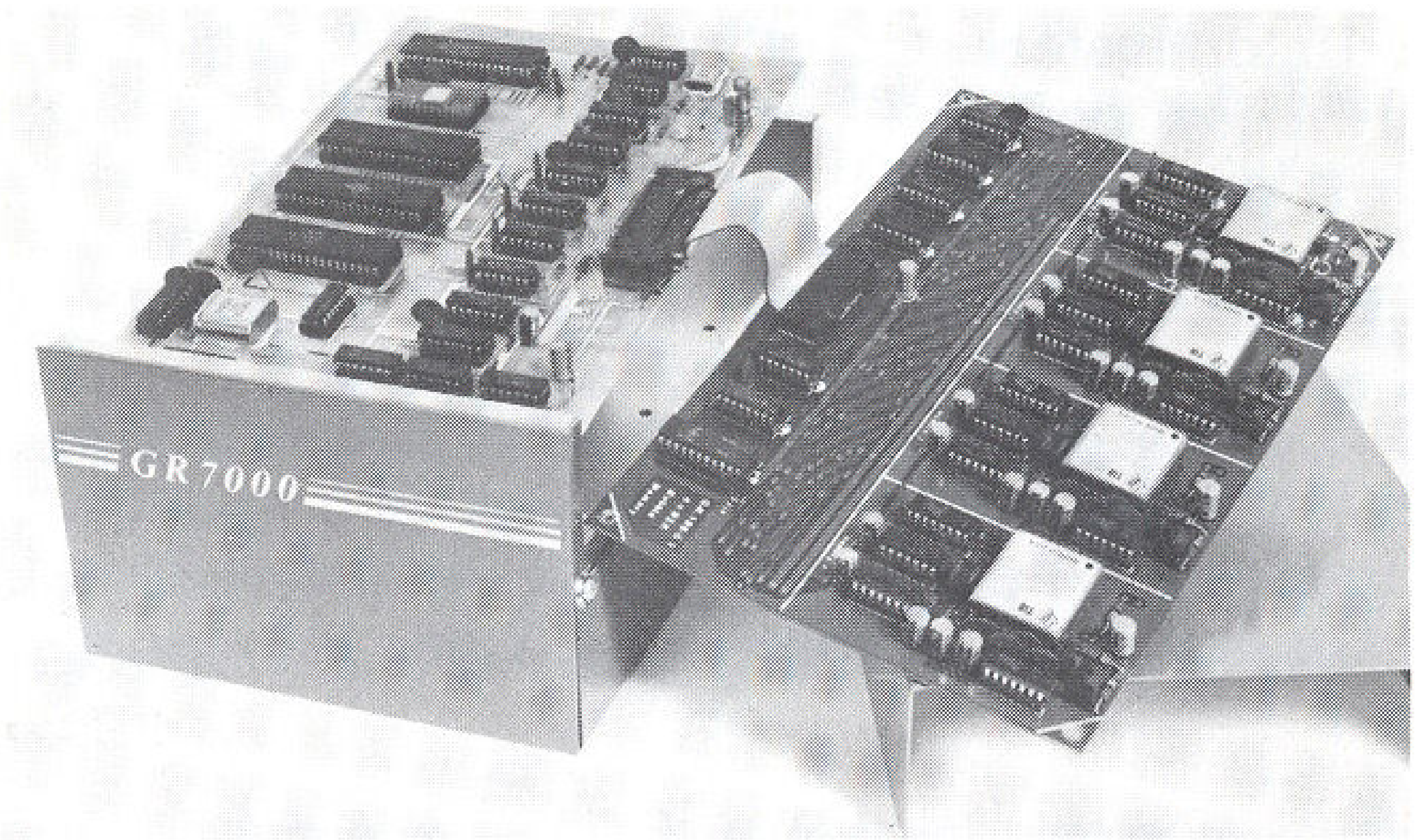
Other storage media

There are methods for storing information digitally other than disk or cassette. Small cassette-type packages called cartridges are used by a number of manufacturers like Hewlett-Packard. These can read data very fast and have access times approaching floppy disks.



The ultra compact Philips Mini-DCR can be easily held in one hand, yet it has a capacity of 128 Kbytes with fast access and a high transfer rate of 6 Kb/s.

Magnetic bubble memories could become more familiar over the next few years. These are used for storing large amounts of data — in the order of millions of bytes. This method of storage uses similar technology to semiconductor memories, except that small magnetic devices are created into a magnetic medium like transistors are in silicon. The memory is addressed serially, like a cassette, and is thus fairly slow — but takes far less power and space than a cassette. However they are still expensive compared to other types of magnetic storage.



The Fabritek GR7000 bubble memory module is a serially interfaced RS232 compatible memory system designed for ensuring data storage in harsh environments.

Output devices

It is becoming impossible to divorce output devices from input devices as time goes by. It is general practice to have both in one unit — the VDU and teletype are examples. Hence what follows must inevitably contain some information that has gone before.

Video output

The VDU is the most common form of video output. The simplest is a domestic television set. Some computer systems, especially those based around a keyboard unit, have a socket for a TV aerial on the back. If this is designed for European sets then all that is required is to tune the set into, say, channel 36 and adjust the brightness and contrast controls. Some of the computer systems that are designed for home use have colour output. In these instances it is absolutely essential that the TV output be the standard for the country of use. It is possible to use systems made for use in, say the UK and Germany without much difficulty. However, France uses a completely different standard and USA and Japan another system again! Black and white poses less problems, but some instability of picture, and maybe less characters on the screen, could occur with systems designed for the US and Japan when used in Europe.

The most foolproof output is called direct video. Instead of converting the computer's video information to the high frequency used by television transmission, the signal is presented in the form used inside the TV. Nearly all TV sets only have an aerial socket. Some sets, designed for use with video recorders, may also have a video socket. Computers give a voltage of 1 volt peak to peak across their video output, so any TV must be able to take this. With 1v pp video it doesn't matter what transmission standards are used in the country of origin — so as long as you only want black and white, everything's OK. Unfortunately there are still 3 different colour standards around the world — so any system using, say the American and Japanese NTSC standard, can only be used with an NTSC receiver. Some American colour computer systems have a permanently attached video monitor — Compucolor for example — or sell a specially converted TV set capable of operating on two or more standards — TI for example with the original 99/4.

Whatever form of output is used, nearly every video display uses similar methods of generating the characters. A screen that can display 25 lines of 40 characters requires 1000 memory locations to store the information. In these so called memory mapped systems, each character position on the screen corresponds to a particular memory location in the video memory. All the computer does is transfer the contents of the memory onto the screen. As each memory location can store one byte — of eight bits — this means that 2^8 or 256 different characters are available. The most common method of encoding characters is the ASCII system. As this only has 7 bits of information, only 128 characters are available. Most personal computer systems add another 128 to make up the number — and they do this by creating graphics characters. There is no standard for these so that PET graphic symbols, for example, have different codes from Sorcerer graphics. The codes are interpreted into actual letters and graphic symbols by a ROM called a character generator. It is clear that all this memory detracts from the amount of memory available for use in the system by the user.

Another method of generating video information is to split the screen into a series of discrete points, say 312 x 210. This requires a lot more memory. Each point on the screen now corresponds to one **bit** in the memory, hence eight points require one byte of memory. 312 x 210 points need 8 Kbytes.

This high density graphics capability is available on a few computers — like the Apple II and DAI. It is a useful facility for educational users — but unless complicated graphical analyses are needed is not really ideal for business use.

Most VDUs in fact take the data in serial format using the RS232C standard. The memory needed to store the information is inside the VDU and does not take up any of the computer memory. As a VDU costs as much as a cheaper computer system however, this method of video output coupled with keyboard input, is really recommended for more expensive bus-structured systems.

Printed output

There are many different types of printer available for computer users. These range from simple printers for around £200 to word processor output at about £2000.

Electric typewriter/TTY

The simplest way of generating printed output is to use a converted electric typewriter. There are a number of these around usually based on the IBM Selectric range. The computer uses software, or a hardware board, to convert the signals into the format used by the internal electronics inside the typewriters. Sometimes these will allow the keyboard to be used to enter information as well.

A Teletype is cheap, if you buy it second-hand. There are lots of used machines on the market to be had for anything from £25 to £300 or more. There are three basic model types: RO (receive only), which has no keyboard but is used as a printer; KSR (keyboard, send and receive), which is the one described above; and ASR — a KSR which also has punch and reader for paper tape. Most Teletype machines are made by the American Teletype Corp, but other firms such as Olivetti also make them.

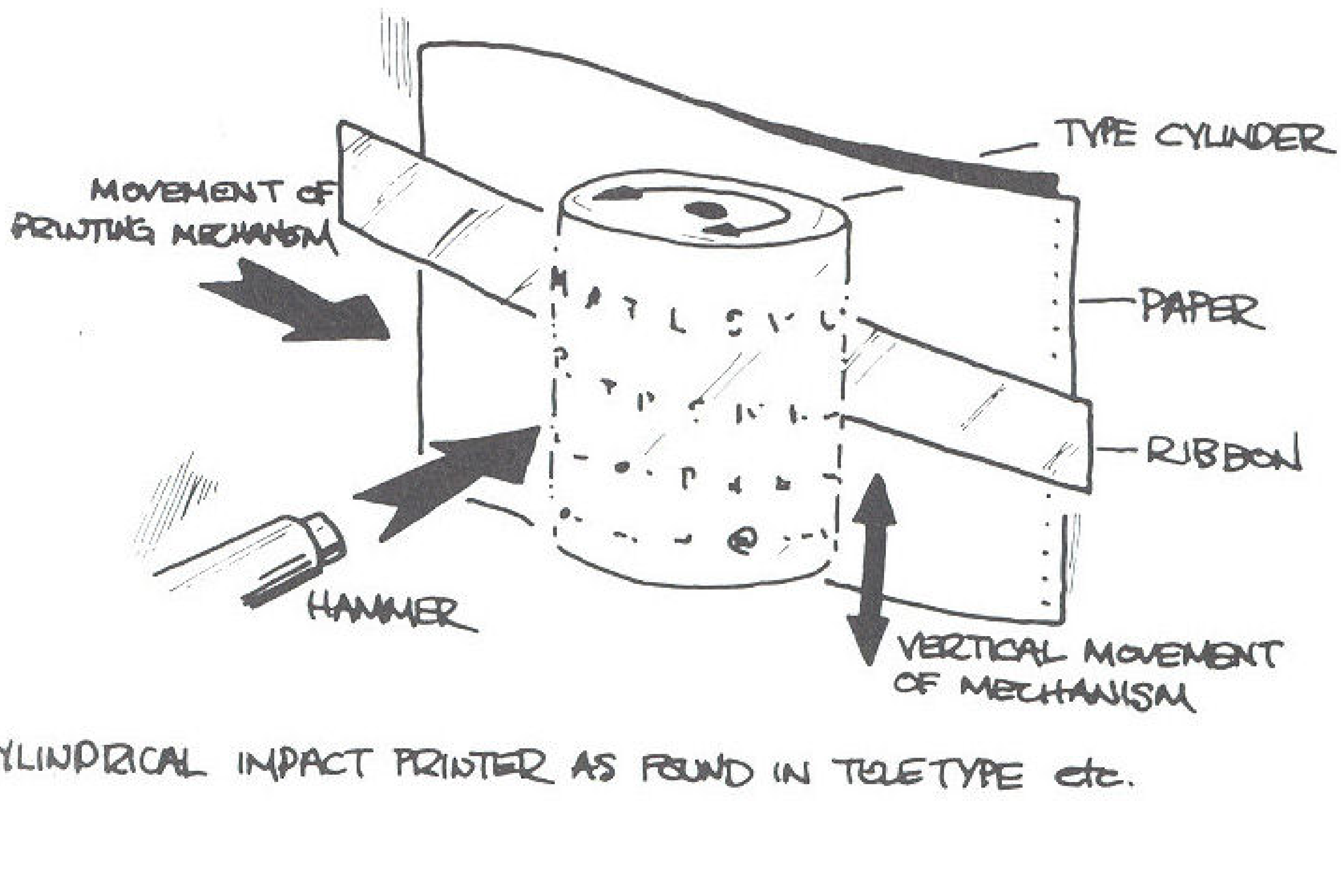
The Selectric machine is an IBM design available in a series of different versions. The spherical type core is common to all of them. This gives very good print quality, it is replaceable, and there are many different type styles.

In its simplest form, the Selectric machine is a normal office electric typewriter. Since the construction is essentially mechanical it must be modified to be able to work with the ASCII signals of the computer. Modification kits are available on the market.

It is somewhat simpler if you can find a second-hand Selectric model 72, 731, 735 or something of the sort. A machine of this type has been used as input/output terminal in various computer systems. The only modification that has to be done is to equip it with an interface for ASCII code.

Several firms have also begun to market Selectric terminals that are reconditioned and modified for ASCII — in other words, completely ready to connect to the computer.

A general word of warning about Selectric machines: the internal mechanics are pretty advanced and may be difficult to repair on one's own. In other words, avoid faulty Selectric machines.

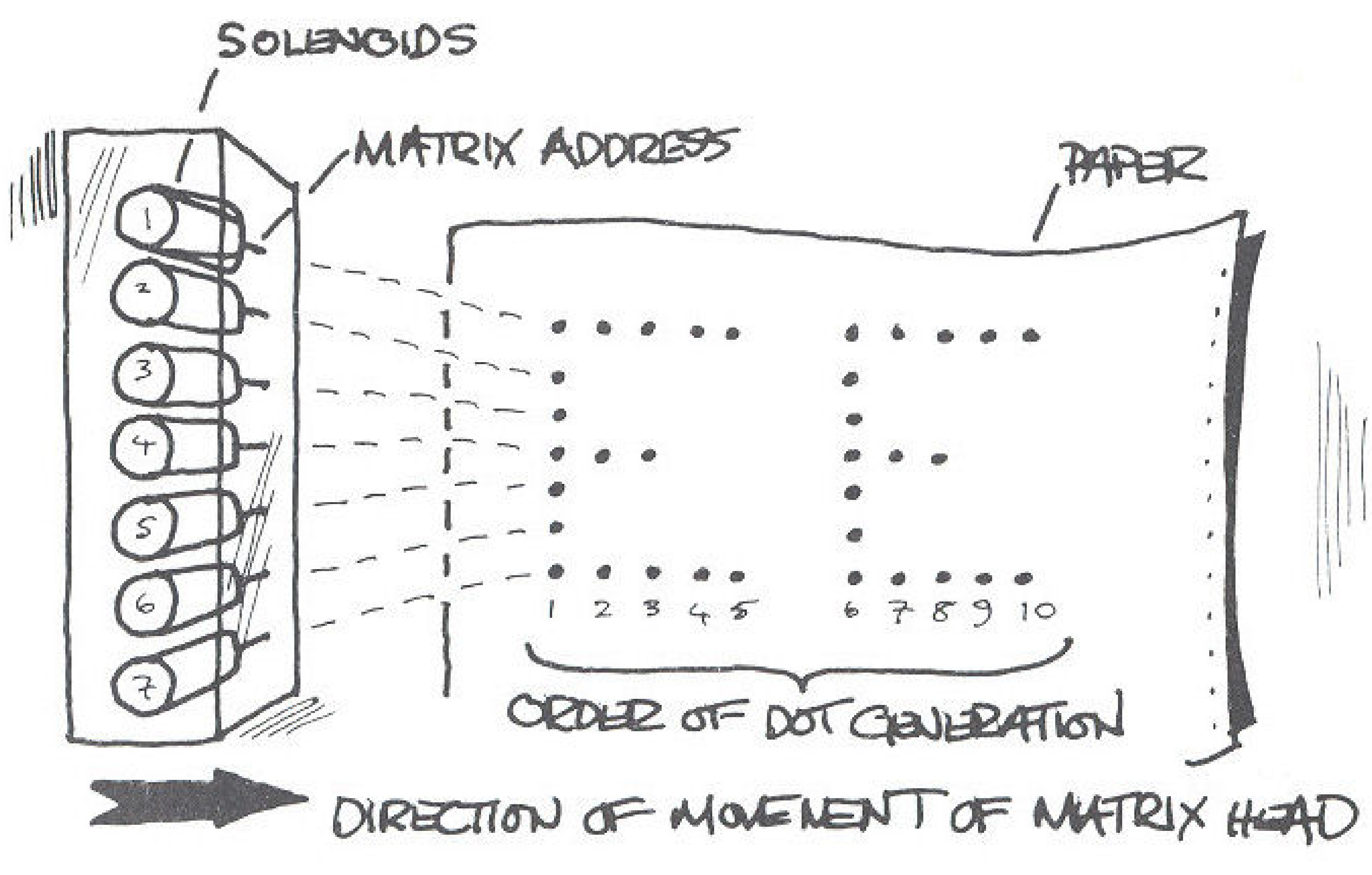


CYLINDRICAL IMPACT PRINTER AS FOUND IN TELETYPE etc.

The Teletype machine is electromechanical in construction with a keyboard for input and an impact printer (a type cylinder) for output on paper which is stored as a roll in the machine. The speed is modest, usually 10 characters a second (110 baud since each character consists of 11 bits — a start bit, eight data bits and two stop bits; in other words series transmission as with the RS-232). There are also Teletypes that are faster — 150 or 300 baud. Other disadvantages are that the machine is loud and usually only prints capitals.

Matrix printers

Another method of producing characters on a page is by printing dots based on say, a 5x7 matrix. Most characters and graphics symbols can be generated this way. The simplest dot matrix printer consists of seven needles in a vertical line that are "fired" onto the paper as the head moves from left to right. Five such "firings" are needed to produce each character. A normal typewriter ribbon means that normal paper can be used. The cheapest dot matrix printer is around £50.



THE WRITING HEAD OF THE MATRIX PRINTER MOVES LEFT TO RIGHT EACH SYMBOL IS BUILT-UP SEQUENTIALLY FROM DOTS, NORMALLY 5-7

It is possible to do quite complicated characters with dot matrix printers as each character can be programmed individually. If the computer system used has a graphics capability a dot matrix printer must be used otherwise these characters will not be printed. And it is here that the non-ASCII standard graphic characters cause problems. Whereas the normal alphabetical and command characters will reproduce on practically any printer with an ASCII interface the graphics do not. A PET computer, for example, *must* have a PET-compatible printer to reproduce its graphics.

It is also important to make sure that the printer can be programmed to reproduce different line widths. The usual 40 characters per line of a small system means a bit of juggling if an 80-line printer is used. Similarly with 64 character systems on a 132 character printer. It is essential that some method of programming the output to take these incompatibilities into account is available.

Daisy wheel printer

If good quality output is required then either a converted Selectric typewriter, or specialised printer, is required. The daisy wheel printer is a form of impact printer that has all the characters around the rim of a circular plastic disk. A hammer hits the appropriate character to produce the printed symbol needed. These are rather expensive when compared to other sorts of printers but give good quality at high speed. The print head can be changed to give different type faces. With most daisy wheel printers it is possible to program them, from the computer software, so that right and left justification with proportional spacing is available. They are designed for word processing applications.

Other printers

The printers mentioned above all use impact technology and rely on ink coming from the typewriter ribbon to create the image on paper. They also use normal paper. There are other ways of getting image onto paper and some printers use these methods.

Heat sensitive paper is used by a number of small printers. These usually have about 40 characters per line. A small dot-matrix heating element moves across the paper, and where a hot spot is created a blue dot appears on the page. They are quick, and because there is no impacting, very quiet. The paper is rather expensive however, and is usually only available in 2″ to 3″ wide rolls.

There are other forms of heat sensitive paper that has an aluminised finish. This is even more expensive than the previously mentioned paper.

Speech Synthesis Unit.

Speech synthesis

Like speech recognition, speech synthesis is becoming more important. It is a lot easier to perform than recognition and takes up far less memory space. Around 200 words can be stored in a 128K byte ROM. The quality is very good, and is in fact used a lot by telephone companies to announce engaged lines, for example. Texas Instruments are pioneers in this field, and they sell a board that can be used in a number of computer systems.

One of the most useful applications of speech synthesis is to announce system failures, or even to give directions on how to use the system! It certainly makes using the computer more interesting.

CHAPTER 5

What can I buy?

The computer system

We've now got to the stage where we know what a computer is, how it works and what's in the boxes. The question now is — how do you actually choose a computer that will serve your needs? This chapter lists the main, generally available, models. Microcomputers can be grouped into four main categories.

First are the single board microcomputers. These originated from the "evaluation" boards that the early microprocessor manufacturers produced to allow engineers the opportunity of familiarizing themselves with the new devices. They have since evolved upwards into self-contained computer systems with keyboards, memory interfacing and even printers. They have also gone down market and can be had for under £30 although at this price they are very basic and are really aimed at the electronics hobbyist who wants to put a simple system together. Also included in this section are a few training computers. These are designed for engineers, scientists and mathematicians who want to learn about machine code programming and interfacing techniques. As these also contain very good self teaching manuals they have been included as some readers may be interested in this area.

Secondly there are the cheaper systems aimed at the home, entertainment or educational use. Like the first category they frequently come as a "naked" circuit board without any casing. This trend has fortunately now ceased and it is even possible to get plastic cases for the early sytems that were produced without them. These systems are very versatile and most support cassette drives, disk drives, video display, memory expansion, printers and modems. They can be developed into quite sophisticated systems, although this category will be dominated by the American TV games manufacturers that have seen a new market evolving in front of their eyes!

Thirdly, there is the "desktop" category. This term was coined some years ago by Hewlett Packard to describe their self-contained units that were "friendly". What they meant by this was that as soon as the unit is powered up it "talks" to you, and can be programmed straight away. This is because the high level language, usually BASIC, is present in ROM and hence does not need loading with a machine code routine, or "boot" before programming takes place. Commodore were the pioneers of this approach with the PET, and the vast majority of the personal computers around follow this approach. They usually also have an integral cassette unit, although the price of disk drives has been dropping and the size of programs has been getting bigger. Hence they are being replaced by integral disk or optional cassette or disk drives.

The original desktop computers had black and white video output, usually on an integral screen. Apple blew this approach apart when the Apple II with its colour capability came on the market. As most homes have colour TV it is now normal for those computers with colour capability to have an output socket that plugs directly into the aerial socket of a domestic colour TV. There is some problem here for US-made computers as Europe uses a totally incompatible system of TV transmission. Therefore most of these computers have been modified for Europe, or have integral or attached TV screens that do not need altering. These problems also occur with those American machines in the second category, although as they rely on TV output they have usually been modified by their European distributors.

Also included in this category are systems which come in separate "boxes" but can be considered as essentially a unitary system.

Finally we come to the systems that are closest to classical computers. This fourth category has two subsections — those based on a bus system and those that are really glorified desktops. the first sub-section are machines that use the S-100 or SS50 buses. The S-100 bus is undoubtedly the most popular — and in its revamped form as the IEEE S-100 standard bus is a very flexible approach. Unfortunately not all components claiming to be S-100 bus based actually implement the standard fully! The SS50 bus only appears with two or three manufacturers and is becoming rather neglected — but it exists. The main advantage of bus structured systems is their flexibility. A system can be configured to particular needs, or expanded easily. However, with plug-in units now available for expanding non-bus structured systems this advantage is not as overwhelming as it was. Most larger, business microcomputers can be grouped in this classification.

Larger desk top systems are capable of supporting large internal memory and large capacity disk units — say 8″ diameter floppy or even hard Winchester-type disks. They are fairly expensive in their basic configuration — and the survey has an upper limit of around £3000 for the minimum useful system.

In the brief survey that follows the following points are emphasised:

Manufacturer: The actual manufacturer, not the distributor, is given. The country of origin is also given. This is important when thoughts of servicing, repair and software back-up are taken into account. In general European computers are easier to get serviced than American, which are better than Japanese.

Brief description: The basic concept of the system is mentioned — with some comment on points of particular interest.

Peripherals: The peripherals available are listed whether from the actual manufacturer, or plug compatible manufacturers.

Keyboard: Number and type of keys.

Display: Number of characters and lines, whether on integral display or on TV output available.

Interfaces: Summary of interfaces available on the system, with optional interfaces if plug-in modules used.

Minimum memory size: Size of RAM for minimum configuration of the system. This is user available RAM and does not include any RAM that is reserved for memory-mapped displays.

Maximum memory size: Maximum user RAM available in largest system.

ROM: List of ROM in minimum system. This usually includes size of monitor and, if applicable, size of any interpreter that resides in firmware.

Power supply: Indication of voltage needed to get minimum system running.

Price: Obviously this will change during the lifetime of this book — if not before it is published! However the prices quoted are approximately those for the minimum available system.

Software support: Summary of the main languages available and the general availability of programs is commented on.

ABC-80

The ABC-80 is made by Luxor in Sweden. It uses a Z80A processor at 1MHz. It is a keyboard based unit with attached video screen. The keyboard has 55 typewriter style keys and the display shows 24 x 40 characters and is viewdata/teletext compatible. The basic system comes with 16K RAM expandable to 40K and 16K BASIC and 2K monitor in ROM. It has a non-standard bus connector as well as RS232C and IEEE 488 interfaces. The system can support a dual floppy disk unit (2 x 160K). There is an internal speaker.

The system is well supported in Sweden, but little software is available in the UK.

Prices: from £738 + VAT

Acorn Atom

The Atom is made by Acorn Computers in Cambridge, England. It is based around the 6502 processor running at 1MHz. The Atom is a keyboard based unit with integral loudspeaker and a 57 key typewriter style keyboard. The minimum RAM available is 2K but this can be expanded to 12K. There are a number of plug-in ROMs, the basic system having an 8K ROM with Integer BASIC and monitor. The ROM capability is expandable up to 126K, including a floating point BASIC. The Atom is available ready-built, or in kit form. The BASIC uses 32-bit arithmetic, and there will be a plug-in ROM to give the new BBC-BASIC, as used on the BBC Microcomputer which is also made by Acorn.

The Atom can also be used with Acorn's Econet networking system, allowing up to 255 units to share a central floppy disk.

The video output is both 1Vpp composite, and UHF and gives a display of 48 x 64 characters, or up to 256 x 192 pixels. A colour generator is also available. A cassette interface, and two 8-bit input/output ports supplement a non-standard Acorn bus interface.

The Acorn Atom has a thriving user group, and is well supported by a number of software houses.

Prices: (2K) kit £120 + VAT
(2K) built £150 + VAT
(12K) kit £220 + VAT
(12K) built £250 + VAT

Acorn Proton

This system has been adapted for the BBC under the name BBC-Microcomputer. It is a keyboard based unit using the 6502 and is designed to be used as the centre of an expandable system.

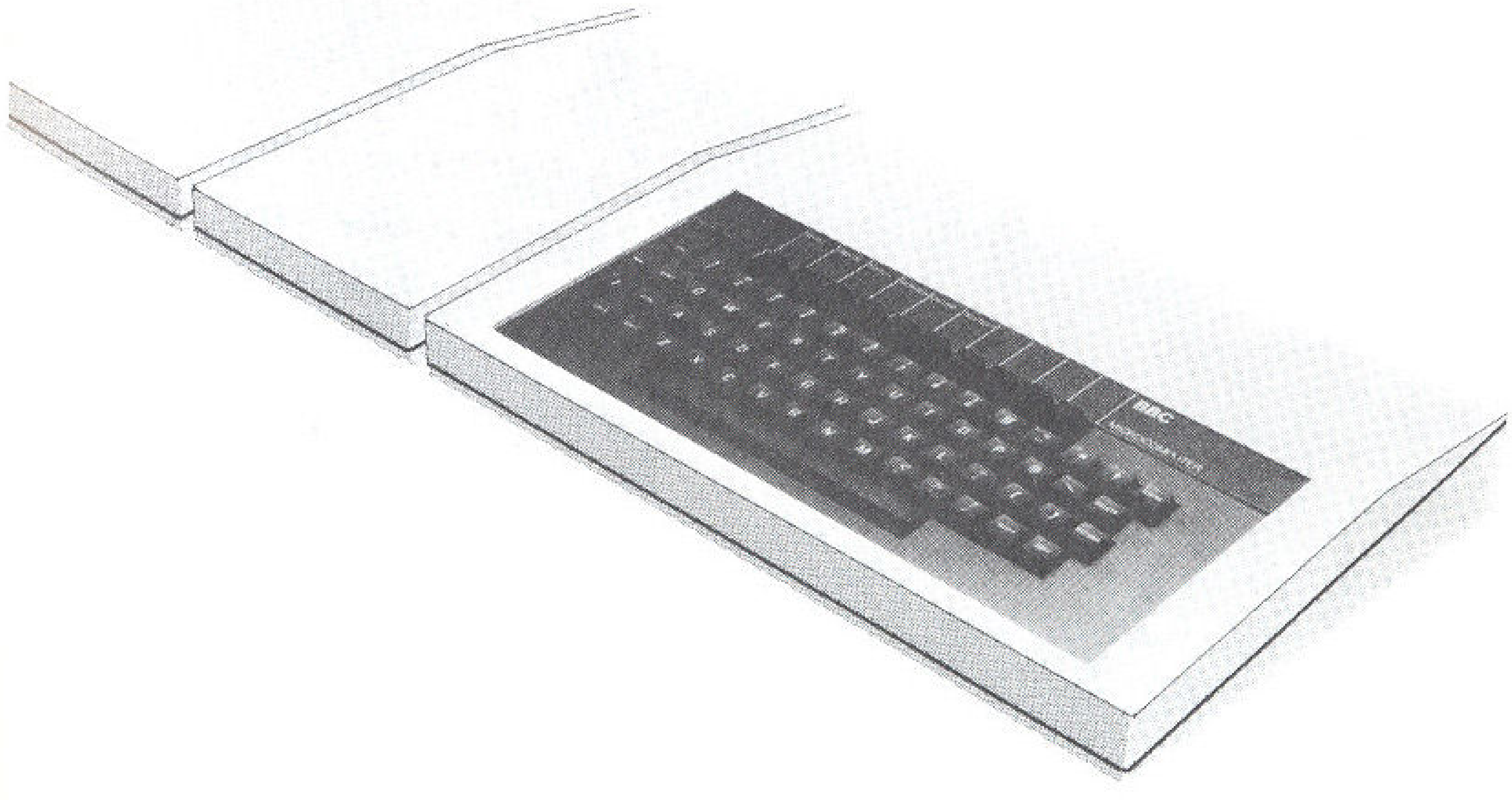
The basic system comes with 16K of RAM expandable on-board to 32K. The BASIC is a special "BBC" version and takes up 32K of ROM. This is expandable to 48K. Three colour video outputs are available — 1Vpp, UHF TV and colour and composite sync. RS232C, cassette and parallel interfaces are standard. It is also possible to use a floppy disk drive, joystick or paddle, and a processor interface that allows a second, Z80, processor unit. This allows the CP/M operating system to be used.

The screen format is 40 x 25 characters, teletext and viewdata compatible, with a maximum graphics format of 640 x 256 pixels, or 80 x 30 characters.

The unit has an integral loudspeaker with synthesiser on-board, and has 70 key typewriter style keyboard with ten user definable keys. It is also possible to interface the unit to the ECONET network.

The BBC are providing many programs in association with the tv series, and many user groups will be set up.

Prices: (16K) £235 + VAT
(32K) £335 + VAT
dual floppy disk (2 + 116 K) £445 + VAT
Disk interface £70 + VAT



AIM 65

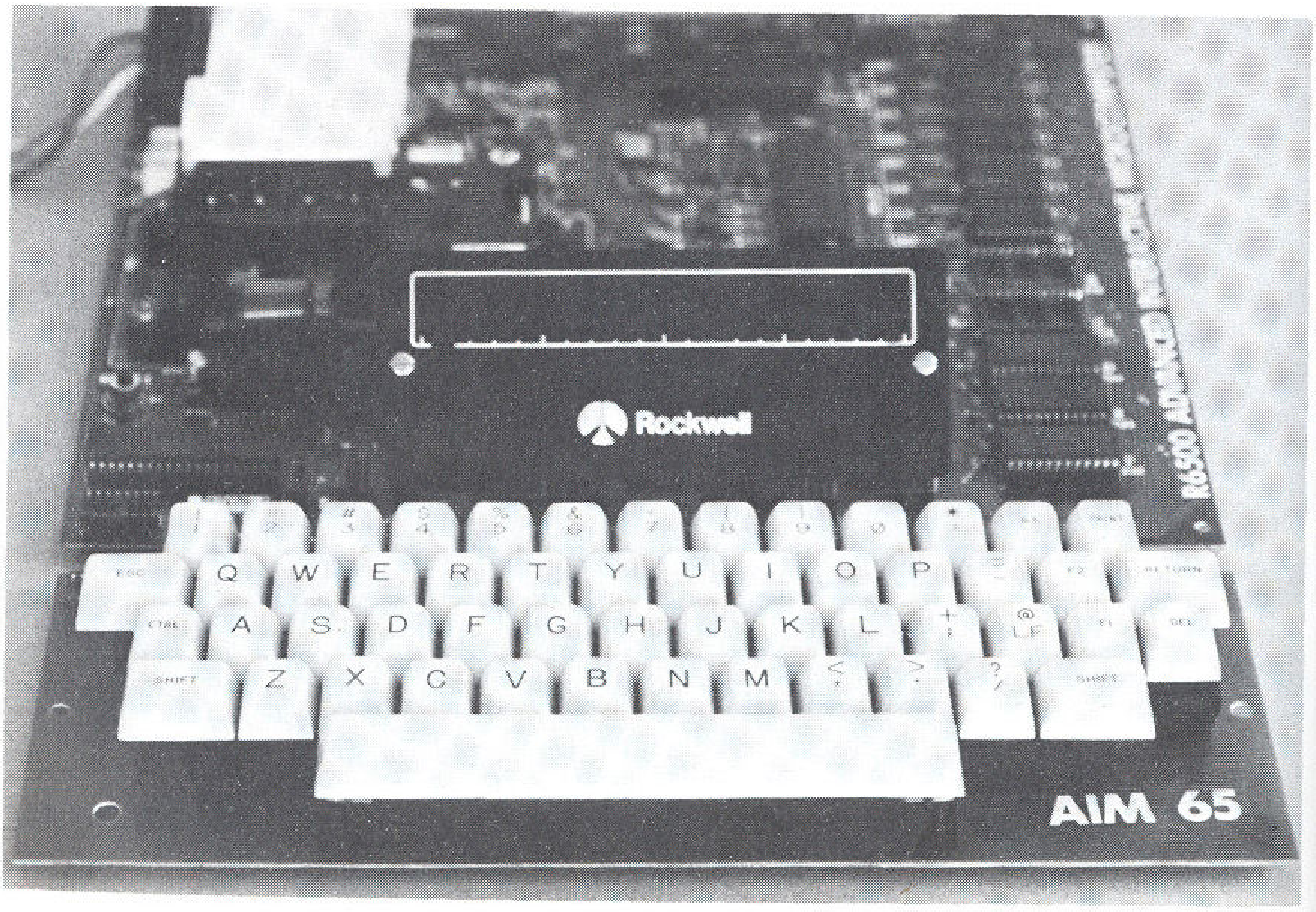
The AIM is a single-board computer made by Rockwell International in the US. It comes with a 54-key typewriter style keyboard, and uses a 20 character led display. It also has an integral 20 column dot matrix (5 x 7) thermal printer. It uses a 6502 microprocessor running at 1MHz and the minimum system has 1K of RAM and 8K of monitor ROM. It can be expanded to 4K of RAM and supports a 1K ROM-based BASIC (Microsoft) interpreter and also an 8K Assembler. The monitor allows full editing and mnemonic machine code entry.

The AIM-65 has two cassette interfaces — one to KIM standard, the other a faster AIM only standard. It also has two parallel input/output ports.

Disks are available from a number of suppliers, as are cases and memory expansion modules.

Rockwell have just announced an upgraded version, the AIM 65/40 with 64K of RAM on board, and two 6502 processors. It also has a 40 column character display and 40 column printer.

Price:	AIM 65 (1K)	£285 + VAT
	(4K)	£ 315 + VAT
	BASIC ROM	£70 + VAT
	Assembler	£59.50 + VAT
	AIM 65/40	£645 + VAT



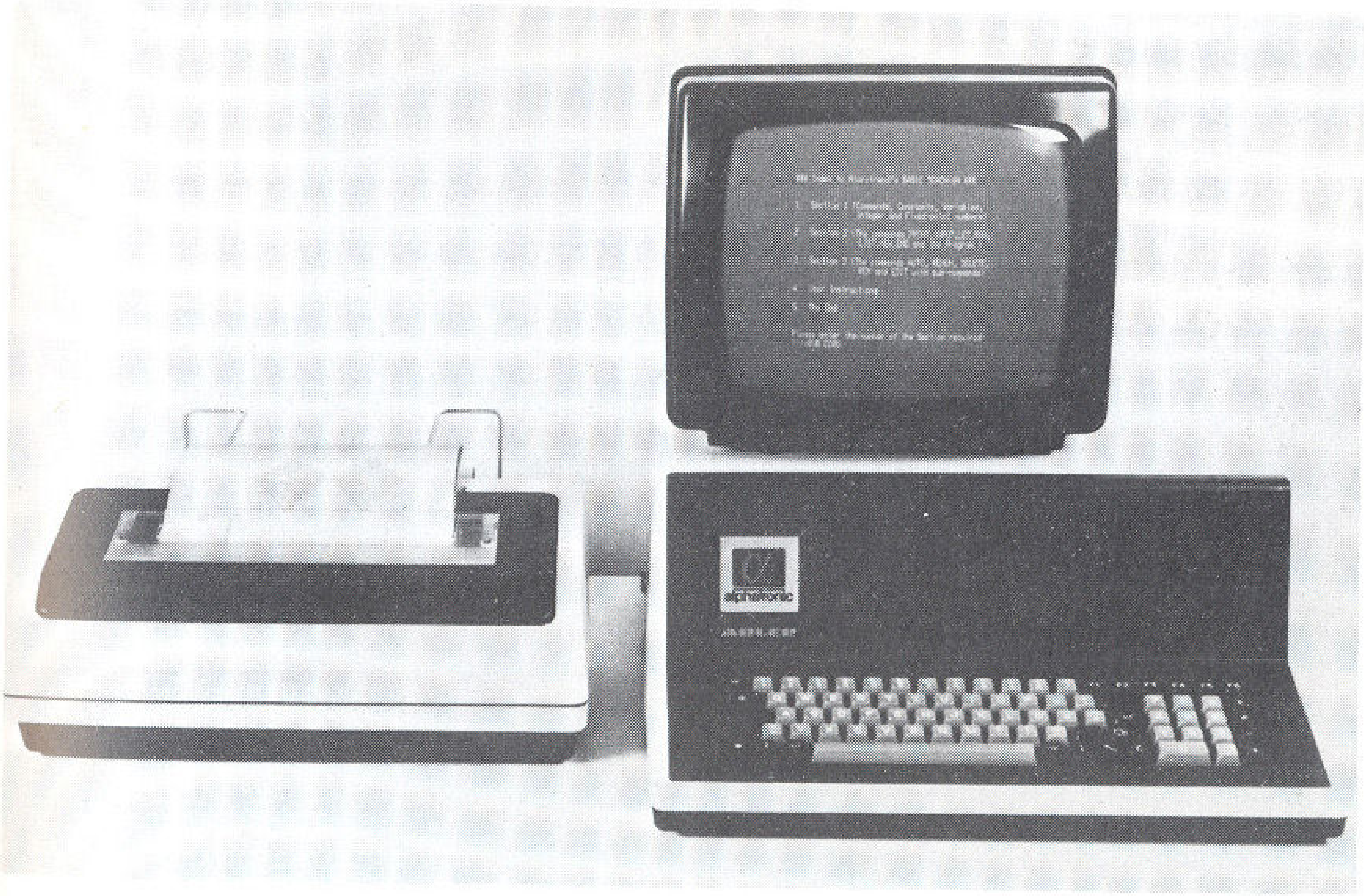
Alphatronic

The Alphatronic is Triumph-Adler's first attempt at the small business/personal computer market. They are hoping to get their system into offices that use other TA equipment like typewriters etc. The Alphatronic is made in West Germany and comes as a two-unit computer, based around the Z80 processor.

The basic unit has integral dual floppy disk drives (2 x 160K) and a 74 key typewriter style keyboard with numeric pad and 6 user definable function keys. The separate display unit has a 12" screen and displays 24 x 80 characters. The unit comes with 48K of user RAM and 12K ROM BASIC. Other languages available include a BASIC compiler, FORTRAN, Pascal and the CP/M operating system.

A number of systems houses have been writing business and commercial software for the machine, and, with its useful interfaces, RS232C and parallel, it seems a useful system for the office.

Prices: from £1550 + VAT



Apple II

The Apple II was one of the first personal computers to capture the imagination of the general, and investing, public. This machine dominates its home market in the US, but is behind the Commodore PET in the UK.

The latest version of the Apple II, as marketed in Europe, is designated the Eurapple Plus.

The system is based around the 6502 processor running at 1MHz. It is a keyboard-based unit with a 53-key typewriter style keyboard. The system can display colour graphics, 280 x 192 pixels, or 25 x 40 characters via a UHF TV interface. The basic system comes with 16K of RAM, expandable to 48K and a 12K ROM containing Microsoft BASIC. There is an integral loudspeaker with sound generator. There are a number of interface slots on the main board that allow RS232C and parallel I/O. Many boards are available giving the system great flexibility. Floppy disk drives (143K), Z80 processor card etc, allow CP/M to be run.

There are many suppliers of both hardware and software in the UK, and, within the limitations of the machine, it is possible to carry out most tasks required of a small computer system.

Prices: (16K) £599 + VAT
(48K) £659 + VAT
RS232C card £130 + VAT
printer card £104 + VAT
single disk drive with controller card £349 + VAT

Atari 400

The Atari 400 is the "baby brother" of the Atari 800, making two of America's best selling personal computer systems. The Atari 400 has a 57 key touch sensitive keyboard, with four sockets for plug-in ROM packs. The system gives a 40 x 24 character, or up to 320 x 192 pixels, colour display. It uses a 6502 processor at 1MHz. There is a built-in loudspeaker with four programmable sound generators. A number of user interfaces give four paddle ports, cassette and UHF TV output and RS23C. The basic unit comes with 16K of RAM, upgradeable to 32K, with 8K BASIC and 10K monitor in ROM.

A whole range of peripherals are available, including dual floppy disk drives (2 x 88K), cassette recorder and printer. The Atari system has a wide range of software — mostly games, but the ROM-pacs are being used with the disk drives to give some business software.

Prices: (16K)£340 + VAT
(32K) £395 + VAT
dual disk drive £325 + VAT
cassette recorder £45 + VAT

Atari 800

This system is essentially the same as the Model 400 — except for the keyboard. The Model 800 has a 57 key typewriter style keyboard — instead of a touch sensitive one, with four sockets for plug-in ROM packs. The system gives a 40 x 24 character, or up to 320 x 192 pixels, colour display. It uses a 6502 processor at 1MHz. There is a built-in loudspeaker with four programmable sound generators. A number of user interfaces give four paddle ports, cassette and UHF TV output and RS232C. The basic unit comes with 16K of RAM, upgradeable to 32K, with 8K BASIC and 10K monitor in ROM.

A whole range of peripherals are available, including dual floppy disk drives (2 x 88K), cassette recorder and printer. The Atari system has a wide range of software — mostly games, but the ROM-pacs are being used with the disk drives to give some business software.

Prices: (16K) £625 + VAT
dual floppy disk drive £325 + VAT
cassette recorder £45 + VAT

Black Box III

Rair is a British company based in London, and first made an impression on the UK with the Black Box computer some four years ago.

The lastest version is the Black Box III, and the basic system — the 3/10 — comes with integral single floppy disk drive (500K or 1M) 32K of RAM and two RS232C ports. The system uses the 8085A processor at 3MHz. The RAM can be extended to 64K. The system is based on a non-standard bus structure, the main unit being an 8-slot motherboard.

The input/output is expandable to 16 interfaces, both parallel and serial. $5\frac{1}{4}''$ and $8''$ hard disks (3.5M, 6.9M, 11.3M, 20.5M) are also available, and a multi-user, multi-computer system can be built up from the individual components.

The CP/M operating is used giving access to BASIC, FORTRAN, COBOL etc.

Prices: 3/10 (32K RAM, single floppy disk) £2000
3/20 (32K RAM, dual floppy disk) £2750

Casio FX-9000P

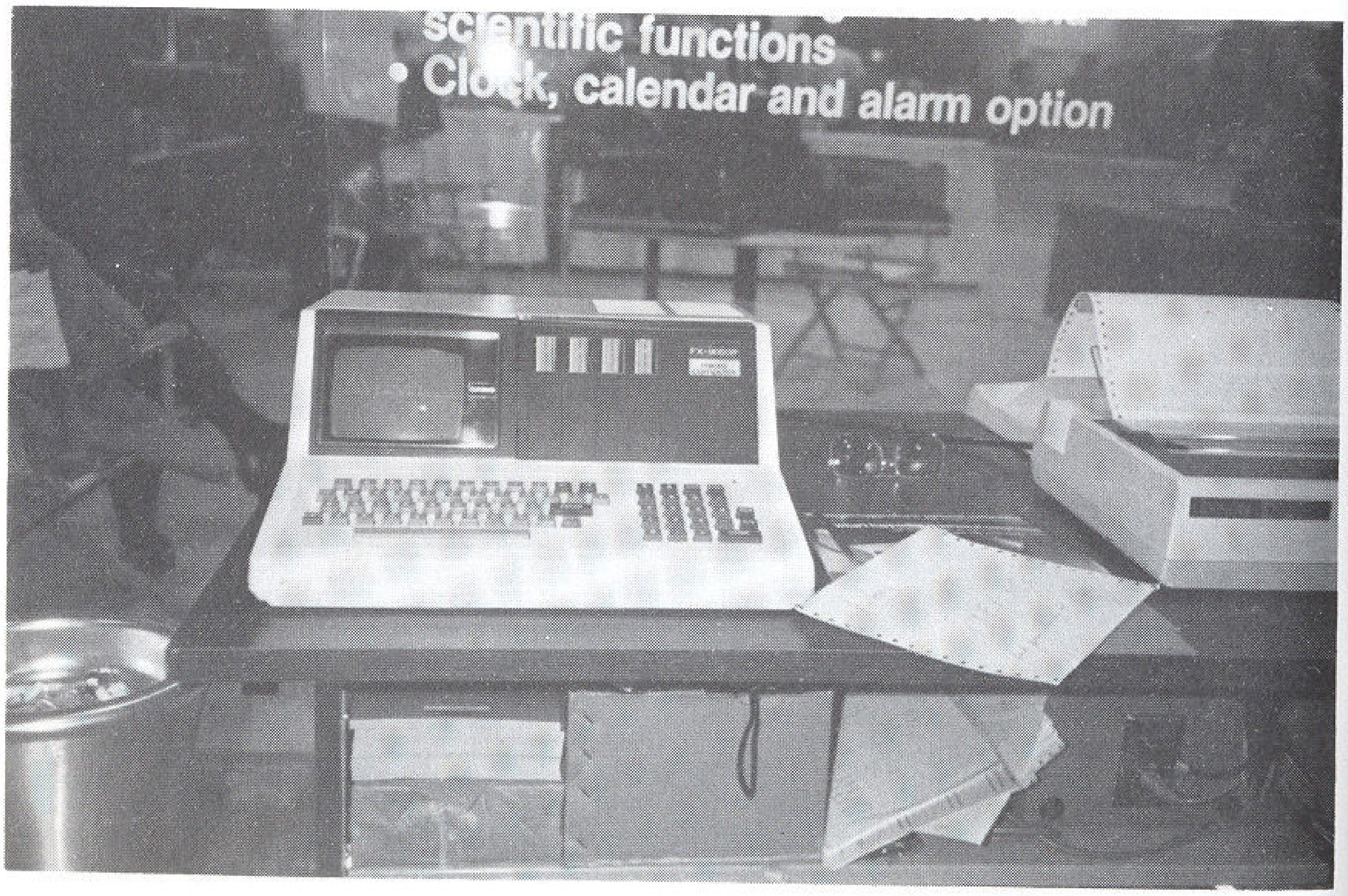
This is Casio's first small desk-top computer, although the company is well known in Japan for its minicomputer range.

The FX-9000P has 5.5" integral video display giving 32 x 16 characters, or 256 x 128 pixels. The 67 keyboard has small calculator style keys with a numeric keypad. The unit has four slots for plug-in RAM, or ROM, modules. The basic model comes with 4K RAM, and this is expandable to 32K. Two RAM modules are available. 16K RAM packs, which lose all information (dynamic) when removed, or 4K RAM packs with back-up batteries that store the information whilst outside the system.

A 12K BASIC interpreter has some graphics control commands. Extended BASIC is an optional extra.

Interface units are available that plug into the main housing and these give RS232C, printer, and floppy disk capability. The system is based around a Z80 processor at 2.75MHz.

Price: (4K) around £500 + VAT



CBM 8032

This system is the "big brother" of the ubiquitous PET, and is essentially an 80 column version of the same machine. As such, most of the comments found there are applicable here. The CBM is an integral system with a 12" video screen housed above the keyboard unit. The display has 25 + 80 characters and the unit comes with a 72 key typewriter style keyboard with a numeric key pad. It uses a 6502 processor at 1MHz and has 8K Microsoft BASIC in ROM. The user RAM is 32K. There are five interfaces — IEEE488, 2 x 8 bit parallel, and two cassette I/O ports.

A whole range of Commodore, and compatible, products are available including floppy disk drives, printers, a - d and d - a converters etc. The CBM probably has more software and peripherals than any other small computer system.

There are many active user groups, and over 250 retailers, around the country. It is now possible to run the CP/M operating system with a special interface unit.

The 8032 is ideally suited for word processing applications and it is possible to expand the RAM to 96K for modelling and other applications.

Prices: 8032 £755 + VAT
dual floppy disks (2 x 180K) £585 + VAT
dual floppy disks (2 x 500K) £755 + VAT

Comart Communicator

Comart were the computer system group that took over the Byte Shop/Computerland chain when it had financial difficulties in the late 70s. They are a large company distributing North Star, and similar equipment — and the Communicator is their first effort at a British-made system.

The Communicator is an S100 bus system based around a main chassis with a ten slot mother board. The basic system has 64K of RAM and uses the Z80A processor. The system comes with three variations of dual floppy disk drive — 2 x 190K, 2 x 390K and 2 x 790K. There are two RS232C interfaces with additional interface boards available.

Other S100 cards available give Viewdata/Prestel capability, 18.7 MB hard disk; with 13.4 MB cartridge back-up.

The system runs the CP/M operating system. A smaller system called the Educator, with networking facilities, is also available.

Prices from: £2295 + VAT

Cromemco System 2

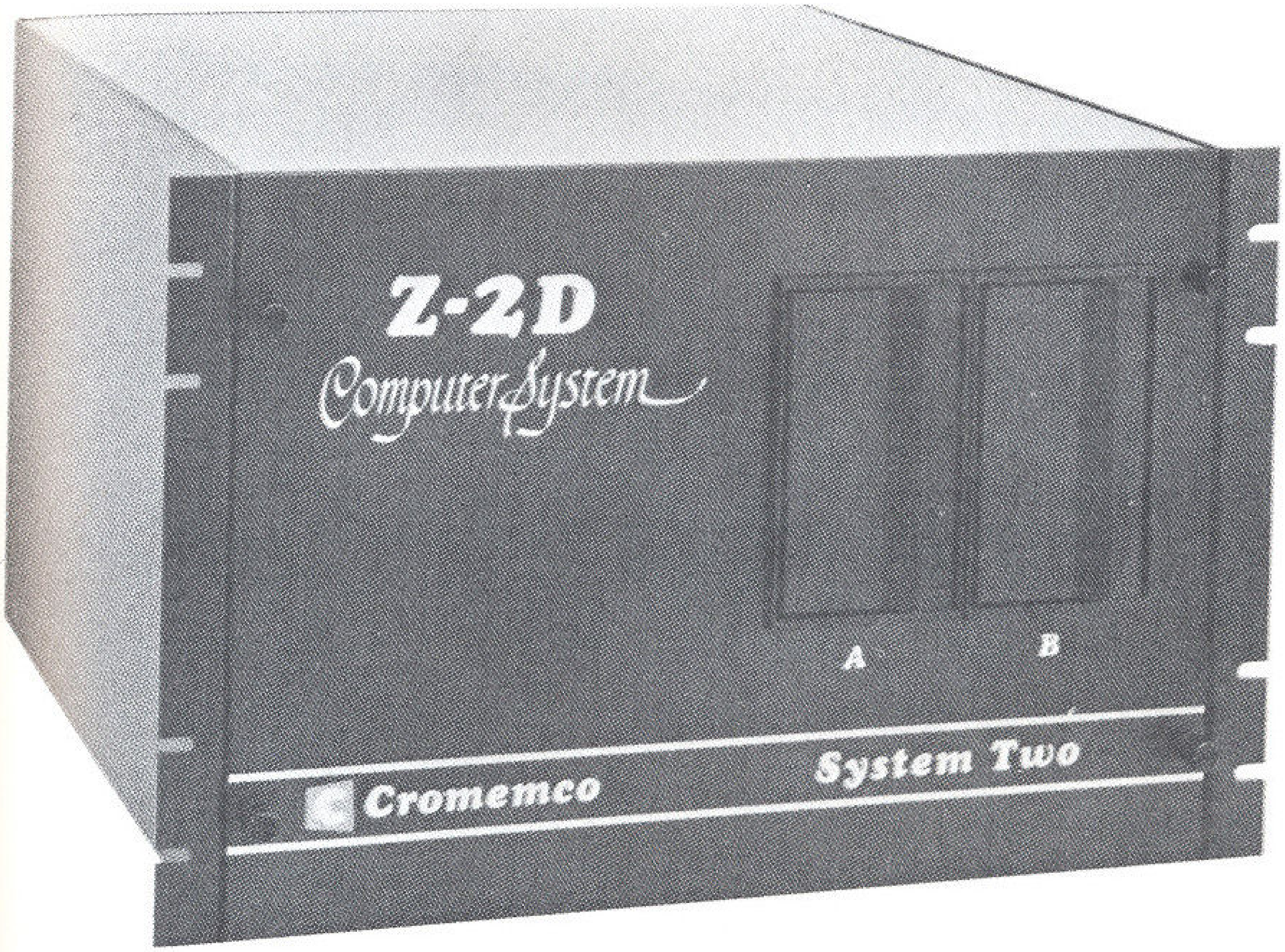
Cromemco make a whole range of systems, the System 2 being one of the most popular.

The basic Z-2 contains a 4MHz Z-80 processor and has a 21-slot S-100 mother board. The Z-2D system takes the basic CPU board and supplements it with 64K of RAM and integral dual floppy disk drives (2 x 184K). There is 1K of monitor ROM, an RS232C interface and a parallel printer interface.

Expansion options include 8″ floppy drives (512K), hard disks (11MB or 22MB), and high resolution graphics.

The system runs CP/M.

Price: Z-2D £2100 + VAT



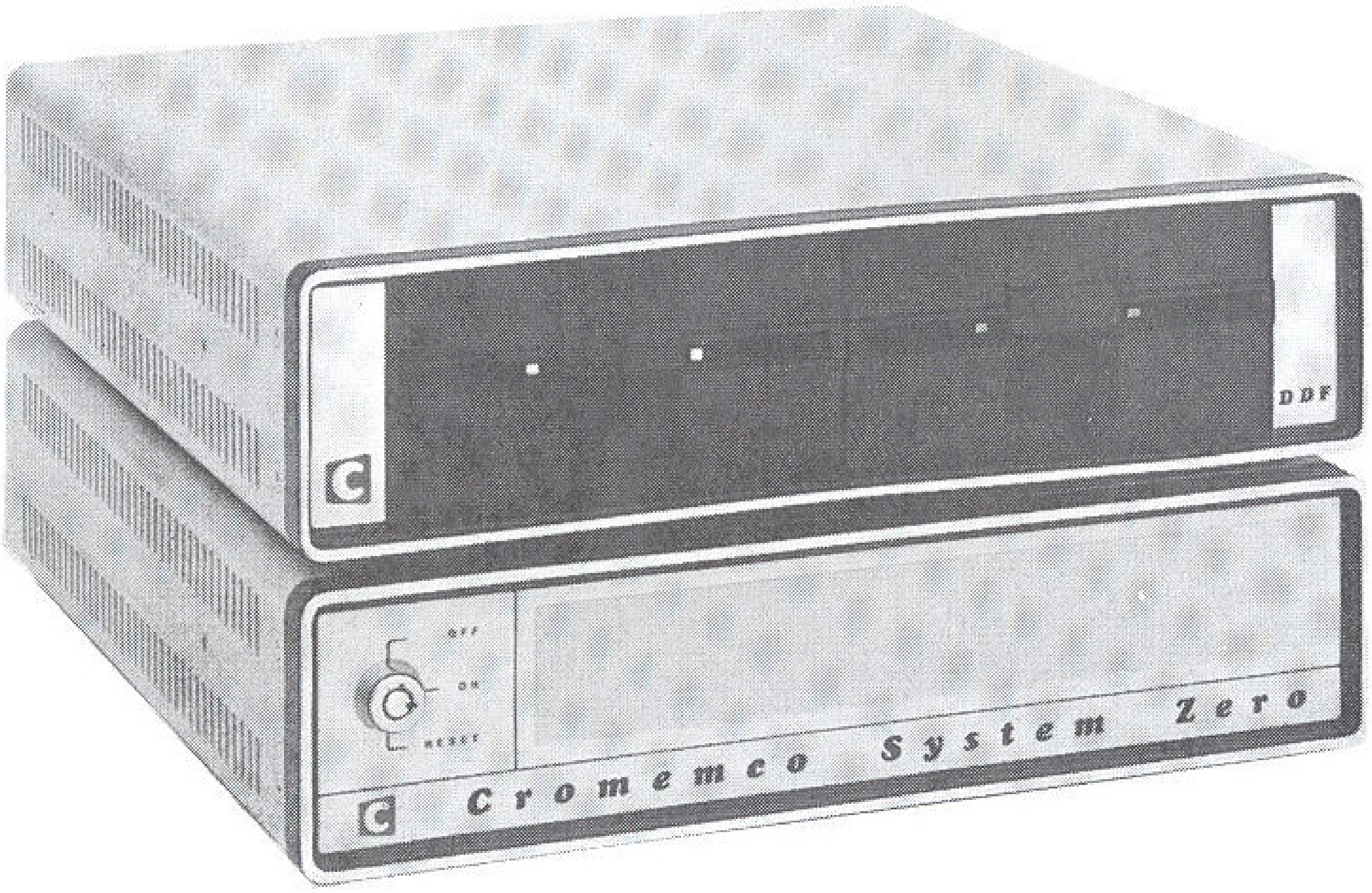
Cromemco System Zero

Cromemco were one of the first American producers of equipment based on the S-100 bus. The System Zero is the latest, and lowest cost, system from this company.

The basic unit is a six-slot S-100 board, with core and power supply. The smallest system with a 4MHz, Z80A board with 1K RAM and 4K ROM monitor. The Cromemco Z80A board gives RS232C and parallel interfaces. The bigger system, designated Zero/D has 64K of RAM, and dual floppy disk drives (2 x 390K). It can user the CP/M operating system. It also has printer drivers and self test diagnostics.

It is also possible to use a version of UNIX, called Cromix if a further 64K of RAM is available.

Prices: System O £587 + VAT
System O/D (64K RAM 2 x 390K disk drives) £2355 + VAT



DAI PC-1

The PC-1 is made by Data Applications in Belgium. It is based around the 8080 processor, and is a keyboard based unit with colour graphics and fully synthesised sound. It has a 57-key typewriter style keyboard and the UHF TV output gives 60 x 24 characters or up to 335 x 255 pixels, in 16 colours. There are two cassettes and one RS232C interface, as well as a non-standard bus expansion connector, used for industrial control and floppy disk units. There are also three audio outputs and two paddle control ports. The unit is available for PAL, SECAM or NTSC colour systems.

The unit is sold in the UK with 48K of RAM and has a 24K BASIC interpreter, with full colour and sound commands. It also has a very flexible editor and machine language facilities. It is possible to use a hardwired 'Maths' chip that speeds maths processing by a factor of ten.

There are many enthusiastic users with a good, Belgian based, user group.

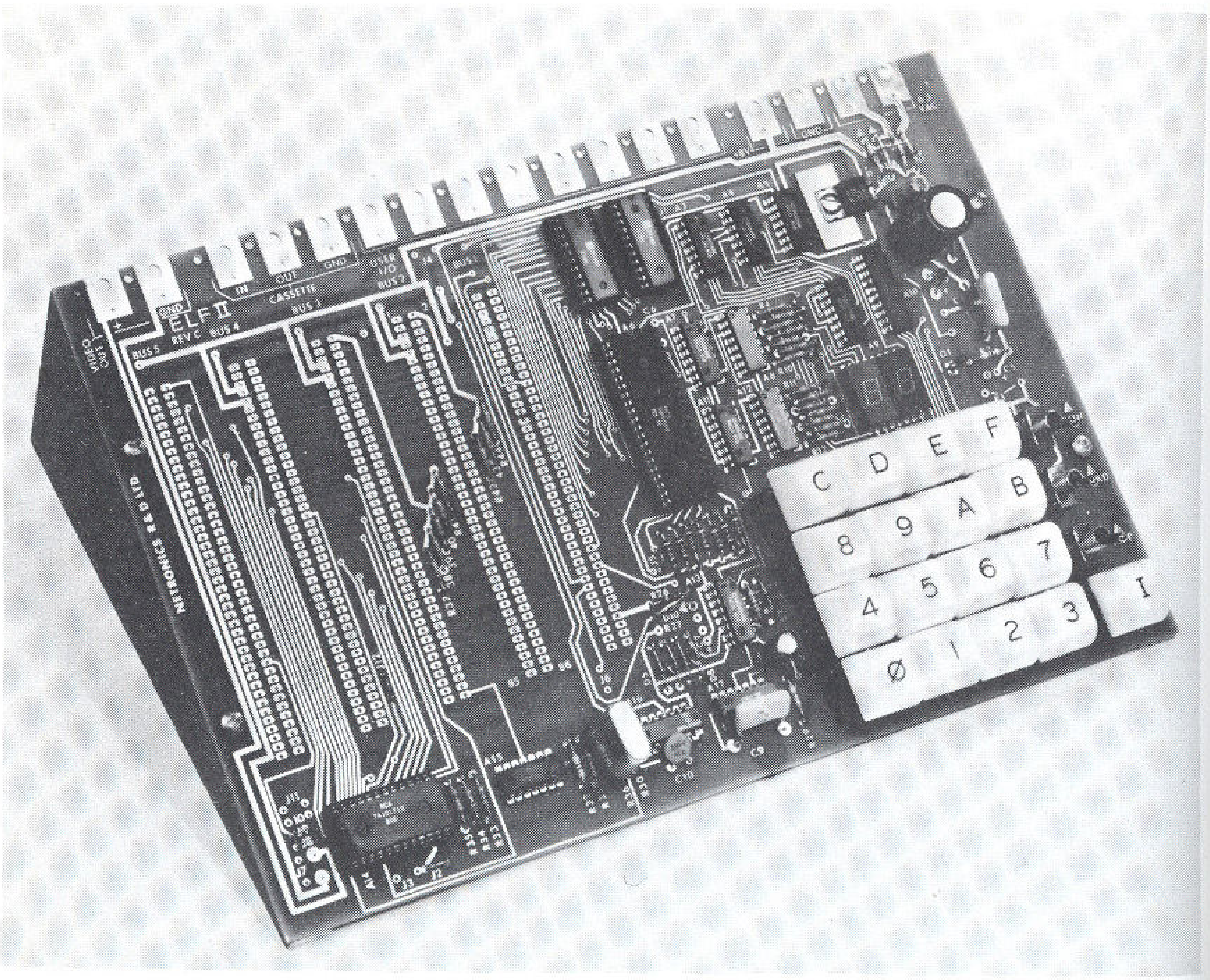
CP/M is available as an operating system.

Prices: (48K) £595 + VAT
dual floppy disk (2 x 170K) £595 + VAT

Elf 2

The Elf 2 is manufactured by Netronics in the USA. It is a single board computer based around the 1802 processor running at 1MHz. It comes as a kit and has a 16-key hexadecimal keyboard for machine code entry. The main board has 5 slots for expansion boards, which include memory, graphics, BASIC etc. A UHF TV output gives a 16 by 16 character display, and RS232 and cassette interfaces are available. One plug-in board will allow use of an alphanumeric keyboard. The minimum RAM configuration is 256 bytes expandable to 64K. The kit comes with a 1K monitor ROM. There are a number of local user groups.

Prices: from £49.95 + VAT

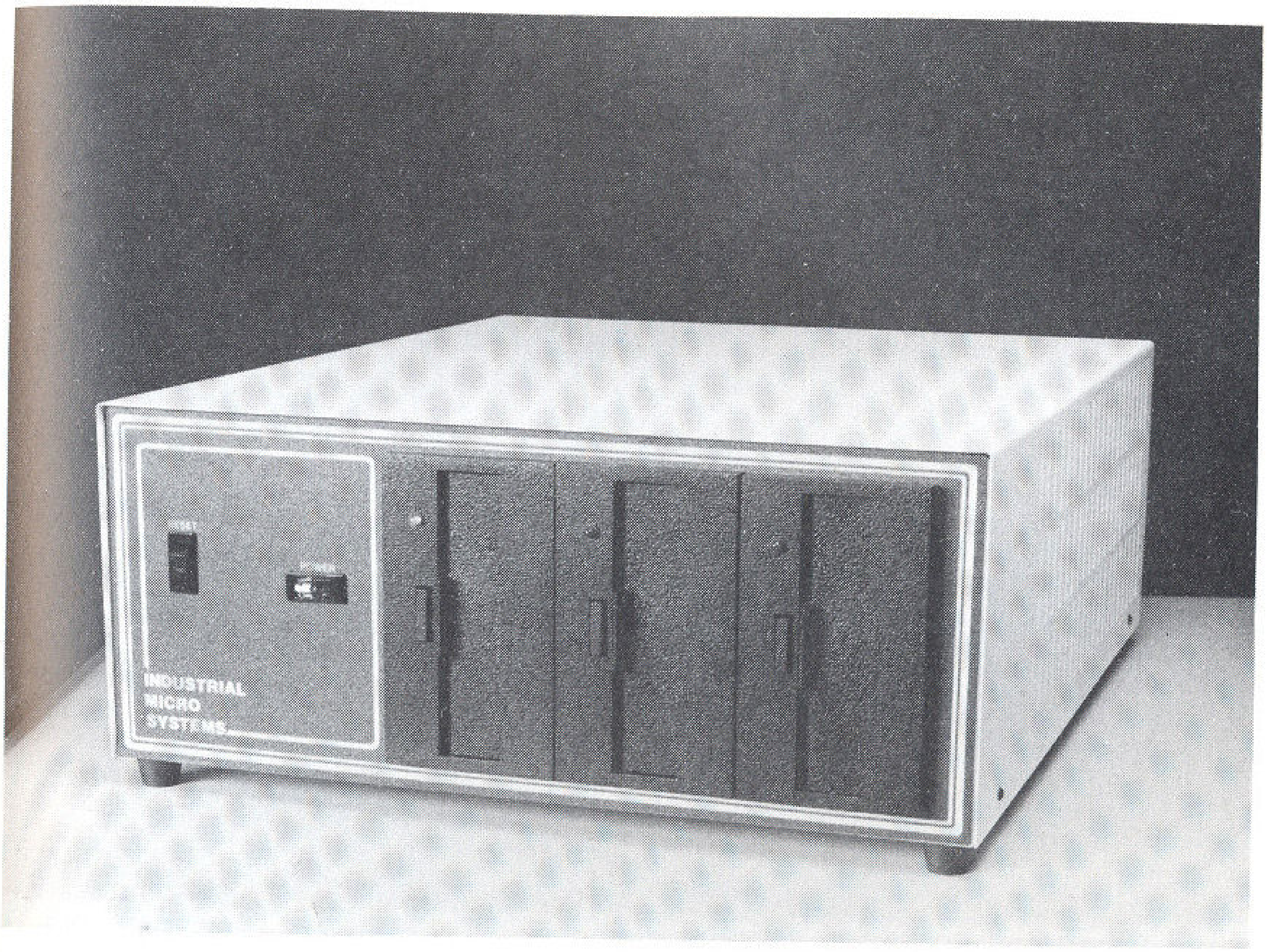


Equinox 5000

The Equinox 5000 system is based around a unit made by Industrial Microsystems in America. It is an S100 bus unit with a 12 slot motherboard. The basic system comes with a 4MHz Z80A processor and 64K of RAM. It also has integral dual floppy disk drives (2 x 170K). The system has two RS232C ports and one parallel port.

The Equinox 5000 system uses the CP/M operating system, and can also support 8" floppy disk drives.

Prices: 2 x 170K drives £1995 + VAT
2 x 340K drives £2232 + VAT



Exidy Sorcerer

The Sorcerer is made by Exidy Corporation in the USA and is a keyboard based unit using the Z80 processor. It comes with 48K of user RAM and a 4K monitor ROM. The keyboard is 79-key typewriter style with numeric keypad. It has RS232C, parallel and cassette interfaces. The video output is both 1Vpp and UHF. The screen can display 30 x 64 characters or 512 x 240 pixels. The Sorcerer uses plug-in ROM pac cartridges to supply the language used, and comes with a Microsoft BASIC.

The system can be expanded with a six card S-100 bus expansion unit. This allows floppy disks (315K) and S-100 boards to be used. 8″ floppy disk drives (1.2MB) and 10M or 20M hard disks can also be added.

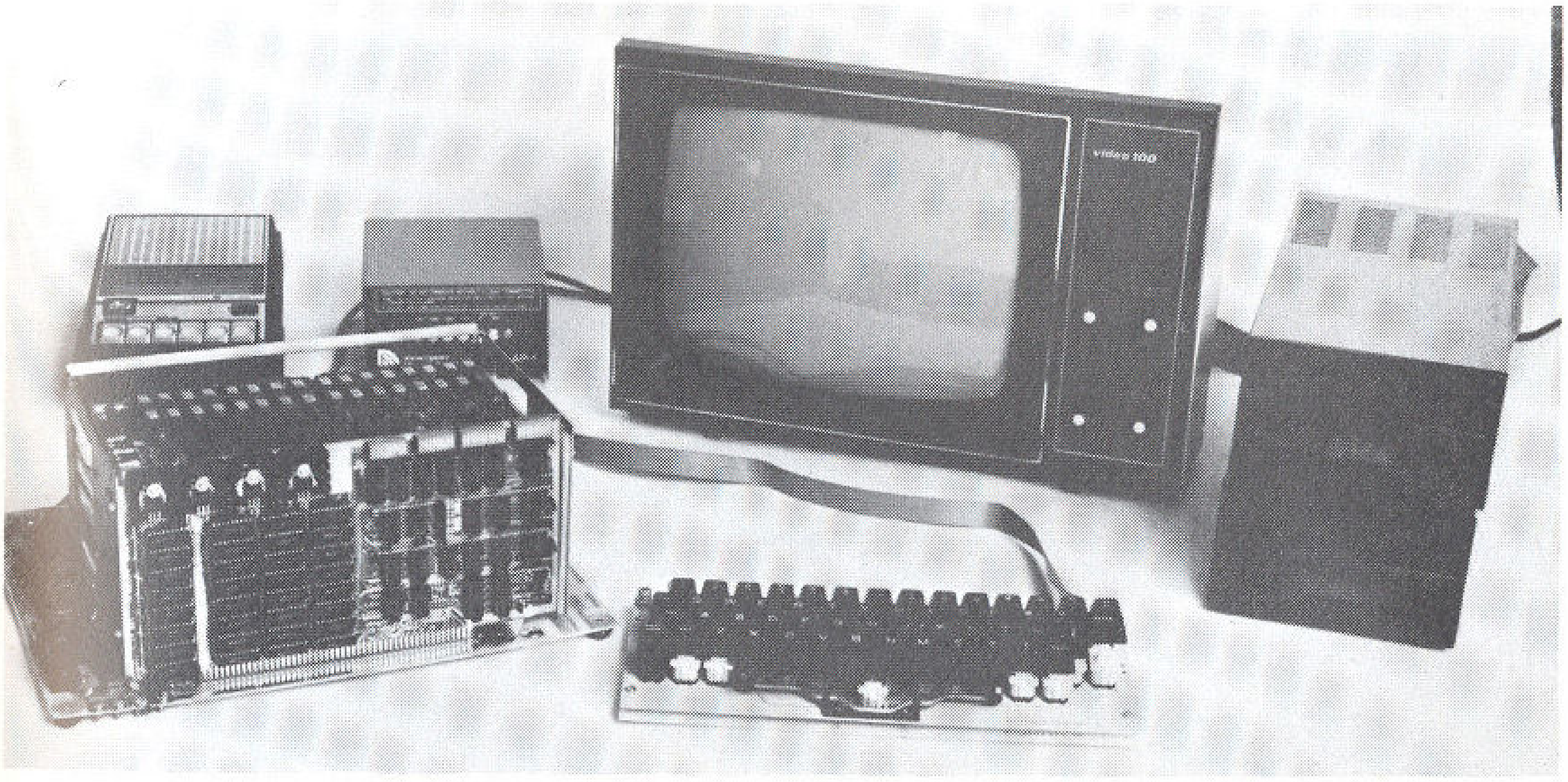
There is a lot of software available, and many ROM pacs, e.g. word processing. With the expansion unit operating, systems like CP/M can be used.

Price: (48K) £695 + VAT

Explorer 85

Made in the US by Netronics, the Explorer 85 uses an 8085 processor at 3.1MHz. It is a kit system based around the S100 bus. The initial system has 4K of RAM, expandable to 64K, and a 2K monitor ROM. A 54 key typewriter keyboard can be interfaced, the UHF video output giving a 32 or 64 by 16 character display. RS232C, 20mA current loop and four eight bit and one six bit parallel interfaces are available. An 8K BASIC (Microsoft) ROM is optional. There is some software support among user groups.

Prices: (4K) kit £327 + VAT
(4K) built £402 + VAT
(16K) kit £140 + VAT
(16K) built £485 + VAT
case £100 + VAT



Gemini

The Gemini is a new British computer made by the people who initially set up Nascom. In fact it owes a great deal to some of the later ideas floated around the Nascom 2. Although available as a series of single boards and plug-in modules for the "kit" enthusiast, it will basically be sold as a complete unit with integral 59 key typewriter style keyboard and dual floppy disk drives (2 x 315K). 64K of user RAM is provided as standard and the computer comes with two Z80A processors — one driving the video and the other doing the number crunching — at 4MHz.

Software provided includes a disk with the CP/M operating system and 24K Microsoft BASIC interpreter. The video output will give 1Vpp video with 80 x 25 or 40 x 25 characters. Interfaces available include two eight-bit parallel, RS232C cassette, and light pen.

Price: £1195 + VAT

HP 83/85

Hewlett-Packard are one of America's biggest electronics companies, and recent diversification into computers has made them one of the largest manufacturers in this area as well. The HP85 is a merging of their well-known calculator technology with their computer expertise.

The HP 85 system is basically aimed at more engineering-type applications. It uses HP's own processor and is a desk top unit having integral keyboard, display, printer and tape cartridge.

The 5″ screen displays 16 x 32 characters, or 256 x 192 pixels. The 93 typewriter style keys include numeric keypad and user definable keys. The thermal printer has a 32 column dot matrix output and can reproduce screen graphics. The mini tape cartridge has fast search facilities and can store 195K.

The basic system comes with 16K of RAM and a 2K monitor ROM. There is a plug-in ROM pack facility that not only allows more RAM to be used (up to 32K) but also high level languages. IEEE488, RS232C and 2 parallel ports are available.

The HP83 is a more orientated version without the printer and cartridge facilities. It is designed to operate with the Epsom MX80 printer and uses HP's floppy disk drive. Visicalc, and many other similar programs are available.

Prices: HP85 (16K) £1830 + VAT
HP83 (16K) £1250 + VAT

IBM Personal Computer

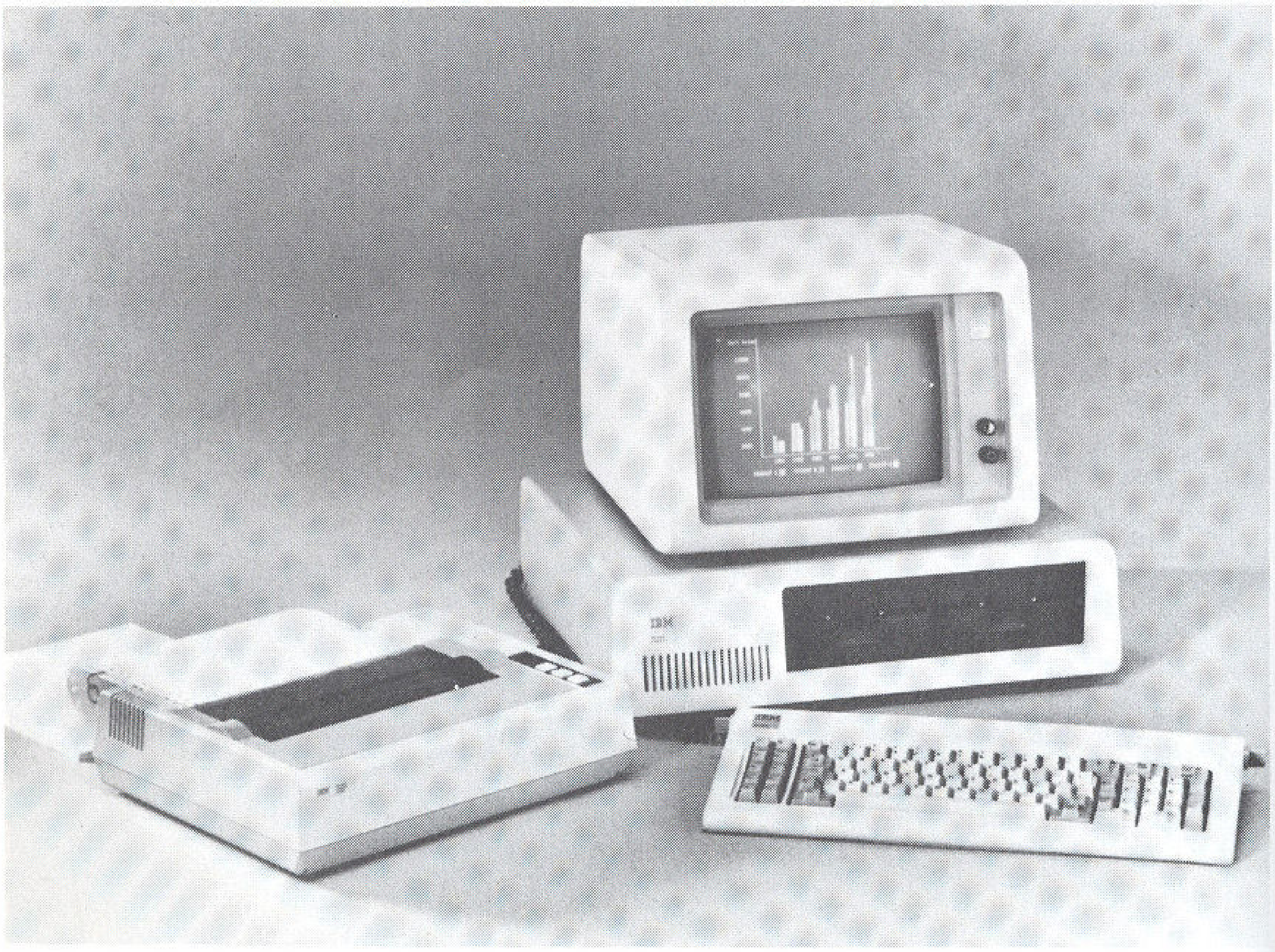
The IBM Personal Computer represents IBM's first entry into the consumer market and is being watched with interest by both small and large manufacturers alike. The unit is a keyboard based system with 83 keys and is attached to the main system by 6 feet of coiled cable. This unit houses a 40K ROM and between 16K and 256K of user RAM. Single or dual $5\frac{1}{4}''$ floppy disk drives can be housed in this unit, each drive giving 160K of memory.

The computer is based around the 8086 16 bit processor and displays 25 by 80 characters on a TV set. Underlining, dual intensity and reverse characters are possible, with 16 colours available in graphics mode.

Interface include ports for games paddles and a printer. RS232C is also available. There is a built in speaker.

BASIC is supplied in ROM with a version of CP/M and USCD Pascal available.

Price: around £1000



IDS Oscar

This British made system comes from Interactive Data Systems of Milton Keynes, Bucks. The system is based around a six-slot S100 mother board housed inside a VDU housing. It uses a Z80A processor at 4MHz and comes with 64K of RAM. The minimal system comes with dual floppy disk drives (2 x 400K) but can be used with an 11Mb hard disk. It comes with the CP/M operating system and 12" video screen displays 80 x 24 characters. The keyboard is separate from the display unit and has 72 typewriter style keys including a numeric keypad.

Price: typical system. (64K RAM, 2 x 400K disk drive) £2495 + VAT

Ithaca DPS 1

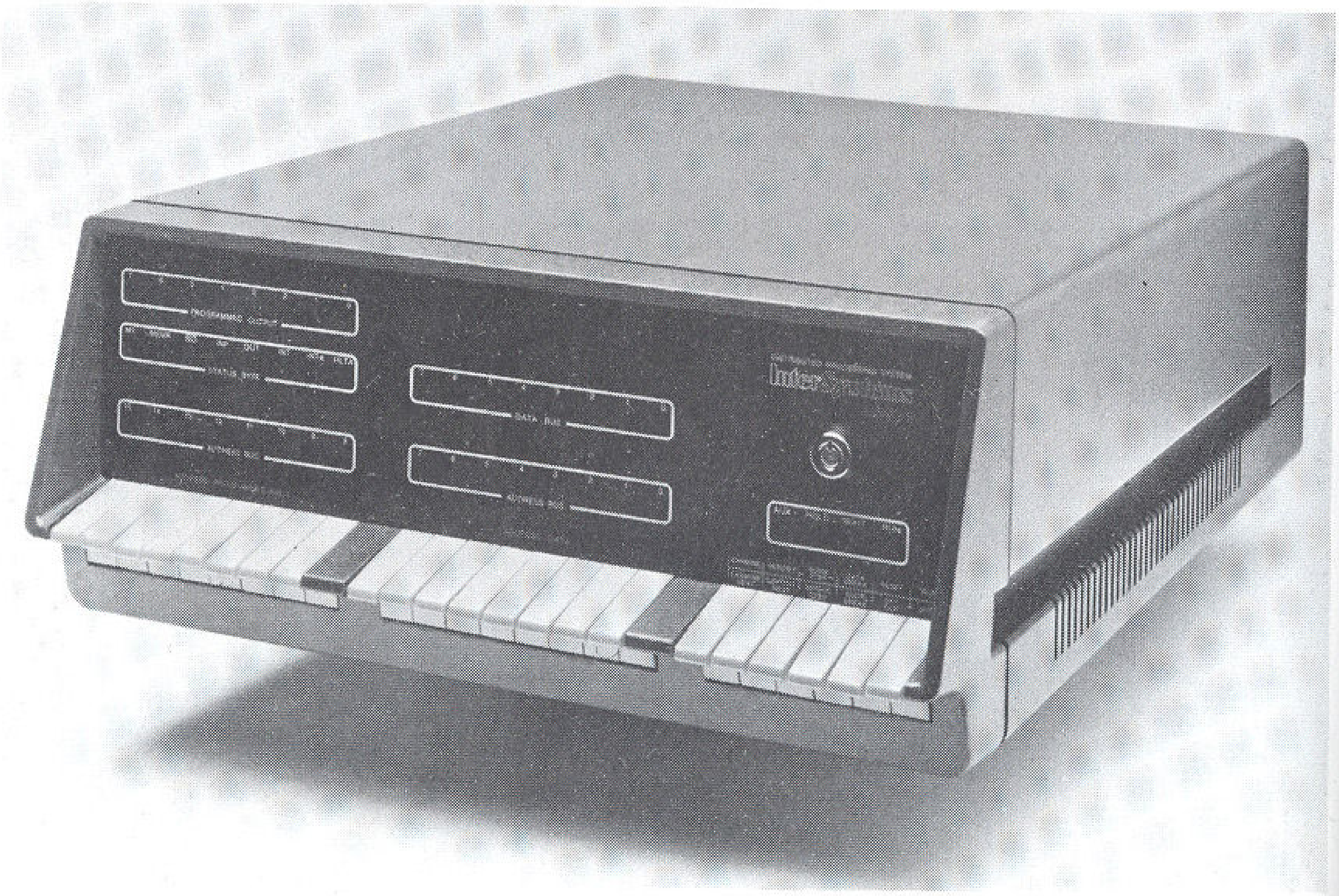
The Ithaca system is considered to be the "Rolls Royce" of S-100-based computers, and has been aimed at the engineering development market. There has been a move during the past year to come out with a version that did not have the engineer-orientated front panel, and present the computer as a business system.

The main unit has a 20-slot S100 bus motherboard which can support 16-bit, as well as 8-bit, processors. The front panel is used to control, or override, the operation of the processor. The minimum system requires 8K of RAM, but like similar systems it is unusual to have less than 64K. There is also a 2K monitor ROM.

The system supports dual 8″ floppy disk drives (2 x 600K) and a hard disk (20M). There are two serial (RS232C) and one parallel port.

With a 16-bit (Z8000) processor the system can take up to 256K of RAM and run the UNIX operating system, as well as the more usual CP/M.

Prices: typical system (64K, 2 x 600K disks, printer, VDU) £3950 + VAT



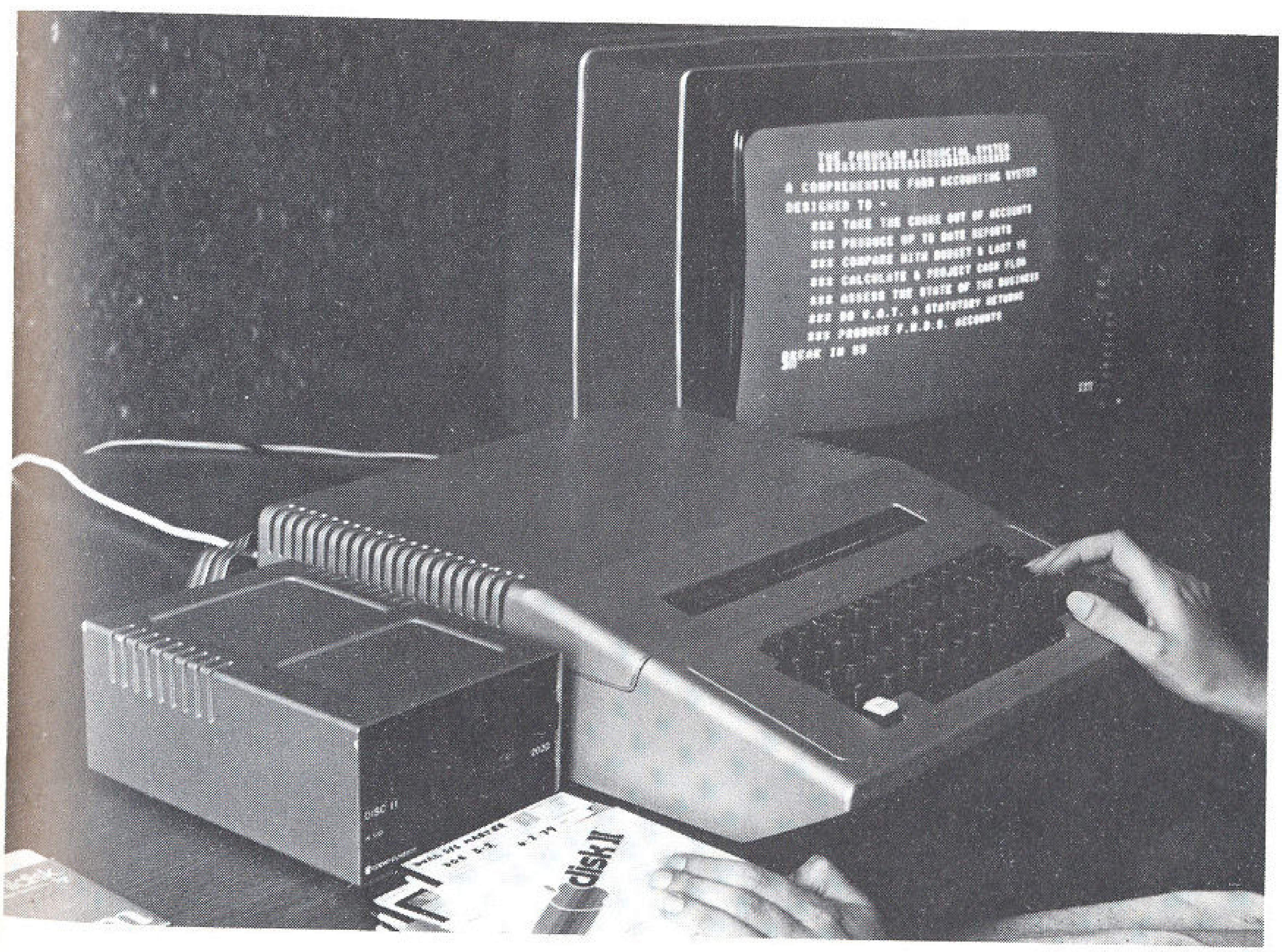
ITT 2020

The ITT 2020 is a version of the Apple II, and is assembled in the UK by ITT Electronics. It is based around the 6502 processor at 1MHz and, to all intents, is a direct copy of the Apple. However, it has been tailored to the European market, and is preferred by some over the original.

Like the Apple II, the ITT 2020 is a keyboard based unit having a key typewriter-style keyboard. The main unit has an eight-slot I/O section that allows a whole range of interface cards to be used. These include dual floppy disk (2 x 116K) printers and video. The standard video output is UHF TV and colour graphics are available — 40 x 24 characters or 28 x 192 pixels. Games paddles can also be used.

It is possible to plug in a second, Z80, processor that allows the CP/M disk operating system to be used. The basic system comes with 16K of RAM upgradable to 48K. Hard disk storage is also possible, and the ITT/Apple can be used on a networked basis. There is a lot of software, many dealers and active user groups.

Prices: (48K) £670 + VAT
floppy disks (2 x 116K) £425 + VAT



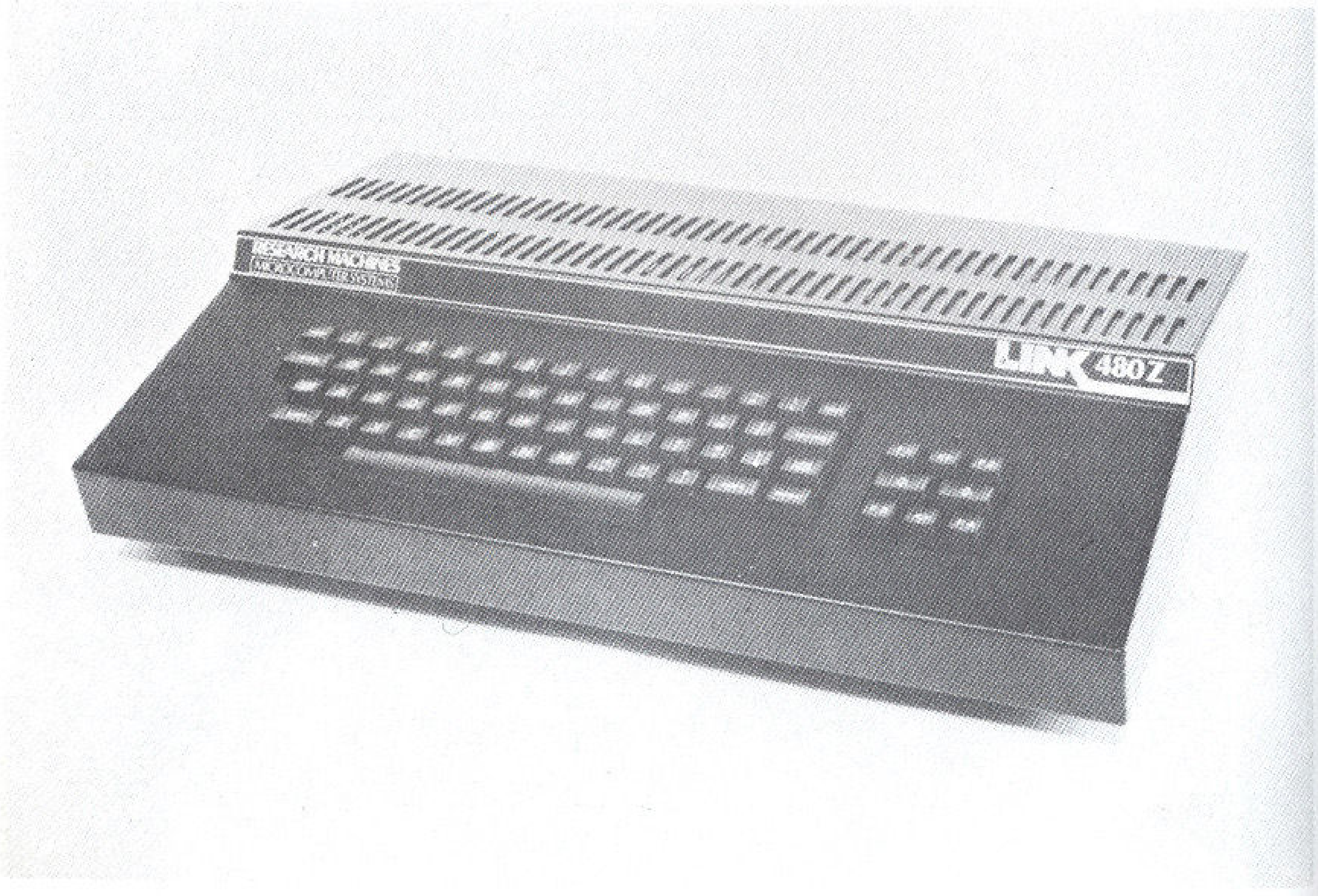
Link 480Z

The 480Z is the baby brother of the 380Z. It runs as a cassette-based, Z80A system and has a minimum 32K of RAM on 8K of ROM-based monitor. The system is keyboard-based and has 65 typewriter-style keys with four cursor-control and four user-definable keys. There are two display modes, 80 by 24 using 1Vpp video, and 40 x 24 using UHF TV output.

Two graphics modes are available, 160 x 72 single-tone or 80 x 72 two-tone. Two cassette interface standards are available and the unit has an integral loudspeaker. One eight bit parallel and one RS32C ports are standard, with a joystick or pushbutton and analogue inputs and output. The ROM can be upgraded to 16K, allowing 16K BASIC in ROM to be used.

Expansion options allow 640 x 192 monochrome, 320 x 192 — four colour — and 160 x 95 — eight colour-graphics. The RAM can be expanded to 64K and an IEEE 488 interface is being developed. Research Machines is also working on a memory expansion module allowing 256K of RAM and a network capability. The 480Z is upwards compatible with the 380Z.

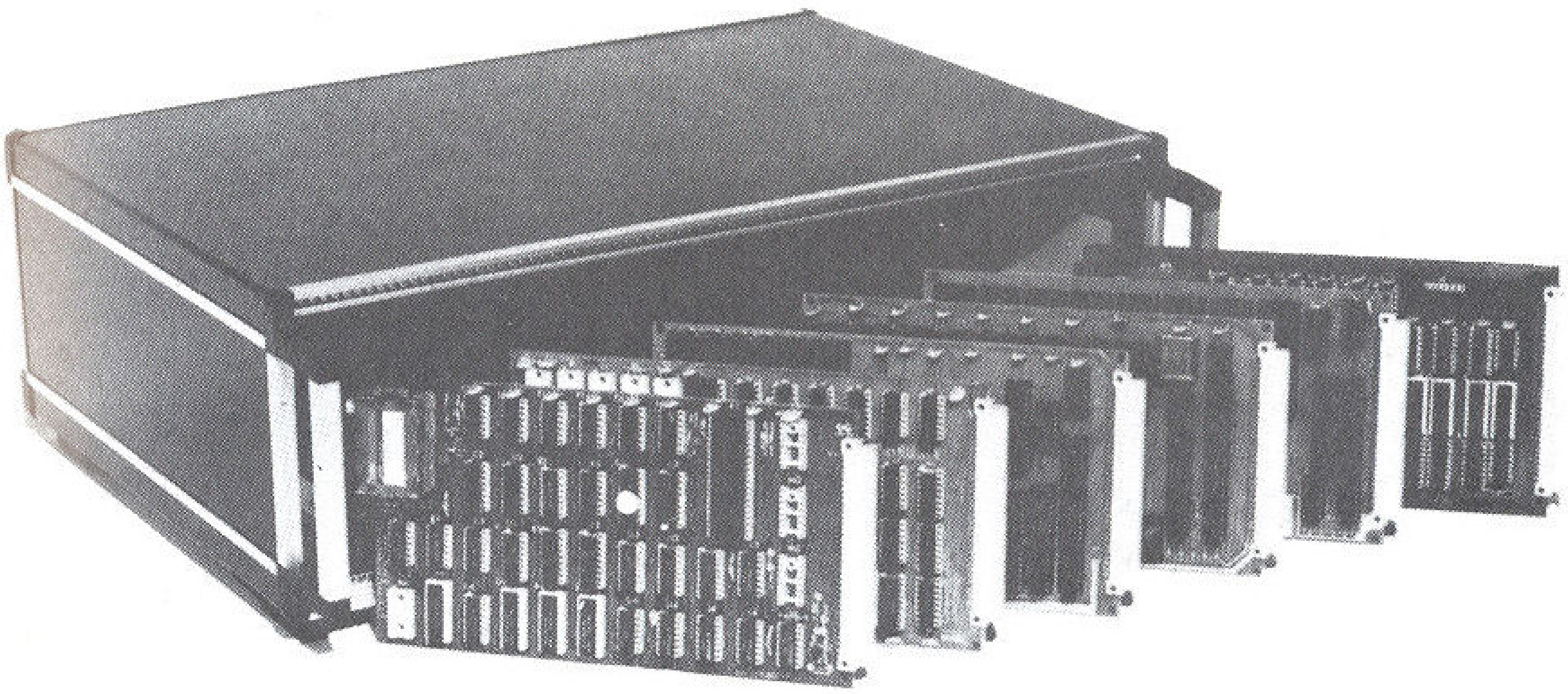
Price: (32K) £550 + VAT



Microtan-65

Manufactured by Tangerine Computer Systems, Ely, Cambridge. The Microtan-65 is a single board computer available in either kit form or ready built. It is based around the 6502 processor running at 1MHz. The basic unit comes with 1K of RAM, expandable to 8K on board. There are a number of input/output ports and a hexadecimal keyboard is optional. The Microtan-65 is essentially a single board control system that can be built up, in a rack, to quite a sophisticated computer system. It can support up to 277K of RAM, and a number of other modules give 10K Microsoft BASIC, RS232C/20mA, a display of 15 x 32 characters, and alphanumeric keyboard. The basic board has a 1K monitor ROM, and other modules give disassemblers etc. Most software support comes from user groups around the country.

Prices: from £69 + VAT



MSI 6800

The MSI system is made by Mid West Scientific Instruments in the United States. It is a bus structured system based on the SS50 bus developed in the early 1970s for the 6800 processor by South West Technical Products — SWTPc.

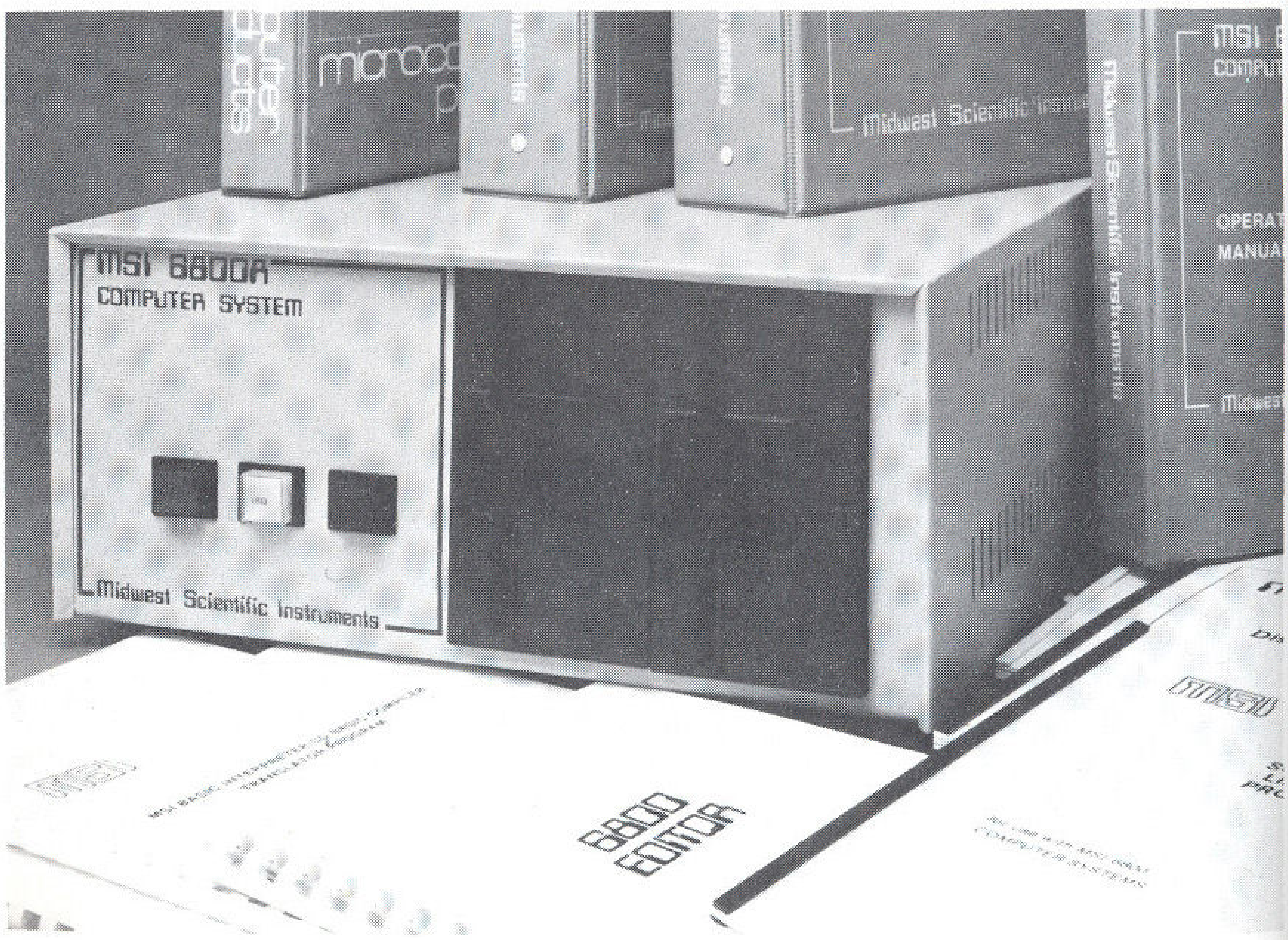
The MSI-6800 uses the 6800 processor at 2MHz. The main system has a 16 slot mother board and is available ready-built, or in kit form for some parts.

The system can support dual floppy disk drives (2 x 160K) which can be integrated into the main unit. It can also support a 76M hard disk unit.

An RS232C interface is provided for a VDU, and cassette, disk and printer ports are available for the I/O slots on the mother board. The minimum system has 8K or RAM, although 64K is more usual. If an optional 6809 processor board is used this can be expanded to 384K. The system uses a 4K monitor ROM.

As the system uses the SS50/6800 bus it can support the CP/M-like operating system FLEX which allows a whole host of high-level languages and programs to be run.

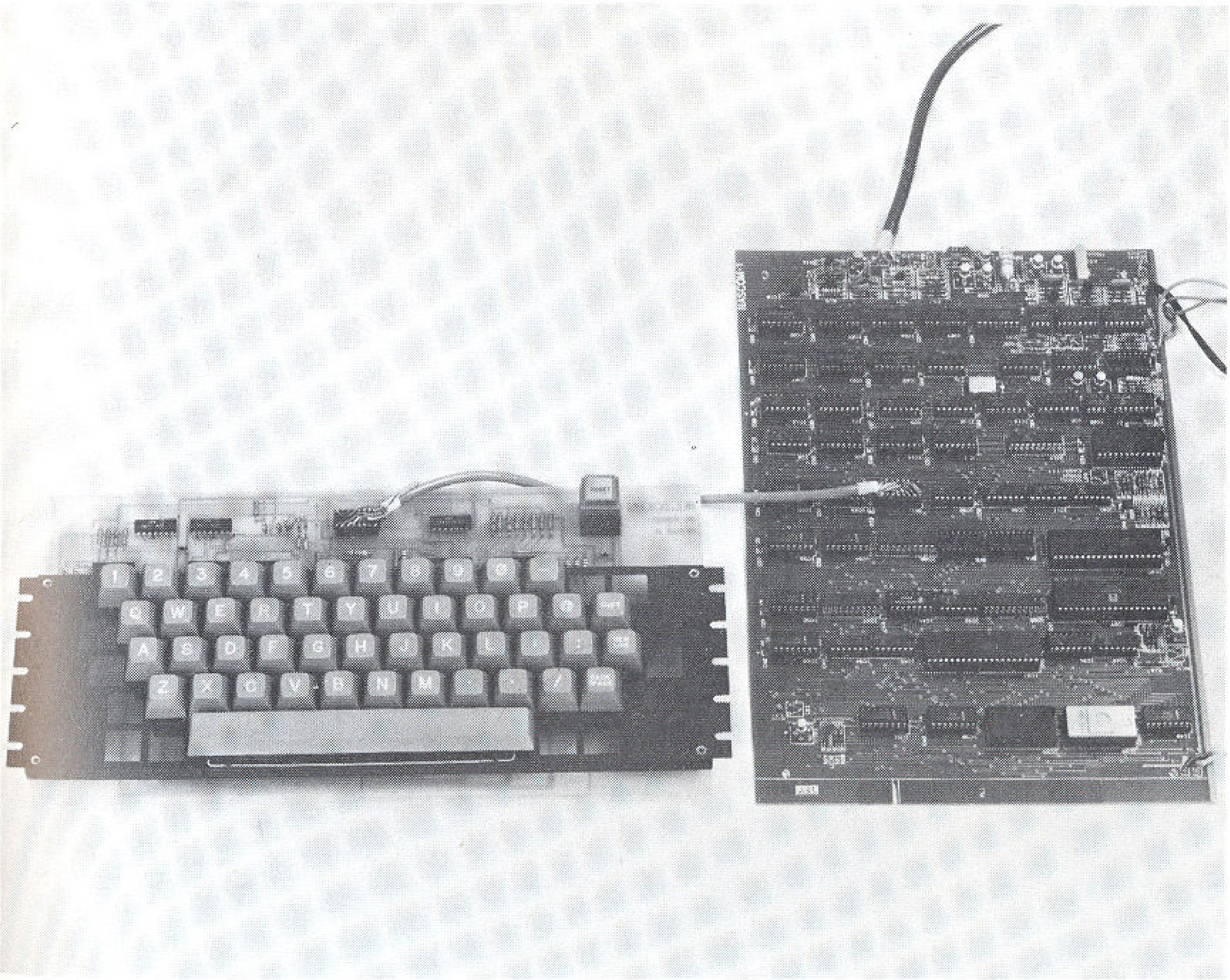
Prices: from £430



Nascom 1

The Nascom 1 is now made by Lucas Logic in the UK. It is one of the most popular computer kits available and is based on a Z80A processor. It is a single board computer with the keyboard attached by cable, and comes as a kit, or ready-built. The keyboard has 47 typewriter style keys. The video output, both 1Vpp composite and UHF, gives a 16 x 24 character display. A whole range of peripherals and expansion boards are available from independent suppliers, as is a vast range of software, and many user groups. A cassette port, RS232C and two parallel input/output ports are available. The minimum RAM configuration is 1K. It has a 1K monitor. Many languages are available including BASIC, C, Forth etc.

Prices: (1K) kit £125 + VAT
(1K) built £140 + VAT



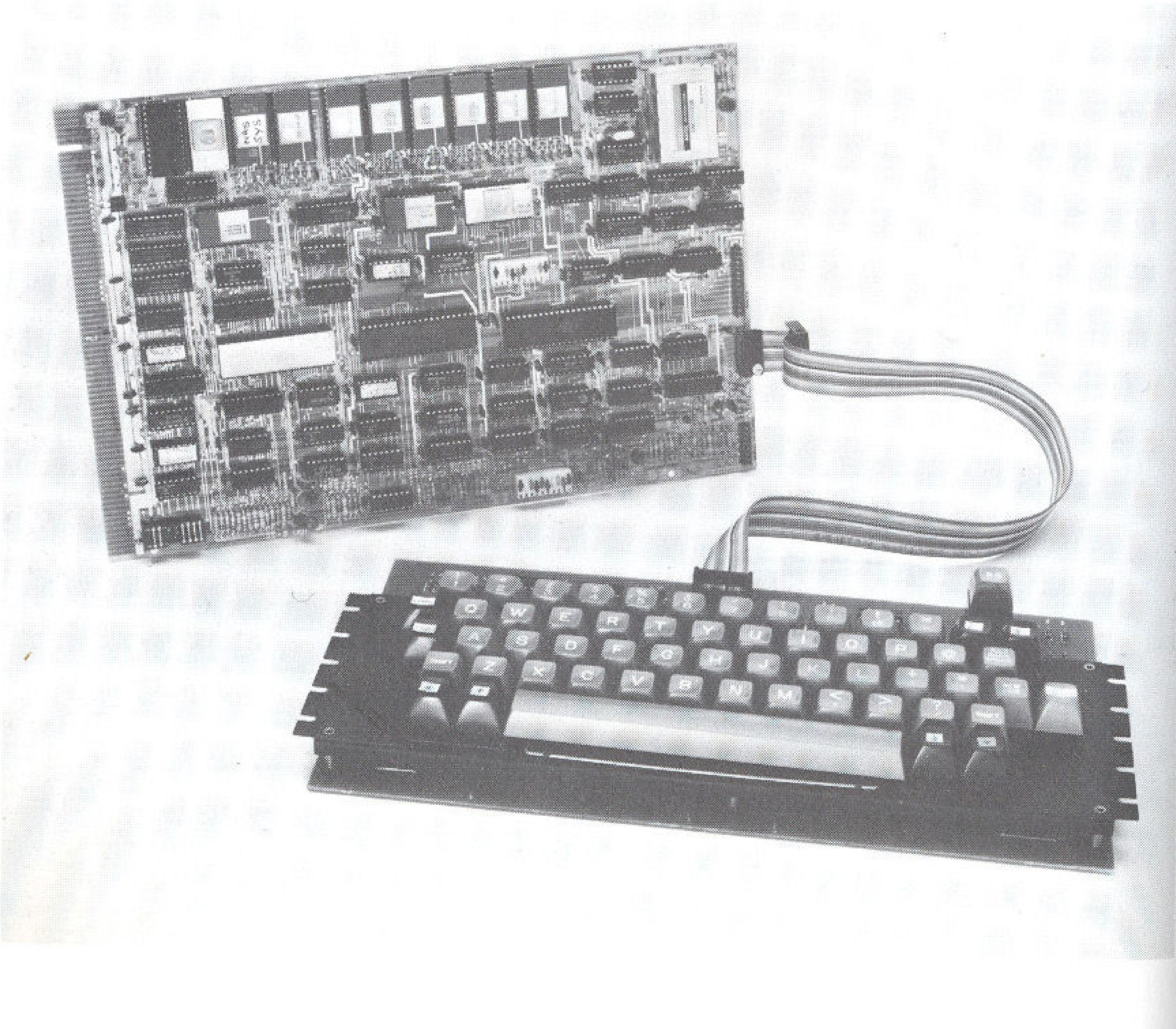
Nascom-2

The Nascom 2 is the big brother of the Nascom 1, Britain's original microcomputer system. It is made by Lucas Logic in the UK. The Nascom 2 is a single board system based around the Z80A at 4MHz. It has a separate 57 key typewriter style keyboard. The basic board comes with 8K of RAM, a 2K monitor ROM and 8K BASIC interpreter. The system can be expanded to 48K of RAM.

There are two eight-bit parallel ports, RS232C and cassette interface. The video output gives 16 x 24 characters. Expansion boards give dual floppy disk capability (2 x 170K). It is possible to purchase a packaged system having dual floppy drives, 16K RAM, case, dot matrix printer and 10" video display.

Like the Nascom 1 this system is well supported by user groups and much software is generally available.

Prices: (8K) kit £225 + VAT
typical system (16K, 2 x 170K disks, VDU, case, printer) £1500 + VAT



NEC PC 8000

NEC claim to have at least 45% of their home Japanese market sewn up with the PC 8000. It is a Z80A-based system housed in a keyboard unit in its basic form.

The typewriter style keyboard has a full set of 56 alphanumeric keys with a numeric keypad and ten special function keys. The system comes with 32K of RAM and 24K of ROM containing a Microsoft-like BASIC interpreter. 1Vpp colour video, or UHF TV output is available, with printer, cassette and expansion interfaces.

The colour graphics have eight colours, and display from 30 to 80 characters per line by 20 or 25 lines. High resolution graphics are possible.

A fixed expansion unit allows disks, (2 x 163K), an extra 32K of RAM and two RS232C, IEEE 488 and one parallel interfaces to be available.

A modular expansion unit gives disk controller and 32K of RAM but also 6 slots for other cards. CP/M can be run with either unit.

Prices: (32K) monochrome display £695 + VAT
typical system (64K, 2 x 163K disks, dual disks, monochrome display, printer) £1950 + VAT

Newbrain

The Newbrain has had a chequered history. Originally announced nearly 2 years ago by Newbury Laboratories, it was in limbo until rescued by the new British Technology Group who have now placed final design and manufacture with the Grundy Group.

The Newbrain is a hand-held unit in three versions. It is based around the Z80A at 4MHz. The main unit has a 62 key calculator style keyboard with a 16-digit alphanumeric display. The outputs include 1Vpp video, UHF TV, and an expansion bus. The video display output gives 24 x 40 characters with the possibility of full viewdata/teletext graphics. The minimum memory size is 2K RAM with up to 20K on board expansion. An external expansion unit allows up to 4M of user RAM to be accessed. A 16K BASIC compiler is available as standard.

The expansion unit can support 2 x 180K floppy disk drives and the RS232C ports a printer and VDU.

As the unit is still very new no details are available as to software support.

Price: from £159 + VAT

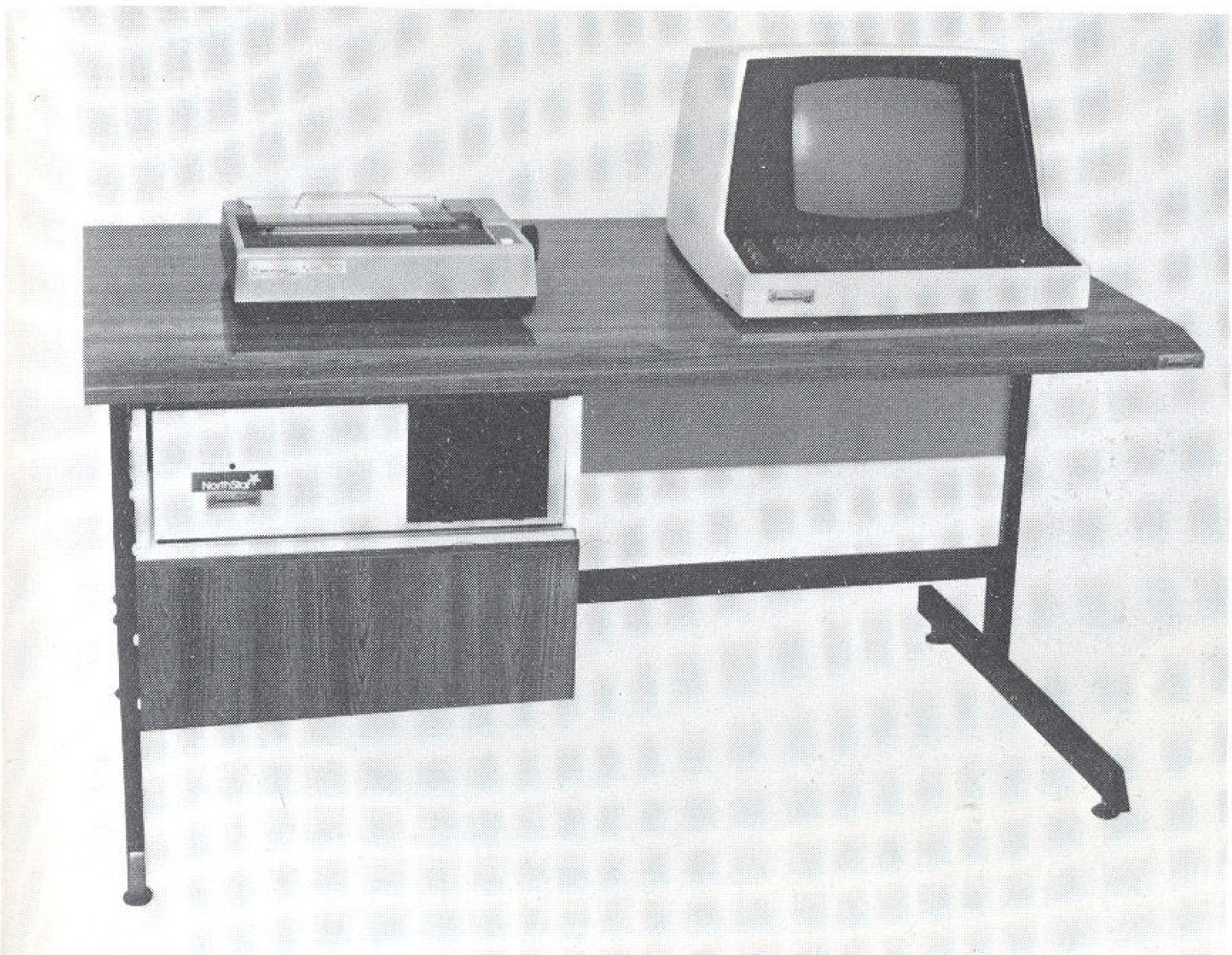
North Star Horizon

North Star were one of the original manufacturers at the start of the personal computer in the United States. The Horizon has been around for a number of years and has been developed and upgraded over this period.

Like other S100-bus systems, the Horizon is based on the Z80A running at 4MHz. The main unit has a 12 slot mother board and has an integral dual floppy disk drive (2 x 180K). The basic system comes with 32K, but usually 64K of RAM and has a small monitor ROM. The interface board has 2 serial and one parallel port.

The system can run the CP/M operating system, thus allowing BASIC, FORTRAN, Pascal, etc., and making much software available.

Price: typical system: (32K RAM + 2 x 180K disks) £1649 + VAT



Osborne 1

The Osborne 1 is essentially a new concept in computers, and has been designed by the guru of the personal computing world, and ex-patriate Briton, Adam Osborne. The Osborne 1 is based around the Z80 processor and comes housed as an integral disk and screen unit with attached keyboard. The whole unit weighs around 20 pounds and has a carrying handle to make it portable.

The screen is a small 5″ monochrome display and gives a window of 24 by 50 characters or a longer display of up to 128 characters. The user RAM supplied as standard is 64K and there is another 2 x 100K available on the dual $5\frac{1}{4}$″ disk drives. Two interfaces are provided; one RS232C and one IEEE 488. A modem with acoustic coupler plugs into a special jack socket. The keyboard is a standard typewriter style unit with numeric keypad. It is attached to the main unit by a 9″ cable. The system comes supplied with the CP/M operating system and BASIC, Word Star and a version of Visicalc.

Price: around £1195 + VAT

OSI C1P/MF

Ohio Scientific, based in Ohio, USA, are the makers of the Superboard II. The Challenger 1P and Challenger 2P-MF are essentially cased versions of this single board system.

The C1P has 8K of RAM and a 53 key typewriter style keyboard. It is based around the 6502 processor and has a 1K monitor ROM and 8K Microsoft BASIC ROM. The video output — both 1Vpp and UHF TV — give 24 x 24 or 12 x 48 characters, or 256 x 256 pixels. A digital-to-analogue converter, RS232C, cassette and joystick interfaces are standard.

The C1P-MF is an upgraded version of the C1P having 20K of RAM and one floppy disk drive (90K). Memory can be expanded up to 32K.

The Superboard, and its derivatives, have good user support and many programs are readily available.

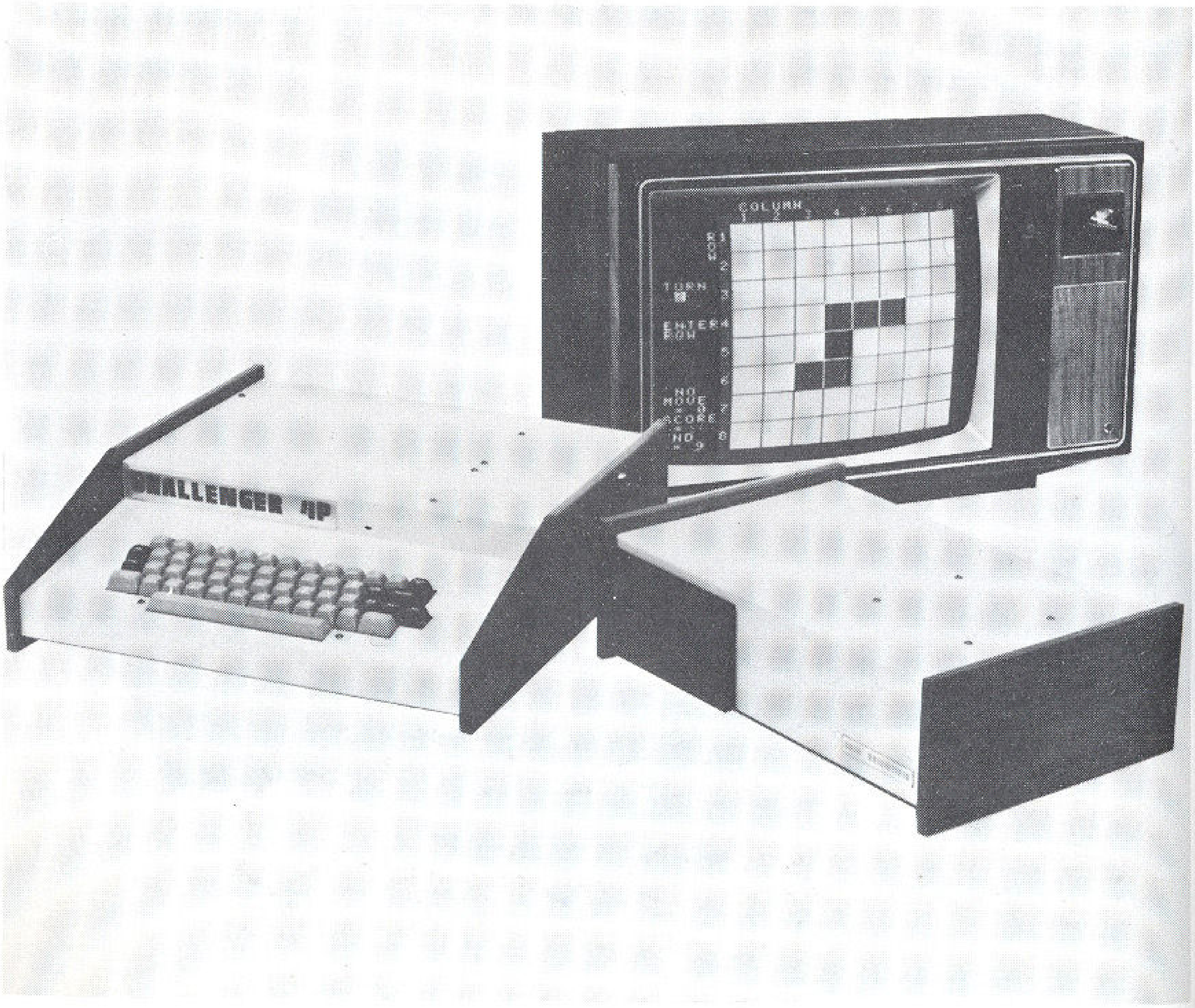
Prices: C1P £250 + VAT
C1P-MF £743 + VAT

OSI C4P-MF

The Challenger 4P-MF is a larger, more versatile version of the Challenger 1. It is based around the same 6502 processor at 2MHz, and 53 key typewriter style keyboard. It comes with 24K RAM as standard and has 8K Microsoft BASIC in ROM.

The screen display can have either 32 x 32 or 64 x 32 characters and can display in 16-levels of grey-scale, or 16 colours with optional colour card. UHF TV ouput, as well as cassette, printer and floppy disk interfaces supplement a digital to analogue converter, two paddle ports and programmable tone generator. The single floppy disk drive, that comes with the system, can store 90K. 8″ floppy disk drives can be added.

Prices: C4P–MF (24K) £1318 + VAT



PET (4008, 4016, 4032)

The latest versions of the PET are basically upgrades of the original 2001 system. This can claim to be the world's first mass produced personal computer and it still has a leading position in Europe, although not in the USA. Commodore now manufacture in both the USA and West Germany, most units sold in Europe coming from the latter.

The PET is an integral system with a 9" (or 12" on some versions) video screen housed above the keyboard unit. The display has 25 x 40 characters and the unit comes with a 72 typewriter style keyboard with a numeric pad. It uses a 6502 processor at 1MHz and has 8K Microsoft BASIC in ROM. The minimum RAM available is 8K, maximum 32K. It is possible — although not advisable — to expand the 16K version to a 32K one. There are five interfaces — IEEE 488, 2 x 8 bit parallel, and two cassette I/O ports.

A whole range of Commodore, and compatible, products are available including floppy disk drives, printers, a - d and d - a converters etc. The PET probably has more software and peripherals than any other small computer system.

There are many active user groups, and over 250 retailers around the country. It is now possible to run the CP/M operating system with a special interface unit.

Prices: 4008 (8K) £378 + VAT
4016 (16K) £462 + VAT
4032 (32K) £585 + VAT
floppy disk drive (2 x 180K) £585 + VAT
floppy disk drive (2 x 500K) £755 + VAT

Philips P2000

This system is not very well known in the UK, but there are some installations. It is made by Philips in Austria and is based around a keyboard unit with integral cassette deck and plug-in ROM modules.

The unit comes with 18K RAM, expandable to 72K, with 4K of ROM. The 74 key keyboard has a standard layout with numeric keypad but has calculator type keys. The mini-cassette unit can store 120K bytes and the unit comes with an RS232C and video (1Vpp and UHF TV) interfaces. The display shows 24 x 40 characters, and colour graphics are possible. The system can support twin floppy disk drives (2 x 140K) and plug-in ROM pacs allow viewdata, BASIC, Pascal, etc.

Prices: from £600 + VAT

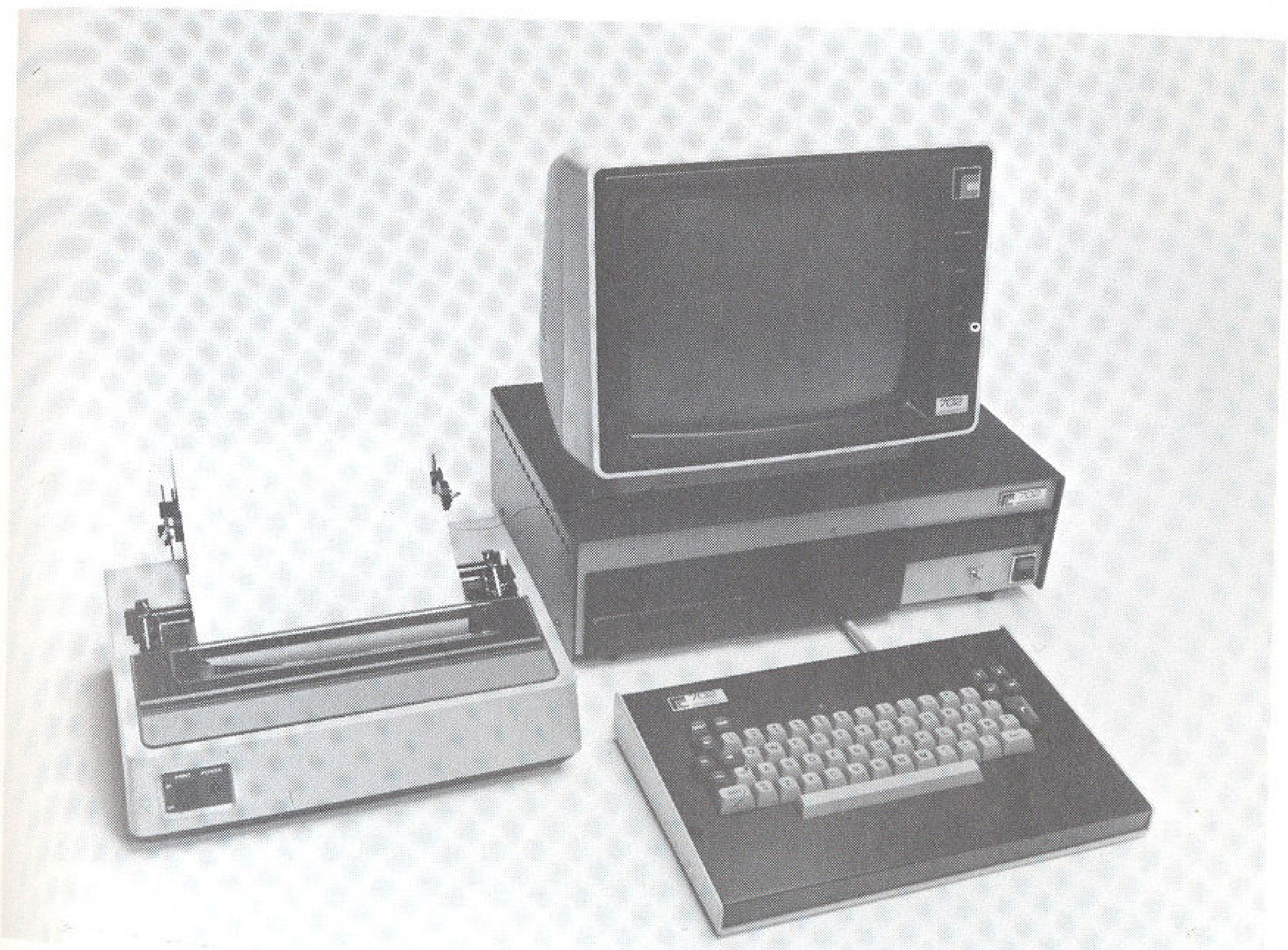
Piccolo RC700

The Piccolo computer is made by A/S Regnecentralen in Denmark. The system is used by most schools in Denmark, and it has the same status as, say, the RM380Z in the UK.

The Piccolo is based around the Z80 processor at 4MHz, and is a chassis-based unit with integral floppy disk drive (380K). The attached keyboard has 62 typewriter keys and the video screen displays 80 x 25 characters. Two RS232C and one parallel interface are provided on the main unit.

One additional floppy disk drive can be inserted into the main unit, and 8″ drives (1. 2M) can be added externally. The basic system comes with 48K of RAM and 2K of monitor ROM. The RAM can be expanded internally to 64K. The system supports CP/M and USCD Pascal, COMAL 80 and assemblers are available.

Prices: (64K) £1680 + VAT



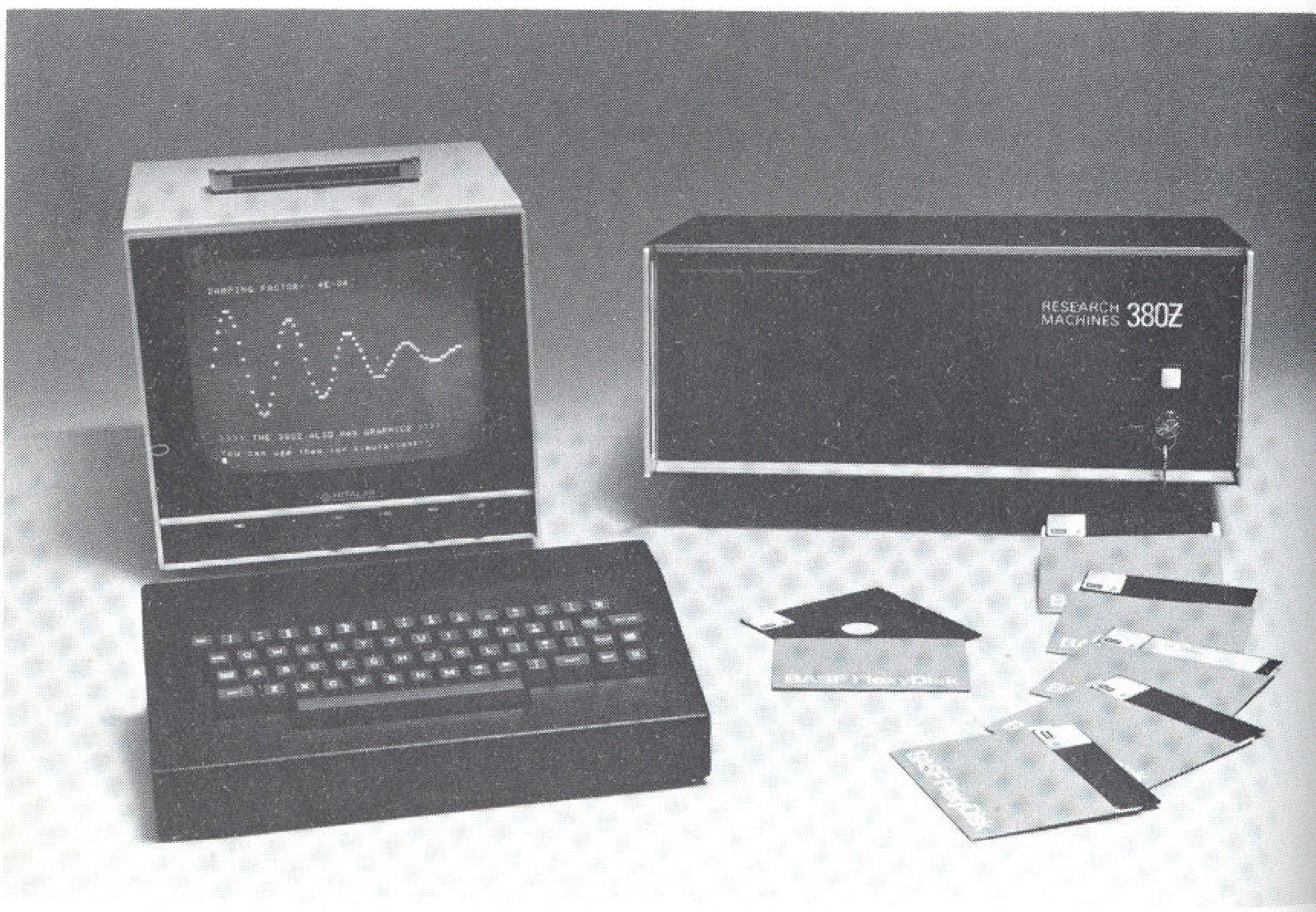
RM 380Z

Research Machines Limited are based in Oxford, England and the 380Z was their first product. It was designed specifically for the education market, and the vast majority of its users are in this area.

It is based around the Z80A processor at 4MHz and has a non-standard bus-structure. The basic system has a typewriter style keyboard detached from the main unit, and comes with 32K of RAM and a 3K monitor ROM. This can be upgraded to 64K of RAM and single or dual floppy disk drives (70K) can be inserted into the main housing. There are optional boards giving high resolution and colour graphics — 319 x 191 pixels in eight colours. The basic unit has UHF TV and 1Vpp video output and cassette and printer interfaces. 8″ floppy drives are also available (2 x 500K).

The unit runs the CP/M operating system. Most of the software is educationally based but, up to now, has been based on a non-standard version of CP/M — this should be rectified in the very near future.

Price: (32K) cassette based £550 + VAT
(64K) cassette based £747 + VAT
typical system: (64K, 2 x 70K disks) £1867 + VAT



Sharp MZ-80B

Made in Japan by Sharp, the MZ-80B is an upgrade of the very successful MZ-80K.

The MZ-80B has integral 9″ video display and cassette recorder. The calculator style keyboard of the MZ-80K has been replaced by a 78 key typewriter style keyboard, with ten user definable keys.

The 9″ display has either 40 or 80 x 25 characters, or 320 x 200 pixels. There is a partial rolling display. The system uses a Z80A processor and the basic unit has 32K RAM and 2K monitor ROM. Like the MZ80K language programs are loaded from a cassette, or disk. The unit needs an expansion unit to drive floppy disks (2 x 280K), printers and other peripheral devices.

Prices: from £1095 + VAT

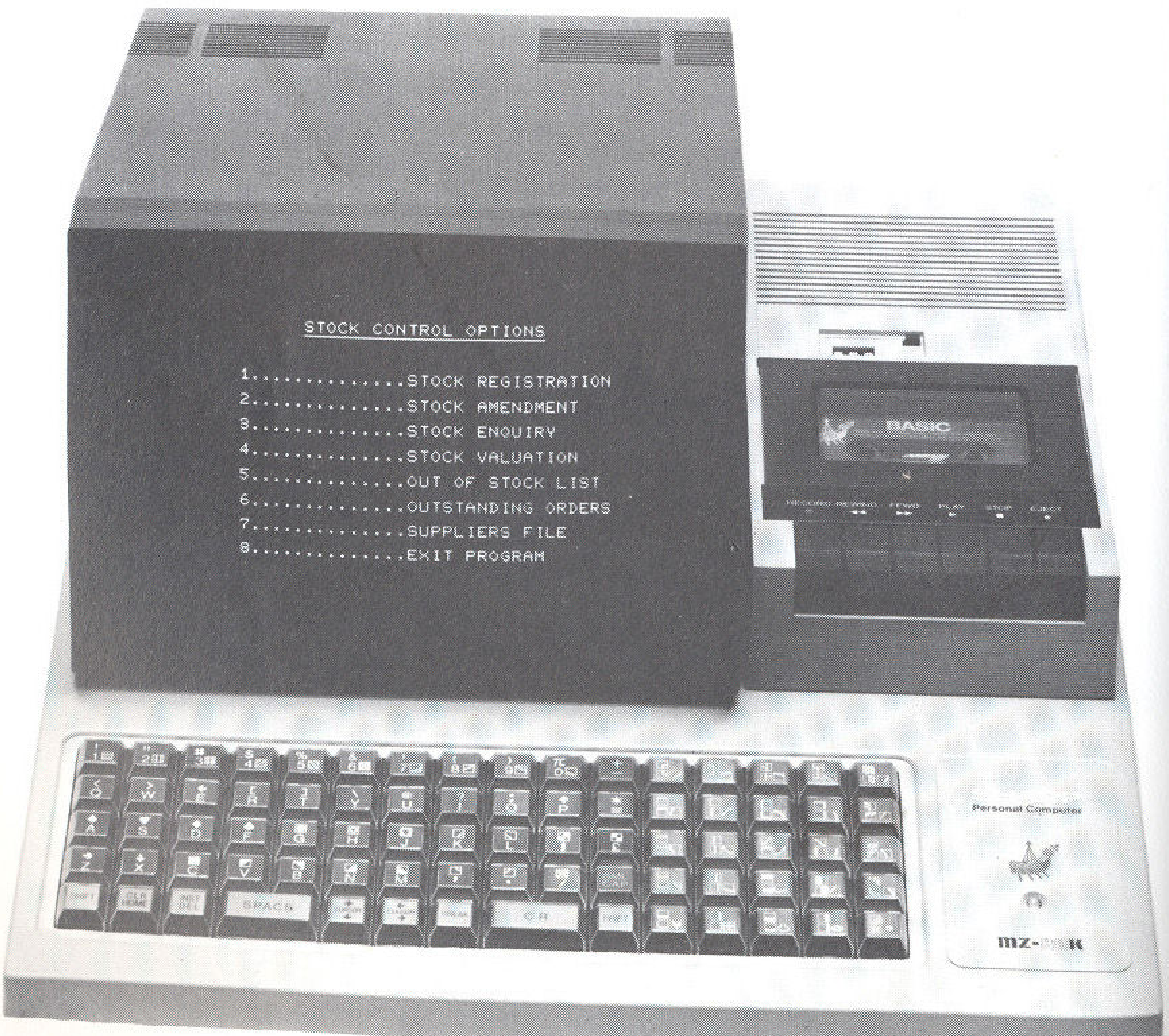
Sharp MZ-80K

The Sharp MZ-80K was one of Japan's first personal computers aimed at the "consumer market". It is still one of the most popular systems in the UK vying for third position in the market place behind the TRS80I/Video Genie.

The MZ-80K is an integral unit having video display and cassette unit built into the keyboard housing. It uses a Z80A processor running at 2MHz. There is a 10" monochrome display showing 25 x 40 characters or 80 x 50 pixels. The system also has a considerable number of graphics characters. The keyboard is a non-standard layout, and use 79 calculator style keys. The system has an integral loudspeaker with three octave tone generator and comes with 20K of RAM in its basic fcrm. There is no resident high level language — this has to be loaded from cassette, or optional floppy disk. There is, however, a 4K monitor to handle the essential machine code routines. The system can be expanded internally to 48K. An expansion interface allows up to five peripherals to be attached — floppy disks, printer, RS232C etc.

There are many programs now available for the MZ-80K and it has a good dealer network, and many user groups. With appropriate interface unit the system can run CP/M operating system.

Prices: (48K) £460 + VAT
Expansion I/O £84 + VAT
floppy disk (2 x 143K) £630 + VAT

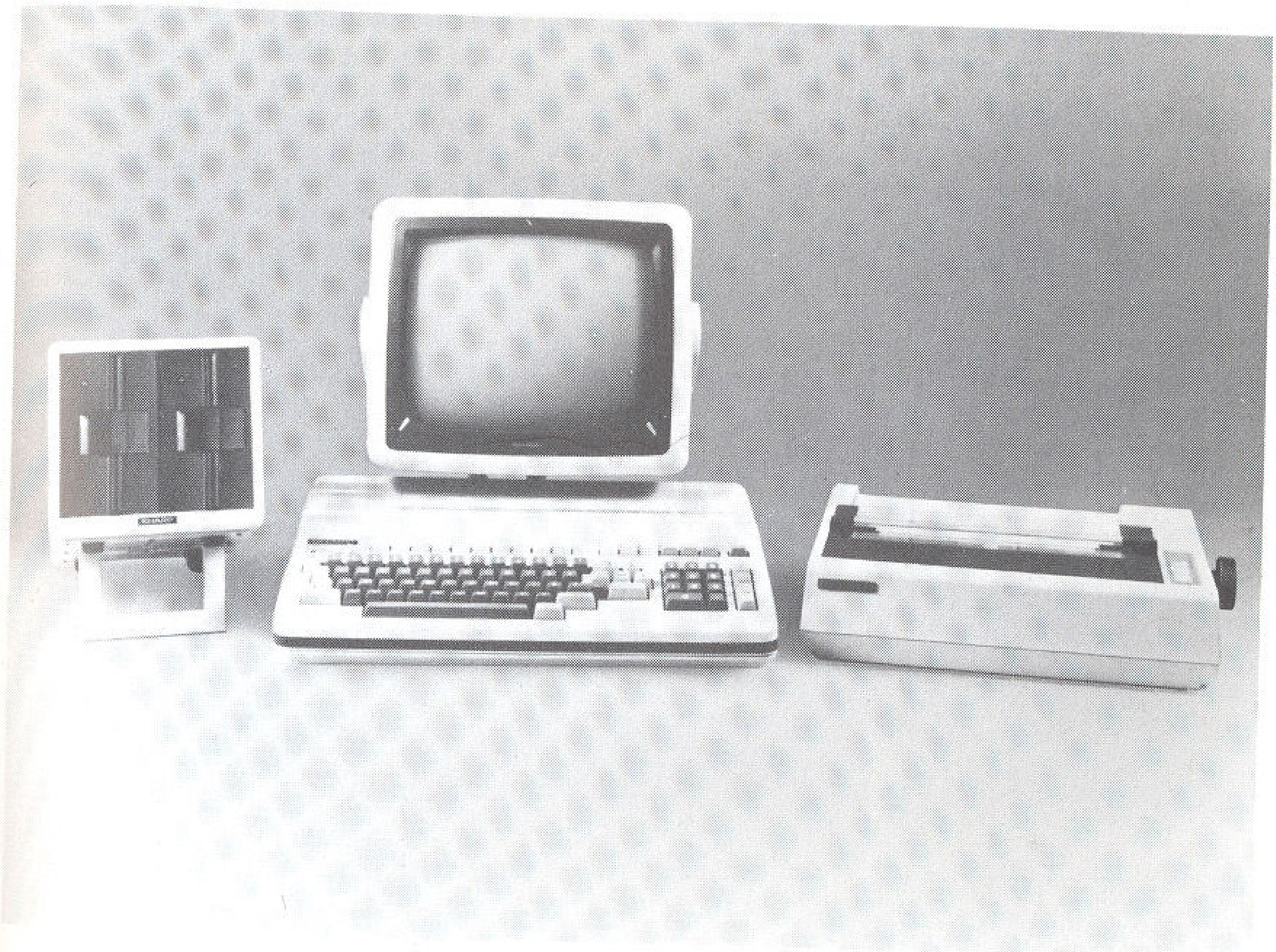


Sharp P3200

This system is the top-end of the Sharp range from Japan. It is aimed at the small business user. The system is based around the Z80 processor, and is a keyboard unit with attached video screen and dual floppy disk drives. (2 x 142K or 2 x 285K). The 92 key typewriter/style keyboard has numeric keypad and ten user definable keys. The system comes with 32K of RAM and 32K of ROM for the BASIC interpreter. It is possible to expand up to 64K of RAM inside the keyboard unit.

The 12" screen has a 80 x 25 character display. Interfaces for cassette and printer are standard. RS232C and other standard interfaces can be plugged in to the main unit.

Price: typical system (32K, 2 x 142K disks, printer) £2950 + VAT



Sig-Net

Shelton Instruments were the people behind the original design of the Nascom 1. Sig-Net is a totally new concept, whereby a series of boards — processor, interface, memory etc. — are interconnected by flexible wires. Each board has its own power supply, and many variations of computer can be evolved from the basic system components. Shelton Instruments are based in London.

The Sig-Net 200 series computer has a number of these modules housed in an enclosure that can have two integral floppy disk drives (2 x 170K). The system comes with 64K of RAM and up to 4K of ROM. It uses a Z80A processor at 4MHz. There are two RS232C interfaces and a 4-channel real-time clock.

A hard disk controller is connected on-board as standard. The system is indefinitely expandable using Sig-Net modules. CP/M 2.2 comes with the computer allowing the CP/M library of software to be used.

Prices: 200/2 £1299 + VAT

Sinclair ZX-81

Manufactured by Sinclair Research in Cambridge, England, the ZX-81, successor to the ZX80, can claim to be Europe's largest selling computer.

The ZX-81 is a small, keyboard based unit with 40 touch sensitive keys. UHF TV output, as well as cassette and power supply sockets, are available. There is one expansion interface that allows a 16K RAM expansion pack to be used. The system comes ready built, or in kit form. The basic unit has 1K RAM and 8K BASIC/monitor ROM. The video gives 24 x 32 characters or 64 x 44 pixels.

The BASIC interpreter carries out dynamic syntax checking on key entry.

A small 40 column electrostatic printer is also available. The system is supported by many user groups who generate most of the available software.

Price: (1K) kit £36.40 + VAT
(1K) built £60.83 + VAT
16K RAM £43.43 + VAT
printer £43.43 + VAT

Sord M100

This is sometimes called the Ace and is made by Sord in Japan. It is a keyboard S-100 bus based system with integral loudspeaker and attached dual floppy disk drives (2 x 350K) and video display. The 12" display has 64 x 256 pixels in 8 colours with an optional board.

The keyboard has 74 typewriter keys with a numeric keypad. Interfaces on the basic system include S-100 bus connector, 2 analogue-digital converters, 2 parallel ports, parallel printer output and RS232C.

The basic configuration comes with 48K of RAM and 8K of monitor ROM.

CP/M is available, thus allowing a vast amount of software to be used.

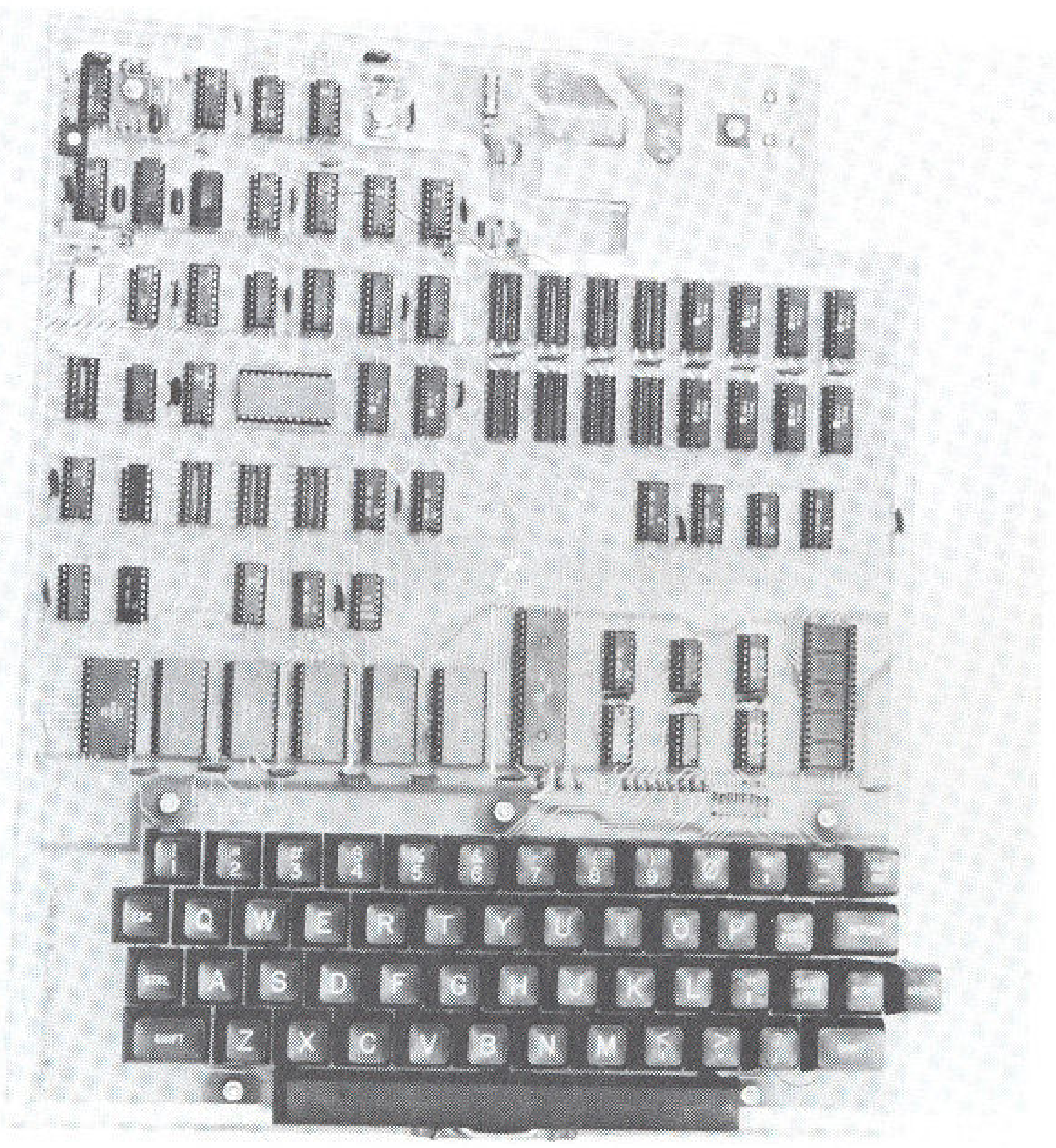
Prices: from £1795 + VAT

Superboard II

Made by Ohio Scientific in the United States, the Superboard was one of the first single board computers aimed at the hobbyist market. It is based around the 6502 processor and the board has an integral 53 key typewriter style keyboard. It has 8K Microsoft BASIC in ROM and comes with 4K RAM expandable to 8K on board, or up to 32K with an expansion board. The interfaces include UHF TV (with optional modulator) 1Vpp video, cassette and a programmable I/O port. There is also an on-board digital to analogue converter and loudspeaker. The screen display is software selectable — either 24 x 24 or 48 x 12 characters. The 1K monitor/character generator ROM gives a full graphics character set based on a 256 x 256 pixel matrix.

There is an optional case, and much of the software is comparable with the UK 101. Many user groups have produced other languages, such as Forth, and there is a lot of software available.

Prices: (4K) £149 + VAT
4K RAM £14.00 + VAT
power supply £23.00 + VAT
8K expansion board £150 + VAT
floppy disk drive (100K) £269 + VAT



Superbrain

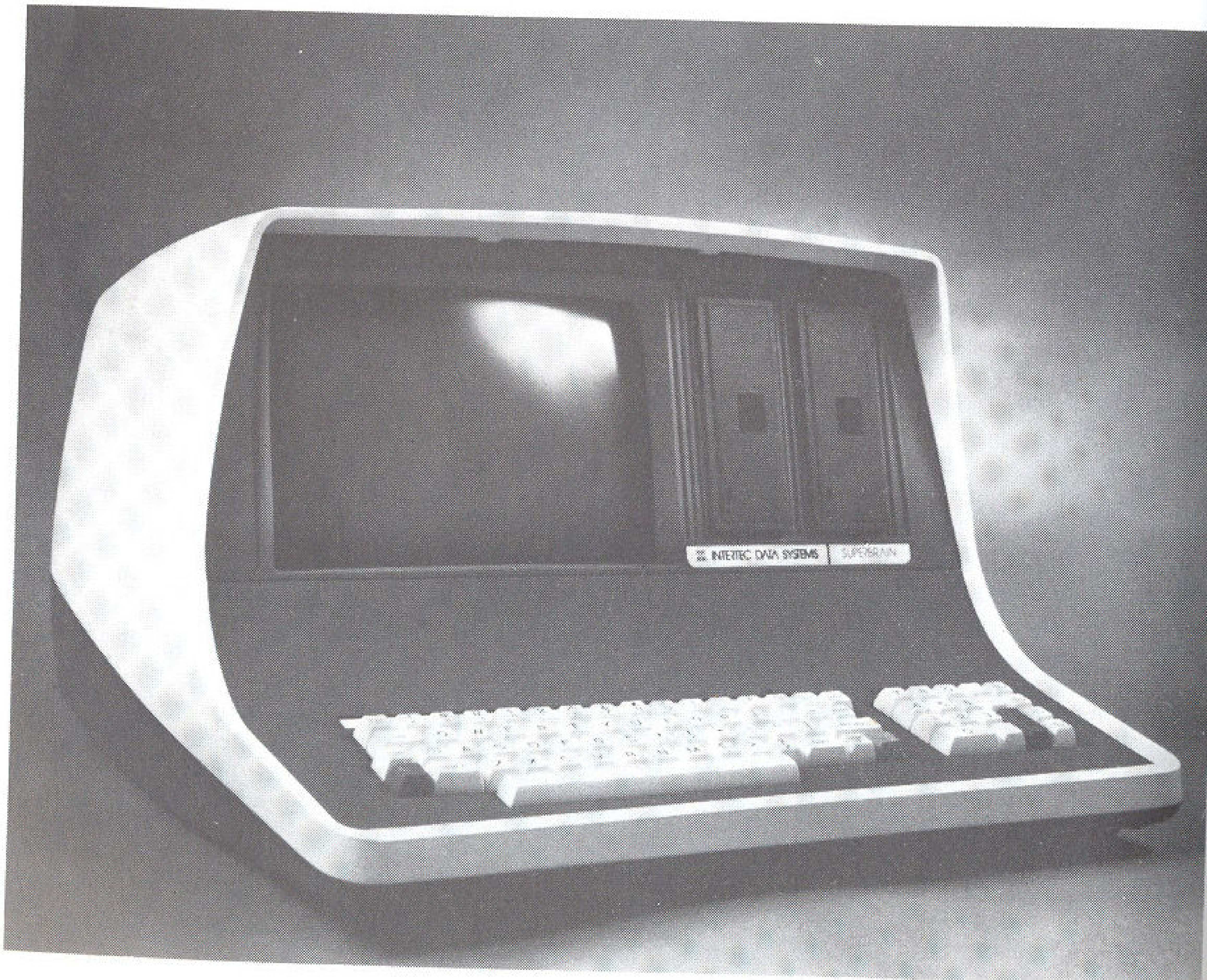
The Superbrain is manufactured by Intertec in the USA. It is an integrated system with keyboard, display and disk drives. The system uses two Z80A processors at 4MHz, one for the main processing, and the other for peripheral activities e.g. disk etc. The dual floppy disk units can be 2 x 175K or 2 x 350K and a 10M CompuStar hard disk can be added to the system. The system comes with 64K of RAM and a 1K ROM monitor.

There is a 12" monochrome display giving 25 x 80 characters. Two RS232C ports and a S-100 bus edge connector are provided. The system runs the CP/M operating system giving BASIC, COBOL, FORTRAN etc.

The keyboard has 80 typewriter style keys including a numeric pad.

There is a lot of software available, especially as the CP/M operating system is used.

Prices: 2 x 175K disks £1950 + VAT
2 x 350K disks £2395 + VAT



SWTPc S/09

South West Technical Products Corporation, SWTPc, of Texas, USA, were the originators of the SS50 bus for the 6800 processor. It was prepared as an alternative standard to the S-100 but never really got off the ground. There are, however, a number of suppliers and the system concept has many supporters.

The S/09 system is based around the pseudo-16-bit 6809 processor and is therefore able to address more memory and perform calculations more quickly, than its 8-bit predecessor. The main system has an 8-slot motherboard, with an 8-slot I/O capability. The minimum system has 8K of RAM, but this can be enhanced to 384K with the 6809 processor.

Three disk options are available, dual floppy drive (2 x 175K) dual 8″ floppy drive (2 x 1.2M) and a 16M hard disk. Disk controllers, as well as printer and RS232C interfaces are provided on appropriate I/O cards. The system uses a 2K monitor ROM.

The Flex operating system is used, with the UNIX-like Uniflex available. Many languages, and lots of software, are readily obtainable.

Prices: from £1250 + VAT

TI 99/4A

This is the updated version of the ill-fated TI99/4. In place of the small keys on the original version are larger, typewriter style keys.

The TI 99/4A is based around the 9900 16-bit processor, and is thus inherently more accurate and faster than the more normal 8-bit systems. The 99/4 is a keyboard based system utilising plug-in ROM pacs. It has full colour graphics and displays 32 x 14 characters or 192 x 256 pixels in 16 colours. The new model has UHF TV output to the PAL European standard.

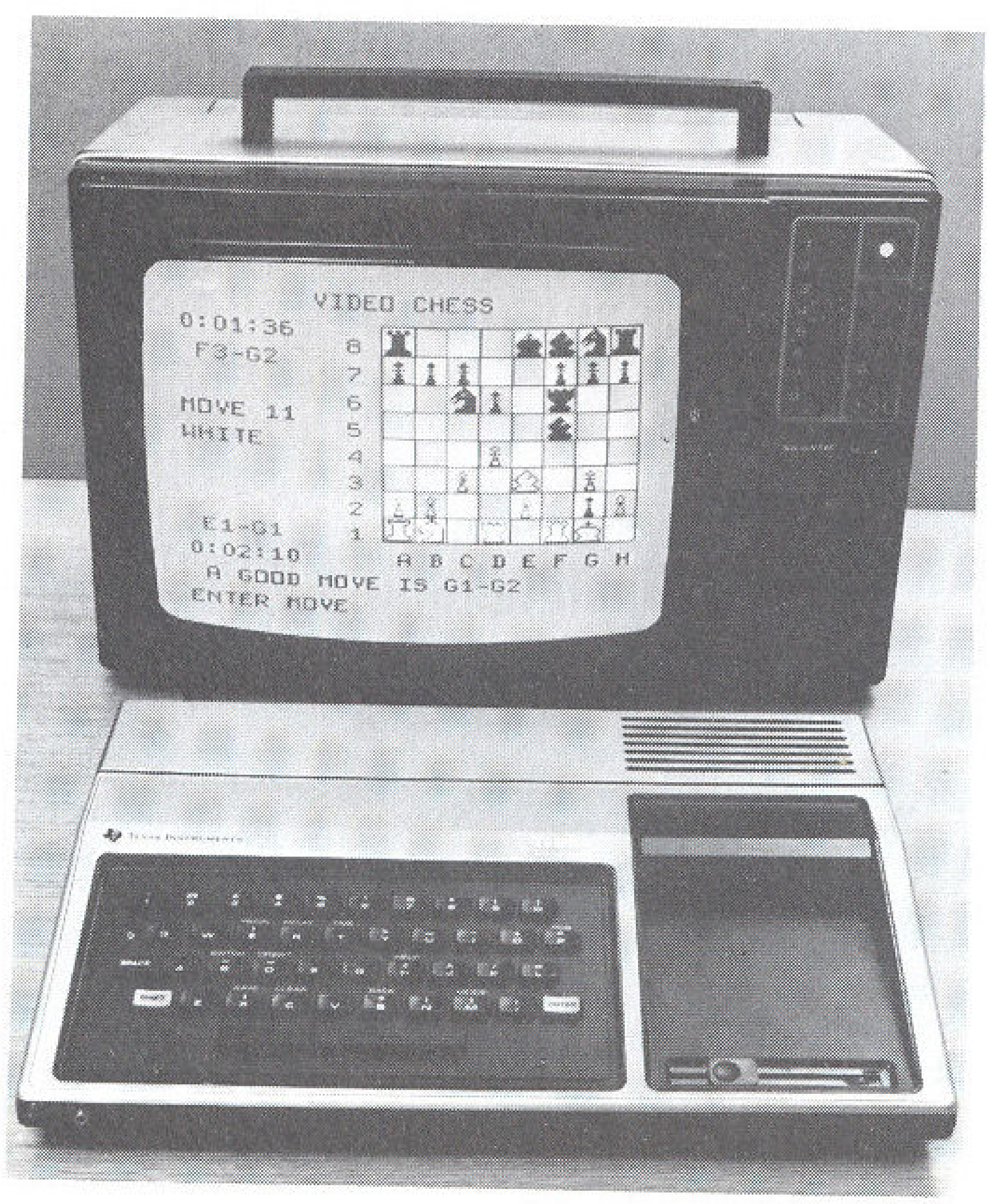
There are a number of outputs including a 44 pin I/O bus that allows expansion units to be connected. These include units for floppy disks (180K), RS232C and joysticks. There is an integral loudspeaker, and a speech synthesiser is available.

The basic unit comes with 16K of RAM, and 14K BASIC and 4.4K monitor ROM. Up to 30K of plug-in ROM pac is allowed.

The system has very sophisticated graphics and allows multiple overlay of graphics symbols.

There is plenty of software available, mostly games, consumer and educational.

Prices: (16K) £399 + VAT



TRS 80/I

This was the first big selling computer, available on every American's doorstep. It is now a bit dated, and Tandy Corporation are phasing its production out whilst giving other companies licences to continue building similar systems.

The system is based around a 1MHz Z80 processor and basically comes as a keyboard based unit. Tandy sell a TV and cassette recorder that make up the essential minimum system. This comes with 4K of RAM upgradeable to 16K. The original "level 1" integer BASIC ROM is not usually available, the floating point "level 2" BASIC ROM taking its place. The system has a 4K monitor.

The keyboard unit has 53 typewriter style keys with a separate numeric keypad. Expansion modules are available allowing up to 48K RAM, floppy disks and printers to be connected. The usual expansion interface also gives RS232 and a parallel I/O port. The video display is 16 x 64 characters with some graphics. The TRS 80/I probably has the largest amount of software, both commercial and non-commercial, of any microcomputer available.

Prices: (4K) with TV + cassette £430.45 + VAT
(16K) with TV + cassette £580 + VAT
floppy disk drive (55K) £339 + VAT
expansion interface £199.95 + VAT

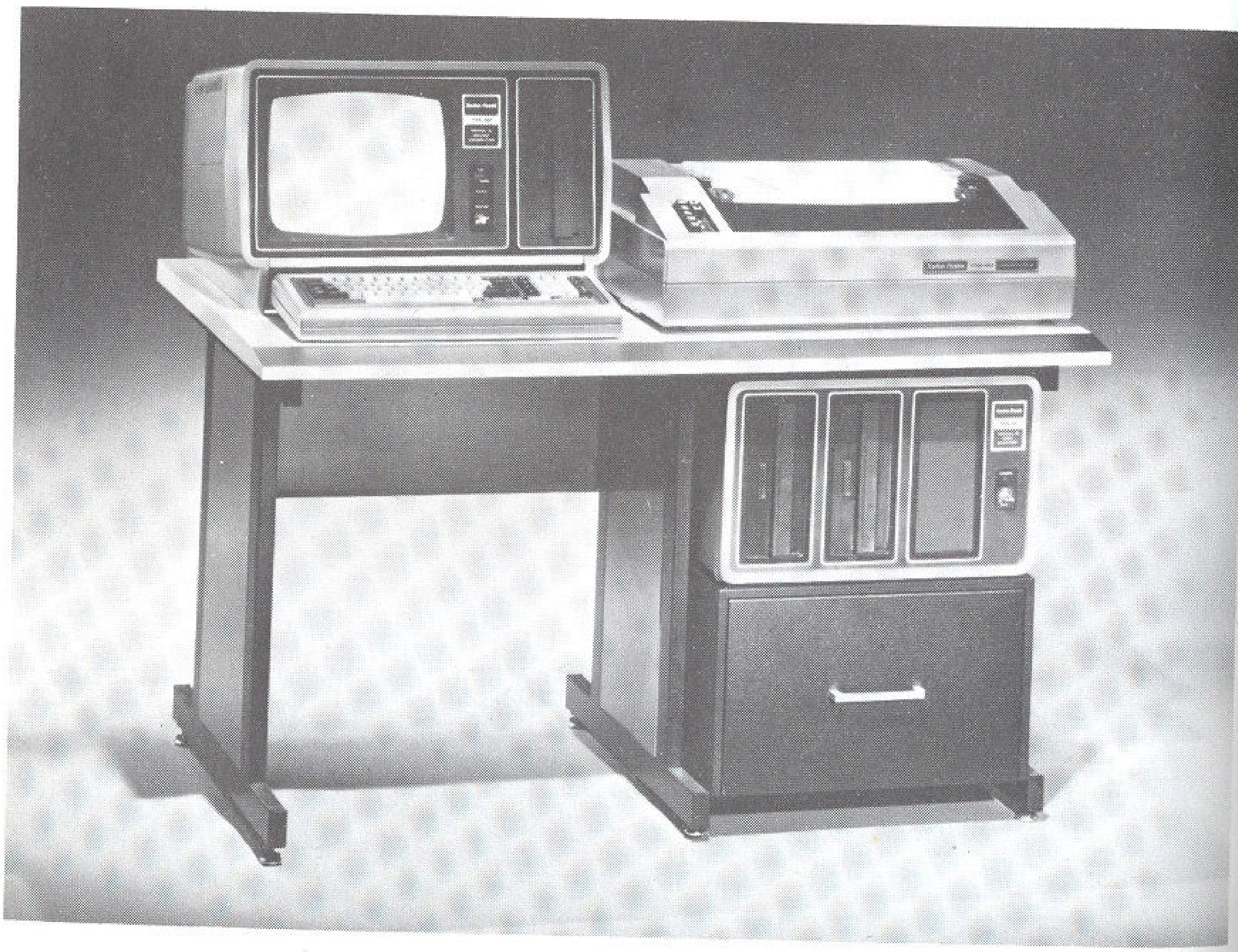
TRS 80/II

The second computer to come from Tandy Corporation in the USA was aimed at the small business market and gives much more facilities than the Model I. The TRS 80/II has a Z80A processor at 2MHz and comes with an integral 8" floppy disk drive giving 500 Kbytes of storage. There is an integral 12" monochrome video screen with 24 x 80 characters.

Three other 8" floppy disk drives may be attached to the system through an expansion port, which is supplemented by 2 RS232C ports and an eight bit parallel interface. The basic system comes with 32K of RAM upgradeable to 64K.

The system has a detachable 76 key typewriter style keyboard with numeric keypad.

Prices: (32K) £2199 + VAT
(64K) £2499 + VAT

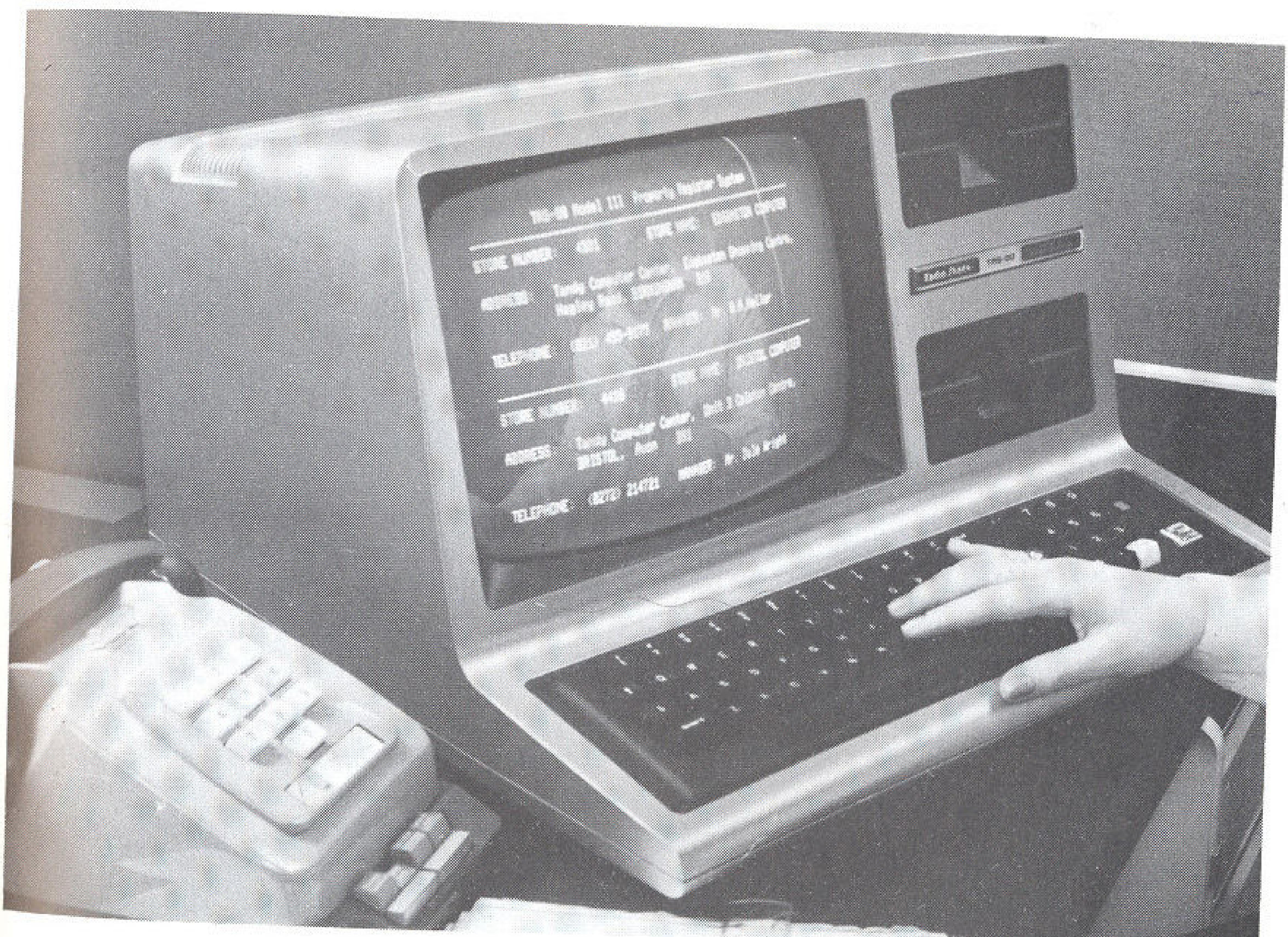


TRS 80/III

This is the latest in the TRS80 computer series, made by Tandy Corporation in the United States. It has an integral 12″ monochrome display and the option of two integral floppy disk drives (175K). The basic system comes with 16K of user RAM expandable to 48K. The BASIC is from Microsoft and takes up 14K of ROM, and is upward compatible from Model I systems. There is a cassette, and printer interface with RS232C and other disk interfaces available as options. The video display is 64 x 16 characters and usual TRS80 graphics are available. The keyboard has 65 typewriter style keys with a separate numeric keypad. The system uses a Z80 processor.

Software from the vast TRS80/I library is available with some modifications.

Prices: (16K) £799 + VAT
2 x 175K disk drives £599 + VAT



Tuscan

Transam are based in London, and have moved on from their original base as a manufacturer of cheap, hobby-type boards. The Tuscan is a complete S-100 based computer system, with integral dual floppy disk drives (2 x 190K).

The main keyboard-based unit has a ten slot S100 motherboard and is based around the Z80 processor at 2 or 4MHz. The system comes with CPU board; video board giving 64 x 16 characters either as 1Vpp video or UHF TV; 8K of RAM and 8K of monitor ROM.

The system can be expanded to 64K of RAM, and can take 8K BASIC in ROM. Most S-100 boards can be used giving optionally graphics, interfaces etc. The CP/M operating system is also available.

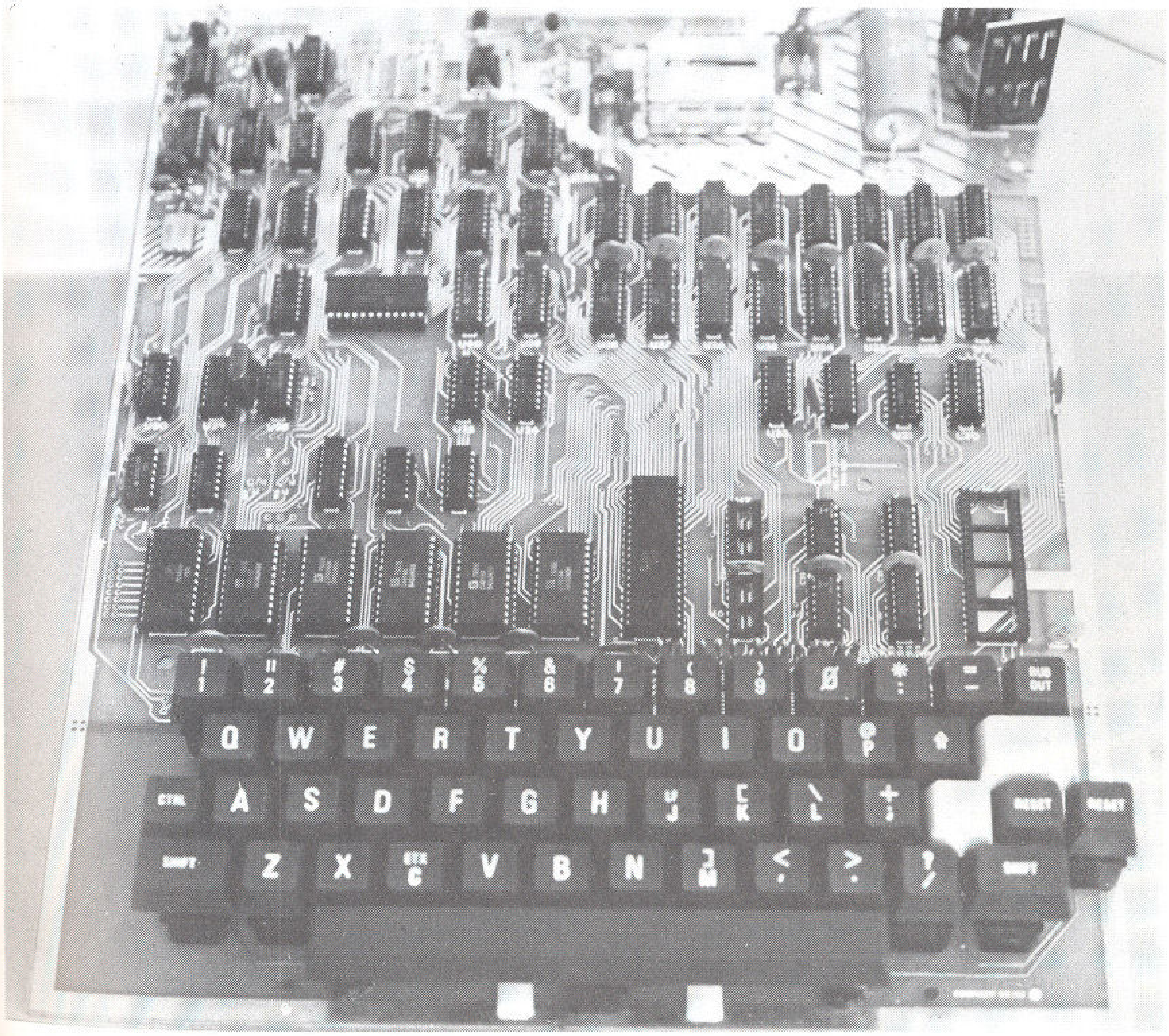
Transam, through their TCL subsidiary, are well known for implementations of high level languages — especially PASCAL. These, and others, can be used on the system.

Prices: typical systems (24K RAM, single floppy disk 190K) £999 + VAT
(60K RAM, dual floppy disk 2 x 190K) £1449 + VAT

UK 101

Made in the UK by Compukit in New Barnet, North London, the UK 101 was originally a copy of the Superboard II. Various legal battles later the UK 101 is now, technically, behind its erstwhile rival. Based around the 6502 processor at 1MHz the UK 101 comes either in kit form, or ready built. It has cassette and TV interfaces, as well as an expansion bus. The keyboard has 52 typewriter-style keys and generates a 48 x 16 character display. 128 graphics characters are available. The minimum RAM is 4K expandable to 8K on-board, or 40K with an expansion board. It has a 2K monitor with 8K Microsoft BASIC as standard. There are many user groups, and plenty of software available. Cases are made by a number of manufacturers.

Prices: (4K) kit £149 + VAT
(4K) built £199 + VAT
(4K) upgrade RAM £15.90 + VAT
case £29.50 + VAT



Vector Graphic MZ

The Vector Graphic MZ is a S-100-bus based system running a Z80A processor at 4MHz. The main unit has an 18 slot motherboard and integral dual floppy disk drives (2 x 315K). The interface board has two RS232C and one parallel ports. The system comes with 48K of RAM and a 4K monitor ROM.

The operating system CP/M can be used, thus providing a whole range of software.

Price: £2595 + VAT

Vector Graphic VIP

This is the cheapest system from Vector Graphic of the USA. It is based around the Vector 3 terminal which has integral 72 key keyboard and 12" video screen. The unit has a 6 slot S100 bus board which comes with a 4MHz Z80A processor board with 56K RAM. This board also has an RS232C interface and three 8-bit parallel I/O ports. The screen displays 80 x 24 characters. The basic unit comes with a single floppy disk drive giving 315 Kbytes of storage. The computer has a 4K monitor ROM.

As it is based on the S100 bus and uses a Z80 processor, the system can run CP/M, and all its associated software like COBOL, PASCAL etc. MBASIC and EXECUPLAN are available from VG, and these give Microsoft BASIC and a form of planning program like Visicalc.

Prices: from £2125 + VAT

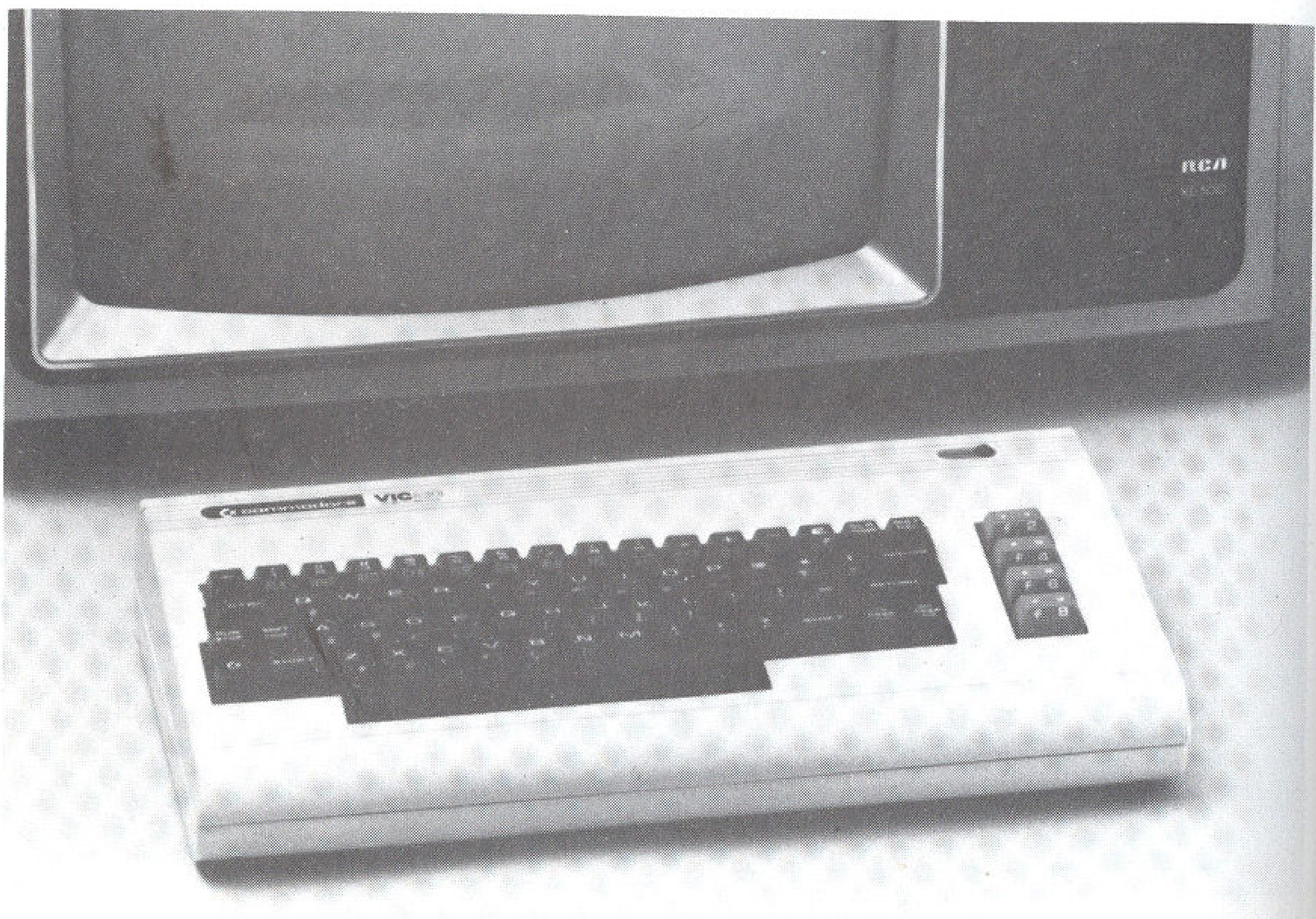
VIC 20

The VIC-20 computer is Commodore's first product in the computer field that represents a challenge to the market below the PET. The VIC-20 is based on the 6502 processor, and can run most of the PET software. The main difference is the 22 x 23 character screen. VIC-20 has 16 screen colours, four internal sound generators using TV loudspeaker, and uses plug-in ROMpacks that can be up to 27K.

The basic unit comes with 5K of RAM, expandable externally to 32K. Standard PET Microsoft BASIC is in an 8K ROM. The system is a keyboard based unit with 66 typewriter style keys, including four user definable keys.

A range of accessories are available including printer, floppy disk (120K), IEEE488 cartridge, joysticks, light pens, etc. Most peripherals should interface with the VIC.

Prices: (5K) £185 + VAT



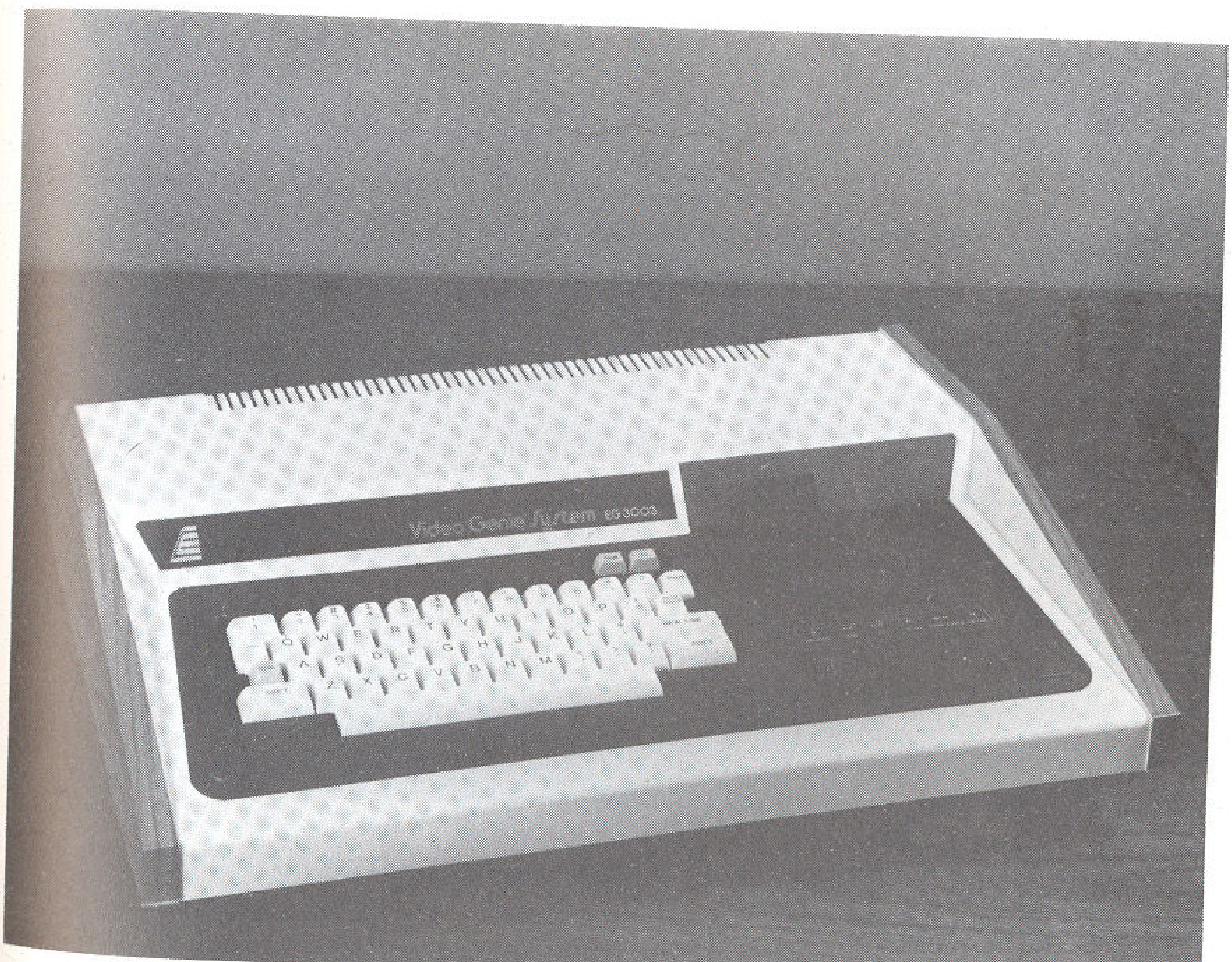
Video Genie EG 3003

The Video Genie is made by EACA in Hong Kong, and is fully compatible with the Tandy TRS 80/I. This had caused some legal problems in the US(!) but the system seems to be outselling the Model I in the UK — mainly because of price.

It is based around the Z80 processor and the basis system comes with 16K RAM and 12K Microsoft BASIC in ROM. It has an integral 51 key typewriter style keyboard and cassette deck. The video output — both 1Vpp and UHF — displays 64 x 16 characters, or 128 x 48 pixels. There is an interface for a second cassette recorder and an expansion interface. This allows an expansion module to be plugged in giving another 64K of RAM and facilities for discs and printer.

The Video Genie runs the vast software library of the TRS 80/I, and has very active user groups.

Price: (16K) £289 + VAT
Expansion unit (with RS232) £215 + VAT
disk (300K) £410 + VAT
32K RAM for expansion unit £130 + VAT



Z-89

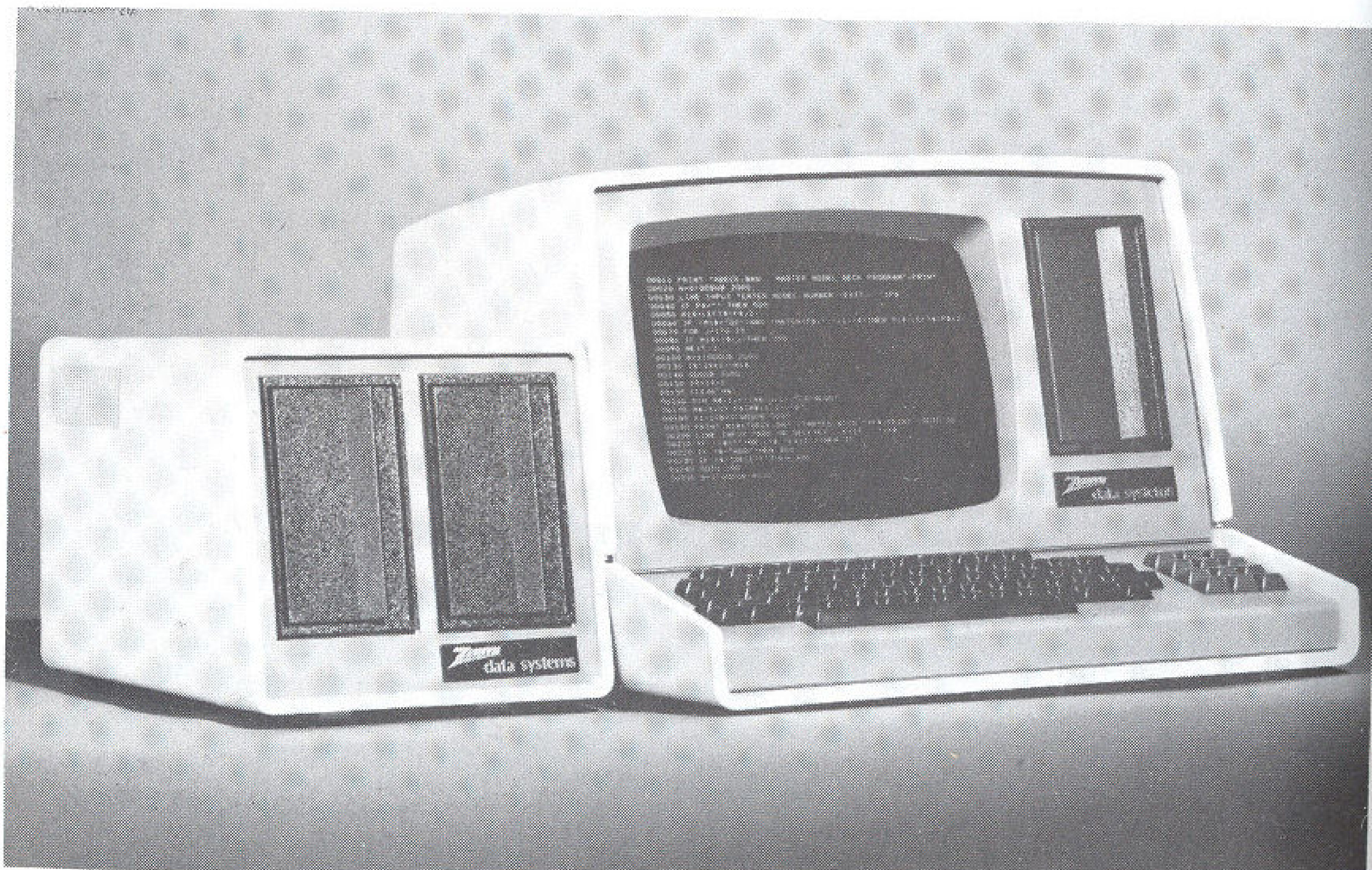
The Z-89 computer is made by the Zenith division of Heath, the American build-it-yourself company.

The Z-89 is based around two Z80 processors at 2MHz and is a desk top unit with integral display, floppy disk and keyboard. The 12″ screen displays 25 x 80 characters or 512 x 256 pixels. The 85 key typewriter style keyboard has a numeric keypad and five user definable keys. The integral floppy disk unit (102K) can be supplemented by three more drives or dual 8″ floppy disk drives externally. (There is a version without integral disk drives designated the H88).

The basic system comes with 48K of RAM expandable to 64K. The second processor runs the graphics capability and this uses 16K of RAM, in addition to the main processor RAM. There is also an 8K monitor ROM.

A number of interface options can supplement the standard RSC232C I/O. There are IEEE488 digitiser pad input and light pen. The system can support the CP/M operating system thus allowing BASIC, FORTRAN etc. to be used.

Prices: £1570 + VAT



CHAPTER 6

How do I choose a system?

We now have a fairly good idea of the systems on the market — their capabilities and how they interconnect with various peripheral devices. As there are as many ideal systems as there are applications i.e. a near infinite number, no system will be ideal. It is worth stressing that it is very unlikely that any computer system will do **all** that you want it to do — some compromise must be sought. Mainly this is in the area of cost. It is possible that the more expensive the system the more likely it approaches your ideal. The trade-off between price and performance, or when you stop looking for something too complex, is up to the individual. Guidelines are possible, and an attempt to have a comprehensive list follows.

The first thing to do is to find out why you want a computer in the first place! Here we come across a circular argument — how do you choose the right system when you've never had a computer before and are not sure what they can do? How do you define your application if you haven't chosen the system? This is where this book, and your local computer retailer comes in. Chat to the retailer, join a club — read the magazines; in fact everything that was mentioned in the earlier chapters.

Once you have got a good idea of the type of system, and amount of money to spend, things become slightly easier.

The obvious starting point is the power of the system. If you just want to learn how a microprocessor system works, and are interested in programming at the machine level — and even controlling electronic apparatus — then the single board computer is your baby. Unfortunately, the accompanying literature is pretty sparse. As the system is really for learning about a specific microprocessor your choice is limited to three or four systems per processor.

With single board computers the questions are fairly obvious, and simply answered. These can be summarized as follows:

a. Is the computer built around a popular processor? This really means the 8080, Z80, 8085, 6800 or 6502. If you choose one of the others, how good is the software support?

b. Is there a good operating system built in to the ROM on the computer board that will let you enter any memory cell you like, look at the contents and alter it? You must of course be able to work through the program one instruction at a time. Reading in the program from paper tape or magnetic tape (is there an interface?) must be possible without difficulty. Is the instruction set simple?

c. Is it a simple matter to connect peripheral equipment via serial or parallel interface?

d. Can you actually afford the storage capacity? Where is the ceiling? Is it in fact simple to expand the single-board computer to a complete system?

e. Does it plug into the mains, or is another power source needed — at additional cost?

f. Can the computer be interfaced to an acceptable bus? What does the additional equipment cost?

g. Are there provisions for using high-level language? BASIC for instance. Most single-board computers do not have this possibility in their basic form. Don't count on being able to use a BASIC interpreter intended, for instance, for an 8080 machine in an 8080-based single-board computer.

Things get a bit more difficult when more complicated systems are required. The basic question usually is — do I need a system that can grow with my needs and inventiveness, or am I getting a system to learn on and then buy a more sophisticated one? Only the user can answer this. If the answer is yes to the first part — expandability — then a system based on an accepted bus standard, or with plenty of peripheral support is required. If the answer is yes to the second part — a simple system to learn on — then a small desk top system may be all that is required.

Expandability and software/hardware support usually determine the choice of system. It is not possible to say that one system is better than another with any objectivity. It all depends on where you live, what your local supplier is like, and whether you have, or want, any software expertise.

The first thing to do is to get to know the supplier. Is he a retailer who has recently come into computers having spent many years selling electrical or consumer electronic goods? If he has a reputation to consider he may not be too hot on software but the servicing backup and help in this area may compensate.

Is he a computer specialist? If so, why does he only stock three or four machines? Why does he consider one better than his competitor up the road, or in the next town? Has he access to software expertise, someone who can help you when you get into a fix?

The main point is — does he know what he's talking about, and is he financially secure? There have been a number of cases recently of computer stores going out of business. Their technical competence was never in question — but the overall management structure was very weak. The last thing wanted is for your supplier, who you will depend on a great deal to start off with, to go bust just when you need him!

This is true for smaller systems that are advertised at discount rates in the magazines. Reading these journals is one way of finding out what is going on. But cheap prices usually means lack of after sales support. It really does pay to shop around in person — go to the exhibitions — and stick with somebody once you've built up trust. If there is a computer club in your locality, a few visits will soon tell you who has a good reputation.

In fact talking to existing users is the best way of getting to know a system. Try and find somebody locally that has solved, or is trying to solve, a similar problem. Their experience can save a lot of time and money.

If I'm sounding a bit vague, then it's because there are as many answers to the question as questioners. A process of elimination will identify a group of three or four machines. This elimination contest will consider the following:

> Cost; physical size; memory size; ease of use; software support; servicing support; reputation of manufacturer; local support.

When these have been considered try out the systems selected, and make your choice. In some ways it's a bit like buying a car, washing machine or hi-fi system — don't think that different considerations apply. They don't!

Ten hints to help you on your way

1 *Become computer conscious.*

Look around you, in your firm or at school — how and where are computers used today? Think about the developments taking place at this moment: the very highly centralized computer centres of the 50s and 60s, with the computers in their ivory towers, are turning at breakneck speed into computers in almost every home. Consider the possibilities this development opens for society and for yourself.

Before you know where you are, your fingers will be itching to play with your own computer ...

2 *Learn more about computers.*

If you are a novice in the computer field it may be wise to read a good introduction to the subject. Keep on reading this book — read several books; look in the bibliography in the appendix. You'll also find there quite a number of useful magazines.

Read product brochures, discuss matters with sales staff and — above all — with other people who are interested in personal computers. Go to some of the computer exhibitions that are becoming quite frequent. The magazines usually advertise these.

3 *Get yourself some computer friends.*

A group of like-minded people is the place to bring up "stupid questions". The place to find crazy ideas and the place to give and find help with tricky problems.

During the last few years, hundreds of computer clubs have been formed in the UK. If you can't find a club in the place where you live, then start one! You're bound to find more interested people at work or school.

4 *Which type are you?*

Are you the type who mainly prefers to **build** a computer system or to **use** it?

It's important to know ... from the start.

There are a number of computer kits available. Most of them are comparatively simple to assemble for a handy person with some sort of experience. But they may be rather time consuming. Sooner or later you get involved in purely electronic questions: trouble shooting, interfacing of different units, or quite simply new design.

If you want to concentrate your interest on programming and using the computer, then you should buy a complete system in full working order.

If possible try out several different computer systems before deciding.

5 *Make up your mind what you want.*

The main question is: how are you going to get hold of some form of computer power? Here are some of the possibilities:

Buy some kind of second-hand commercial computer.

Borrow computer time at work, at school, or from an acquaintance.

Invest in a personal computer of your own of the type that has been developed mainly for private use.

Evaluate these alternatives and compare them. They all have their pros and cons.

6 *Check your wallet.*

Go through your budget, and cut your cloth according to your purse. When you see all the good things coming on to the personal computer market it's all too easy to lose your head.

Make up a financing plan **before** you start buying your computer stuff. Try to see what you will need to invest a year or so ahead. Finances **are** important. You will soon find yourself with the same kind of costs as a yachtsman, a motor-boat owner, a motorcycle enthusiast or a hi-fi convert.

7 *Look ahead.*

At the present rate of development, today's ideas will be out of date in a year or even a month and prices are falling. You are getting more and more for your money. But that's no reason for waiting. Try instead to get an idea of how developments are going. That way you'll stand a good chance of avoiding expenditure on things that have no future.

The fastest development is taking place on the electronics side, in highly integrated semiconductor technology. There are new and better processors, but above all larger and faster semiconductor memories.

On the peripheral side things are moving slower. Keyboards still cost the same as last year; so too do tape recorders and cassettes. However, prices are beginning to drop on printers, and mass storage, such as floppy disks, are getting cheaper all the time.

8 *Try to find a partner.*

Sooner or later you will need help — in finding a fault, in learning some operating trick or other, or in developing some specific application.

Maybe you will need some programs.

At such times, two's good company! Especially if you find a good friend who has the same system and ways of looking at things. This will save you both a lot of trouble. And you will both make faster progress. (Some people even club together to buy a system. If you are on really good terms then this is quite a good way of sharing the capital costs.)

9 *When the system is up and running — start using it.*

Once your new computer system is set up and everything seems to be working properly, you may perhaps ask, "What do I do now?"

The answer is "Use it, of course!"

All too often, it seems, many personal computers are used to run test programs or various types of demonstration program. This is rather like looking at the test picture on the TV or scraping, painting and equipping the boat without ever sailing it! For many people, the reason for this kind of hesitation may be the lack of suitable program material.

Take it easy! Now you've come this far you have started to look at life differently. You've got a working computer system at your fingertips. Applications which those without a computer cannot even imagine will occur to you. Not everybody wants to learn how to program. If you do, then have a look at what other people are doing. Most colleges, polytechnics, computer clubs etc. run courses on BASIC programming.

Call them and see what's going on. When you do start programming, remember that often the best way to program is plagiarism!

10 *Get other people interested in computer technology.*

Look at your own changing interest in computers. Once it was only lukewarm — maybe even unconscious; now you are an active computer user, full of enthusiasm at all the possibilities this technology places at your disposal. Nobody's a better advertisement for personal computers than yourself.

Demonstrate your own system to friends and acquaintances. Get hold of a suitable set of demonstration programs. Tell people about them and answer their questions. In return, you may get some new ideas for applications for your own system.

CHAPTER 7

What can I do with it?

There are many fascinating applications for cheap micro-computers. It is practically impossible to make a complete list — new ideas are flooding in all the time. Each application of personal computers places individual demands on equipment and programs. A personal computer system for book-keeping etc. has quite a different design from the computer you buy to control the central heating at home.

Games

It is quite fashionable among so-called computer experts to run down the playing of games on a computer. This is rubbish! One of the easiest and most pleasant ways of learning how to program is to get hooked on a game, and then want to learn how it works. Among the reasons given by many computer professionals for joining Computer Clubs is that it is not possible to play games on the computer at work! Certainly programming for graphics, or presenting information in a way that anybody can understand it, is a very important part of computer development. The greatest boost to speech synthesis, and integral sound capabilities, was the need to amuse the operator!

Games do provide light relief though, and the value of this must not be underestimated. Many a sceptical, or concerned, user has been converted to the computer by such means. If the computer you have chosen is to be used in the home, it most certainly must have a games capability. That will mean ability to have games joysticks attached, and possibly sound output. A colour graphics video output is also desirable.

Computer games can also be "mind stretching". Most desktop

systems can now play a passable game of chess, backgammon or bridge. In the latter case it's just a matter of turning the machine on and loading the program. You don't even have to hunt around for three other people!

Many magazines devote a large proportion of their editorial space to games programs, and this is usually a way to keep your library up to date with the latest developments. Program libraries available commercially on cassette or disk are usually fairly expensive.

The great thing for both educational and fun use is the interactive nature of the computer. In computer aided instruction (CAI) it will not make you go faster than your learning capability. In games you are playing with "somebody" who will usually prove a worthy opponent — at whatever level of difficulty you agree on!

Education

One of the major areas of interest, and one place where children are being introduced to aspects of computing, is in education. Unlike the Americans most Europeans put educational use of the personal computer above most others. With systems like the Sinclair ZX81, costing under £100 and the Commodore PET at under £500, schools and colleges are able to enter computing for a modest amount. Even where budgets are being slashed, the need for a computer is high on most school's equipment requirements.

The use of computers in schools is limited by the knowledge of the teacher. Cheap computers, though, can be bought by students and used in the home and this is where a lot of adults will use them for the first time. It's the calculator effect all over again — and with computers now available for the price of a calculator five years ago (in real terms) they will probably have a similar effect.

Programs for use in a school environment can be divided into five categories. Drill and practice; simulation/modelling; games; tutorials; and a combination of these four.

Drill and practice programs are used to develop and master skills associated with specific objectives. Maths, spelling, syntax

Geometry being taught here with the help of the PET computer.

etc, are typical examples. The student is expected to continue with the program until the program has been mastered. Unfortunately students soon get tired or frustrated with this type of program.

With simulation/modelling programs, which are more difficult to write, real world events are simulated or modelled. It is not necessary for the students to physically encounter the actual problem. Data collected from real events is often entered into the programs, decision making skills are developed and group discussion evolves.

Games programs are fun — as previously mentioned. If educational games programs are well designed they assist students in developing their thinking skills.

Tutorial programs are probably the most difficult to write. Students are expected to acquire specific knowledge through well designed frame sequences. They are usually designed for a specific student.

There are thousands of these programs around. Most, unfortunately, are rubbish as software development has not kept up with hardware. In America there are many suppliers with long lists of computer aided instruction — CAI — software but little has reached across the Atlantic. The software that is available is fairly poor at the moment, although there are some good programs available in the maths and science area. This is to be expected, of course, but it is only a matter of time before programs for language learning, for example, get better.

Business Use

When we come to consider business use, the ways of using a computer are virtually infinite. Many domestic finance programs exist for keeping a check on your bank balance; calculating income tax returns; acting as an address and telephone file; simple diary-keeping; and even cataloguing books and records!

In a commercial environment the investment of a few thousand pounds will usually pay for itself within one year.

Book-keeping — in all its forms — is an obvious job for computers. You only have to look at the development in the business world: large and medium-sized companies make use of computer technology. Inexpensive personal computers now make it possible for even small companies (including one-man businesses) to rationalize their book-keeping, order processing, current-account ledgers, invoicing, salaries, stock accounting, budget, etc. Many of the people who have embraced personal computers are small businessmen: doctors, consultants, shopkeepers, stockbrokers, garage proprietors and others.

One thing is clear, however: relatively simple personal computer systems will not get you far. This is because, for the system to work in a rational and convenient manner, it needs a lot of storage. The handling of data files separately from the application program itself is also necessary. Many personal computer BASIC interpreters can't manage this. Above all, well written — and easily modifiable — software is needed.

The Apple Stock Quote Reporter receives a current display of stock prices and allows the operator to place a corresponding bid. The Reporter also ascertains the closing price, the high and low price for the day, and the current volume of the stock. Stock portfolios can be easily changed to suit the individual operator.

The minimum system for a small company should contain the following:

A fast CPU based around a common microprocessor;

At least 32K but preferably 64K of user RAM;

A reasonable printer. If letters, invoices or other "official" stationery are going to be printed, a daisy-wheel type printer is essential;

A video display unit with at least 64 characters/line;

Dual floppy disks with a capacity of more than 200K;

A good disk operating system that is "portable", or at least popular and well-supported, e.g. CP/M; Flex; PET; Apple etc.

A decent, flexible BASIC interpreter;

And plenty of applications software, carefully tested and with good documentation.

When choosing software it is very important to consider the following; ease of tailoring and accounting system used. In the first case most programs will need tailoring for your particular use. If a £100 program is going to cost £500 to get going for your particular application, it is going to be cheaper to buy a £400 program and pay £100 for tailoring. In general if a program needs modifying by more than 10% forget it.

It is sometimes worthwhile paying for a program to be written especially for you. In most cases this will cost as much as the machine. But if you can justify it in terms of eventual saving then it is probably the best thing to do. Do not think that a £15 invoicing package will be capable of reproducing a £15,000 mainframe package. It won't!

The second point, accounting system used, is very important. Many American packages conform to American accounting procedures and will not be any use in Europe. This is specially true with packages for calculating tax. In fact for VAT calculations many programs sold in Europe do not meet the Inland Revenue standards. If you want to use your computer for tax calculations find out whether it is acceptable to the tax man first.

Word processing

Next to business accounting, word processing is the fastest growing area of use. Information handling is being revolutionized by the personal computer.

It may be simple enough to write a letter by typing but an attractive and faultless text is considerably more difficult for most of us.

A word processing system built around a cheap personal computer makes the whole procedure much simpler and reduces frustration.

The concept of word processing means that you type the text in the normal way on an electric typewriter or VDU connected to the computer. With the assistance of the computer you then correct and edit the text until it is exactly as you want it. After this the computer supplies a perfect printout, with as many copies as needed, and the text can be stored for later use.

Professional word processing systems usually cost anything up to £10,000 or even more. For personal use you can get a long way for considerably less.

This is what's needed ...

Let's see what is needed to turn the personal computer into a word processing system in mini form.

First of all: *suitable terminal equipment* for handling input, editing and printout of text. There are several possible arrangements for such terminal equipment. One example: an electric typewriter of the IBM Selectric type modified for use with a computer system plus a video printer. A VDU with a decent printer will also be acceptable.

Text input takes place at the keyboard, editing on the screen and the output comes on the printer.

The printer is really the bit that costs money in word processing systems, especially if you want attractive copy.

And then you need a suitable program — a *text editor.* This makes it possible to produce the text body on the screen on command, to correct spelling errors, erase, add, exchange sentences or paragraphs and get a printout made. There are many different types of text editors, both simple and more complex ones.

One word processing program which is becoming very popular, especially on CP/M machines, is Wordstar.

Wordstar has on-screen visual text composition and dynamic justification and re-margination. There is no need to hit "return" at the end of a line, for example, as Wordstar moves you to the next line. The preceding line is re-displayed justified to the left and right margins. You can centre in a line with one keystroke and set boldface or underline even in mid paragraph. The text can be rejustified to new margins when necessary. Dynamic pagination shows the printer page breaks during text entry, correction or review. Page headings, page numbers etc. can be easily entered. Selected pages, pauses between pages and headings can be inserted during printing. This comprehensive system costs around £200 and is available on an 8″ disk.

"Complete Business System" from Micropower.

One example of how you can have practical use of your own word processing system: letters. For a letter, you write your rough copy, edit it on the VDU until you are satisfied, and then get a fair copy printed out. When it is a question of a large number of letters that are basically the same, you also input an address list of the persons who are to be sent the letters. It is also possible to input particulars for each individual case. In this way each letter will be a personal one despite the fact that the printout has been done with the aid of the computer. Very time-saving.

In the mass memory of the system you can store a number of standard phrases which can be called up when needed and put together, for example for a contract. A newspaper article or a memorandum can be composed and edited without the heap of unsuccessful attempts filling the waste paper basket. The paper only comes into the picture in the final phase.

A lot of letters we get have been addressed by a computer. The personal computer can be used for the same purpose: for storing, updating and printing out names and addresses. It may be a list of members or customers — or perhaps even a list of Christmas cards to be sent.

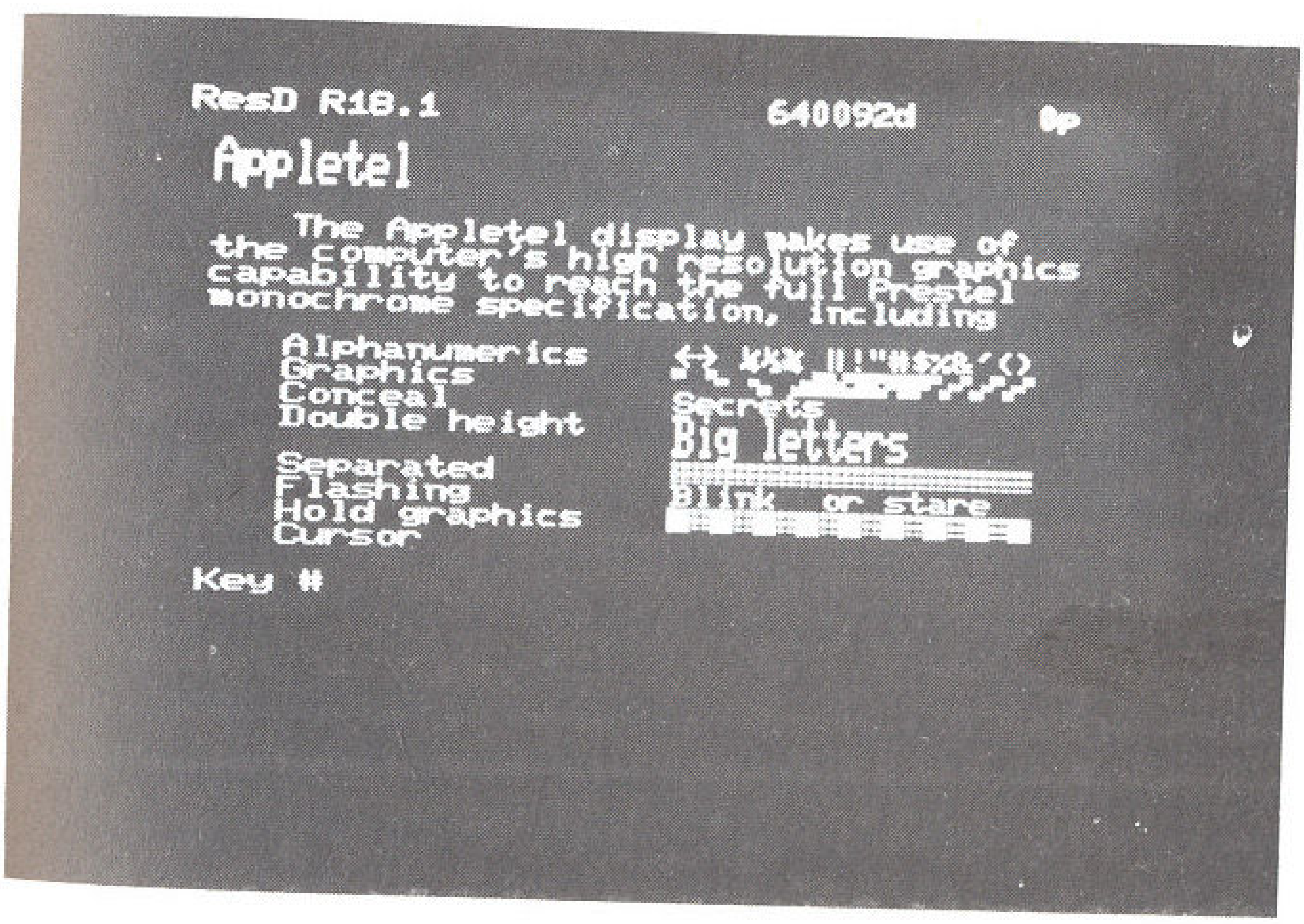
The word processing system — described above — can with advantage also be used for mailing registers. For the sake of mailing registers alone, though the software does not need to be all that advanced. A good mailing-register program should of course contain search routines so as to make it possible to find a certain person in the register, or to print out all those with a certain code designation or postal code, or to sort in alphabetical order (good for lists of members) and so on.

Another very popular word processor is Word-Pro. This is available from Commodore and is designed to run on their very successful PET computer. Although initially designed as a "home" computer — and unfairly dismissed by its competition as a toy! — the PET, and similar computers, make a good starting point for a small system. Word-Pro does most of the things that Wordstar does — and a few more. Unfortunately it is only designed for the PET, and is therefore not portable. This is the decision that has to be made when choosing a system for business use — portability and expandability — or off-the-shelf. It is similar to buying a hi-fi system — the choice is between separate units, with all the problems of matching impedances and leads hanging around, or a music centre with everything in one box.

Information handling

This is going to be the growth industry of the 80s — and the personal computer is at the heart of the exercise. The Post Office and broadcasting authorities are spending millions on giving us an information retrieval system that will revolutionise our lives. Prestel, Viewdata, Ceefax, Oracle, Antiope, Captain or ... — call it what you will they are all basically similar in concept. Many personal computers will soon have the capability to retrieve and display the information stored by the central computer. Apple have a teletext/viewdata card that plugs into one of the I/O ports. Some personal computers designed in Europe have been designed from the start to plug into the national and international networks. One problem in Britain is the need to get PO approval before any device can be plugged into the telephone system.

An interesting system is made by B&B Computers called the BEE. This is designed around a viewdata module. An internal modem is at the heart of the system. Other companies are getting on the bandwagon, and S-100 cards are available, as well as units for Commodore PETs and TRS-80. To appreciate and use teletext or viewdata to the full, however, colour must be available and this does rule out many American or Japanese systems.



Apple II Plus Computer + Appletel Prestel Package. Shown here is a typical Prestel screen display which is enhanced by Apple's high resolution graphics capability.

Once you've got a computer that can handle these information retrieval systems, is it useful? It must be possible to integrate the information into the applications required. Expenditure of a couple of hundred pounds on a teletext decoder would be better spent on having it inside a domestic TV. If the TV is used as the display unit, however, then it is better to have the decoder inside the computer. Most people will use viewdata/teletext as a simple data base. It is possible to use it in a quasi-interactive mode — really interrogating the data base to selectively get information. In fact many large firms are getting their own viewdata systems to hold their company information.

Multi-user systems

A multi-user system allows a number of different users running different applications to use the computer at the same time.

There are a number of methods to set up a multi-user system for business use. The simplest is to connect a number of microcomputers — say PETs or Apples — together using various networking devices.

There are two basic types of network:

A "ring" network is a loop of wires which connects all the computers on the loop. Data for transmission is put into the ring at one point — the source — and taken off at another — the destination. That means that each point — or node — on the ring needs to have some form of identification — or specific address.

Each node has to recognise information destined for it and also generate information in such a way that the desired destination recognises it. It is usual to allow any two or more points on the ring to converse with each other. The destination therefore needs to know who is sending the information and there can be two successive transmissions to one destination.

The other form of network connects the computers in the system in a pattern rather like the spokes in a wheel. Thus a central controller is needed through which each computer must communicate.

Interfacing computers in the network via disk drives is one of the most popular methods for personal computers.

Two systems are based on the Corvus Winchester disk. The simplest is called the Miracle and is constructed around the Corvus Constellation network. In addition to his own dedicated computer and associated memory, each user has access to a central hard disk memory, without interference from other users.

Miracle also makes it possible for similar computers to communicate with each other in the network and for peripherals to be shared.

The Miracle system can connect a wide range of micros in any combination. These include the Apple, Alpha, Altos, Superbrain, TRS-80 Models I and II, Digital Equipment LSI-11 and all S-100 bus computers.

Sharing

Peripherals — such as printers, VDUs, devices for speech output and voice recognition, colour graphics, light pens and digitisers — can also be linked. A single-level Miracle network can have up to eight computers and a two-level network allows up to 64 computers to share the disk memory.

A device called the Constellation shared data multiplexer is used to connect the computers in the system to 10 or 18 megabytes of Corvus hard disk memory. Using four disk drives, the total capacity of a Miracle system can be extended to 72 megabytes. Corvus will work with existing software and give a disk accessing speed which is normally 20 times faster than floppies.

Miracle makes use of a range of computer languages, from BASIC, COBOL and FORTRAN to Pascal, ALGOL and APL.

The latest local networking facility from Corvus is called Omninet. It can link up to 64 different or identical microcomputers or varied peripherals.

According to Corvus, Omninet is 100% compatible with the Constellation. Corvus anticipates that many users of its large installed base of Constellation units will opt to combine that system with Omninet.

A similar system, from Zynar — called Cluster/One Model A or Nestar — is also designed for the Apple. Information can be transmitted from machine to machine directly, without going through disk files and without the need for modems.

Shared access

All stations share access either to private data or to a common read-only library of programs and data, and they can update databases simultaneously.

All information can be shared among users or, using the protection mechanism of the system, can be password-protected so that only one designated individual or group has access to special data. As an example, a number of people may have permission to read specific data but may not have the ability to modify that data.

In the Nestar Model A network, a standard Apple with 48 kilobytes of processor memory can function as the dedicated central controller/mass storage manager, with connections to the Nestar storage subsystems and to other Apples which serve as user stations. Each station is connected to the ClusterBus which can be up to 300 metres of 16-wire ribbon cable or standard round cable.

Programs written in languages supported by Apple — Applesoft, BASIC, Integer BASIC, machine language, Pascal — will function without change in the Model A. Any hardware in the Apple universe obeying Apple standard conventions can be used. Most applications in automation, accounting, legal or small business use can be put to work immediately without reprogramming.

The Model A has utility programs for initialising new floppy disks, making back-up copies and copying certain data selectively for off-line retention. Program files, data files, binary files, picture files and random access on sequential text files may all be saved on the disk.

Nestar supports multiple printer stations. An application program can run in a user station and can search for files prepared for printing. It can then print those files with any Apple-compatible printer. The system can be upgraded in the field by adding more individual stations, up to 65 units, at any time.

Nestar consists of a compact metal box with two double-sided 8″ floppy disks with formatted available storage of 1.26 million bytes; the necessary electronics which plug into the central controller Apple; the ClusterBus communication card; the communication cards required for each Apple station; and all necessary software. Also available as an option is the Nestar hard disk drive with either 16.5 or 33 million bytes of formatted available storage in a single unit.

The user station communication cards contain 2 kilobytes of ROM, 1 kilobyte of RAM and all necessary bus electronics. They are compatible with standard Apple interface cards, including those for mini-disks, serial and parallel printers, modems, sound and graphics tablets.

The ClusterBus communications card plugs into any of the peripheral connectors inside the Apple and is perceived by the Apple to be an Apple mini-disk controller card and thus will be initialised automatically when the Apple is turned on.

The Apple is not the only microcomputer with at least two network options. The PET has the Mu-PET and the KC Netkit. Both use a hardware approach, with the Netkit having some software in ROM — firmware.

Essentially the KC Netkit is a hardware/firmware package which allows a relatively inexperienced programmer, assisted by 10 new BASIC commands, to achieve configurations without recourse to tedious machine-code routines. The PET can act as a smart or dumb terminal, which can accommodate virtually any protocol and character conversion the user may require.

Additionally, by allowing the serial port to equate with the keyboard, the PET can exchange data and program files freely and can also be controlled remotely, opening a wide range of high-speed networking and also, more important, be grouped with others to achieve greater real-time number-crunching.

The firmware includes re-location routines which avoid clashes with other devices demanding access to memory. The expansion bus, on which the Netkit unit is placed, is duplicated, and the only way one can recognise that an apparently standard PET has been modified is a small connector clipped unobtrusively into the expansion port.

Mu-PET allows up to eight PETs to share one or more Commodore disc drives and any compatible printer. The hardware runs via the standard PET-IEEE bus and each PET accesses the disk as if it were the only user.

Tandy has a fairly primitive system which allows a teacher in a classroom to switch each computer into a master disk unit. That is obviously not so sophisticated as other manufacturers' products and should not be considered as a true network.

There are many other developments which will allow multi-user access to information. Systems using the disk operating system CP/M 2.2 should be converted easily to its multi-user equivalent MP/M. As that is a slightly different concept from a dedicated system based around a specific microcomputer, it will not be considered here.

More expensive microcomputer systems — costing around £7,000 or more — usually have the multi-user system built in.

Controlling things

Most of the systems considered in this book have at least one programmable input/output port. This allows information to be sent to and from the computer in the ones and zeros needed by the CPU, and thus lets all sorts of things to be connected up to the computer.

Simple things like switches need little interfacing as they are essentially binary devices. Thus burglar alarm systems lend themselves easily to computer control. In the other direction things can be switched on and off fairly simply by pulses — suitably amplified — coming from the computer. Those systems with internal timers or clocks obviously help there.

If other signals are required special interfaces called analogue-digital converters are needed. These convert varying voltages into equivalent digital signals as required by the computer. Thus it is possible for the computer, for example, both to simulate sound and also to manipulate and process it.

PERSONAL COMPUTING FOR PROFESSIONALS — The HP-85 computer from Hewlett-Packard, is designed for personal use in business and industry by professionals such as engineers, scientists, accountants, and investment analysts.

Computers with control buses — like the IEEE 488 — can be used to control, and get information to and from various instruments. Thus they become intelligent controllers in a laboratory.

One area where personal computers can be fun is in robotics. A number of companies sell small robots that can be operated from small systems. Intelligent machines are obviously going to become more important in the future and, again, a small computer system is ideal as an entry into this area.

Making money

One factor in choosing a system, or even buying one in the first place, is the possibility of earning some money with it! There is still a lack of good programs at the moment. This is the first area to look at. If you're good at programming then the purchase of a well-known and popular system could be the key to a prosperous future! Many people are now making a living writing software for the major software distributors. These companies work on a straight royalty basis. 10% on a £10 program works out a reasonable amount even if only 10 a week are sold!

Many local retailers do not have the resident software expertise they need. If you are good at tailoring — or are an accountant, for example, with programming ability — then your local retailer will probably welcome you with open arms.

Advertise! Most of the computer monthlies have special advertising slots for the one-man band. These cost about £20 and usually pay for themselves quickly. Contact your local computer clubs. If they have a relatively "high profile" in your locality then those with software problems will contact them at sometime for help. If the secretary or chairman knows your name and area of expertise then he will pass it on.

If you have hardware experience then helping people get their kits working is one area of need. Or selling the "boxes" that are needed to interface various systems together. A number of companies are doing very well selling interfaces that convert, say IEEE 488 signals into RS232C, and vice versa. Small control systems are also worth investigating — as well as voice recognition and speech synthesis. Most of the early work in this area was done by one or two engineers in their spare bedroom or garage. It certainly isn't worth trying to design a computer system from scratch now — not unless you have a few hundred thousand pounds to invest anyway. There are just too many suceessful companies around.

Examples of pesonal computers in use

Some examples of the uses to which personal computers have been put will illustrate how versatile a system can be. It is clear that the number of uses is only limited by the human imagination.

i) A college in London has adapted a Nascom 1 computer so that severely handicapped people can communicate effectively. Two foot operated switches move a cursor around a matrix of letters and characters. These can then be assembled on the video display showing the message required.

ii) The PET computer has been programmed to carry out many of the functions in hotel reservations and billings. The software allows the PET to keep a record of room availability — with immediate viewing and blocking for up to 12 blocks of rooms, 400 days ahead with up to 99 rooms per block. Entering a date gives screen display of availability of all blocks for that date and following 6 days. Guest billing is also available and caters for up to 200 rooms. Final accounts include service charge and VAT. A printer can be used as an integral part of the system.

iii) The best example of financial modelling is Visicalc, a forecasting program that was a worldwide best seller. Virtually any problem that can be entered in tabular form can be solved using Visicalc. A recalculation feature allows the user to change only one figure for the program to adjust all the others automatically. This enables one to ask a series of 'What if?' questions such as 'What if interest charges increase by 3%?' or 'What if sales fall by 12½%?'

Visicalc is essentially a general purpose utility and it is widely used for budgeting and financial planning as well as in engineering and scientific applications. It has been described as software which allows you to program without having to learn a programming language. (More sophisticated utilities are under development, including a number of program generators which, given precise instructions, will automatically write programs to suit your purposes. But contrary to some claims, these program generators do require some knowledge of systems analysis in order to prepare the correct instructions.)

iv) Solicitors are using a number of packages to carry out the following applications: time recording and costing; financial accounting; debt collection and word processing. Other areas being considered are information retrieval, conveyancing support and probate support. The time recording and costing relies on each fee-earner maintaining a daily or weekly time-sheet. The information on the time sheets is fed into the computer and stored in a client/ matter sequence. A fee-earner can then, subsequently, enquire on any individual matter to see how much time has been spent — and how much it costs! Solicitors have found their income has been underestimated by up to 50% on non-computerised systems. The legal accounting system involves a vast number of different types of transactions and these are governed by very strict rules. The computer can make this process easier — and more accurate. Debt collection is a natural spin-off from accountancy and the uses of word processing, especially when dealing with repetitious documents, are obvious.

v) A public relations consultancy had a package written for them to handle most of their daily routines. This includes an activity planner, mailing system and accounting tasks. The activity planner gives a daily check-up, client by client, with matters pending according to client, date and type of activity. Clients are sent end of month summaries as activity reports. The mailing system allows up to 300 names and addresses to be categorized in up to 100 ways. Hence selective mailings are possible. The accounting tasks include clients' costing accounts. Output is available on a dot-matrix or daisy-wheel printer, and can be used in conjunction with a word processor package.

vi) Estate agents have been offered a package on the Apple II. There are routines to handle the entry of new property details, amend and update these details, enter potential purchaser details and a sales checklist. The program also gives a home sale negotiation analysis, and allows routine letters and transfer of records to archives. It is claimed to handle an unlimited number of records.

vii) Telesoft has been developed by the IBA for use with adapted TV sets receiving Oracle teletext transmissions. This allows schools and colleges access to programs held by the IBA's central computer and broadcast with as-normal teletext characters. When the TV set receives the software it puts it into RAM and this becomes available for use on any computer plugged in to it. It is designed mainly for educational use at the moment.

viii) A kitchen outfitters are using an Apple II to help design layouts for kitchen designs. It replaces the traditional drawing board. The computer gives both speed and accuracy and also appeals to customers. The Apple's high resolution graphics are used, with sub-assembly plans held on disc. This allows customers to move units around and see them displayed on a screen immediately. Five programs are available: plan and design the kitchen; price and design a kitchen for any range of any manufacturer; draw and calculate tile requirements; draw the plan of the kitchen on to a graph plotter or daisy-wheel printer; or design the worktops. To draw an average square or L-shaped room takes two minutes and around 15 minutes to design the plan.

ix) A pharmacist in Hertfordshire is using a PET computer to keep track of his prescriptions. Not only does it do stock control, but also prints labels — in a way that customers can read easily! There are plans to link the dosage with a data bank so that over-prescription is avoided. Most of the information is held in code, so that the operator can produce a complete label with only a few key strokes. The pharmacist is also investigating the possibility of using the computer to check for drug incompatibilities.

x) A group of independent lorry drivers in the States have got together to use a computer based information system to help them to be more efficient. A central office uses the computer to link the various truck stops — and the drivers who frequent them — with customers having loads for carrying. The system interfaces directly with the telephone network. A customer calls the computer and uses the telephone keys to communicate their latest load information — both for new jobs and updates on old jobs. The computers, located at key truck stops display the information on a monitor. The drivers then scan the monitor, identify jobs they may be interested in hauling and person to contact. There's even an up-to-date information listing where and when fuel is available on the road ahead. The truck driver is saving time and money and has immediate access to available work, wherever he may be. And the customers are getting maximum exposure to the independent lorry drivers.

Well, these are just a few of the uses for personal computers. You'll agree that they are very varied — and the surface has only just been scratched!

Now that you've been fired with enthusiasm, go out, get yourself a system and contribute to the fund of knowledge available!

APPENDIX A

Binary arithmetic — or thinking in 1's and 0's

Almost as soon as we get to school, we have to learn the decimal number system with digits, 0-9. This means that we know how much 27 is, or 56, or whatever number.

The computer is designed to calculate with the binary number system which has two digits, 0 and 1. Despite this, the computer has no difficulty in expressing numbers like 27 or 56.

This is how it works:
$27_{10} = 11011_2$ Marvellous isn't it? (The little 2 or 10 denotes the number system used.)
How can it be 11011? Let's first take a look at how the number 27 is built up in the *decimal number system* ...
27 in decimal actually means $2 \times 10^1 + 7 \times 10^0$, where 10^1 is 10 and 10^0 is 1. Similarly, 10^2 is 100 and 10^3 is 1000. (The little number indicates the number of zeros after the one.) This is sometimes called the multiplier. We can therefore make a simple table:

Multiplier	10^1 (10)	10^0 (1)
Digit	2	7
Product	20	7

which, added together, give 27.

Similarly, the number 1234_{10} really means:
$1 \times 10^3 + 2 \times 10^2 + 3 \times 10^1 + 4 \times 10^0$.

Exactly the same method is used with the binary system, except that the 10 is replaced by a 2. Therefore 11011_2 really means:

$$1 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \quad \text{or}$$
$$1 \times 16_{10} + 1 \times 8_{10} + 1 \times 2_{10} + 1 \times 1_{10}$$

which should give 27_{10}.

A practical method of converting 27 to a binary number is as follows:

27/2 = 13 remainder 1
13/2 = 6 remainder 1
6/2 = 3 remainder 0
3/2 = 1 remainder 1 ↑ READ UP
1/2 = 0 remainder 1

1 1 0 1 1

Then what does the decimal number 56 become in binary code? The same method gives a quick answer: 111000.

Adding binary numbers

There are four extremely simple rules for addition: 0 + 0 = 0, 0 + 1 = 1, 1 + 0 = 1, and 1 + 1 = 10. we express the last rule like this "one plus one is one zero".

This table illustrates the addition of binary and decimal numbers:

Binary		*Decimal*	
0 + 0 =	0	0 + 0 =	0
0 + 1 =	1	0 + 1 =	1
1 + 1 =	10	1 + 1 =	2
10 + 1 =	11	2 + 1 =	3
11 + 1 =	100	3 + 1 =	4
100 + 1 =	101	4 + 1 =	5
101 + 1 =	110	5 + 1 =	6
110 + 1 =	111	6 + 1 =	7
111 + 1 =	1000	7 + 1 =	8
1000 + 1 =	1001	8 + 1 =	9
1001 + 1 =	1010	9 + 1 =	10

Octal, Hexadecimal — or how do you pronounce 1001000111110010?

Even if the computer thrives on such an expression, it's very inconvenient for human beings to deal with!

It's much simpler if you make the binary number a number with the base 16 or 8, i.e. sixteen or eight symbols instead of the 2 for the binary system or the 10 for the decimal system.

A system with the base 8 is called *the octal number system* and has eight symbols: 0 1 2 3 4 5 6 7.

The base 16 gives what is called the *hexadecimal system.* This has sixteen symbols: 0 1 2 3 4 5 6 7 8 9 A B C D E F.

This is how these number systems correspond to the binary system:

Binary	Octal	Binary	Hexadecimal
000	0	0000	0
001	1	0001	1
010	2	0010	2
011	3	0011	3
100	4	0100	4
101	5	0101	5
110	6	0110	6
111	7	0111	7
		1000	8
		1001	9
		1010	A
		1011	B
		1100	C
		1101	D
		1110	E
		1111	F

It is simple to convert a long binary number, say 11101011, to octal form:

11 101 011

3 5 $3 = 353_8$

We just group the binary number in threes, and express each group of three as its equivalent octal number.

Converted into hexadecimal form:

1110	1011	Here we group the binary number in fours and then express each group of four as its equivalent hexadecimal.
E	$B = EB_{16}$	

How about the long binary number in the section heading?

Well it's certainly easier to remember $91F2_{16}$ than $1001\ 0001\ 1111\ 0010_2$!

Generally it's wisest to learn to use the hexadecimal number system, or hex as it's usually called. The octal form is less common. This is because each symbol in hex expresses four binary bits. Since most computers are constructed to handle data in groups of eight bits (i.e. one byte), one byte can be simply and exactly expressed by two hex symbols; 00_{16} is a byte with nothing but zeros, FF_{16} is a byte with nothing but ones.

The addresses consisting of sixteen bits can conveniently be expressed by four hex symbols, e.g. FA01.

The ASCII Code (a practical use of binary)

ASCII, the American Standard Code for Information Interchange, is the standardised code normally used for communication between the computer and the peripheral units. While human speech is relatively sophisticated a computer only understands machine language, i.e. zeros and ones. ASCII serves as a translation between man and machine (and vice versa).

If we strike the A key on the keyboard, the internal circuitry of the keyboard effects a transformation, according to the ASCII rules, into machine language: 1000001. Striking the M key gives binary 1001101.

And in the other direction, if the terminal printer is to print for example the digit 8, the computer has to feed it with the binary code 0111000.

The letter combinations in the table are machine instructions which are not printed out.

TABLE A–1. ASCII Code

most significant bits

				b_7 b_6 b_5 →	0 0 0	0 0 1	0 1 0	0 1 1	1 0 0	1 0 1	1 1 0	1 1 1
bits b_4 ↓	b_3 ↓	b_2 ↓	b_1 ↓	row ↓ column →	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	'	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	″	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	★	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	;	K	[	k	{
1	1	0	0	12	FF	FS	,	<	L	\	l	¦
1	1	0	1	13	CR	GS	—	=	M	]	m	}
1	1	1	0	14	SO	RS	.	>	N	^	n	~
1	1	1	1	15	SI	US	/	?	O	—	o	DEL

least significant bits

NUL Null
SOH Start of Heading (CC)
STX Start of Text (CC)
ETX End of Text (CC)
EOT End of Transmission (CC)
ENQ Enquiry (CC)
ACK Acknowledge (CC)
BEL Bell (audible or attention signal)
BS Backspace (FE)
HT Horizontal Tabulation (punched card skip) (FE)
LF Line Feed (FE)
VT Vertical Tabulation (FE)
FF Form Feed (FE)
CR Carriage Return (FE)
SO Shift Out
SI Shift In

DLE Data Link Escape (CC)
DC1 Device Control 1
DC2 Device Control 2
DC3 Device Control 3
DC4 Device Control 4 (Stop)
NAK Negative Acknowledge (CC)
SYN Synchronous Idle (CC)
ETB End of Transmission Block (CC)
CAN Cancel
EM End of Medium
SUB Substitute
ESC Escape
FS File Separator (IS)
GS Group Separator (IS)
RS Record Separator (IS)
US Unit Separator (IS)
DEL Delete

CC: Communication control
FE: Format effector

IS: Information separator

APPENDIX B

Interface standards

S-100

The original interface called the S-100 bus was proposed by the makers of the Altair computer. It was rapidly adopted by other manufacturers — but most introduced their own variations. A few years ago the Institute of Electronic and Electrical Engineers (IEEE) in the United States started work on a definitive standard for the bus. This expanded the original idea, and provided pin allocations so that the new generation of 16-bit microprocessors would run on it. The IEEE standard allows up to 16 16-bit processors to run simultaneously — and consequently has a good interrupt handling capability. Figure B1 shows the pin allocations, with brief descriptions of each pin usage.

Fig B.1
S-100 bus pin list

PIN NO.	SIGNAL & TYPE	ACTIVE LEVEL	DESCRIPTION
1	+8 VOLTS (B)		Instantaneous minimum greater than 7 volts, instantaneous maximum less than 25 volts, average maximum less than 11 volts.
2	+16 VOLTS (B)		Instananeous minimum greater than 14.5 volts, instantaneous maximum less than 35 volts, average maximum less than 21.5 volts.
3	XRDY (B)	H	One of two ready inputs to the current bus master. The bus is ready when both these ready inputs are true. See pin 72.
4	VIO*(S)	L	Vectored interrupt line 0.

PIN NO.	SIGNAL & TYPE	ACTIVE LEVEL	DESCRIPTION
5	VI1*(S)	L	Vectored interrupt line 1.
6	VI2*(S)	L	Vectored interrupt line 2.
7	VI3*(S)	L	Vectored interrupt line 3.
8	VI4*(S)	L	Vectored interrupt line 4.
9	VI5*(S)	L	Vectored interrupt line 5.
10	VI6*(S)	L	Vectored interrupt line 6.
11	VI7*(S)	L	Vectored interrupt line 7.
12	NMI*(S)	L	Non-maskable interrupt.
13	PWRFAIL*(B)	L	Power fail bus signal.
14	DMA3*(M)	L	Temporary master priority bit 3.
15	A18 (M)	H	Extended address bit 18.
16	A16 (M)	H	Extended address bit 16.
17	A17 (M)	H	Extended address bit 17.
18	SDSB*(M)	L	The control signal to disable the 8 status signals.
19	CDSB*(M)	L	The control signal to disable the 5 control output signals.
20	GND (B)		Common with pin 100.
21	NDEF		Not to be defined. Manufacturer must specify any use in detail.
22	ADSB*(M)	L	The control signal to disable the 16 address signals.
23	DODSB*(M)	L	The control signal to disable the 8 data output signals.
24	o (B)	H	The master timing signal for the bus.
25	pSTVAL*(M)	L	Status valid strobe.
26	pHLDA (M)	H	A control signal used in conjunction with HOLD* to co-ordinate bus master transfer operations.
27	RFU		Reserved for future use.
28	RFU		Reserved for future use.
29	A5 (M)	H	Address bit 5.
30	A4 (M)	H	Address bit 4.
31	A3 (M)	H	Address bit 3.
32	A15 (M)	H	Address bit 15 (most significant for non-extended addressing.)
33	A12 (M)	H	Address bit 12.
34	A9 (M)	H	Address bit 9.
35	DO1(M)/DATA1(M/S)	H	Data out bit 1, bidirectional data bit 1.
36	DOO(M)/DATAO(M/S)	H	Data out bit 0, bidirectional data bit 0
37	A10 (M)	H	Address bit 10.
38	D04(M)/DATA4(M/S)	H	Data out bit 4, bidirectional data bit 4.
39	DO5(M)/DATA5(M/S)	H	Data out bit 5, bidirectional data bit 5.
40	DO6(M)/DATA6(M/S)	H	Data out bit 6, bidirectional data bit 6.
41	D12(S)/DATA10(M/S)	H	Data in bit 2, bidirectional data bit 10.
42	D13(S)/DATA11(M/S)	H	Data in bit 3, bidirectional data bit 11.
43	D17(S)/DATA15(M/S)	H	Data in bit 7, bidirectional data bit 15.
44	sM1 (M)	H	The status signal which indicates that the current cycle is an op-code fetch.
45	sOUT (M)	H	The status signal identifying the data transfer bus cycle to an output device.
46	sINP (M)	H	The status signal identifying the data transfer bus cycle from an input device.
47	sMEMR (M)	H	The status signal identifying bus cycles which transfer data from memory to a bus master, which are not interrupt acknowledge instruction fetch cycle(s).
48	sHLTA (M)	H	The status signal which acknowledges that a HLT instruction has been executed.
49	CLOCK (B)		2 MHz (0.5%) 40-60% duty cycle. Not required to be synchronous with any other bus signal.
50	GND (B)		Common with pin 100.
51	+8VOLTS (B)		Common with pin 1.
52	-16VOLTS (B)		Instantaneous maximum less than -14.5 volts, instantaneous minimum greater than -35 volts, average minimum greater than -21.5 volts.
53	GND (B)		Common with pin 100.
54	SLAVE CLR* (B)	L	A reset signal to reset bus slaves. Must be active with POC* and may also be generated by external means.
55	DMAO* (M)	L	Temporary master priority bit 0.
56	DMA1*(M)	L	Temporary master priority bit 1.
57	DMA2*(M)	L	Temporary master priority bit 2.
58	sXTRQ*(M)	L	The status which requests 16-bit slaves to asset SIXTN*.
59	A19 (M)	H	Extended address bit 19.
60	SIXTN*(S)	L	The signal generated by 16-bit slaves in response to the 16-bit request signal sXTRQ*.
61	A20 (M)	H	Extended address bit 20.
62	A21 (M)	H	Extended address bit 21.
63	A22 (M)	H	Extended address bit 22.
64	A23 (M)	H	Extended address bit 23.
65	NDEF		Not to be defined signal.

PIN NO.	SIGNAL & TYPE	ACTIVE LEVEL	DESCRIPTION
66	NDEF		Not to be defined signal.
67	PHANTOM*(M/S)	L	A bus signal which disables slave devices and enables phantom slaves—primarily used for bootstrapping systems without hardware front panels.
68	MWRT (B)	H	pWR*—sOUT (logic equation). This signal must follow PWR* by not more than 30 ns.
69	RFU		Reserved for future use.
70	GND (B)		Common with pin 100.
71	RFU		Reserved for future use.
72	RDY (S)	H	See comments for pin 3.
73	INT*(S)	L	The primary interrupt request bus signal.
74	HOLD*(M)	L	The control signal used in conjunction with pHLDA to co-ordinate bus master transfer operations.
75	RESET*(B)	L	The reset signal to reset bus master devices. This signal must be active with POC* and may also be generated by external means.
76	pSYNC (M)	H	The control signal identifying BS_1.
77	pWR*(M)	L	The control signal signifying the presence of valid data on DO bus or data bus.
78	pDBIN (M)	H	The control signal that requests data on the DI bus or data bus from the currently addressed slave.
79	AO (M)	H	Address bit 0 (least significant).
80	A1 (M)	H	Address bit 1.
81	A2 (M)	H	Address bit 2.
82	A6 (M)	H	Address bit 6.
83	A7 (M)	H	Address bit 7.
84	A8 (M)	H	Address bit 8.
85	A13 (M)	H	Address bit 13.
86	A14 (M)	H	Address bit 14.
87	A11 (M)	H	Address bit 11.
88	D02(M)/DATA2(M/S)	H	Data out bit 2, bidirectional data bit 2.
89	D03(M)/DATA3(M/S)	H	Data out bit 3, bidirectional data bit 3.
90	D07(M)/DATA7(M/S)	H	Data out bit 7, bidirectional data bit 7.
91	D14(S)/DATA12(M/S)	H	Data in bit 4 and bidirectional data bit 12.
92	D15(S)/DATA13(M/S)	H	Data in bit 5 and bidirectional data bit 13.
93	D16(S)/DATA14(M/S)	H	Data in bit 6 and bidirectional data bit 14.
94	D11(S)/DATA9(M/S)	H	Data in bit 1 and bidirectional data bit 9.
95	D10(S)/DATA8(M/S)	H	Data in bit 0 (least significant for 8-bit data) and bidirectional data bit 8.
96	sINTA (M)	H	The status signal identifying the bus input cycle(s) that may follow an accepted interrupt request presented on INT*.
97	sWO*(M)	L	The status signal identifying a bus cycle which transfers data from a bus master to a slave.
98	ERROR*(S)	L	The bus status signal signifying an error condition during present bus cycle.
99	POC*(B)	L	The power-on clear signal for all bus devices; when this signal goes low, it must stay low for at least 10 msecs.
100	GND (B)		System ground.

(B) = Bus signals
(S) = Slave signals
(M) = Master signals

Preliminary—Subject to Revision

When looking at S-100 bus based equipment it is essential that *all* the boards follow the pin allocation at least! The standard also defines the minimum and maximum time for signals on the bus, and their interrelationship with one another. These are not considered here — but it is reasonable to assume that those manufacturers who use the correct pin allocations and claim to support the bus, do so properly. It is worth checking this with the retailer.

RS232C

The RS232C standard was specified by the Electronics Industry Association (EIA) and defines both the electrical and physical specifications for bit-serial transmission. It also defines the handshaking signals used to control standard telephone connection equipment and standard modems (modulator-demodulator).

The main signal lines are transmit data and receive data. With ground these are the three wires necessary to configure any system. The bit rate is defined in bauds, or bits/second. The standard rates are:

50, 75, 110, 150, 300, 600, 1200, 2400, 4800, 9600, and 19,200.

Teletypes usually use 110, 150, or 300 baud, whereas video display units use 600 or 1200, in small systems.

The voltage on the lines varies between plus and minus 12 v, although the voltage can go down as low as 3 v and still work reasonably well. A 25-pin connector is used, and the connections are shown in Figure B2. Only the primary fifteen connections are shown, the other ten being used for data and control paths for a second serial channel running at much lower speed. Although the second channel is hardly ever used, it is useful for providing the information for the modems connected at each end of the communication link.

Fig B2.

Pin Number	RS232C name	Description
1	AA	Protective ground
2	BA	Data transmitted from terminal
3	BB	Data received from modem
4	CA	Request to send
5	CB	Clear to send
6	CC	Data set ready
7	AB	Signal ground
8	CF	Carrier detector

Pin Number	RS232C name	Description
9		unassigned
10		unassigned
11		unassigned
12	SCF	Secondary received line signal detect
13	SCB	Secondary clear to send
14	SBA	Secondary transmitted data
15	DB	Tx. timing (from modem)
16	SBB	Secondary received data
17	DD	Rx. timing (from modem)
18		unassigned
19	SCA	Secondary request to send
20	CD	Data terminal ready
21	CG	Signal quality detector
22	CE	Ring detector
23	CH/CI	Data rate selector
24	DA	Tx. timing (from terminal)
25		unassigned

When used with modems on domestic telephone links the lines designated "request-to-send", "clear-to-send", "data-terminal-ready" are used to control the modem link.

Sometimes manufacturers specify a V24 interface instead of RS232C. This is essentially the same, the only differences being in the pulse shape. Another standard is the 20 mA current loop. This again is a serial transmission standard, but the voltage range can be up to ± 80 v. In this case the current passing through the circuit is held constant at 20 mA.

IEEE-488

The IEEE-488, or general purpose, interface bus — GPIB — was developed some eight years ago around a standard proposed by Hewlett-Packard in the United States. There are now around 1000 products using the bus — mostly in instrumentation. A typical system consists of a controller (e.g. the PET) and up to 14 devices (e.g. printers, recorders, etc.). Each peripheral device is designated a "listener" or a "talker". Listeners are usually printers, x-y plotters etc., whereas talkers are counters or meters. It is possible to have devices that are both listeners and talkers and the PET floppy disk is an ideal example. It is very important that the peripheral used is identified as a listener, talker or both and the GPIB is very specific in identifying whether the message is for an interface unit or an actual device. The ATN line allows the PET to set up all the relevant devices into their listen or talk modes by addressing specific devices one after the other, and using the status control lines before releasing the devices to begin the operating cycle using the full 8-bit data bus.

The 16 signal lines in the passive interconnecting GPIB cable are grouped into three sets, according to their function — Figure B3. Eight DATA lines carry coded messages in bit-parallel, byte-serial form to and from devices with each byte being transferred from one TALKER to one or more listeners. Data flow is bi-directional in that the same lines are used both to input program data and to output measurement data from an individual device. Data is exchanged asynchronously, enabling compatibility among a wide variety of devices. All interface messages (to set up, maintain, and terminate an orderly flow of device-dependent messages) are 7-bit coded. Device-dependent messages may be from 1 to 8 bits; however, the codes containing printable characters of the ASCII (American Standard Code for Information Interchange) code set are most commonly used and messages containing numbers are typically presented in scientific notation (FORTRAN-type) format.

Three DATA BYTE TRANSFER CONTROL (handshake) lines are used to effect the transfer of each byte of coded data on the eight DATA lines.

The five remaining GENERAL INTERFACE MANAGEMENT lines ensure an orderly flow of information within the GPIB system. One of these is called the "attention" line.

Several listeners can be active simultaneously, but only one talker can be active at a time. Whenever a talk address is put on the data lines (while attention is low), all other talkers are automatically unaddressed.

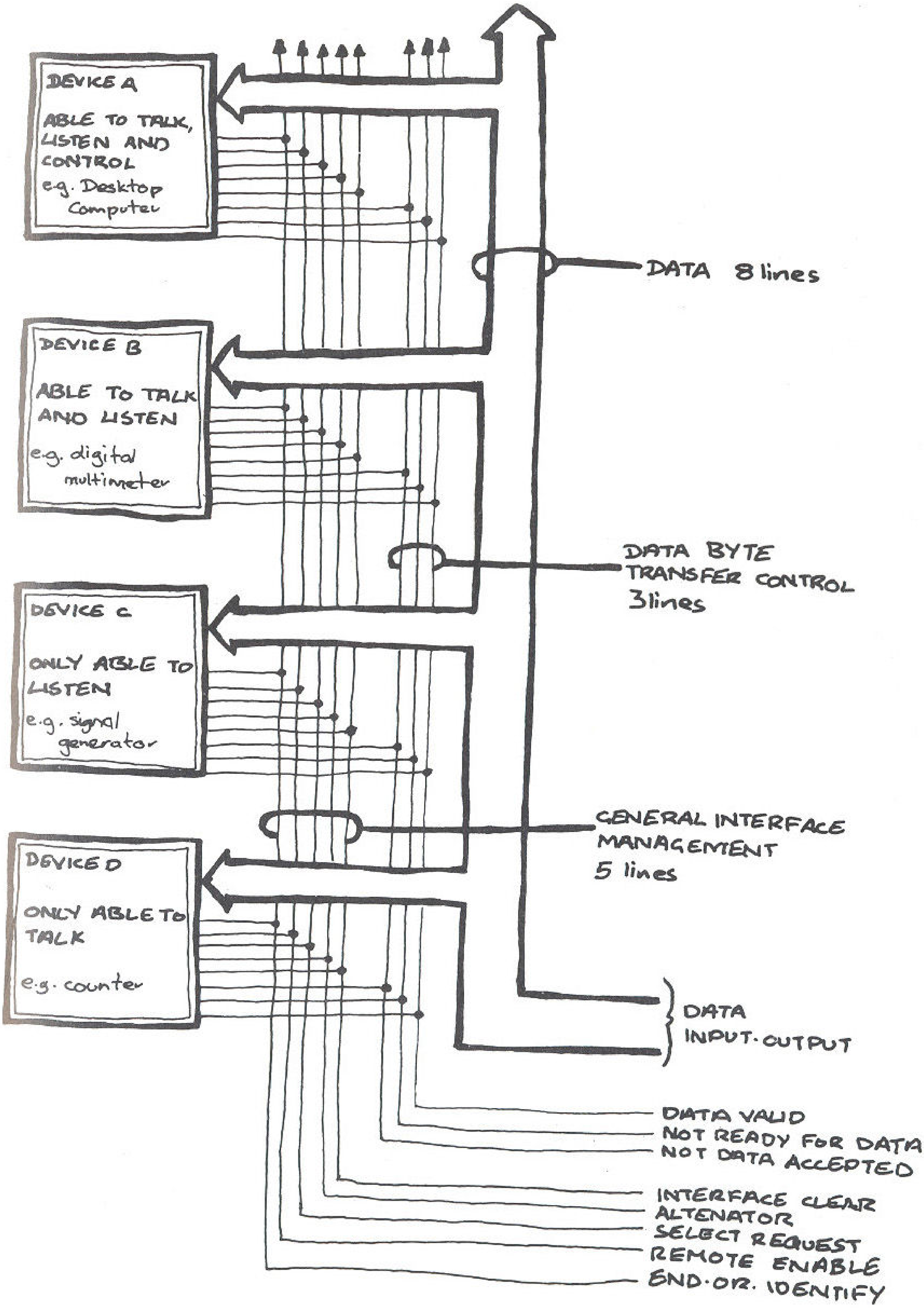


FIG B3. INTERFACE CONNECTIONS AND BUS STRUCTURES

APPENDIX C

Manufacturers/Distributors

Acorn
ACORN COMPUTERS LTD
4a Market Hill, Cambridge CB2 3N1
COMPUTER SYSTEMS: Atom, Proton (BBC Microcomputer)

Apple
APPLE COMPUTER INC
10260 Blandley Da Cupertino, Cal 95014, USA
Dist: MICROSENSE COMPUTERS LTD
Finway Road, Hemel Hempstead, Herts HP2 7PS
COMPUTER SYSTEM: Apple II

Atari
ATARI CONSUMER DIVISION
1265 Borregas Avenue,
PO Box 427, Sunnyvale, Cal 94086, USA
Dist: INGERSOLL LTD
202 New North Road, London N1
COMPUTER SYSTEMS: Atari 400, 800

British Micros
BRITISH MICROS LTD
Unit Q2, Imperial Way, Watford, Herts
COMPUTER SYSTEMS: Gemini 801

Casio
CASIO ELECTRONICS LTD
28 Scrutton Street, London EC2A 4TY
COMPUTER SYSTEMS: fx9000

Comart
COMART
PO Box 2, St Neots, Cambs PE19 4NY
COMPUTER SYSTEMS: Communicator, Educator

Commodore
COMMODORE BUSINESS MACHINES
818 Leigh Road, Slough
COMPUTER SYSTEMS: CBM/PET, VIC20

Compukit
COMP COMPUTER COMPONENTS
14 Station Road, New Barnet, Herts EN5 1QW
COMPUTER SYSTEMS: UK101

Cromemco
CROMEMCO INC
280 Bernardo Avenue, Mountain View, Ca 94043, USA
Dist: COMART
PO Box 2, St Neots, Cambs PE19 4NY
COMPUTER SYSTEMS: System Zero/Two

Dai
DATA APPLICATIONS INTERNATIONAL
16B Dyer Street, Cirencester, Glos GL7 2PF
COMPUTER SYSTEMS: DAI PC-I

Eaca
EACA INTERNATIONAL LTD
8-11 Fls. 13 Chong Yip St, Kwun Tong, Kowloon, Hong Kong
Dist: LOWE ELECTRONICS
Chesterfield Road, Matlock, Derby DE4 3HE
COMPUTER SYSTEMS: Video Genie

Exidy
EXIDY CORP
1234 Elko Drive, Sunnyvale, Ca 94086, USA
Dist: LIVEPORT DATA PRODUCTS
The Ivory Works, St Ives, Cornwall
COMPUTER SYSTEMS: Exidy Sorcerer

Grundy
Somerset Road, Teddington,
Middlesex TW11 8TD
COMPUTER SYSTEMS: Newbrain

Hewlett-Packard
HEWLETT-PACKARD LTD
King Street Lane, Winnersh, Wokingham, Berks
COMPUTER SYSTEMS: HP83/85

IBM
IBM GENERAL BUSINESS DIVISION
PO Box, 41, North Harbour,
Portsmouth PO6 3AU
COMPUTER SYSTEM: IBM Personal Computer

IMS
INDUSTRIAL MICROSYSTEMS INC
628 N Eckhoff, Orange, Ca 92668, USA
Dist: EQUINOX COMPUTER SYSTEMS LTD
Kleeman House, 16 Anning Street, New Inn Yard, London EC2A 3HB
COMPUTER SYSTEM: Equinox 5000

Interactive Data
INTERACTIVE DATA SYSTEMS LTD
14 Heathfield, Stacey Bushes, Milton Keynes, Bucks MK12 6HP
COMPUTER SYSTEM: Oscar

Intertec
INTERTEC DATA SYSTEMS
2300 Broad River Road, Columbia, S Carolina 29210, USA
Dist: Numerous in UK
COMPUTER SYSTEM: Superbrain

Ithaca
ITHACA INTERSYSTEMS INC
1650 Hanshaw, Ithaca, NY 14850, USA
Dist: ITHACA INTERSYSTEMS (UK) LTD
Coleridge Lane, Coleridge Road, London N8 8ED
COMPUTER SYSTEM: DPSI

ITT
ITT CONSUMER PRODUCTS (UK) LTD
Chesterhall Street, Basildon, Essex
COMPUTER SYSTEM: ITT 2020

Lucas
LUCAS LOGIC LTD
Welton Road, Wedgnock Industrial Estate, Warwick CV34 5PZ
COMPUTER SYSTEM: Nascom 1, 2

Luxor
Dist: DATORMARK LTD
Fox Oak, Seven Hills Road, Walton-on-Thames, Surrey KT12 4DG
COMPUTER SYSTEM: ABC 80

MSI
MIDWEST SCIENTIFIC INSTRUMENTS
220 W Cedar, Oalthe, Kansas 66061, USA
Dist: SEED
Portland House, Coppice Side, Brownhills, West Midlands
COMPUTER SYSTEM: MSI 6800/09

NEC
NIPPON ELECTRIC COMPANY
NEC Building
33–1 Shiba-Gochome, Minato-Ku,
Tokyo 108, Japan
Dist: IBR MICROCOMPUTERS
Unit 57, Suttons Ind. Park, London Rd, Earley, Berks
COMPUTER SYSTEMS: PC8000

Netronic
NETRONICS R & D LTD
333 Litchfield Road, New Milford, CT06776, USA
Dist: HL AUDIO LTD — NETRONICS
255 Archway Road, London N6 5BS
COMPUTER SYSTEMS: Explorer/85; Elf 2

North Star
NORTH STAR COMPUTERS INC
1440 Fourth Street, Berkeley, Ca 94710, USA
Dist: Numerous in UK — COMART
COMPUTER SYSTEM: Horizon

Osborne
OSBORNE COMPUTER CORP
2650 Corporate Avenue, Hayward CA 94545, USA
Dist: COMART
PO Box 2, St Neots, Cambs PE19 4NY
COMPUTER SYSTEM: Osborne 1

OSI
OHIO SCIENTIFIC INC
1333 S Chillicotte Road, Aurora, Ohio 44202, USA
Dist: OHIO SCIENTIFIC (UK) LTD
Middlegreen Estate, Middlegreen Road, Langley, Berks
COMPUTER SYSTEM: Superboard II, Challenger 1P/4MF

Philips
PHILIPS BUSINESS SYSTEMS
Arundel Great Court
8 Arundel Street, London WC2R 3DT
COMPUTER SYSTEM: P2000

Rair
RAIR COMPUTERS
30-32 Neal Street, London WC2H 9PS
COMPUTER: Black Box 3

Regnecentralen
A/S REGNECENTRALEN
Lautrupbjerg 1, DIC 2750, Ballerup, Denmark
Dist: REGNECENTRALEN UK LTD
CAP House, 9-12 Long Lane, London EC1
COMPUTER SYSTEMS: Piccolo

Research Machines
RESEARCH MACHINES LTD
209 Cowley Road, Oxford
COMPUTER SYSTEM: RM380Z, LINK 4802

Rockwell
ROCKWELL INTERNATIONAL
Microelectronic Devices, PO Box 3699, Anaheim, Ca 92803, USA
Dist: PELCO (ELECTRONICS) LTD
Regency Square House, 26-27 Regency Square, Brighton, Sussex BN1 2FH
DATATEXT
Nasmyth Road, Southfield Ind. Estate, Glenrothes, Fife KY6 2SD
COMPUTER SYSTEM: AIM65

Sinclair
SINCLAIR RESEARCH LTD
6 Kings Parade, Cambridge CB2 1SN
COMPUTER SYSTEM: ZX81

Sharp
SHARP CORPORATION
Osaka, Japan
Dist: SHARP ELECTRONICS (UK) LTD
107 Hulme Hall Lane, Manchester M10 8HL
COMPUTER SYSTEM: Sharp MZ80K/B, P3200

Shelton
SHELTON INSTRUMENTS LTD
22-26 Copenhagen Street, London N1 0JD
COMPUTER SYSTEM: Sig-Net 200/2

Sord
SORD COMPUTER SYSTEMS INC
Isoma No 2 Bld 42-12 Nishi Shunizoiwa, 4-chrome Katsushika-ku, Tokyo, Japan 124
Dist: MIDAS COMPUTER SERVICES LTD
2 High Street, Steyning, Sussex BN4 3GG
OPOLLO COMPUTERS
17 Market Place, Penzance, Cornwall TR18 21G
COMPUTER SYSTEM: M100

SWTPc
SOUTH WEST TECHNICAL PRODUCTS CORP
219W Rhapsody, San Antonio, Texas, USA
Dist: SWTP
38 Dover Street, London W1
COMPUTER SYSTEM: S/09

Tandy
TANDY CORP
Fort Worth, Texas 76102, USA
Dist: TANDY CORP (UK)
Bilston Rd, Wednesbury, West Midlands WS10 71N
COMPUTER SYSTEM: TRS 80/I, 80/II, 80/III

Tangerine
TANGERINE COMPUTER SYSTEMS LTD
Forehill, Ely, Cambs
COMPUTER SYSTEMS: Microtan 65

TI
TEXAS INSTRUMENTS LTD
Menton Lane, Bedford
COMPUTER SYSTEMS: TI 99/4A

Transam
TRANSAM COMPONENTS LTD
59/61 Theobalds Road, London WC1
COMPUTER SYSTEMS: Tuscan

Triumph-Adler
TRIUMPH-ADLER (UK) LTD
Alphatronic Division, 27 Goswell Road, London EC1M 7AJ
COMPUTER SYSTEMS: Alphatronic P1/P2

Vector-Graphic
VECTOR-GRAPHIC INC
31364 Via Colinas, Westlake Hill, Ca 91361, USA
Dist: ALMARC DATA SYSTEMS LTD
906 Woodborough Road, Nottingham NG3 5QS
COMPUTER SYSTEMS: Vector Graphic VIP/MZ

Zenith
H/S DATA SYSTEMS
Schlumberger Products Group
Hilltop Road, St Joseph, MI 49085, USA
Dist: ZENITH DATA SYSTEMS
Gloucester GL2 6EE
COMPUTER SYSTEMS: Z89

APPENDIX D

Computer Clubs in the UK
Regional Groups

Avon

BRISTOL COMPUTING CLUB
Leo Wallis,
6 Kilbernie Road,
Bristol,
Avon BS14 OHY

0272-832453

£4 sub, 3rd Wednesday monthly meetings, Independent club.

BRUNEL COMPUTER CLUB
S.W. Rabone,
18 Castle Road,
Worle,
Weston-Super-Mare,
Avon BS22 9JW

0934-513068

Alternate Wednesdays meetings, skilled and non skilled sub groups, use of Tech equipment.

Berkshire

THE THAMES VALLEY AMATEUR COMPUTER CLUB
Brian Quarm (0276 22186)
OR Brian Steer (0753 20034)

Meetings are on the first Thursday of every month and from November on, that will be at "The Southcote", Southcote Lane, off the Bath Road, Reading, Berks Starting time, 7.00pm.

Cambridgeshire

PETERBOROUGH COMPUTER CLUB
Trevor Marchant.

0733 76681.

Meets on first and third Mondays each month at Adult Education Centre, Brook Street, Peterborough.

Cheshire

CHESHIRE COMPUTING CLUB
W. Collins,
37 Garden Lane,
Chester,
Cheshire

NORTH-WEST COMPUTER CLUB
John Lightfoot,
135 Ashton Drive,
Frodsham,
Warrington,
Cheshire WA6 7PU

0928 31519.

Cleveland

CLEVELAND MICROCOMPUTER CLUB
J. H. Telford,
13 Weston Crescent,
Norton,
Cleveland

0642 550061

DARLINGTON COMPUTER CLUB
L. Boxell,
8 Vane Terrace,
Darlington

0325 67766

Business & Word Processor section meets Fridays 7.30. Scientific & Recreational Saturdays 10.00.

Cornwall

CORNISH RADIO AMATEUR CLUB
Bob Reason,
24 Mitchell Road,
Camborne,
Cornwall TR14 7JH

Computer section meets on the second Monday of every month at the SWEB Social clubroom. Pool, Redruth. New members welcome.

Derbyshire

DERBY MICROCOMPUTER SOCIETY
Mike Riordan,
172 Blagreaves Lane,
Littleover,
Derby

0332 769440

Meets fortnightly at Derby Lonsdale College, Uttoxeter Road, Derby.

Devon

EXETER AND DISTRICT COMPUTER CLUB
Doug Bates,
3 Station Road,
Pinhoe,
Exeter,
Devon

0392-69844

Monthly meetings, special interest groups.

PLYMOUTH AND DISTRICT AMATEUR COMPUTER CLUB
Keith E Gould,
Willoby House,
Meavy Lane,
Yelverton,
Devon PL20 6AL

082285 2575.

Subscription £5.00 pa. Meetings last Wednesday monthly.

Dorset

BOURNEMOUTH AREA COMPUTER CLUB
Peter Hills,
54 Runnymede Ave,
Bournemouth,
Dorset BH11 9SE

02016 6547

Meets monthly at the Kinson Community Centre.

Durham

NORTHEAST PETS
Jim Cocallis,
20 Worcester Road,
Newton Hall Estate,
Durham

0385-67045

The group meets on the 3rd Monday of each month (at 7.30pm) in: Room A102, Ellison Bldgs, Newcastle Polytechnic, Newcastle-upon-Tyne.

Essex

THE COLCHESTER MICROPROCESSOR GROUP

Meetings held at the University of Essex on the 2nd and 4th Wednesdays of each month — 7.30pm start. Membership is open to all, on payment of £5 annual sub (£1 for full-time students). Contact:
The Information Centre at the University on the evening of the meeting.

SOUTH EAST ESSEX COMPUTER SOCIETY

Holds monthly informal computer evenings plus lectures. Open to anyone over 14. Contact:
R. Knight at 0702 218456.

SPRINGFIELD COMPUTER CLUB
Steve Cousins,
1 Aldeburgh Way,
Springfield,
Chelmsford,
Essex CM1 5PB

0245 50155.

Gloucestershire

CHELTENHAM AMATEUR COMPUTER CLUB
M. P. Pullin,
45 Merestones Drive,
The Park,
Cheltenham,
Cloucestershire GL50 2SU

0242–25617

4th Wednesday monthly meeting at NGCT, Prom programming, 6800 and 6809 systems.

Hampshire

SOUTHAMPTON AMATEUR COMPUTER CLUB
Paul Dorey,
c/o Dept of Physiology,
The University,
Southampton,
Hampshire SO9 3TU

Andy Low (0703) 555605 ext 34

Meets 8pm 2nd Wed each month (not July — Sept) at Medical Science Building. Bassett Cres. East, Southampton. £3 pa, OAP, & students £2. Newsletter & special int. groups; 2 yrs old, 80 members soon setting up another club in Portsmouth area.

Hertfordshire

HARPENDEN MICROCOMPUTER GROUP
David M. James,
5 Ox Lane,
Harpenden,
Hertfordshire

05827 5366, evenings

HARROW COMPUTER GROUP
Bazyle Butcher,
16 St. Peter's Close,
Bushey Heath,
Watford,
Hertfordshire W82 3LG

01-950 7068

Alternate Wednesday meetings at Harrow College of FE or Travellers Rest, Kenton, Magazine library.

Kent

GILLINGHAM USER GROUP
A. Aylward,
194 Balmoral Road,
Gillingham

0634 56830

INDEPENDENT PET USERS' GROUP SOUTH-EAST REGION
164 Chesterfield Drive,
Sevenoaks,
Kent TN13 2EH

Sevenoaks (0732) 53530

MEDWAY AMATEUR COMPUTER AND ROBOTICS ORGANISATION
Mrs C. Webster
13 Ladywood Road,
Cuxton,
Rochester,
Kent

0634 78517

NORTH KENT AMATEUR COMPUTER CLUB
B. J. Biddles,
3 Acer Road,
Biggin Hill,
Kent TN16 3SP

09594 71742

Lancashire

CHORLEY COMPUTER CLUB
Chris Hicks,
131 Market Street,
Chorley,
Lancashire

02572 78376 or 71875

NORTH LANCASHIRE USER GROUP
Denise Green,
550 Midgeland Road,
Blackpool

0253 692261

NORTH-WEST GROUP AMATEUR COMPUTER CLUB
Ken Horton,
50 Lymfield Drive,
Worsley

061 228 6333, Ext. 372

YAMCO
Nigel Sutcliffe,
1 Suncliffe Road,
Higher Reedley,
Nr. Burnley,
Lancashire

0282-67677

Leicestershire

LEICESTERSHIRE PERSONAL COMPUTER CLUB
Jill Olorenshaw,
c/o Arden Data Processing,
Municipal Buildings,
Charles Street,
Leicester

0533 22255

Meetings held the 2nd Monday in each month, at Leicester Univeristy and Loughborough University alternately. They start 7pm. Membership is £2 per annum £1 for under 16s.

Lincolnshire

GRIMSBY COMPUTER CLUB
Meets fortnightly on Mondays at 7.30pm. Contact:
Jenson Lee,
29 Park View,
Cleethorpes

0472 32559

LINCOLNSHIRE MICROPROCESSOR SOCIETY
Eric Booth,
Bishop Grossetest College,
Newport,
Lincoln

0522 27347

London

EAST LONDON AMATEUR COMPUTER CLUB
Fred Linge,
01-554 3288

Meets 7-10pm. on 2nd & 4th Tuesdays monthly at Harrow Green Library, Leytonstone, London E11.

MEDICAL MICRO USERS' GROUP
Patrick Dixon,
c/o Medicom,
14 Broadway,
London W13

01-579 5845

NORTH LONDON HOBBY COMPUTER CLUB
Robin Bradbeer,
Dept of Electronics and Communications Engineering,
Polytechnic of North London,
Holloway Road,
London N7 8DB

01-607 2789 ext. 2052

200 members, £20 sub, £5 students subs, Last Wednesday monthly meetings, Hardware Mondays — Software Thursdays weekly, Other specialists groups, Various systems, Access to college facilities, Newsletter "GIGO", ZX-81, PET, Commercial User Groups, Education Workshop — Wednesdays.

SOUTH-EAST LONDON MICROCOMPUTER CLUB
Peter Phillips,
61 Craigerne Road,
London SE3

01-853 5829

SOUTHGATE COMPUTER CLUB
Panos Koumi,
33 Chandos Avenue,
Southgate,
London N14 7ES

01-882 2983
or Alan Toothill on 01-360 7014 (home)
or 01-882 6111 Ext. 2281 (work)

Manchester

MANCHESTER COMPUTER CLUB
David Wade,
28 Hazel Road,
Altrincham,
Cheshire WA14 15L

061 941 2486

Meetings 1st and 3rd Thursday monthly in the Computer Science Building, Manchester Univeristy.

Merseyside

MERSEYSIDE MICROCOMPUTER GROUP
J. S. Stout,
Dept. of Architecture,
Liverpool Polytechnic,
53 Victoria Street,
Liverpool L1 6EY

051-236 0598

Several sub-groups including: 380Z Users Group (Alan Pope on 051-924 2470): Computer Education Society (Mr M Trotter on 051-652 1596): SC/MP Special Interest Group (Bob Perrigo on 051-677 6716): PET Special Interest Group: 6800 and 77/ 8

Middlesex

RICHMOND COMPUTER CLUB
Robert Forster,
18A The Barons,
St. Margarets,
Twickenham,
Middlesex

01-892 1873

2nd Monday monthly meeting at Richmond Community Centre, Equipment supplied by members.

WEST LONDON PERSONAL COMPUTER CLUB
G. J. Brain,
81 Rydal Crescent,
Perivale,
Middlesex

01-997 8986

Norfolk

ANGLIA COMPUTER USER GROUP
Jan Rejzl,
128 Templemere,
Sprowston Road,
Norwich NR3 4EQ

Northampton

PERSONAL COMPUTER USERS' CLUB
J. R. Jackson,
Mereway Upper School,
Mereway,
Northampton NN4 9BU

0604 63616

Nottinghamshire

NOTTINGHAM MICROCOMPUTER CLUB
P. C. McQuoney,
28 Seaford Avenue,
Wollaton,
Nottingham,
Nottinghamshire

Keith Swainson (Membership),
9 Brayton Crescent,
Bullwell,
Nottingham,
Nottinghamshire (enclose an SAE)

0602 286709

1st Monday monthly meeting except Jan, Aug and Sept, Course at local poly, Special interest groups, Newsletter, Visitors 50p per meeting, Meetings at Trent Poly.

Oxfordshire

OXFORDSHIRE MICROCOMPUTER CLUB
Stephen Bird,
139 The Moors,
Kidlington,
Oxfordshire OX5 2AF

08675-6703 Evenings except Wednesday

2nd, 3rd and 4th Wednesday monthly meetings, Courses available, Newsletter.

OXFORD UNIVERSITY MICROCOMPUTER SOCIETY
Philip Taylor,
St. John's College,
Oxford

0865 47671

Staffordshire

THE AMATEUR COMPUTER CLUB OF NORTH STAFFS
I. Roll,
16 Hill Street,
Hednesford,
Staffordshire WS12 5DJ

05438-4363

Surrey

CROYDON MICRO/SMALL COMPUTER GROUP
Vernon Gifford,
111 Selhurst Road,
London SE25 6LH

SURREY MICROPROCESSOR SOCIETY

(SUMPS) Covering Surrey plus bits of South London and other adjacent counties. Anyone interested in joining, call Mike on 01-642 8362.

SUNBURY COMPUTER CLUB

S. N. Taylor,
8 Priory close,
Sunbury-on-Thames TW16 5AB

Meets last Tuesday each month. 40p/meeting £4pa (under 18 ½ price)

THAMES VALLEY AMATEUR COMPUTER CLUB
Brian Quarm,
25 Roundway,
Camberley,
Surrey

0276 22186

Meetings are on the 1st Thursday of every month and from November on, that will be at "The Southcote", Southcote Lane, off the Bath Rd, Reading, Berks. Starting time, 7.00pm.

Sussex

CRAWLEY COMPUTER CLUB
John Fieldhouse,
18 Seaford Road,
Crawley,
West Sussex

0293 543509

MACKENZIE MICRO CLUB
Howard Pilgrim,
42 Compton Road,
Brighton,
Sussex BN1 5AN

0273-561982

Weekly meetings, PET system, Library, Newsletter.

MID-SUSSEX MICROCOMPUTING CLUB
Bernard Langton,
228 St. Leonard's Road,
Horsham,
West Sussex RH13 6AU

0403 61156

Tyne & Wear

NEWCASTLE PERSONAL COMPUTER SOCIETY
John Bone,
2 Claremont Place,
Gateshead,
Tyne & Wear

0632 770036 (home) or
0632 781412 Ext. 236 (work)

Meets first Tues. each month in Room D103, Newcastle Polytechnic. Over 60 members sub £5.00. Several sub-groups inc. PET, TRS-80 and S100 (Last one meets weekly).

West Midlands

BIRMINGHAM COMPUTER CLUB
Fortnightly meetings planned but venue not yet fixed.
Contact:
Dr. M. Bayliss,
021-743 7197

MIDLAND AMATEUR COMPUTER CLUB
Roy Diamond,
27 Loweswater Road,
Coventry CV3 2HJ

0203 454061

WEST MIDLANDS AMATEUR CC
John Tracey,
100 Booth Close,
Brierley Hill,
West Midlands

0384 70097
Meet 2nd & 4th Tuesdays each month at Elmfield School, Love Lane, Stourbridge.

Worcestershire

WORCESTER AND DISTRICT COMPUTER CLUB
D. J. Stanton,
55 Vauxhall Street,
Rainbow Hill,
Worcester WR3 8PA

0905 22704

Meets 2nd Monday monthly at 8pm, Old Pheasant Inn, New Street, Worcester.

Yorkshire

DARLINGTON COMPUTING CLUB
L. Boxell,
8 Vane Terrace,
Darlington DL3 7AT

0325 67766

LEEDS AND DISTRICT BRANCH OF THE BRITISH COMPUTER SOCIETY
Dave J. Sheppard,
20 Green Lane Close,
Overton,
Near Wakefield WF4 4SE.

0924 270419

PENNINE AND DISTRICT COMPUTER CLUB
Douglas R. Bryant,
26 Mill Hey,
Haworth,
West Yorkshire BD22 8NA

0535 43007

Open at both 26 and 51 Mill Hey, Haworth, W. Yorks. each Sat & Sun 10am to 10pm. systems books, magazines, members shop,

SOUTH YORKSHIRE PERSONAL COMPUTING GROUP
S. P. Gray,
11 The Meadway,
Sheffield S17 3EB

0742 351440

2nd Wednesday monthly meetings, Sheffield University.

WEST YORKSHIRE MICROCOMPUTER GROUP
P. R. Clark,
Suite 204,
Crown House,
Armley Road,
Leeds LS12 2EJ

0532 450667

YORK COMPUTER CLUB
S. Wilson,
1 Grange Close,
Skelton,
York YO3 6YR

0904 470464 after 6pm.

Scotland

CENTRAL SCOTLAND COMPUTER CLUB
James G. Lyon,
78 Slamannan Road,
Falkirk FK1 4NF

0324 22430

THE GRAMPIAN AMATEUR COMPUTER SOCIETY
M. Basil,
Orton Cottage,
Burnside,
Lumphanan
Kincardineshire
Grampian Region

033 983 284

SCOTTISH AMATEUR COMPUTER SOCIETY
Alistair Macpherson
6 Curriehill,
Castle Drive,
Edinburgh 14

031 449 6658

Wales

GWENT AMATEUR COMPUTER CLUB
Ian Hazell,
50 Ringwood Hill,
Newport,
Gwent NPT 9EB

0633 277711

Covering the Gwent and Cardiff areas, the club has its own computer room and technical library. Meetings are held once a week on Wednesdays at 10 Park Place, Newport.

SWANSEA AND SOUTHWEST WALES ACC.
Paul Griffiths,
1 Prescelli Road,
Penlan,
Swansea SA5 8AF

0792 583897

Free membership, Last Friday monthly meetings, Software exchange, Newsletter soon.

Ireland

BELFAST AMATEUR COMPUTER CLUB
John Peacocke,
22 Wheatfield Gardens,
Belfast 14

0232-749379

Meetings held at end of month, Affiliated to UK ACC

COMPUTER EDUCATION SOCIETY OF IRELAND

A voluntary organisation that consists of a national body and an expanding number of local branches. Their brief is to monitor computer education in Ireland.
National CESI (£3pa) — Dairmuid McCarthy, 7 St. Kevin's Pk, Kilmaoud, Blackrock, Co. Dublin.

Cork branch (£1 extra) — Michael Moynihan, Colaiste an Spioraid Naomh, Bishopstown, Cork.

Dublin branch (£1.50 extra) — Jim Walsh, CBS Naas, Co. Kildare.
Limerick branch (£1 extra) — Sr Lourda Keane, Convent FCJ, Laurel Hill, Limerick.
Waterford branch (£1 extra) — Mr Hugh Dobbs, Newtown School, Waterford.
Kilkenny branch (£1 extra) Sr Helen Lenehan, Presentation Secondary School, Kilkenny.

Specialist groups

APPLE

UK APPLE USERS' GROUP
5b The Poultry,
Nottingham NG1 2HW

0602 583254

BAUD, BRISTOL APPLE USERS AND DABBLERS
Geoff Smythe,
Datalink Microcomputer Systems Ltd,
10 Waring House,
Redcliffe Hill,
Bristol BS1 6TB

0272 213427

APPLE USERS' GROUP
S. F. Proffitt,
The Granary,
Hill Farm Road,
Marlow Bottom,
Buckinghamshire

06284 73074
01-750 7298, day

APPLE/ITT 2020 USERS' GROUP
John A. Sharp
20 The Glebe,
Garston,
Watford,
Hertfordshire WD2 6LR

09273 75093

APPLE MUSIC SYNTHESIS GROUP
Dr. David Ellis,
22 Lennox Gardens,
London SW1
Interested in ALF, Mountain Hardware, Alpha Syntauri and Soundchaser systems. Enclose SAE.

CBM/PET

WEST LANCASHIRE PET USERS' CLUB
D. W. Jowett,
197 Victoria Road East,
Thornton,
Blackpool FY5 3ST

0253 869108

PET USERS' EDUCATIONAL GROUP
Dr. Chris Smith,
Dept. Physiology,
Queen Elizabeth College,
Campden Hill Road,
W8 7AH

01-937 5411, Ext. 429

INDEPENDENT PET USERS' GROUP
Geoff Squibb,
108 Teddington Park Road,
Teddington,
Middlesex

01-977 2346

COMMODORE PET USER CLUB
M. Gulliford,
818 Leigh Road,
Slough Industrial Estate,
Slough,
Berkshire

0753 74111

NORTH-EAST PET USERS' GROUP
Jim Cocallis,
20 Worcester Road,
Newton Hall Estate,
Durham

0385 67045

INDEPENDENT PET USERS' GROUP
G. A. Parkin,
Robert May's School,
West Street,
Odiham

025 671 2700

DERBY & DISTRICT BRANCH OF IPUG
Raymond Davies,
105 Normanton Road,
Derby DE1 2GG

(day) 0332 41025 (eve) 0332 514016
Meets monthly in Derby.

SOUTHERN USERS' OF PET ASSOCIATION (SUPA)
Howard W. Pilgrim,
42 Compton Road,
Brighton,
Sussex BN1 5AN

0273 5611982

INDEPENDENT PET USERS' GROUP (IPUG)
57 Clough Hall Road,
Kidsgrove,
Stoke-on-Trent,
Staffordshire

PET USERS IN WEST LANCS
David W. Jowett,
197 Victoria Road East,
Thornton,
Blackpool FY5 3ST

0253 869108
Meetings on the third Thursday of each month at Arnold School, Blackpool.

TRS-80

INDEPENDENT TRS-80 USER GROUP
Mike Costello,
17 Langbank Avenue,
Rise Park,
Nottingham NG5 5BU

TRS-80 USERS' CLUB
Mike Collins,
3 Altofts Gardens,
Ventnor,
Isle of Wight

TRS-80 USER GROUP
Michael Dean,
22 Roughtons,
Galleywood,
Chelmsford

0245 76127

TRS-80 USER GROUP
Dr. S. Tetlow,
3 Highbury Close,
Springwell,
Gateshead NE9 7PU

0632 462532

TRS-80 — NORTH WEST GROUP
Melvyn D. Franklin,
40 Cowlees,
Westhoughton,
Bolton,
Lancashire BL5 3EG

0942 812843
Meetings last Wednesday monthly
(not Dec)

NATIONAL TRS-80 USERS' GROUP
J. S. Wellsman,
292 Caledonian Road,
London N1

01-607 0157

NATIONAL TRS-80 USERS' GROUP
Brian Pain,
40a High Street,
Stony Stratford,
Milton Keynes,
Buckinghamshire

0908 566660

TRS-80 INDEPENDENT USER GROUP
Contact:
Mike Bayliss, 021-743 7197

TRS-80 LEVEL 1 USER GROUP
N. Rushton (LIUG),
123 Roughwood Drive,
Northwood,
Kirkby,
Merseyside L33 9UG.
Software libary and quality newsletter (write for details and free copy). £5.00 pa.

TRS-80 MEDICAL & LABORATORY USERS NEWSLETTER
Dr. N. Robinson,
The Residency,
Northwick Park Hospital,
Harrow,
Middlesex
Free quarterly newsletter detailing interests, programs & applications. Send SAE & details of interests.

ZX 80/81

NATIONAL ZX-80 USERS CLUB
Tim Hartnell,
44-46 Earls Court Road,
London W8 6EJ

ZX-80 USERS' CLUB
David Blagden,
PO Box 159,
Kingston-upon-Thames,
Surrey KT2 5YQ
£6 UK £10 O/S

ORPINGTON ZX 80/81 COMPUTER CLUB
Meeting each Friday. Contact:
R. A. Pyatt,
23 Arundel Drive,
Orpington,
66 20281

Research Machines

WEST MIDLANDS RML USER GROUP
Peter Smith BECC,
Camphill Centre,
Stratford Road,
Birmingham B11 1AR

RESEARCH MACHINES USERS' GROUP
Tony Crowle,
134 Howard Street,
Oxford

380Z USER GROUP NORTHERN HOME COUNTIES
Inc. Herts, Cambs, Oxon. Contact:
Sheridan Williams,
35 St Julian's Road,
St. Albans,
Herts AL1 2AZ

NORTH-EAST RML 380Z USERS' GROUP
M. Hatfield or R. Reed,
Computer Unit,
Northumberland Building,
The Polytechnic,
Newcastle-upon-Tyne NE1 8ST

0632 26002 ext. 268 office hours
Meets monthly at Micro-Electronics Education Centre, The Polytechnic, Newcastle-upon-Tyne.

OSI

OSI U.K. USER GROUP
Richard Elen,
12 Bennerley Road,
London SW11 6DS

OHIO SCIENTIFIC U.K. USER GROUP
Tom Graves,
19a West End,
Street,
Somerset BA16 0LQ

0458 45359

NASCOM

INTERNATIONAL NASCOM MICROCOMPUTER CLUB
The Secretary,
c/o Oakfield Corner,
Sycamore Road,
Amersham,
Buckinghamshire HP6 6SU

OTHERS

CP/M USERS' GROUP U.K.
D. Powys-Lybbe,
11 Sun Street,
London EC2M 2PS

01-247 0691

NATIONAL TI-58/59 CLUB
R. M. Murphy,
Dept. of Electrical Engineering,
University College Swansea,
Swansea,
South Wales

MINICOMPUTER USERS IN SECONDARY EDUCATION — MUSE
R. Trigger,
48 Chadcote Way,
Catshill,
Bromsgrove,
Worcestershire

USCD SYSTEM USER SOCIETY
John Ash,
Dicoll Data Systems Ltd,
Bond Close,
Kingsland Estate,
Basingstoke,
Hants RG24 0QB
Existing special interest groups include industrial application, word processing, real time, business applications and forward planning.

6502 USERS' CLUB
Joe Manifold,
16 Bun Yam Close,
Pirton,
near Hitchin,
Hertfordshire

0462 18522

TANGERINE USERS' GROUP INTERNATIONAL
Microtan 65 Users',
R. B. Green,
16 Iddesleigh Road,
Charminster,
Bournemouth,
Dorset BH3 7JR

SHARP PC-1211 USERS' CLUB
Jonathan Dakeyne,
281 Lidgett Lane,
Leeds LS17 6PD
£5

SHARP MZ-80 USERS CLUB
Paul Chappell,
Computer Centre,
Yeovil College,
Yeovil,
Somerset BA21 4AE
Free membership: Extensive library and facilities. Send for details on meetings & Newsletters (SAE please).

SHARP MZ-80K USER GROUP
Contact:
Joe L. P. Seet,
16 Elmhurst Drive,
Hornchurch,
Essex RM11 1PE

04024 42905

COMPUCOLOR II USER GROUP (UK)
Bill Donkin,
19 Harwood Avenue,
Bromley,
Kent

01-460 2626 (eve)
Quarterly newsletter: Hardware and software advice: Program library and exchange: links with other CCII national groups.

FORTH INTEREST GROUP U.K.
H. Dobson,
c/o 38 Worsley Road,
Frimley,
Camberley,
Surrey GU16 5AU

02516 6254

MANCHESTER ATOM USERS' GROUP
Clement M. Rutter,
3 Leopold Avenue,
Withington,
Manchester M20 8JG

061 434 3092

CENTRAL PROGRAM EXCHANGE
Mrs J. Brown,
Dept. of Computing and Mathematical Sciences,
The Polytechnic,
Wulfruna Street,
Wolverhampton WV1 1LY
£25 Europe £40 O/S

0902 27371 Ext. 56

ACORN ATOM USER GROUP
T. G. Meredith,
Sheerwater,
Yealm View Road,
Newton Ferrers,
South Devon
£4

BIRMINGHAM 7/66 USER GROUP
Sue Dunn,
Newbear Computing Store,
First Floor Offices,
Tivoli Centre,
Coventry Road,
Birmingham B26

021 707 7170

LEVEL 1 USER GROUP
N. Rushton,
123 Roughwood Drive,
Northwood,
Kirkby,
Merseyside L33 9UG

UK PILOT USER GROUP
Alec Wood,
Wirral Grammar School,
Cross Lane,
Bebington,
Wirral L63 3AQ

POWERTRAN USERS CLUB
Phillip Probetts,
50 Cromwell Road,
Wimbledon,
London SW19 8LZ

01-540 3713
Annual subscription £6.50. UK membership, £8.00 for members abroad, which includes a monthly newletter.

INTERNATIONAL SHARP USER GROUP
Graham Knight,
108 Rosemount Place,
Aberdeen

0224 630526
1,400 members in 31 countries £3 sub includes MZ-80K Space Invaders Cassette and newsletters.

UK 101 USER GROUP
Adrian Waters,
117 Haynes Road,
Hornchurch,
Essex

01–494 0490

COSMAC USERS CLUB
James Cunningham,
7 Harrowden Court,
Harrowden Road,
Luton,
Bedfordshire LU2 0SR

0582-423934

Software library, Newsletter.

PDP 8 USERS GROUP
Nigel Dunn,
21 Campion Road,
Widmer End,
High Wycombe,
Buckinghamshire

0494-714483

Free membership, Full information on all "8" hardware, Software, Spares, Newsletter.

PDP 11 USERS GROUP
Pete Harris,
119 Carpenter Way,
Potters Bar,
Hertfordshire EN6 5QB

0707-52091

Free membership, Information service.

HEATHKIT USER GROUP
John Smithson,
Heath (Gloucester) Ltd,
Bristol Road,
Gloucester GL2 6EE

0452-29451

Large worldwide membership, Software, Hardware, technical advice, Quarterly magazine "REMARK".

UK INTEL MDS USERS GROUP
Lewis Hard,
29 Chaucer Road,
Bedford,
Bedfordshire

0234-41685

Free membership, Annual meeting, Newsletter.

ITHACA S100 USERS GROUP
Dave Weaver,
16 Etive Place,
Condorrat,
Cumbernauld,
By Glasgow G67 4JF

02367-36570

SAE for membership, Discount available, Promotes exchange of information and software.

MK14 SCMP USERS GROUP
Geoff Phillips,
8 Poolsford Road,
London NW9 6HP

01-200 6209
or 01-207 2000 Ext. 233

Newsletter.

INDEPENDENT NASCOM USERS CLUB
Jason Twell,
15 Damside Street,
Lancaster,
Lancashire

0524-33596

Software and hardware support, Bi-monthly newsletter.

EXIDY SORCERER USERS GROUP
Andy Marshall,
Micro 44,
44 Arthurs Bridge Road,
Woking,
Surrey GU21 4NT

04862-66084

£5 sub, Program exchange, Newsletter.

SORCERER PROGRAM EXCHANGE CLUB
Colin Morle,
32 Watchyard Lane,
Formby,
Nr. Liverpool

070-48 72137

£5 sub, International membership £12. Newsletter.

TRITON USER GROUP
Nigel Stride,
Transam Ltd.,
12 Chapel Street,
London NW1

01-402 8137

1200 members, £4 sub, Software exchange, Newsletter.

CHELMSFORD *TRS 80* USER GROUP
Michael Dean,
22 Roughtons,
Galleywood,
Chelmsford,
Essex

0245-76127

Free membership, Last week monthly meetings. Newsletter soon.

COMP 80 NEWSLETTER
D. L. Probetts,
50 Cromwell Road,
Wimbledon SW19 8LZ

01-540 3713

Newsletter and information service monthly £6.50 UK £8.00 O/S.

77/68 USERS GROUP
c/o Newbear Computing Store,
40 Bartholomew Street,
Newbury,
Berkshire

0635-30505

500 members, Free years sub on purchase then £1.50, Quarterly newsletter.

9900 USERS GROUP
Chris Cadogan,
21 Thistle Downs,
Northway Farm,
Tewkesbury,
Gloucestershire GL20 8RE

Free membership, Software and data libraries.

01-494 0490

AMATEUR COMPUTER CLUB:

The Amateur Computer Club is the national body co-ordinating club activities in the UK. Individual membership is possible at £4.50 per annum. Membership secretary:
Rupert Steele,
St John's College,
Oxford OX1 3JP

For regional activities, contact:

AMATEUR COMPUTER CLUB 2650 LIBRARY
Roger A. Munt,
51 Beechwood Drive,
Feniscowles,
Blackburn,
Lancashire

02542 2341

EXETER AND DISTRICT AMATEUR COMPUTER CLUB
Doug Bates,
2 Station Road,
Pinhoe,
Exeter

0392 69844

THE AMATEUR COMPUTER CLUB OF NORTH STAFFORDSHIRE
I Roll,
16 Hill Street,
Hednesford,
Staffs WS12 5DJ

05438 4363
or 0785 3251 Ext. 441 work

APPENDIX E

Magazines in English ... UK/USA

'68 MICRO JOURNAL

'68 Micro Journal, 6131 Airways Blvd, Chattanooga TN 37421 $14.50 (12 issues)

Aimed at 6800 micro users: similar to Micro:6502 Journal.

73 MAGAZINE

73 Inc, Pine Street, Peterborough NH 03458 $62 (12 issues)

An amateur radio magazine with a regular section on microcomputers.

BYTE

Byte Publications Inc, 70 Main Street, Peterborough NH 03458 $43 (12 issues)

The world's first magazine devoted exclusively to personal computing with a staggering, estimated world-wide circulation of 180,000 copies. It is orientated to hardware rather than software and it is aimed at the experienced hobbyist or professional engineer rather than the novice. Since being taken over by McGraw-Hill it is beautifully produced, professionally laid out, and averages 500 pages per issue.

COMPUTE — The journal of progressive computing

Small System Services, Inc, PO Box 5406, Greensboro, North Carolina 27403, $42.50 (12 issues)

Subtitled the "6502 resource magazine for PET, APPLE, ATARI, KIM, SYM, AIM and OSI owners", this magazine is a combination of three popular newsletters that have been available for a number of years: 6502 User Notes, The PET Gazette and the PET User Notes. *Compute* is essential reading for anybody with a 6502 based machine.

COMPUTER AND VIDEO GAMES

ECC Publications, 30-31 Islington Green, London N1 £10.00 (12 issues)

A new magazine aimed to cater for the growing computer video market. Too new for informed comment.

COMPUTER MUSIC JOURNAL

Peoples Computer Company, Menlo Park CA 94024 $11.00 (4 issues)

This is devoted to "high quality musical applications of digital electronics". A wide ranging magazine covering not only the theoretical aspects of digital signal processing but also the more mundane topics like reviews of particular musical instruments. Its articles are quite successful attempts to make comprehensible material from what can be a difficult field.

COMPUTING TODAY

Modmags Limited, 145 Charing Cross Road, London WC2H 0EE £11.25 (12 issues)

Originally a supplement to the fine constructional magazine *Electronics Today International, Computing Today* was expected to fill the need for a good, solid personal computer *constructional* magazine. It is the typical blend of introductory articles for the beginner with software listing mainly in BASIC and machine code.

CREATIVE COMPUTING

Creative Computing, PO Box 7890M, Morristown NJ 07960 $29.00 (12 issues)

Its subtitle — "the magazine of computer applications and software" reflects its contents very accurately. Aimed at the personal user, it has the usual mixture of introductory articles, product reviews and news about the industry. It is a very good source of games software and has regular columns devoted to the APPLE, PET and TRS-80 etc.

DR DOBB'S JOURNAL OF COMPUTER CALISTHENICS AND ORTHODONTIA

People's Computer Company, Box E, 1263 El Camino, Menlo Park CA 94025 $42 (12 issues)

Apart from the *Byte*, probably the most famous of the computer hobbyist magazines. It specializes in systems software and its articles will invariably include the full documentation and source code. It used to carry no paid advertising which implied that its product reviews were genuinely independent evaluations. The advertising policy has changed but there has been no discernible change in the quality of its reviews. This is by any standards a good quality journal.

EDUCATIONAL COMPUTING

ECC, 30-31 Islington Green, London N1 £6.50 (10 issues)

The first UK magazine specifically devoted to the use of computers in education. Since its recent change in editor it has endeavoured to cater for the interests of the further and higher education sector. It is primarily orientated to the secondary school market, with a fair number of its contributors being practising teachers. There is obviously a market for a specialist educational computer magazine; however, it is still not clear whether this one has really found the right formula.

IEEE MICRO

IEEE Computer Society, Los Alamitos CA 90720 $23.00 (4 issues)

One of the Institute of Electrical and Electronic Engineers magazines published by their Computer Society. Aimed at computer specialists.

INTERFACE AGE

McPheters Wolfe & Jones, 16704 Marquardt Avenue, Cerritos CA 90701 $35 (12 issues)

A typical mixture of introductory articles, product reviews and news about the industry. It has a strong orientation with regular articles for the small businessman. In general its regular features — on micros and the law, micros and mathematics, etc. are of a high standard. Each year it publishes indexes to the software and hardware available in the USA.

KILOBAUD MICROCOMPUTING

Wayne Green Inc, 80 Pine Street, Peterborough NH 03458 $62 (12 issues)

One of the best value for money magazines for the hobbyist. Each issue has a large number (over 20) of good solid articles for the hobbyist or small businessman. It has regular columns on most common microcomputers. Wayne Green's rather idiosyncratic view of life pervades the magazine.

MICRO: The 6502 Journal

MicroInk Inc, PO Box 6502, Chelmsford MA 01824 $33 (12 issues)

The 6502 is reputed to be the most widely used micro in the world. It is used in the KIM, PET, APPLE, ACORN, etc. Fortunately there are now two good quality journals specifically aimed at 6502 users. This one and *Compute* complement each other in that *Micro* concentrates more on software and interfacing techniques than *Compute*. It also carries a bibliography of 6502 articles which is reasonably comprehensive.

MICROCOMPUTER PRINTOUT

PO Box 48, Newbury RG16 0BD £11.40 (12 issues)

Although exclusively devoted to the PET, it appears that it will soon widen its scope to cover other 6502 based machines. It has been a consistently good source of information about the Commodore product range and many of its foibles! It has achieved a remarkable improvement in its layout over the last year. The main problem for the magazine is that its main competitor *Compute* is now in a league of its own in terms of quality of articles. Furthermore it is a little "thin" for price.

MICROPROCESSORS AND MICROSYSTEMS

IPC Science and Technology Press Limited, PO Box 63, Westbury House, Bury Street, Guildford, Surrey GU2 5BH £55 (10 issues)

Not really a hobbyist magazine but designated for practising engineers. However it features a wealth of practical information and is worth looking at if you're an experienced hobbyist.

MICROSYSTEMS — THE CP/M AND S100 USER'S JOURNAL

Microsystems, PO Box 789-M, Morristown NJ 07960 $10 (6 issues)

A bi-monthly journal aimed specifically at those using the CP/M operating system and the S100 bus.

PERSONAL COMPUTER WORLD

Sportscene Publishers (PCW) Ltd, 14 Rathbone Place, London W1P 1DE £9.00 (12 issues)

Europe's first magazine for personal computing has re-established itself as unquestionably UK's premiere magazine in this field. It now has a nice balance between hardware and software features while its news column by Guy Kewney is consistently entertaining and informative. PCW's main strength however lies in the quality of its regular contributors: Malcolm Peltu's book reviews, Sheridan William's computer clinic, Sue Eisenbach's features, etc. It is a pity however that the general layout and organisation of the magazine is very poor.

PERSONAL COMPUTING

Hayden Publishing Inc, 50 Essex St, Rochelle Park NJ 07662 $32.00 (12 issues)

Since the takeover by Hayden Publishing, *Personal Computing* has aimed for a more business orientated reader. Each issue feature a theme, e.g. word processing, database management systems etc. Software listings, as usual, primarily in BASIC.

PRACTICAL COMPUTING

IPC Electrical Electronic Press Ltd, Quadrant House, The Quadrant, Sutton, Surrey SM2 5AS £14.00 (12 issues)

It was predicted that Practical Computing would become the premiere British personal computer magazine. Unfortunately it has deteriorated over the last year and although still a good magazine it does not appear to have attracted such good quality contributors as its main competitor (*Personal Computer World*). It has regular columns devoted to the ZX80/81, TANDY, APPLE, PET and the 6502 microprocessor.

RECREATIONAL COMPUTING

People's Computer Company, 1263 El Camino, Box E, Menlo Park CA 94025 $26 (6 issues)

Previously called *People's Computers* it was originally aimed at the school-user but is now more relevant to the average hobbyist. It is especially good on all aspects of computer games and is generally a "fun" magazine. It has a regular column on the PET.

ROBOTICS AGE

Robotics Publishing Corp, 5147 Angeles Crest Hwy, La Canada CA 91011 $15 (4 issues)

A useful magazine for those wishing to build their own robots, or take part in one of the many "Micromouse" competitions. Can be a bit technical.

SYNC

c/o 27 Andrew Close, Stoke Golding, Nuneaton CV13 6EL £10 (6 issues)

Glossy American publication aimed at ZX80 and ZX81 users. Compares favourably to *Compute* in quality.

WINDFALL

Europress Ltd, Europa House, 68 Chester Street, Hazel Grove, Stockport SK7 5NY £12.00 (12 issues)

Specialist magazine aimed at Apple II and ITT 2020 users. Very glossy production with many useful hints and good advice.

YOUR COMPUTER

IPC Electrical and Electronic Press, Quadrant House, The Quadrant, Sutton, Surrey SM2 5AS £6.00 (12 issues)

Aimed at the teenage games playing market and published by the people that bring you *Practical Computing*. Good for ZX81 freaks and has lots of games listings.

ZX81 INTERFACE

National ZX80 and ZX81 Users Club, 44-46 Earls Court Road, London W8 6EJ £8.50 (12 issues)

Specialist user group magazine with lots of games and utility routine listings.

Compiled by Mike O'Reilly, of the Library and Information Service, The Polytechnic of North London. (Mike is probably the UK's leading expert on personal computer publications. RTB)

APPENDIX F

Bibliography of selected microcomputer books

Prepared by Mine of Information Ltd, 1 Francis Avenue, St Albans, from a large private database on the basis of accuracy, relevance and value for money.

The prices shown are approximate 1981 levels and are subject to fluctuation. All the titles are available through Mine of Information Ltd at prevailing prices, or direct from the publishers.

GENERAL

The Future With Microelectronics by Iann Barron and Ray Curnow (Open University Press 1979) 244pp £4. Paperback edition of a study report completed in early 1978 and commissioned by the British Government. Worth ploughing through the turgid officialese for the insights and breadth of vision.

The Mighty Micro by the late Christopher Evans (Gollancz 1980) 222pp £2. This intelligent and exciting analysis of the computer revolution was serialized on television.

The Making of the Micro by the late Christopher Evans (Gollancz 1981) 118pp £6. Illustrated history of the computer enhanced with irreverent anecdotes.

The Computer Book by Robin Bradbeer, Peter de Bono and Peter Laurie (BBC 1981) about £9. Part of the new BBC course in computer literacy scheduled for 1982, the Year of Information Technology.

Mindstorms: Children, Computers, Powerful Ideas by Seymour Papert of M.I.T. (Harvester Press, 1980) £10. An eminent mathematician explains how to use computers in the classroom so that even maths is enjoyable: less teaching and more learning.

The Beginner's Book by Adam Osborne (Osborne/Mcgraw-Hill 1979) 300pp £7. Volume Zero in the Classic series "Introduction to Microcomputers" takes the hand of those who would rather not know. American bias.

Basic Concepts — Second Edition by Adam Osborne (Osborne/McGraw-Hill 1980) 439pp £11. Volume 1 of the series, now a standard textbook in hundreds of courses in the USA. Hardware bias. Very good.

Scelbi's Secret Guide to Computers by Russell Walter (Scelbi 1980) 96pp £5. A short course involving the use of the BASIC computer language, pitched to appeal to schoolchildren.

1001 Things to Do With Your Personal Computer by Mark Sawusch (Tab 1980) 335pp £5. Full of ideas with sketchy programs to fire you into action.

Your First Computer by Rodney Zaks (Sybex 1980) 258pp £7. Interesting introduction to the commercial use of microcomputer systems from a best-selling author. Get a second opinion before actually deciding anything.

Business System Buyer's Guide by Adam Osborne (Osborne/McGraw-Hill 1981) 165pp £8. Practical handbook from the guru himself helps you evaluate the signifiant features of competing systems.

Microcomputer Primer by Waite & Pardee (Sams 2nd edition 1980) 384pp £10. Aimed at the computer hobbyist who is interested in electronics.

The Personal Computer Book — Second Edition by Robin Bradbeer (Gower Publishing 1981) 250pp £6. A good introduction to low-price computing, includes a market survey and a recursive bibliography.

Home Computers Can Make You Rich by Joe Weisbecker (Hayden 1980) 122pp £5. Describes money-making opportunities in the field of personal computing.

The Microelectronics Revolution edited by Tom Forester (Blackwell 1980) 589pp £7. Over forty papers on the subject requiring no prior technical knowledge. Sponsored by the Department of Industry.

Choosing and Using a Business Microcomputer by Bradbeer, Miles, Allason and Webb (Gower Publishing 1982) £12.50.

LEARNING BASIC

Illustrating Basic by Donald Alcock (Cambridge University Press 1978) 134pp £3. Immensely popular hand-written script that keeps to a standard, portable subset of Basic. Spiral bound to lie flat. Highly recommended.

Microcomputers and the Three R's by Christine Doerr (Hayden 1979) 117pp £7. Course material at the secondary level with advice for teachers on how to introduce and manage microcomputers in a school. Bias to the PET.

Programming in Basic for Business by Bosworth & Nagel (SRA 1977) 224pp £8. At last a book that ignores trigonometry in favour of payroll, stock control and file handling.

Basic and the Personal Computer by Dwyer & Critchfield (Addison-Wesley 1978) 438pp £12. Attractive and enjoyable, includes an Eight-Hour Wonder course. Lots of examples.

Basic With Style by Nagin & Ledgard (Hayden 1978) 134pp £5. In the Programming Proverbs series — for those who like to get it right. Sound advice with humour.

Advanced Basic by James Coan (Hayden 1977) 184pp £8. Applications and problems with a strongly mathematical bias. If you are determined to learn Basic in the same way, try *Basic Basic* by the same author.

Data File Programming In Basic by LeRoy Finkel & Jerald Brown (Wiley 1981) 338pp £9. Teaches file handling in TRS-80, Microsoft and Northstar dialects of Basic.

Basic Programming Primer by Waite & Pardee (Sams 1978) 239pp £8. A beginner's book that is not tied to any particular make of computer.

A Bit of Basic by Dwyer & Critchfield (Addison-Wesley 1980) 184pp £6. Includes the Eight-Hour Wonder Course and sections on the TRS-80 and Apple II. This is an updated, shorter version of *Basic and the Personal Computer* by the same authors.

The Basic Handbook — Second Edition by David Lien (Compusoft 1981) 480pp £15. Newly updated encyclopaedia of dialects of Basic to help you understand and convert other people's programs. Essential reference book for the hobbyist.

Executive Computing by John Nevison (Addison-Wesley 1981) 319pp £5. Chatty casebook shows the development of useful working programs in Basic.

The Little Book of Basic Style by John Nevison (Addison-Wesley 1978) 151pp £5. Subtitled "How to Write a Program YOU Can Read". Nineteen rules for neat presentation to reduce maintenance costs later.

Programming for Poets in Basic by Conway & Archer (Winthrop 1979) 323pp £9. Aims to teach a raw beginner (the "poet" in the title) how to understand Basic programs.

Basic Programming — Third Edition by Kenneny & Kurtz (Wiley 1980) 333pp £10. A straightforward machine independent manual from the original founders of Basic.

BASIC PROGRAMS

Computer Programs That Work! by Lee, Beech & Lee (Sigma Technical Press 1980) 120pp £5. Third edition with micro implementation notes for PET, Apple and TRS-80. School science and educational games in a slim volume.

Basic Computer Games edited by David Ahl (Creative Computing Press 1978) 185pp £6. The classic from microcomputer maker DEC, adapted to Microsoft for this microcomputer edition.

More Basic Computer Games edited by David Ahl (Creative Computing Press 1979) 185pp £6. Another 84 games to add to the repertoire of your micro.

Some Common Basic Programs by Poole & Borchers (Osborne/McGraw-Hill 1979) 195pp £12. Third edition of seventy-six tested programs in finance, maths and statistics.

General Ledger — CBASIC by Poole & Borchers (Osborne/McGraw-Hill 1979) 178pp £16. Documentation and listings for a business package to run under CP/M using CBASIC-2. Now on diskette in the CP/M User Library. American accounting standards.

Accounts Payable and Accounts Receivable — CBASIC by Lon Poole (Osborne/McGraw-Hill 1979) 357pp £16. Documentation and listings for a business package to run under CP/M using CBASIC-2. Now on diskette in the CP/M User Library. American accounting standards.

Stimulating Simulations by C. W. Engel (Hayden 1979) 98pp £5. Twelve entertaining simulation programs including navigation, business and fantasy.

Practical Basic Programs edited by Lon Poole (Osborne/McGraw-Hill 1980) 176pp £12. Forty documented programs in financial management and statistics, following on from "Some Common Basic Programs".

PASCAL

A Practical Introduction to Pascal by Wilson & Addyman (Macmillan 1978) 148pp £5. An excellent and inexpensive programmer's primer in a gentle British style.

Beginner's Guide for the UCSD Pascal System by Kenneth Bowles (BYTE/McGraw-Hill 1980) 204pp £8. From the originator of UCSD Pascal.

Introduction to Pascal Including UCSD Pascal by Rodnay Zaks (Sybex 1980) 419pp £12. An attractive book with plenty of example programs.

The Pascal Handbook by Jacques Tiberghien (Sybex 1981) 482pp £12. Alphabetical encyclopaedia of Pascal and many of its dialects such as UCSD Pascal.

Pascal With Style by Ledgard, Hueras & Nagin (Hayden 1979) 210pp £6. Another in the Programming Proverbs series, rather pre-empted by the nature of Pascal. Useful so that "somebody else can figger out what they have done and why".

Pascal User Manual and Report — Second Edition by Jensen & Wirth (Springer-Verlag 1975) 167pp £8. The international reference work on the Pascal language.

Introduction to Pascal by Welsh & Elder (Prentice-Hall 1979) 300pp £9. Sober textbook with 17 case study programs.

Programming in Pascal — Revised Edition by Peter Grogono (Addison-Wesley 1980) 384pp £8. One of the leading American Pascal primers.

Microcomputer Problem Solving Using Pascal by Kenneth Bowles (Springer-Verlag 1977) 563pp £10. Teaches the UCSD de facto microcomputer standard Pascal with extensions for graphics and string handling.

The Byte Book of Pascal — Second Edition edited by Blaise Liffick (Byte/McGraw-Hill 1979) 333pp £18. Collected reprints from BYTE magazine including the Tiny Pascal compiler.

MICROCOMPUTER SYSTEM DESIGN

How to Build a Computer-Controlled Robot by Tod Loofourrow (Hayden 1978) 132pp £5. For hobbyists, create an inhuman intelligence on wheels, based on a KIM-1.

Introduction to Microprocessors by Lance Leventhal (Prentice-Hall 1979) 640pp £10. Paperback edition as student textbook. Concentrates on the 8080 and 6800.

Microprocessor Interfacing Techniques by Lesea & Zaks (Sybex 1979) 420pp £8. Third edition covers peripherals, analogue circuitry, bus standards, trouble-shooting. Case study is a 32 channel multiplexer (concentrator).

CRT Controller Handbook by Gerry Kane (Osborne/McGraw-Hill 1980) 215pp £6. Describes most of the currently available LSI chips for managing visual display units.

S-100 Bus Handbook by Dave Bursky (Hayden 1980) 257pp £11. Explains the S-100 bus and gives details of components, circuits, boards and suppliers.

Interfacing to S-100/IEEE 696 Microcomputers by Sol Libes & Mark Garetz (Osborne/McGraw-Hill 1981) 317pp £11. Up-to-date and detailed handbook for anyone interested in the technicalities of the S-100 bus.

Complete Handbook of Robotics by Edward Safford (Tab 1978) 358pp £6. A treasurehouse of information on mobility, sensors, power, servo, radio control etc.

Communicating with Microcomputers by Ian Witten (Academic Press 1980) 163pp £7. Deliberately ambiguous title for this stimulating book, ranging from computer buses to speech synthesis and recognition.

Microprocessors for Measurement and Control by Auslander & Sagues (Osborne/McGraw-Hill 1981) 310pp £12. Industrial case studies with specific solutions. Programs in Basic, Fortran, Pascal, C and assembly level coding for the 8080 and PDP-11.

Osborne 4 & 8-Bit Microprocessor Handbook by Osborne & Kane (Osborne/McGraw-Hill 1981) 1226pp £16. Facts and independent opinions on almost all 4 and 8 bit microprocessor chips, this bound edition updates and replaces the relevant sections of the original Volume 2 in the series "Introduction to Microcomputers" entitled "Some Real Microprocessors".

Osborne 16-Bit Microprocessor Handbook by Osborne et al (Osborne/McGraw-Hill 1981) 791pp £16. Companion to the Osborne 4 & 8-Bit Microprocessor Handbook.

Some Real Support Devices by Osborne et al (Osborne & Associates 1978/79) 1500pp £45. Looseleaf edition with binder and a year's update pages. Facts and independent opinions on many memory and peripheral chips including communications protocol devices and floppy disk controllers.

SOFTWARE METHODS & MANAGEMENT

Software Engineering for Micros by T. G. Lewis (Hayden 1979) 156pp £6. Subtitled "The Electrifying Streamlined Blueprint Speedcode Method". How to get assembly language programs up and running quickly and reliably with a fascinating tailpiece on the possibilities for hardware assistance.

Writing Interactive Compilers & Interpreters by Peter Brown (Wiley 1979) 265pp £13. An amusing and very practical book for hobbyists. Paperback edition planned for late 1981 should be much cheaper.

Fundamental Algorithms by Donald Knuth (Addison-Wesley 1979) 634pp £11. One of the standard DP reference books now available in paperback. Very strong on storage allocation: trees, arrays, lists, queues.

Microprocessor Programming for Computer Hobbyists by Neill Graham (Tab 1977) 382pp £6. Handy reference book for data structures, searching, sorting and arithmetic routines.

Successful Software for Small Computers by Graham Beech (Sigma Technical Press 1980) 210pp £6. Teaches structural programming in Basic with solutions to common problems. Approved by Tandy for the TRS-80.

Microcomputer Management & Programming by Carol Anne Ogdin (Prentice-Hall 1980) 348pp £13. Comprehensive handbook for all aspects of a development project.

Computer Language Reference Guide by Harry Helms (Sams 1980) 109pp £5. Slim booklet that gives just the flavour of Algol, Basic, Cobol, Fortran, Lisp, Pascal and PL/I. Worth a quick skim at the library.

Using Forth — Second Edition by Rather, Brodie & Rosenberg (Forth Inc. 1980) 165pp £15. Manual for polyForth showing how a threaded language allows you to build up a complex application from simple components.

CP/M

Using CP/M: A Self-Teaching Guide by Fernandez & Ashley (Wiley 1980) 243pp £7. Carefully structured for quick understanding of the main points.

The CP/M Handbook with MP/M by Rodnay Zaks (Sybex 1980) 321pp £12. The first best-seller on CP/M is still a good comprehensive reference book for programmers.

Osborne CP/M User's Guide by Thom Hogan (Osborne/McGraw-Hill 1981) 283pp £11. Throughly up-to-date textbook on CP/M, including DDT commands.

CP/M Primer by Murtha & Waite (Sams 1980) 92pp £8. Expensive and erratic, contains some pearls.

UNIX/C

The C Programming Language by Kernighan & Ritchie (Prentice-Hall 1978) 228pp £11. Describes the main language behind the UNIX 16-bit operating system.

UNIX System: The Bell System Technical Journal edited for Bell Laboratories (Bell Labs 1978) 416pp £3. Twenty papers with UNIX as the central theme.

16-BIT MICROS

The 8089 I/O Processor handbook and the 8289 Bus Arbiter by Adam Osborne (Osborne/McGraw-Hill 1980) 110pp £6. Describes the powerful I/O chip to go with the 8086 microprocessor.

The 8086 Primer by Stephen Morse (Hayden 1980) 205pp £7. Subtitled "An Introduction to its Architecture, System Design and Programming".

The 8086 Book by Rector & Alexy (Osborne/McGraw-Hill 1980) 610pp £13. Describes the hardware, interfacing and programming of the 8086, and its close relative the 8088 as used in the IBM Personal Computer.

68000 Microprocessor Handbook by Gerry Kane (Osborne/McGraw-Hill 1981) 114pp £6. Details of the architecture, timing etc. for the chip they said could not be made.

Programming the Z8000 by Richard Mateosian (Sybex 1980) 297pp £12. Compact description of the Z8000 family with programming details and examples.

Z8000 Assembly Language Programming by Lance Leventhal et al (Osborne/McGraw-Hill 1980) 911pp £15. Comprehensive textbook with the instruction set laid out and explained with examples, algorithms and exercises.

8080/8085

8080/8085 Software Design Book 1 by Larsen, Rony & Titus (Sams 1979) 334pp £8. Explains how to do it with almost 200 program examples.

8080/8085 Software Design Book 2 by Titus, Larsen & Titus (Sams 1979) 348pp £8. More application examples.

8080A/8085 Assembly Language Programming by Lance Leventhal (Osborne/McGraw-Hill 1978) 494pp £12. Presents the instruction set in detail, with hints, examples and complete algorithms. Pricey.

Guide to PL/M Programming for Microcomputer Applications by Daniel McCracken (Addison-Wesley 1978) 262pp £9. A high-level language based on PL/I, designed by Intel and implemented on the Intellec MDS.

Introduction to 8080/8085 Assembly Language Programming by Fernandez & Ashley (Wiley £8). This Self-Teaching Guide is good value for beginners.

The 8080/8085 Microcomputer Book by Intel (Wiley 1980) 603pp £13. Comprehensive and authoritative source of facts and ideas.

Z80

The Z80 Microcomputer Handbook by William Barden (Sams 1978) 304pp £8. One of the earliest Z80 books and still very popular. Hardware, architecture, instruction set, flags, interrupts, interfacing, subroutines.

The Z80 Microprocessor: Programming and Interfacing by Nichols (Sams 1979). Two volumes at 300pp and about £11 each. Examples are based on the Nanocomputer.

Z80 Instruction Handbook by Nat Wadsworth (Scelbi 1978) 117pp £4. Overpriced pocket sized reference book on the Z80 architecture and instruction set.

Z80/8080 Assembly Language Programming by Kathe Spracken (Hayden 1979) 168pp £7. Programmer's introduction to the Z80 with tables of Intel/Zilog mnemonics.

8080/Z80 Assembly Language: Techniques for Improved Programming by Alan Miller (Wiley 1981) 318pp £8. A single reference source for the programmer with ten useful Appendices and a chapter on linking to CP/M.

Z80 Software Gourmet Guide & Cookbook by Nat Wadsworth (Scelbi 1979) 322pp £11. Contains useful assembly language routines including a floating-point package.

Programming the Z80 by Rodnay Zaks (Sybex 1979) 624pp £12. Latest reprint is nominally the third edition. Readable and well presented.

Z80 Assembly Language Programming by Lance Leventhal (Osborne/McGraw-Hill 1979) 623pp £13. An accurate and reliable textbook with examples and algorithms.

6800/6809

6800 Assembly Language Programming by Lance Leventhal (Osborne/McGraw-Hill 1978) 472pp £12. Describes the instruction set in detail, with examples and algorithms.

Basic Microprocessors and the 6800 by Ron Bishop (Hayden/Motorola 1979) 268pp £8. Complete course from a definition of "voltage" up to a 6800 remote dump program. Not really for the faint of heart.

6809 Assembly Language Programming by Lance Leventhal (Osborne/McGraw-Hill 1981) 563pp £13. The instruction set for this advanced 8-bit micro introduced, explained and demonstrated.

6502

First Book of Kim by Butterfield et al (Hayden 1978) 176pp. A guide to the KIM-1 and program listings to type in to make it do something interesting.

6502 Assembly Language Programming by Lance Leventhal (Osborne/McGraw-Hill) 623pp £13. Probably the best book in its class: instruction set, examples, algorithms.

Programming the 6502 by Rodnay Zaks (Sybex 1980) 387pp £10. Popular third edition, easy to read.

6502 Applications Book by Rodnay Zaks (Sybex 1979) 278pp £10. A stimulating source of ideas on using the 6520, 6522, 6530, 6532 I/O chips with the 6502.

6502 Software Gourmet Guide & Cookbook by Robert Findley (Scelbi 1979) 204pp £9. Assembly language routines including a floating-point package.

6502 Software Design by Leo Scanlon (Sams 1980) 267pp £9. Assembly language explained with examples using the AIM-65 computer.

Programming & Interfacing the 6502 with Experiments by Marvin de Jong (Sams 1980) 414pp £11. Applied to the KIM, SYM and AIM computers.

COMMODORE PET/CBM

The PET and the IEEE-488 Bus (GPIB) by Fisher & Jensen (Osborne/McGraw-Hill 1980) 243pp £11. Describes how to interface the PET via the IEEE bus to the outside world.

32 Basic Programs for the PET Computer by Rugg & Feldman (Dilithium Press 1979) 267pp £12. A beginner's book with programs for the home, education, games, graphics, maths and miscellaneous. Patchy.

PET/CBM Personal Computer Guide — Second Edition by Osborne & Donahue (Osborne/McGraw-Hill 1980) 506pp £12. Clear and comprehensive manual, includes the 8000 series.

The PET Revealed — Second Edition by Nick Hampshire (Computerbits 1980) 186pp £10. Essential reading for PET programmers if you can tolerate the typeface and grammar. New edition rumoured.

Library of PET Subroutines by Nick Hampshire (Computerbits 1980) 141pp £10. About fifty valuable routines mostly in Basic for input, screen handling, disk sort algorithms etc.

PET Graphics by Nick Hampshire (Computerbits 1981) 218pp £12. Includes a machine code graphics package.

APPLE II

32 Basic Programs for the Apple Computer by Rugg & Feldman (Dilithium Press 1981) 284pp £12. Applesoft programs for the home, education, games, graphics, maths and miscellaneous.

Practical Microcomputer Programming: the 6502 by Walter Weller (Northern Technology Books 1980) 459pp £22. Assembly language development on the Apple II.

Computer Graphics Primer by Mitchell Waite (Sams 1979) 184pp. Starts with a general review of graphics concepts and computers and goes on to describe how to create graphics on the Apple II.

Apple Machine Language by Don & Kurt Inman (Reston 1981) 296pp £9. Goes from PEEK and POKE up to the use of the Apple Mini-Assembler.

Apple II User's Guide by Lon Poole et al (Osborne/McGraw-Hill 1981) 385pp £12. Popular single volume takes you from switch-on, via Basic and advanced programming to the machine language monitor.

TANDY TRS-80 MODEL I

TRS-80 Basic: A Self-Teaching Guide by Albrecht, Inman & Zamora (Wiley 1980) 351pp £7. Introduces and reinforces each new concept for a good grounding in Level II Basic.

More TRS-80 Basic: A Self-Teaching Guide by Inman, Zamora & Albrecht (Wiley 1981) 280pp £7. Continues with more advanced topics including graphics and file handling techniques.

TRS-80 Interfacing Book 1 by Jonathan Titus (Sams 1979) 187pp £8. Learn to control external devices through Level II Basic statements.

TRS-80 Interfacing Book 2 by Titus et al (Sams 1980) 254pp £8. Advanced projects including remote control, in assembly language and Basic.

32 Basic Programs for the TRS-80 Level II Computer by Rugg & Feldman (Dilithium Press 1980) 267pp £12. Ranges from home, education and games to graphics and maths programs.

Microsoft Basic Decoded and Other Mysteries by James Farvour (IJG 1981) 310pp £18. Annotated dis-assembly of the major portions of the ROM with chapters on the detailed data structures and software methods. Tremendous value if you need it.

SINCLAIR ZX81

Getting Acquainted with Your ZX81 — Second Edition by Tim Hartnell (Database Consultancy 1981) 120pp £5. Eighty programs interspersed with hints, tips and explanations.

The ZX81 Pocket Book by Trevor Toms (Phipps Associates 1981) 136pp £5. Cheerful course for new ZX81 owners, covers a lot of ground with many ideas including an Adventure Master program.

Machine Language Programming Made Simple for Your Sinclair ZX80 & ZX81 by Beam Software (Melbourne House 1981) 160pp £11. First two-thirds of the book is a primer for assembly language on the Z80 microprocessor, followed by details and examples for the Sinclair computers.

The ZX81 Companion by Bob Maunder (LINSAC 1981) 131pp £8. Graphics, real time techniques, data structures and file processing, educational programs and a dis-assembled listing of the Monitor.

ACORN ATOM

The Acorn Atom Magic Book (Timedata 1981) 80pp £5. Programs, hints and tips to get the most out of your Atom.

Getting Acquainted with Your Acorn Atom by Tim Hartnell (Database Consultancy 1981) 100pp £5. Looks like a useful addition to the hobbyist's library.

APPENDIX G

Glossary

A

access time: the time it takes to obtain information from a storage device.

accounts payable/receivable: American software/documentation uses the terms 'purchase ledger/sales ledger' respectively. As American accountancy procedures are different to ours, most US software has to be altered to European and UK practice.

acoustic coupler: converts pulses of sound from a telephone line into the digital signals computers understand, and vice versa. It is a form of modem (viz) that the telephone handset plugs into.

A/D, analogue-digital converter: used for converting analogue electrical signals to digital form. The opposite is called a digital/analogue converter (DAC).

address: a digital number stating a certain specific storage location in the computer memory.

algorithm: a prescribed set of well-defined rules or methods for the solution of a problem. The input of an algorithm into the computer takes the form of a sequence of instructions.

alphanumeric: a code containing the digits 0 to 9 and the letters of the alphabet.

ALU, Arithmetic Logic Unit: that part of the computer which performs basic mathematical operations such as addition, subtraction, multiplication and division of binary numbers.

analogue: means that a physical quantity can be represented analogously to a voltage, or other variable physical quantity. Example: a velocity can be presented in analogue fashion by the voltage across a meter.

ASCII: American Standard Code for Information Interchange. A standardized code used extremely frequently for data transmission. The code comprises 128 large and small letters, digits and some other special characters. Each of these is coded with its own unique 7-bit binary number.

assembler: a computer program that translates a program written in symbolic assembly language into machine language (binary code).

assembly language: a programming language built up with memory codes or mnemonic codes designed to facilitate programming ("mnemonic" means "assisting the memory"). Examples: ADD meaning "add", SUB meaning "subtract". Programs coded in assembly language are converted by a so-called assembler into machine code which the computer can understand. The process is called assembly.

asynchronous: asynchronous communication means that when an operation is finished it starts the following one. The opposite is *synchronous* procedure, which requires control by a clock signal.

B

backup: a procedure, or facility, that allows users to retain information in the event of a failure.

BASIC: Beginner's All-purpose Symbolic Instruction Code, a procedure-orientated programming language built up of simple words and abbreviations. This language is very commonly used in the personal computer world and is best suited to systems with so-called interactive terminals, i.e. where the operator converses with the computer.

baud: a baud is the unit of signalling speed, which is approximately equal to one bit per second (a bit of a binary signal element).

BCD: Binary Coded Decimal. The binary equivalents to the digits 0 to 9 consist of groups of four bits each.

benchmark: a program to test and compare different computers for speed, programming simplicity, etc.

binary: a number system with only two digits, 0 and 1, i.e. the base is 2. Computers work fundamentally with binary numbers.

bit: an abbreviation for binary digit. The smallest unit of data in a computer. Eight bits make one byte.

bit parallel: a method for simultaneous transmission of all the bits in a group by parallel conductors, one for each bit in the group.

bit serial: refers to the sequential transmission of the bits in a group — i.e. one by one — via one single conductor.

bootstrap: a short program routine read into the computer at the time of starting up. The bootstrap instructions tell the computer where to look for data and what to do with it. Sometimes abbreviated to "boot".

bubble memory: a new type of memory which has a large storage capacity despite its small dimensions.

bug: a fault in the program or computer.

bureau: a company that runs other people's work on its computer. Of limited benefit to most small businesses.

bus: a number of conductors forming the communication path for data, addresses and control signals between different units (processor, memory, etc.) in a computer.

byte: a group of usually eight bits which is treated as a unit and stored at a storage location.

C

character set: the collection of numbers, letters, graphics and symbols that are used by a computer.

chip: the piece of silicon that makes up a transistor or integrated circuit; also often used to apply to the whole integrated circuit (IC).

clock: an electric pulse-generator synchronizing all the signals in a computer.

CMOS: Complementary Metal Oxide Semiconductor, a family of digital integrated circuits characterized by extremely low power consumption and high complexity, but also by moderate working speed and sensitivity to static electricity.

COBOL: The Common Business Orientated Language — a high level programming language. Designed for commercial applications; CIS COBOL is one of the more common versions for microcomputers.

compatible: one item of equipment is said to be compatible with another if in certain respects it is constructed to the same standard. The term is also used for programs to express the possibility of running the program in a different model of equipment from that for which it was made.

compiler: a specialised program that translates a high level language program, in toto, into the machine language of the computer.

computer: a system which can receive data and, without human intervention, execute the usually complicated processing of the data, and also produce the results in the desired form. A computer consists of CPU, memory and input and output units.

CP/M: a disk operating system (viz) that has become a world wide de-facto standard, thus allowing software packages to be run on many different systems.

CPU: Central Processing Unit — the heart of any computer system.

CRA: Computer Retailers Association; a grouping of retail outlets whose code of conduct is a step towards establishing standards for selling and aftersales service.

CRT: Cathode Ray Tube; the glass tube in any video system, although it sometimes is used to describe a complete video display unit.

cursor: a square, dash, or other symbol, used on video display units to indicate where the next character is to be written.

D

D/A: digital-to-analogue.

daisywheel: a form of printhead that has the characters embossed around the edge of a petal-like disk. A daisy-wheel printer is used for high quality printer output.

data: representation of facts or ideas in a formalised manner which can be processed or transferred by persons or machines.

data base: a file of data organised so that users can call on an up-to-date pool of information.

database management system: (DBMS) a software system for designing, setting-up and managing a data base.

debugging: the detection and correction of faults in software and hardware.

device: often used synonymously with "Peripheral equipment".

digital: refers to a method of representing all the quantities in a problem by the binary numbers 0 and 1; a digital circuit is an electronic circuit working in principle as a switch, i.e. with ON/OFF positions.

DIP: Dual Inline-Package, the name for the most commonly used chip form for integrated circuits.

disk: a rotating plastic medium for storing data. The plastic disk may be solid or flexible, and is coated with a magnetic oxide. Flexible disks come in $5\frac{1}{4}''$ or $8''$ diameters, hard disks in $5\frac{1}{4}$, $8''$ and $12''$ diameters.

disk storage: a method for rapid mass storage of problems and data. Data is written (stored) or read (fetched) via the read/write head which usually searches over the rotating disk for the right location.

display: a presentation unit of some kind like a CRT or other unit which can show characters.

distributed processing: the use of a number of small microcomputers, interlinked so that they share the same data bases, program memory or even larger mainframe computer.

DMA: Direct Memory Access, a method of transferring data quantities directly between a peripheral unit and the computer memory, without going via the CPU. This is a way of increasing the speed and thus the efficiency of the system.

DOS: Disk Operating System, a program enabling the disk storage to store and output data.

dot matrix: the technique used for building up characters from a matrix of dots, which make up a rectangular pattern. Used by video display units and matrix printers.

dual processor: a computer system based around two CPUs. Usually one handles the main processing involving data whilst the other handles input, output and, maybe, the disk access and video display.

duplex: a construction designed to allow transmission in both directions at once.

dynamic memory: a type of semiconductor memory where the presence or absence of an electric charge in a small capacitor represents the logic status in the binary memory cell. A dynamic memory must periodically be "refreshed".

E

edit: to prepare data for a later operation.

editor: a program used for editing other programs, to revise, adapt and correct or supplement them for correct running and documentation.

electronic office: sometimes called "the office of the future", or "automated office"! Basically an office structure that uses computer-based systems to handle all the information processing in an integrated manner.

EPROM: an erasable, programmable ROM; this means that it is possible for the user to "zero" the program and then reprogram it.

execute: to carry out an operation in a program, to "run" a program in the computer.

external store: also called a mass memory. A memory with considerably greater capacity than a working memory, but often with considerably longer access time. Data is often transmitted in blocks between memories. Examples: floppy disk, magnetic tape.

F

FET: Field Effect Transistor. A certain type of transistor (unipolar instead of bipolar) characterised by small dimensions, lower power requirements and low price.

fetch; the CPU fetches an instruction from the memory and decodes it.

file: a collection of data that can be considered in some way complete and is treated together.

firmware: a program (micro-instructions) stored in a permanent ROM.

floppy disk: also called diskette. A mass memory in the form of a soft plastic disk enclosed in a protective square envelope. The storage of data is magnetic. There are two sizes of floppy disk: standard — 200mm or 8″ diameter and a smaller mini disk that is $5\frac{1}{4}$″ diameter. See also disk storage.

flowchart: a graphic representation of a system or program. used as an aid to efficient programming or system design.

format: refers to the structure of data.

FORTRAN: FORmula TRANslator — a high level language developed for scientific use and the "father" of BASIC.

G

gate: a logic element with two or more inputs and one output. The state of the output depends on the logic state of the input signals. The connection between incoming and outgoing signals is described in what is called the truth table of the circuit.

graphics: output that is not alphanumeric, usually symbols or pictorial characters.

H

handshaking: control signals making it possible for two electronic circuits to synchronize their work.

hard copy: a printout of data on paper.

hard sectored: method used on some disk drives that positions the read/write head by a series of holes punched into the disk.

hardware: the actual apparatus of a computer system in contrast to software.

hexadecimal: the hexadecimal number system has 16 as its base and comprises the digits 0-9 and letters A, B, C, D, E and F.

high-level language: a programming language which is largely independent of the type of computer being used; the most used high-level language is BASIC.

I

IEEE-488: a standard interface consisting of sixteen parallel lines as defined by the Institution of Electronic and Electrical Engineers (IEEE) in the United States.

Input/Output: also known as I/O. General designation of peripheral equipment units making it possible for the computer to communicate with the outside world.

instruction: a set of characters defining a certain specific computer operation.

instruction repertoire: the different instructions with which a certain computer can work.

integrated circuit: also known as IC. A semiconductor circuit in micro form; on a small silicon chip a few millimetres square there are a large number of active and passive semiconductor elements dovetailed and connected in such a way that the circuit is given a specific electronic function.

interface: a connection between two units of apparatus with different functions.

internal store: a memory built into the computer; it is on-line i.e. directly controlled by the computer.

interpreter: a program controlling the execution of another program without the necessity for the latter to have been compiled or assembled first.

interrupt: this means that present program execution is interrupted in favour of a routine with higher priority; when this routine has been run, the computer returns to the original program at the point where the execution was interrupted.

J

jump: a deviation from the normal sequence of orders followed by a computer to execute instructions.

K

K: read as "kilo". It means 1024, e.g. a memory with storage space for 2K bits manages 2048 bits, Cf. k, which represents 1000, e.g. 1000 g = 1 kg.

keyboard: a set of keys similar to that on a typewriter, used for the input of data into the computer.

key-to-disk: a data entry technique whereby data goes directly from the keyboard to a disk file.

keyword: a word in a file that is used to retrieve its contents.

L

line number: used by some high level languages to indicate the start of a new instruction in a program.

line printer: a printer that prints a line at a time, as compared to a serial printer that prints a character at a time.

loader: a program controlling the operation of the peripheral equipment while other programs are being read in to the computer memory.

logic: the study of fundamental principles and applications of specific portions of symbolic logic, connection theory and other related methods used for instance for the designing of equipment for computers.

logic circuit: a coupling up of logic elements giving a certain specific function.

LSI: Large Scale Integration, a highly complex integrated circuit which can execute complicated operations. On a chip only a few millimetres square there are several thousand transistors.

M

machine language: the fundamental language with which a computer works; instructions are in groups of ones and zeros.

matrix printer: a printer that uses a single, or row, of needles to print characters using dot matrix techniques.

Mega: (M) means one million; as in 12M bytes i.e. 12 million bytes.

memory: a general term for any unit intended for the storage of binary data.

memory map: a graphic picture of the total storage in the computer and the disposition of this space: the map starts with the address 0000 (hex) at the bottom and goes up to the address FFFF.

micro: prefix signifying one millionth. Example: one micro-second: one millionth of a second.

microcircuit: another name for an integrated circuit.

microcomputer: a computer family characterised by being built up with only a few integrated circuits on a circuit board, with a word length usually not exceeding 16 bits, and with a low price.

microprocessor: an extremely complex integrated circuit giving the function of a central processing unit in a computer.

Microsoft: An American company that developed the first acceptable BASIC interpreter for microprocessor based systems. Now used as a standard by which other implementations are measured.

mnemonic code: data instructions written in concise, easily remembered symbolic or abbreviated forms, e.g. SUB which may symbolise "subtract".

modem: an item of equipment for signal conversion in data transmission: it is an abbreviation for modulator/demodulator.

monitor: a program — a typical part of a large operating system — which supervises or controls the operation of the computer system e.g. the input and output of data.

MOS: Metal Oxide Semiconductor, a field effect transitor characterised by extremely high ingoing resistance.

mother board: a large circuit board to which one can plug in a number of smaller board modules.

multi-access: a system that allows several users to access at the same time.

N

nano-: prefix signifying one thousand millionth. Example: one nanosecond is one thousand millionth of a second (in common usage this is called one billionth).

network: a system of interconnected microcomputers.

nibble: a system of four binary digits — half a byte!

nonvolatile: when this term is used of computer memories it signifies that stored data is permanently retained even if the supply voltage should be turned off. Examples: nonvolatile memories include magnetic tape, disk storage and ROM.

O

object program: the binary form of a source program (the source program is what has been written by the programmer).

octal: a number system with 8 as its base, consisting of the digits 0-7.

OEM: Original Equipment Manufacturer. A company that takes individual parts of a system, plugs them together, and sells a complete unit.

on-line: refers to a control system whereby one unit is controlled by another without manual intervention, e.g. a data terminal is often on-line with regard to the computer.

operating system: software making it possible for the user to communicate with the computer in a convenient way.

output: what you get out of the computer!

P

package: a package is a program developed for a particular application, hopefully with good operating manuals.

paper tape: a tape made of paper, usually 25mm wide, with punched holes in specific code positions, used as a data carrier.

parallel I/O: a method of transmitting all the bits in a bit group at the same time, for example a whole byte at once: this is done via parallel conductors, one for each bit.

password a string of characters allowing restricted access to programs or files.

peripheral equipment: general designation of equipment used with a computer but not part of the CPU. Examples: printers, keyboards, video terminals, and floppy disk units.

plug-in: any additional module, e.g. a memory board, that can be connected to the computer bus structure by means of the bus connector.

port: a communication channel through which data can pass into or out of a computer.

printed circuit: a fibreglass board with a circuit pattern of thin copper conductors, to which electronic components can be added; when it already contains these components it is called a circuit board.

printer: an item of electromechanical equipment which is fed with signals from the computer and prints out graphic characters on paper; the opposite is a plotter.

printout: printed output from the computer.

program: a complete sequence of computer instructions needed to solve a specific problem with the aid of the computer. (Can be spelt programme.)

PROM: Programmable Read Only Memory, a permanent ROM which can be programmed by the user. Sometimes also called a "write once memory". See also ROM, EPROM.

proprietary software: a copyright program sold on a commercial basis.

R

RAM: Random Access Memory, a memory where the access time depends on the location of the data in the memory.

real time: a way of running a system where the operations in the computer are carried out so fast in relation to a physical system that the result of the operations can be used by the physical system.

record: a group of related information or items of data.

register: a temporary storage place in a processor.

reset: to return a counter to the original position, usually the zero position.

reset: a key or button that re-initiates the computer system following a system "crash".

ROM: Read Only Memory, a memory where one can only read and not write.

routine: a sequence of data needed to effect a certain specific function.

RS-232: a standard specifying the electrical properties for input and output of data from peripheral units.

S

S-100: a standard bus consisting of 100 parallel lines as defined by the IEEE in the United States.

semiconductor store: a memory consisting of active bistable semiconductor elements: the internal store of a mini- or microcomputer is a semiconductor store.

serial I/O: a method of transmitting data between the computer and a peripheral unit one bit at a time via a single conductor.

simplex: a connection designed to allow transmission in one direction at a time.

soft-sectored: a method of partitioning a disk using software, not physically on the media itself, as in hard-sectored.

software: a systematic collection of programs and associated documentation concerning the use of a data processing system. Examples are operating systems, monitors, compilers, editors, auxiliary programs and user programs.

software documentation: program listing and/or manuals telling you how to use the software.

source language: the language in which the programmer writes the program and from which translation is made.

source program: a program — either in the form of a list or stored in a memory unit — written in a language (source language) other than machine language and requiring translation by means of an assembler, compiler or interpreter program.

static memory: a semiconductor memory of the RAM type which does not require refresh pulses.

statement: in some programming languages, a designation of certain types of instructions.

string: an ordered sequence of signs or characters.

subroutine: a number of instructions which execute part of the work in a later program; a subroutine can, for example, be started by a command from the main program.

synchronous: used of a method of transmitting serial binary data between for instance the computer and the peripheral equipment; the transmission takes place at a fixed speed, and transmitter and receiver are synchronized with a clock.

syntax: rules determining how programming statements are constructed. Sometimes defined as the "grammar" of the language.

syntax error: message from the interpreter or compiler that a program has a "grammatical" error.

system: a collection of hardware and software that performs a particular, or general computing function.

system software: software that enables a programmer to develop and run applications programs.

systems analysis: the procedure for analysing the phases of all the activities of an organisation, and then developing detailed procedures for implementing a computer system.

T

tape: usually refers to magnetic tape and sometimes to punched paper tape.

Teletype: this is the everyday term for a teleprinter although it is actually a printer made by the Teletype Corp. in America. This machine is often used as an I/O device in computer systems and has a keyboard and printer, and sometimes also a paper-tape reader and punch.

terminal: an I/O unit connected to a computer.

time sharing: a computer system where the CPU time and other system resources are shared by many users with many different tasks: some time sharing systems work so that the users contact the computer by telephone.

transaction: any event which requires a record to be generated in a computer system.

turnkey: a system started by the "turning of key". This usually "boots" the system and is thus immediately available for use.

U

UART: Universal Asynchronous Receiver Transmitter, an often used LSI circuit for transmitting and receiving digital data.

utility program: a program which is often part of the operating system and is used to facilitate running and testing of written programs.

V

validation: checking input to ensure that it is correct.

VDU: video display unit.

video: that part of a TV signal which transmits the information (intensity, colour and synchronization) required for the formation of an image on a TV screen. The standard video output has a magnitude of 1V peak to peak (1Vpp).

volatile: refers to a property of a computer memory whereby stored data disappears when the supply voltage is turned off. Example: a RAM is volatile.

W

Winchester: a hard disk system developed by IBM. Usually refers to a hermatically sealed, non-removable disk unit.

word: a group of binary symbols treated by the computer as a single unit of information: the word length is determined by the design of the computer. A typical microcomputer has 8 bits to the word: In an 8 bit system the term byte has the same meaning as word.

word processing: the automatic processing of text. This is possible using a dedicated system — a word processor; or by using software on a general computer system.

APPENDIX H

Some Hints on Kit-built Systems

There has been a great deal of interest recently in kit-built computers. At one time it was predicted that they would fade away but this has not proved correct. With the introduction of machines like the Sinclair ZX81 and the UK101, kit building has shown itself to be very popular.

There are a number of reasons for building a computer from a kit:

The price is lower — around 20%

It's interesting, and instructive; and there is a certain amount of pride when the thing actually works!

It is sometimes easier to repair a system that you have built yourself — especially if you learnt how it works in doing so.

Unfortunately there are a number of pitfalls to be avoided. It is not just a matter of going to the shop and buying the bits.

Most systems are sold by mail order — make sure that the manufacturer has stocks before you send any money. Also check that the system actually exists and is not just a figment of his (or his advertising manager's) imagination!

You must also check that it is a good design — and that it actually works. As this is difficult for the inexperienced, ask other people who have bought one. Don't be the first in your group to buy a new system. Let someone else have all the headaches first!

Another area of weakness in most kits is the instruction manual. It is usually written by somebody with years of experience who may have difficulty writing for a novice.

Are the components of reasonable quality? Some kit suppliers use "seconds" — chips which do not meet their specifications and are really rejects. Again, a reputable firm will not risk their reputation — it may cost you a bit more but in the long run it will be worth it.

What happens if the computer doesn't work? Is there a good backup — in information, trouble-shooting, and even repair facilities?

Tools

Some basic tools are required for building any kit. These are:

A light electric soldering iron (say 15-25 watts) with a fine bit.

Fine solder with a resin flux core: NOT acid type flux.

A sharp pair of side cutters.

A magnifying glass — very useful when looking for short circuits and inspecting solder joints.

Something to get solder off components that are in the wrong place or the wrong way round. Desoldering braid or a solder-sucker will be needed as double sided printed circuit boards with plated through holes are used.

If you have no experience of soldering, spend some money on buying cheap components and a pcb and get some practice. Don't use your computer as a way of learning to solder!

Follow the instructions carefully. This may be obvious, but the fact is that most people are careless. Read through everything from start to finish before you begin work. There are often important corrections to be noted and these are usually given on an errata sheet. During the work, follow the instructions, one after the other, and then you won't have to make unnecessary mistakes.

Have a good place to work in. You need good lighting, a good sitting position, and plenty of space so that you can keep all components and papers in order. Don't work when you are tired.

Take your time. Work quietly and methodically, spreading out the assembly over several evenings instead of doing a "quick job" that turns out wrong. Trouble-shooting can actually be very time-consuming and frustrating.

Ask questions. If the instructions are ambiguous on any point, don't guess. Take nothing for granted. Ask someone who knows!

Watch out for solder shorts. This is one of the most common mistakes. The circuit boards used for computers are usually highly complex and there's not much room between the conductors. Drops of solder flying around can very easily cause a short circuit. Keep the tip of the iron clean!

Use IC sockets. Integrated circuits (IC) with their many connector pins are troublesome to exchange. Invest in IC sockets if these happen not to be included in the kit. This makes life much easier when it comes to jobs such as trouble-shooting or repairs.

Handle the components carefully. Turn them the right way up — components such as transistors, integrated circuits, electrolytic condensers, diodes and others often tend to finish the wrong way up. **Double check**! Some integrated circuits are constructed with what is known as MOS technology and are sensitive to static electricity. Follow the instructions carefully.

Get hold of reference literature. The manuals do not always contain enough information about the components, for instance the socket connection of integrated circuits, colour-marking of resistors and so on. A good handbook is a valuable aid.

Check your work carefully. Before current is switched on you must carefully check that everything is connected according to the description. Look for unintentional soldering bridges with a magnifying glass. This is a small price to pay when you think of how disastrous a wrong connection can be.

If, after all this effort, the system doesn't work, what can you do? Well, don't put the system into its full casing until it does. It is very frustrating taking panels off, or unscrewing bits and pieces. It is also easier to check what's wrong if you can get at all the parts. If it is mains powered, however, be **very** careful when prodding about the power supply area.

Quite a bit can be done without having to resort to expensive equipment. Check the mains fuse — are the voltage regulators getting warm? That usually indicates something may be wrong. Check that the video display, or printer, is connected properly — and working. If things get **very** hot, then switch off and check for short circuits, or components the wrong way round. The microprocessor and memory chip tend to get warm — if they don't, check back along the power rails.

If simple feeling, pushing, and looking achieves nothing, get onto your supplier. If you send it back then expect a charge for the service. Most suppliers will **not** repair half-made kits though — so don't give up halfway through!

The Personal Computer Book

Second, Updated and Revised Edition

by Robin Bradbeer

Suddenly, everybody's talking about personal computers: on TV, in the press, and on the radio we are hearing words like "personal computing", "hobby computer" and "home computing".

The personal computer has arrived. Today's personal computers are the introduction to something which, within only a few years, will bring about radical changes in our lives both at home and at work and will become as natural a part as the TV, stereo or typewriter . . .

This book has been written to explain the many possibilities for the personal computer—and also the pitfalls. It should be read before a single penny is spent on personal computer equipment, but will remain an invaluable reference even after installation of the chosen system.

This new edition has been completely updated to take in all recent significant changes and includes a survey of over sixty different models of microcomputers currently on the market. The detailed appendices have all been updated and extended.

Reviews of the First Edition:

". . . (the author) aims to provide the novice with a straightforward introduction to personal computers and he succeeds admirably . . . Robin Bradbeer's book provides all the information in one place, in a sensible order and in a consistent, clear style." **Practical Computing**

". . . sets out in a very entertaining and readable form the facts on owning a computer . . . the final chapters and incredibly detailed appendices are worth the price of the book on their own . . ." **Computing Today**

About the Author:

Robin Bradbeer is editor of **Educational Computing** magazine, a Senior Lecturer in the Department of Electronic and Communications Engineering at the Polytechnic of North London, and is regarded as a foremost expert in the microcomputing field.

Gower Publishing Company Limited,
Gower House, Croft Road,
Aldershot, Hampshire GU11 3HR

ISBN 0 566 03423 9

Gower

Cover design by Sampson/Tyrrell Designers