CET

USPEC 32

user specification

MARCH 1980

A Guide to the Selection of Microcomputers

General Introduction to USPECs

1. The Working Party on Standards and Specifications

The Council for Educational Technology's Working Party on Standards and Specifications for Educational and Training Equipment has members drawn from all areas of education and training.

The Working Party was set up:

(a) to give the user a guide to the facilities and performance he should look for when purchasing equipment

(b) to create a means of transmitting user requirements to manufacturers

(c) to provide a concerted user voice in British standardizing committees and through them, international committees.

To help achieve these aims the Working Party decided that its first task was to produce a series of User Specifications (USPECs).

2. USPECs

Generally, each USPEC sets out to define a range of acceptable values for normal use for equipment or codes of practice.

Wherever possible USPECs specify acceptable requirements to ensure safe and easy handling by untrained personnel.

Where different machines are available every effort is made, through wide consultation, to ensure that the same cassette, tape, slide, film, etc, can be used without modifying the machine.

Where appropriate, USPECs issued after November 1974 are expressed first in lay terms for the benefit of the non-technical user and then in more detail in the technical section.

3. Production of USPECs

Outline proposals for a USPEC are prepared by a sub-group of the Working Party, often with the assistance of coopted experts. Following discussion in Committee the sub-group prepares a draft for circulation to some 400 users for detailed comment. These are considered during the production of a second draft, copies of which are then normally sent to the manufacturers' associations (BEEA —British Educational Equipment Association, and ICETT —Industrial Council for Educational and Training Technology) for their comments before a final draft for publication is submitted to the Working Party.

4. Revision of USPECs

USPECs are revised at two-yearly intervals or earlier if there are technical developments or variations in British or international standards which require changes to be made.

COUNCIL FOR EDUCATIONAL TECHNOLOGY FOR THE UNITED KINGDOM
Working Party on Standards and Specifications for Educational and Training Equipment
3 Devonshire Street London W1N 2BA Telephone 01 580-7553 636-4186

CONTENTS **Page**

Section 1 :	Introduction	3
Section 2 :	Putting the Microcomputer to Use	4
Section 3 :	The Computing Facilities to Support the Different Kinds of Educational Applications	7
Section 4 :	The Equipment (Hardware)	9
Section 5 :	Software	13
Section 6 :	Documentation from the Supplier	15
Section 7 :	Support for Hardware and Software	16
Section 8 :	Glossary of Terms Used in this USPEC	17
Appendix :	Typical Cost Figures as at Date of Publication	21

1. INTRODUCTION

1.1 Many schools and colleges are contemplating the purchase of one or more *microcomputers.* For those who are making such decisions the expanding range of microcomputer applications seems relevant to, and a necessary part of, many aspects of school and college curricula; the descending price of such units makes their purchase within the reach of modest budgets. For those who have been using large computers the use of a microcomputer avoids the wastage of telephone charges or the delay of batch processing and provides facilities at a low cost which can be made available continuously throughout each day. Furthermore, the new machines show every sign of being very reliable.

1.2 Before choosing a microcomputer it is necessary to identify and carefully consider the needs for which it is required. The system that is purchased will determine the limits of what it can be used for and this is why it is important to select one which will meet existing needs (and budget) and which will be easily expandable to meet forseeable needs.

1.3 In this document the word 'microcomputer' (or computer) will be used to refer to a 'microprocessor based computer system'. Descriptions of various systems are given in Section 2.

1.4 The danger already exists, however, that microcomputers will be bought in the belief that finding the purchase price is the only difficulty. One of the most important messages of this document is that the real problems arise after the computer has been bought. Without the necessary detailed sets of instructions (the *software*) the electronic equipment (the *hardware*) will be useless. Good software has to be obtained in order to do something useful with the microcomputer and this issue must not be overlooked when selecting a system. One of the first considerations is to establish where the software can be obtained. This is discussed further in Section 3.

1.5 Above all, the purchaser must realize that to choose a microcomputer system and develop its use in isolation is a road to almost certain disaster. Support is an essential requirement for both hardware maintenance and future software development. Sharing experience and expertise is most important if a school or college is to make good use of the microcomputer it obtains. This is discussed further in Section 7.

1.6 In an area of technology which is in its infancy and developing with unprecedented speed, it is extremely difficult to provide guidance to would-be purchasers. Whichever microelectronic equipment is purchased today, one thing is certain; by tomorrow something better and cheaper will be available. There are at least three common reactions to this problem:

(i) buy nothing at the moment; wait for the better things of tomorrow

(ii) be diverse in purchasing so that the reported experience of both good and poor equipment will help later purchasers and possibly influence manufacturers to improve their products; this strategy means some wastage for those early in the field

(iii) make a carefully considered judgement of present equipment and then select a system which will allow expansion to meet future educational needs: for example, future additional features could be:

(a) colour display
(b) further peripherals; eg, printer, additional visual display units and disc drives
(c) more powerful processor
(d) larger memory
(e) alternative means of input, eg, light pen, spoken commands
(f) alternative means of output, eg, speech, interfacing with measuring instruments.

Having made the decision, standardize on this system for at least two or three years.

1.7 If reaction (i) is pursued to its logical conclusion, there will not be any progress for at least five years and teachers will not be able to make any progress in discovering the most worthwhile uses for computers.

Reaction (ii) has certain things to recommend it but in both the long and short term it will be very wasteful of that most precious resource — teachers' time. Each school could find itself working in technological isolation and unable to benefit from and share experiences. What is more, the majority of teachers could find that the required technical support is not available; in particular, the isolation could lead to serious maintenance problems. The value of collaboration in the areas of software development and educational applications cannot be overstressed.

Reaction (iii) is strongly recommended as the most appropriate path to follow. Cooperation on a local basis can be both fruitful and invigorating and a local authority policy to purchase a particular microcomputer for schools and colleges is most desirable. Local meetings can then be arranged to exchange experience and resolve problems. Individual local authority schools contemplating the purchase of a microcomputer should consult the adviser responsible, no matter what source of finance is being used for the purchase. The adviser, with access to advice from local micromputer specialists and other advisers, should be able to provide recommendations and support. In the absence of such local authority coordination, it is advisable to attempt to coordinate work with other schools and colleges in one's own region. Advice can also be sought from a number of organizations with expertise in the field (CET can give details in this respect).

1.8 It is strongly recommended that time be set aside by the school or college to enable teaching staff to become familiar with this new technology. Wherever possible teaching staff should attend locally organized in-service training courses. As a second best, allocating time for teachers to acquire for themselves the skills needed to operate the microcomputer would promote an effective introduction to its uses.

1.9 This USPEC raises key issues concerning the selection of microcomputers for educational applications. Its format and coverage acknowledges the different levels and types of experience held by those entering the field. Readers are encouraged therefore to use this USPEC flexibly, perhaps selecting in advance a pathway through the text which seems appropriate to their specific situation. To aid this strategy an extensive glossary has been referenced to the key-words presented in italics throughout the text.

2. PUTTING THE MICROCOMPUTER TO USE

In this section a number of uses of microcomputers in education are categorized, while in Section 3 the type of hardware and software needed to support each of these categories is described. Four typical microcomputer systems are identified (see opposite) and explained further in Section 4.

Microcomputers can be used for the following purposes.

2.1 Introducing children to the general concept of computers including the teaching of CSE or O-level courses on computing: such courses normally include a section on how computers work (very elementary) and sections on computer applications and their implications and effect on society.

> See Systems B, C and D and Section 3.1

2.2 Supporting an A-level course in computing: these courses are in-depth studies of computer systems and emphasize the structure of computers as well as their applications.

> See Systems C and D and Section 3.2

2.3 Supporting *computer assisted learning* activities in a wide range of subjects extending from the sciences, through geography and history, to modern languages: the applications anticipated are those which do not involve the manipulation of large quantities of data.

> See Systems B, C and D and Section 3.3

2.4 Implementing programs that involve large sets of data: these include *computer managed learning*, assessment processes and some of the larger computer assisted learning packages. Information retrieval and storage techniques are also included in this category.

> See System D and Section 3.4

2.5 Enabling *real time on-line data capture, analysis and control*, associated with experiments in science or technology. Automatic data capture techniques could, in addition, be used with other subjects such as social sciences. An additional input/output interface will be needed for this type of application.

> See Systems A, B, C and D and Section 3.5

2.6 Assisting school administration, using word processing within the school office, school records and some timetabling activities.

> See System D and Section 3.6

2.7 It is very important to realize that during the regular use of the microcomputer, time is required for:

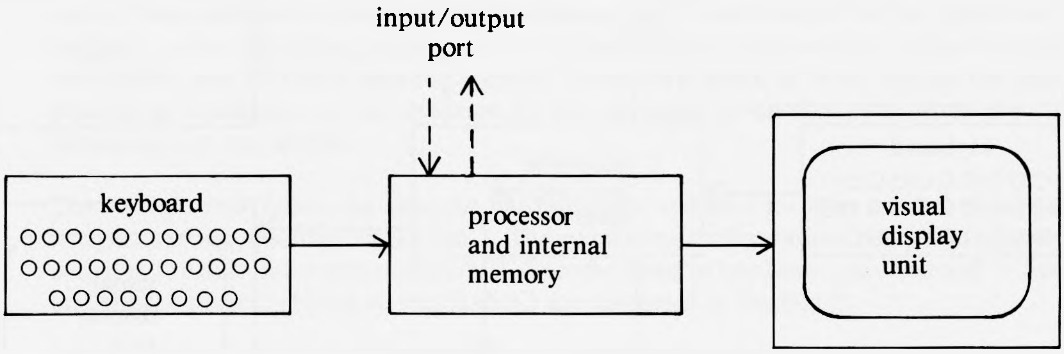
(a) internal staff training

(b) software development

(c) systems management (which includes documentation, archiving and scheduling).

The emphasis in the applications above has been on secondary school and college work. It is likely that future developments will provide us with valid applications at primary level but there is little experience of this at the moment.

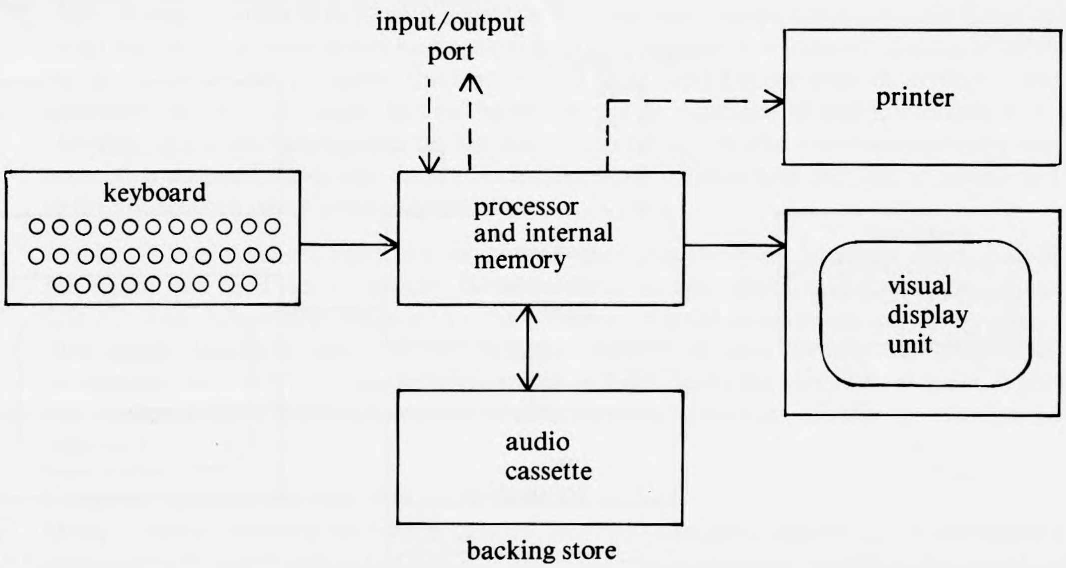
SYSTEM A



The most basic computer consists of the above components, the *keyboard, processor* and *internal memory* and *visual display unit*, and may all be contained in a single box. However, this system has the serious drawback in that *programs* are lost from the system when the power is switched off and they have to be retyped when the machine is switched on again. In practice, some form of *'backing store'* is essential.

The dotted input/output *port* is needed for some applications, eg, 2.5.

SYSTEM B

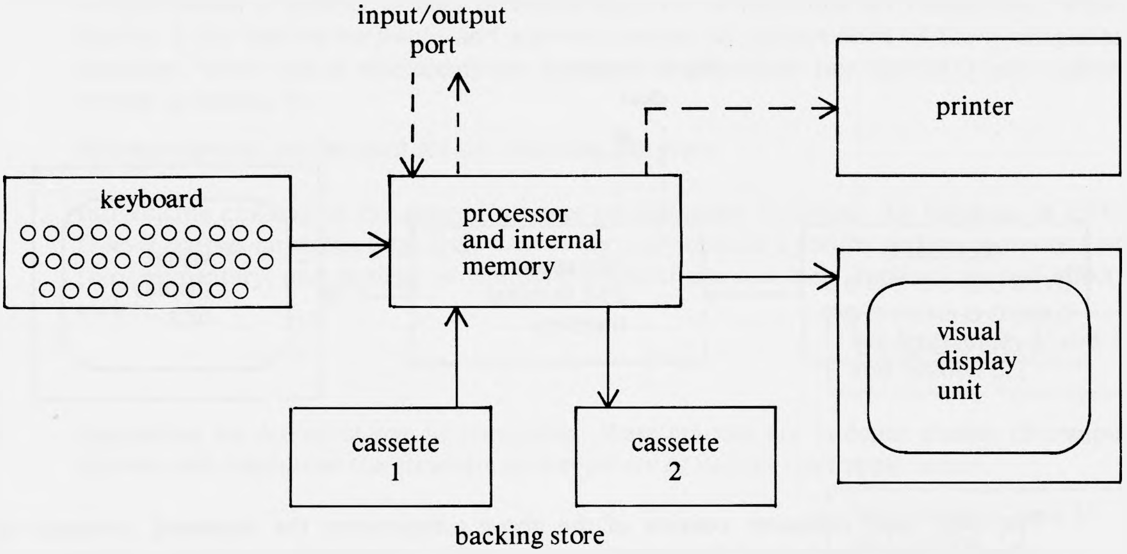


This system contains the cheapest form of *backing store*, the *audio-cassette*, which is sometimes built into the microcomputer or comes as a separate unit (ie, a suitable domestic recorder). However, it is rarely practical to handle large amounts of *data* using a single cassette and thus the storage of *lists* and *files* should be avoided.

The *printer* is shown as a highly desirable additional output device.

System B is suitable for applications 2.1 and 2.3.

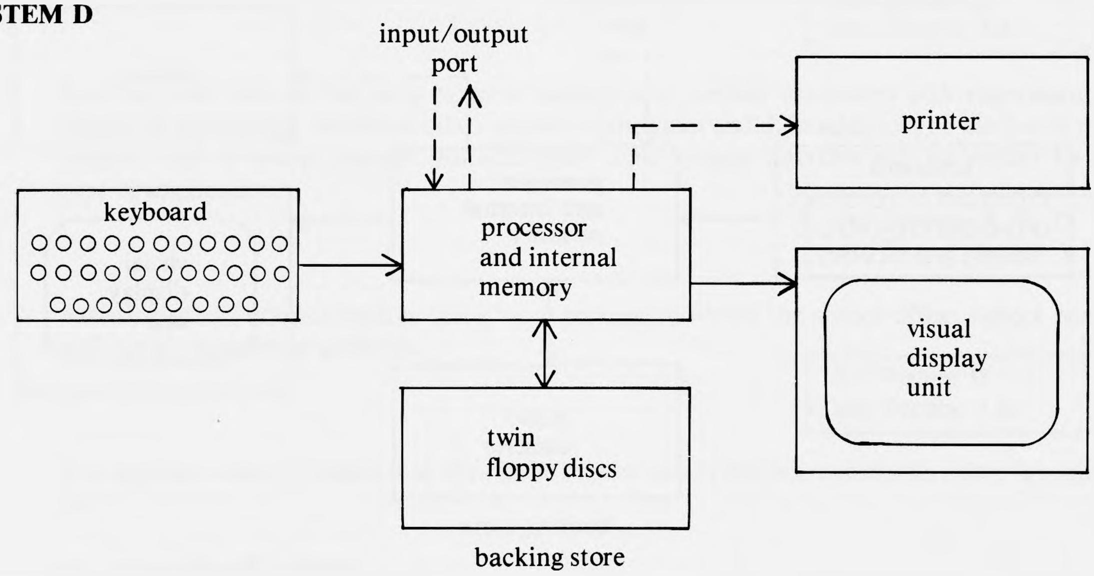
SYSTEM C



This system has the provision of two (or twin) cassettes which can be controlled (turned on and off) by the microcomputer. It enables *data* to be automatically read in from one cassette, processed and then stored (or recorded) on the other cassette. This permits large files of data to be processed.

System C is suitable for applications 2.1 (for business stock control type of transactions), 2.2 (but not adequate on its own for this), 2.3 and the simpler forms of 2.4.

SYSTEM D



This system uses twin *floppy discs* which enables sophisticated data handling where large amounts of *data* can be retrieved and recorded in fractions of a second; any item on the file can be *accessed* in two or three seconds, whereas with cassettes it may take several minutes.

It is recommended that twin floppy discs are used as the difference in cost (a few hundred pounds) between the single and twin mini-floppy systems greatly enhances the flexibility of the system.

System D is suitable for applications 2.1 to 2.6.

3. THE COMPUTING FACILITIES TO SUPPORT THE DIFFERENT KINDS OF EDUCATIONAL APPLICATIONS

While reading this section it should be borne in mind that the microcomputer is usually used in one of three different ways: as a demonstration facility with output to the school television monitor so that the whole class can see; interactively by one or two students working by themselves; and by batch whereby students complete a piece of work during the lesson, or possibly as homework, to be processed by the computer at another time when the students themselves are not present.

The sections that follow describe the hardware and software facilities needed to support the categories of use described in Section 2. The computing configurations needed to support these activities are also illustrated in Section 2. Specific items of hardware are discussed in Section 4 and the languages referred to (eg, BASIC) are discussed in Section 5.

3.1 Familiarization and computer studies to CSE/O-level

HARD WARE

Although System B, together with a printer, would be sufficient for courses of this nature, System C is desirable since it enables the students to write and run *programs* illustrating the typical application of computers in business and commerce. However, one microcomputer is really insufficient for large classes although the *low-resolution graphics* available on many microcomputers open up opportunities for student project work. The students are motivated by applications which use graphics but many examination authorities demand that project work in this area is documented with a permanent copy of the graphics produced. One way of achieving this permanent copy is by photographic snapshots of the screen.

SOFT WARE

Most courses at this level use BASIC language and the better the version of BASIC the more satisfactory the course can be. If BASIC is not used, some other suitable *high-level language* must be provided.

3.2 A-level courses in computing

HARD WARE

It is essential that students at this level are given direct experience of handling *files* of *data.* System D would provide all the computing resources needed for a course of this level, or alternatively, Systems B or C with access to a *mainframe computer* to give additional support could be used. A *printer* would be essential both for *program* development and for evidence of a successful run of any *program*. Students of this level need a more detailed course on machine structure and such a course can be based on the *architecture* of the microcomputer itself. However, some microcomputers do not allow the user to descend into the *machine code* or its associated *assembly language*, in which case the machine structure part of the course will need to be undertaken using alternative facilities.

SOFT WARE

Ideally, A-level students need to study a *high-level programming language* other than BASIC but again this need not necessarily be undertaken on the school microcomputer. Courses at CSE/O- and A-level are enhanced by the presence of good *applications software*, particularly that which illustrates some of the problems associated with commercial large-scale data processing. Such software can be homegrown or built up by the school as it gains experience, but it is more likely to arise as a result of collaboration with other schools, as recommended in Section 1.

3.3 Computer assisted learning with small amounts of data

HARD WARE

Many *computer assisted learning* programs, such as those developed by the 'Computers in the Curriculum Project'* will just fit into a cassette-based microcomputer, such as System B, even if the total storage available is as low as *16K bytes.* A modest configuration such as this is not too heavy and can easily be moved to the appropriate department in the school. However, as is explained in Section 4, cassette tapes are slow to use and do not always read successfully first time so that a teacher has to be fairly dedicated to use a computer under these circumstances. This is particularly a problem if the computer is going to be used for group demonstration despite the motivating illustrations that even *low-resolution graphics* can provide. Many teachers find microcomputers such a powerful facility that they are prepared to put up with some of these inconveniences.

*Schools Council project, whose packages consist of the computer program and printed material for the teacher and student.

SOFT WARE The *applications* software needed to support *computer assisted learning* in schools is still not plentiful and even that which exists may not be in a form which is suitable for a particular microcomputer. It is not easy to write good *computer assisted learning* material and it is a time-consuming exercise. Some of the simpler applications in mathematics and science can be successfully written within the school but progress will necessarily be slow unless teachers have adequate time to produce such CAL packages.

3.4 **Computer assisted learning with large amounts of data and computer managed learning**

HARD WARE Currently there is little experience of this type of work on a microcomputer but indications suggest that the applications which were developed for *mainframe computers* are capable of being rewritten for a microcomputer system, such as System D (or System C for simpler uses). For *computer managed learning* particularly, the speed of the *printer* may be critical. Also some method of coping with the input of the student data needs to be achieved and at the present, despite its drawbacks and obvious expense, *a mark-sense card reader* is a way of achieving this.

SOFT WARE Packages associated with field studies subjects, such as geography, biology or history, could come into this category, as could information retrieval techniques. However, little *software* currently exists in this area and its writing involves a level of expertise to which the school could not normally aspire unless teachers are released from teaching duties specifically for this purpose.

3.5 **Real-time on-line data capture and control**

HARD WARE Work of this nature in schools is still in its infancy but the possibilities look attractive. There are a number of alternative approaches; one way is to use special purpose *hardware* to perform one particular task, such as a *data logger* (a device which can be set to take experimental readings at predetermined times). However, there are many other possibilities and a minimal system such as System A with the right input/output *port* can be used with *analog-to-digital converters.*

SOFT WARE The amount of *software* needed for many of these applications is not large and even if it has to be written at *assembly language* level it should not prove too difficult and tedious. However, if a fully fledged microcomputer is used it should be possible to use BASIC or some other *high-level language* for the programming aspects of this work. Many versions of BASIC allow the user to descend into *machine code* to enable processes not normally available in a *high-level language* to be carried out.

3.6 **School administration**

HARD WARE System D would be essential and so would a high-quality *printer.* It is doubtful whether a microcomputer can successfully be shared between the school office and other educational uses. At the same time there is neither the experience nor the *software* currently available to dedicate a microcomputer to school administration. However, the situation is changing rapidly, and school administration will be able to benefit from the current advances being made in the area of *word processing.*

SOFT WARE Two particular problems are potentially solvable in terms of the school-based microcomputer. These are the processing of school examination results and school records and problems associated with the school timetable. However, there are problems associated with these applications and a school should not embark on using a microcomputer for these applications without expert advice and support. The development of *word processor* software is taking place rapidly and it should be possible to adapt such advances to the benefit of the school administration. However, even when this has been achieved, secretarial training will have to be undertaken and this is not a problem that should be underestimated.

3.7 In the preceding paragraphs applications of microcomputers in the various educational activities have been described. A single microcomputer in a school will not provide the facilities necessary to cope adequately with all the activities mentioned. Those schools that have been used to using batch facilities or links to mainframes would be well advised to retain these links and supplement them with the facilities offered by microcomputers. In particular, regional support can be provided by the use of microcomputers as terminals to mainframes in order that a local authority can provide software support for all its schools from one point. The access to local central mainframes does not need to be continuous but does allow access to library material, etc.

4. THE EQUIPMENT (HARDWARE)

4.1 Microcomputer systems

4.1.1 As already emphasized, because the hardware determines what the system can be used for, it is important to make a choice which will satisfy existing needs and foreseeable needs.

4.1.2 The microcomputer should be as simple to operate as possible. This implies that when first plugged in the machine should respond, in plain English, and immediately guide the user in deciding what to do next, for example, by providing a list of alternatives. Before the system begins to communicate sensibly it should not have to be necessary to make too many connections or press too many buttons. However, to ensure that the system can be expanded it will be necessary to have certain connectors which provide access to the electronic circuitry.

4.1.3 It is important that the system should be upgradable and expandable without rendering the existing components redundant.

4.1.4 The *keyboard* is the means by which the user enters programs and data into the computer. It should be like a standard typewriter keyboard, with large, easy-to-use keys. It is an advantage to have both upper- and lower-case letters with the numerals still available when the key shift is in operation. There should also be some special symbols and keys, such as 'RETURN'. The keyboard will receive a great deal of use and it is advisable to ensure that it will be hardwearing.

4.1.5 The *processor/internal memory* box in the diagrams of Section 2 contains the microprocessor chip, *read only memory* (ROM), *random access memory* (RAM) and other integrated circuits. It is the work-horse of the system. The processor, in executing the program which is stored in the memory, does the arithmetic and other data handling required by the program. A small amount of ROM ensures that the system retains its fundamental starting up instructions when it is switched off. ROM and RAM requirements are dealt with below.

4.1.6 The *visual display unit* (VDU) is normally a small television display and this is dealt with in more detail below. It is desirable that there is provision for either a conventional television receiver or an additional large screen monitor to be connected to the microcomputer to enable a group of people to see the display from a distance.

4.1.7 The *backing store* is the device whereby programs and data can be stored and kept for later use. There are two forms which are of interest to schools and colleges at present — cassettes and discs. The cheapest form is the *audio cassette*, whereas the *floppy discs*, which are like small records which rotate inside a square plastic envelope, are more expensive but much more useful. These are dealt with in more detail below.

4.2 ROM and RAM requirements

The amount of memory (ROM and RAM) in the microcomputer will determine how large and complex a program (with associated data) it can run. At least *8K* (*8192*) *bytes* of RAM should be provided for the user's programs and data.

Typically, at least an additional 8K of memory is needed to store a *high-level language* (such as BASIC). Sometimes the BASIC language is stored in ROM and the user can then work in BASIC as soon as the computer is switched on; if this is not the case, a further 8K of RAM is needed into which BASIC can be loaded.

Thus, the minimum requirements are:

8K of user RAM + 8K of BASIC in ROM + 'Starting up' ROM

or,

16K of RAM (8K for the user and 8K into which BASIC can be loaded) + 'Starting up' ROM.

Extra RAM will be needed for systems which use discs.

4.3 **Visual display**

4.3.1 The visual display will be a television display in either monochrome or colour. The computer should generate between 40 and 72 typographical characters per line, preferably in both upper- and lower-case. 40 characters per line gives a very clear image which can be easily read and is similar to the *teletext* and *viewdata* displays. However, 40 characters is rather too few for those applications which need a lot of data to be displayed. The display should show at least 24 lines of text on the screen.

4.3.2 There can be a saving in cost if the microcomputer can be connected to a conventional domestic television receiver rather than having the display in the main unit. However, the quality of the picture seen on a standard television screen may be inferior to that obtainable on a monitor, especially in the case of colour displays. It is not possible to display more than about 64 characters per line using a domestic television receiver.

4.4 **Graphics**

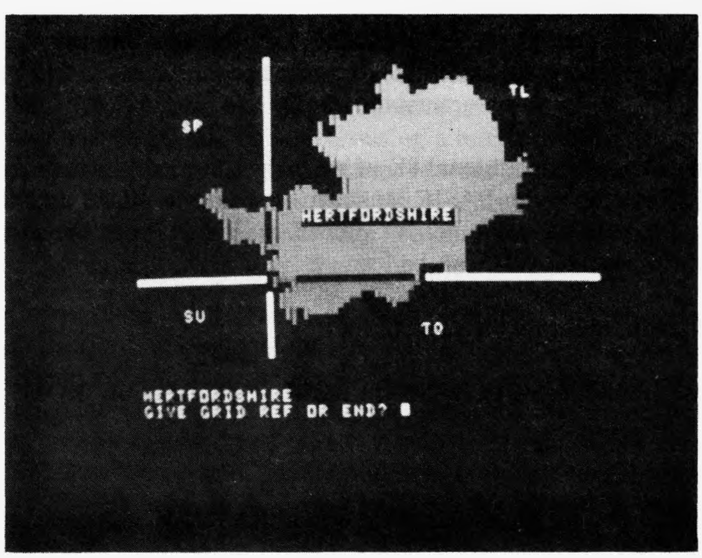
There are several types of graphics available on microcomputers, as detailed below.

4.4.1 *Fixed shapes* such as hearts and diamonds. These have a limited use, often being used for games.

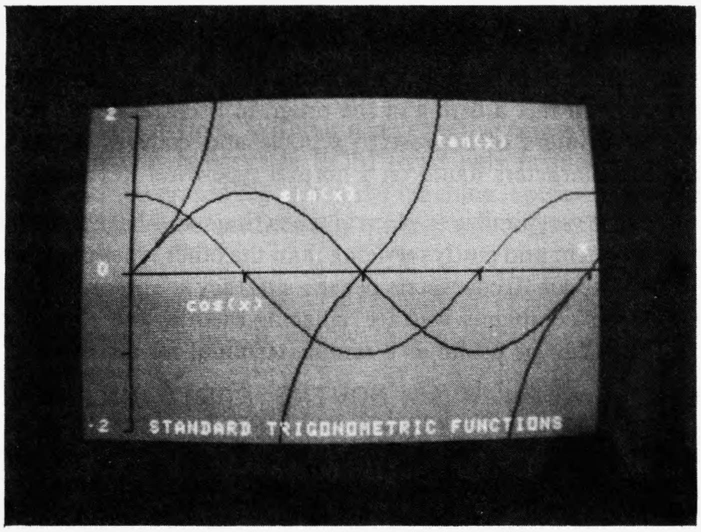
4.4.2 *Low-resolution graphics* of the type seen on teletext and viewdata displays. Simple graphs and diagrams can be displayed. In these the 'cells' normally occupied by each of the 40 characters on the 24 lines are divided into 6 smaller cells, for example,

One character cell becomes six smaller cells.

Thus, the display consists of 80 (40 x 2) boxes horizontally by 72 (24 x 3) boxes vertically. A typical page of display is shown below.



4.4.3 *Medium-resolution graphics* (often called high-resolution by the overzealous) have between 80 and 250 horizontal picture points. 'Pictures' and good graphs can be drawn with appropriate programs, as the following illustrates.



4.4.4 *High-resolution graphics* should imply more than 250 horizontal picture points. Resolution of 1000 horizontal picture points can only be obtained on expensive displays at present.

4.5 **Backing store**

4.5.1 *Audio-cassettes* provide a cheap means of storing programs and data. However, they are inconvenient to use because access to the various programs and items of data can only be obtained by searching through the tape and this is a time-consuming process. It is a considerable advantage to have a tape counter and, even though more expensive, advisable to use short cassettes for programs which are required to be used regularly.

In addition, programs may take several minutes to transfer to and from cassettes. Recordings produced by one make of microcomputer are unlikely to be readable by a different make. If the recorder is not built into the microcomputer, it is advisable to choose one which has been recommended by the manufacturer for the system. Twin cassettes enhance the system considerably because input and output processes can take place in quick succession.

4.5.2 *Floppy discs* give rapid access to programs and data and enable large programs to be loaded (fed into) the microcomputer in seconds. However, they are more expensive than cassettes.

They are available in two sizes:

*Mini-floppies** are 5¼ inches in diameter and can store at least 70K bytes of data, which is equivalent to about 2000 lines of BASIC program.

*Standard floppies** are 8½ inches in diameter and can store at least 250K bytes of data (about 7000 BASIC program lines).

Some systems permit both sides of each disc to be used thus doubling the capacity. There is little standardization of disc systems at present, so discs recorded on one system are generally not usable on another.

4.5.3 It is important to note that the efficiency of the disc system provided with each microcomputer will depend critically on the quality of the software incorporated. Advice should be sought from those who have had experience using a number of *disc-operating systems.*

*Typical costs are provided in the Appendix.

4.5.4 Another form of backing store, which is expected to become increasingly important, is the plug-in ROM module. However, this can only carry a program or data which is required to be read into the microcomputer, for example, BASIC language; it cannot be used to store data or programs created by the user or as an output store.

4.6 Printers

For reasons described in Section 3, a printing device giving a permanent copy of the output from the program, or in fact a listing of the program itself, is highly desirable. There are, broadly, four types of printers of interest to schools and colleges at present which are capable of producing the characters found on a normal typewriter keyboard.

4.6.1 *Teleprinters** are very similar to electric typewriters and have many moving parts which require much more frequent and costly servicing than the other types of printers listed. Because they are noisy and slow (about 10 characters per second) they are not a first choice but may be picked up at no charge from companies who are replacing them by more modern equipment. As they have a keyboard they can be useful as a remote terminal for a computer.

THERE ARE ONLY 15 MOVING PARTS ON THE DAISY PRINTER

- COMPARED WITH ABOUT 2000 ON A GOLFBALL TYPEWRITER

4.6.2 *Dot matrix printers** have a vertical column of 7 or 9 needles which hit a typewriter ribbon and thus mark the paper. Characters are formed from rows of dots as shown below. Printers with only 7 needles are unable to print 'descenders' such as 'gjpqy' with their tails below the line and untrained readers often find such print difficult to read speedily. The quality of the print may not be suitable for some applications. Dot matrix printers are quite fast (often over 100 characters per second) and are suitable for heavy use. Costs vary over a range of 3:1 with some of the cheaper printers representing very good value. Running costs can be quite high because of the need to use nylon ribbons. Some of these printers also incorporate a keyboard and can be used with a computer in a similar way to a teleprinter.

Sample character 

There are only 15 moving parts on the daisy printer
- compared with about 2000 on a golfball typewriter

4.6.3 *Daisy-wheel printers** have a daisy-shaped wheel embossed with letters at the end of each petal. This wheel rotates rapidly and is hit by a hammer whilst in motion. These printers produce a high quality print at a slower, but quite acceptable, speed (50-100 characters per second). An additional advantage is the possibility of changing typefaces in a similar way to a golfball typewriter. Prices tend to be high but this must be accepted if very high quality is absolutely necessary.

There are only 15 moving parts on the daisy printer
- compared with about 2000 on a golfball typewriter

4.6.4 *Thermal printers** have a printing head with hot areas which moves across specially sensitized paper leaving characters formed in dots. The paper is many times the cost of ordinary paper. These printers are reasonably small and quiet. They seldom have more than 40 characters per line and their speed is 30-60 characters per second. The quality of the print is adequate and machine costs are moderate.

THERE ARE ONLY 15 MOVING PARTS ON THE DAISY PRINTER
- COMPARED WITH ABOUT 2000 ON A GOLFBALL TYPEWRITER

4.6.5 In general, it can be said that the quality of the output and the speed of the output of printers depends on how much is paid for them.

4.7 Mark-sense card readers*

These machines can be attached to a microcomputer but they are not in common use because there are both hardware and software problems in the interfacing requirements. (See also 3.4, glossary and appendix.)

*Typical costs are provided in the Appendix.

4.8 One of the performance parameters of hardware is the rate at which data is transfered. This is referred to as the *baud* rate of transmission and normally represents the number of *bits* per second, but the term is used rather loosely.

5. **SOFTWARE**

5.1 Software is the general name given to all the programs which enable a computer to be used. It exists at three levels.

The most fundamental software, the first level, is built into the machine by the manufacturer and is known as the *operating system.* Without it no effective communication with the machine whatever would be possible. It is based in ROM (read only memory) which means that the operating system can never be erased (eg, when the machine is switched off) and that a part of the system's memory can never be used for any other purpose (see 5.2).

The second level of software is known as a *compiler* or an *interpreter* (each works in a different way) and is described as, for example, a PASCAL compiler or a BASIC interpreter, according to the language with which it deals. It enables a computer to run programs written in a high-level language (see 5.3).

The user of the computer writes his programs, designed to carry out specific tasks, in a high-level language which was originally defined to be independent of any particular system. This is the third level and these programs are known as *applications software* and are usually loaded into RAM (random access memory). The contents of RAM are not permanent and the user can change them whenever he wishes (see 5.4).

Most *high-level languages* unfortunately exist in a number of incompatible versions which means that each microcomputer tends to have its own compiler or interpreter for each language, so that programs written for one system can rarely be run on another without prior amendment. It is important to ascertain before making a purchase just what high-level language each system will support and in what version (see 5.5).

The above account has been kept deliberately simple since a full description of the nature of software lies well outside the scope of this USPEC.

> If the above description of software has not left you with a reasonably clear picture of the relationships between the various types, the following analogy of a very familiar facility, the automatic telephone system, may be of some help.
>
> When you lift the receiver you hope and expect the telephone system to respond with a signal — the dialling tone. This tells you that you have established contact with the system which now expects you, in your turn, to dial in a number. After this the system replies with a ringing tone or an engaged signal (both of which imply that it acknowledges that your dialled number is a valid one) or a number unobtainable tone (which implies that the system regards the dialled number as invalid — perhaps it never existed, or is discontinued, or belongs to a telephone which is out of order). All these noises have been built into the system of the Post Office as a basic level of very restricted communication, which is fundamentally similar to the *operating system* of a microcomputer.
>
> Imagine that you have in fact called the number of a large firm. When the firm's operator answers your call you can communicate with her much more extensively than you could with the Post Office's system, but still with restrictions. It would, for example, be of little use asking her the performance of certain electronic intregrated circuits, but she should be able either to put you in touch with the appropriate engineer or to tell you when and where he will be available. This second and more extensive communication corresponds with the *compiler* or *interpreter* associated with the *high-level language* you have chosen.
>
> When finally you are talking to the engineer, you can speak without restriction, provided still that each of you is familiar with all the terms used by the other. This level is that of the *applications* software.

5.2 **Operating system**

This is normally supplied by the manufacturer but certain proprietary operating systems can be obtained from software houses for popular makes. However, neither are likely to point out its weaknesses. Advice should be sought, if possible, from someone who has had experience of an individual system to ascertain whether it has been easy to use. Does it give the services needed? Is communication with it simple? Does it load other software easily and does it display helpful messages while it is doing so? A blank screen can be most disconcerting and, since mistakes are inevitable, speedy, clear and unambiguous messages are most desirable.

5.3 **Compilers and interpreters**

These also come from the manufacturer and can only be assessed in action. A distinction between compilers and interpreters is made in the glossary. Compilers are usually faster to process data but may be more difficult to use; they can sometimes check the whole program before it is running and this facility can be most helpful. In compensation for their slower speed, interpreters sometimes allow *error messages* to be displayed even as each single line of a program is typed in.

5.4 **Applications software**

This can be begged, borrowed, bought or written by the user. Remember always that the multiplicity of versions of the languages might mean that some or all of the programs available will not run on the system being considered. So if, for example, it is intended to use the materials of the 'Computers in the Curriculum Project' (written in BASIC), make sure that the mode in which they are distributed can be read by the system and that the program can be executed by the system. Another method, but much more tedious, is to obtain a copy of the *listing* of the program required and type it in from scratch making whatever modifications are necessary on the way. If international or national standard versions of a language do not exist, it is always possible to define a local standard. Schools working closely together can gain tremendous advantages in this way by producing collections of programs which each of them can run without any prior amendment.

5.5 **High-level languages**

Many languages are available but the ones listed below are the most relevant to teachers in secondary education.

BASIC

This is the language most commonly available with microcomputers and is very widely used in secondary education. Unfortunately, it exists in many distinct versions and programs are therefore unlikely to be transferable from one system to another. The only internationally accepted standard is that of ANSII Minimal BASIC, which is inadequate for anything other than elementary use.

There are several different versions called 'Extended BASIC' which come from different manufacturers and again they are not compatible or suitable for all applications. They vary in the facilities which they provide and care must therefore be taken to find out precisely what each will do.

The great advantage of BASIC as an educational language is that it is usually an interpreted language which enables a large number of errors to be detected and corrected immediately. BASIC can sometimes be obtainable as a compiled language and though it is then more difficult to use, this may be preferable where it is intended to do very complex calculations rapidly. Most currently available computer assisted learning material is written in BASIC. However, BASIC is not a good language for long programs.

COBOL
This is most widely used in commerce and therefore has good, realistic file handling properties. The implementations of COBOL adhere closely to an accepted standard which eases the problem of transferability between systems. It is, however, both verbose and clumsy, particularly when used for small programs and therefore is quite unsuitable for O-level work. In particular, being a compiled language, it does not provide adequate *error messages* for beginners.

FORTRAN 4
This was designed as a scientific language. It has a commonly accepted standard and is, therefore, transferable. It has clear advantages for scientific work but requires a high degree of skill for effective use. It is not suitable for use below A-level work and has the usual problems of a compiled language in that its *error messages* are not very good. It also does not handle text well.

PASCAL
This looks like becoming the first widely available structured *programming language* for microcomputers. It is being increasingly used in programming courses in higher education and therefore presents the advantage that school leavers who have used it will be well prepared for these courses.

PILOT
This has been specifically designed to enable highly conversational branching programs to be easily implemented. It is not widely used yet but could be of considerable importance.

6. DOCUMENTATION FROM THE SUPPLIER

6.1 Good documentation is essential and should be sufficiently comprehensive to enable:

(i) the system to be checked on arrival both for completeness and to see that it is in full working order

(ii) the naive user to operate the system adequately

(iii) the competent user to exploit the system fully

(iv) future applications to be planned, including additional items of hardware needed.

6.2 To meet the above at least the following should be provided in four clearly defined categories.

Category 1
(a) A full list, including serial numbers, revision or version numbers, of all hardware and software supplied to the user.

(b) A layman's guide, in simplest terms, to setting up the system.

(c) A layman's guide to executing simple diagnostic tests — such as memory tests — as soon as the system is set up.

Category 2
A brief explanation of the use of all the component parts (such as keyboards and floppy disc units) and of each item of software supplied. These explanations would normally only be a few paragraphs long — one paragraph to explain what can be done, and one or two to give simple instructions on use.

Category 3
Category 2 needs to be supplemented by much fuller documentation which should include:

(a) full circuit diagrams, layout and component lists (though the user is cautioned not to invalidate the terms of the warranty)

(b) full *source listings* of the monitor and operating system

(c) explanation of the detailed structure of the operating system with full explanation of subroutines which could be of use to the experienced user

(d) full operating instructions for all high-level software supplied with the system: the instructions should include a memory map and details of any subroutines available to the user

(e) details of any modifications needed to give compatibility with other programs supplied (such as the *text editor*), which should be clearly set out

(f) a full explanation of all *error messages*

(g) a full and effective index.

Category 4
The supplier would be wise to include a section giving detailed introductions to software and hardware features which are available from himself or other sources. The addresses of hardware and software suppliers in each county of sale would also be of value. Lastly, details of service arrangements should be included.

7. SUPPORT FOR HARDWARE AND SOFTWARE

7.1 Effective support is vital if the system is to be used fully and effectively by many staff and pupils.

7.2 Maintenance should be provided by the manufacturer or an agent. There are various levels of maintenance contracts which may not be essential for the electronic components themselves but are advisable for electromechanical components.

7.3 The local education authority through local organizations is encouraged to have available a good selection of well documented programs. The documentation should be for the teacher as well as the pupil. This again implies the importance of a county standard system.

7.4 The user will need a short set of clear instructions to enable naive users to cope with the system. The instructions should tell them where to get more information if needed. Users will need advice on:

(a) the range of material available and its characteristics

(b) how to integrate the computer into their teaching most effectively

(c) how they can either write (or get written) programs that they can see the need for. It needs to be remembered that this will involve many man hours.

7.5 Active encouragement should be given to staff to evaluate available material.

7.6 Providing software support for schools across an LEA is not a trivial matter. Although there are many enthusiastic software writers, there are significant problems which can arise with any attempt to maintain and distribute this software even amongst owners of the same machine. The following are the more important problems.

(a) Even slight changes in the computer configuration can be sufficient to prevent software running without modification.

(b) The user documentation needed to enable software to be used by people other than the originator must be of a high quality. Even when this exists, it is not always easy to copy it for distribution with the software.

(c) Inevitably software bugs will appear after distribution has taken place. A method of correcting these bugs, possibly by redistributing the software, has to be found.

(d) The software has to be catalogued so that users know what is available.

One way of distributing software is to store it all on a large computer to which all the users have access. With the right interface and communicating software it is then possible to arrange for the individual users to connect their microcomputer to the main computer via the telephone so that individual applications programs can be transmitted from the main computer into the microcomputer. Once the transfer is complete, the telephone link can be broken and the microcomputer can run the program in stand-alone mode. Additionally, the program can be stored locally for subsequent use.

This method of keeping a library of micro-software for a group of users is operational for at least one LEA and it has demonstrated, through use, that it is capable of overcoming many of the problems listed above. Experience shows that a system for software distribution along these lines can work but it involves a good deal of time and expertise to set up and operate.

8. GLOSSARY OF TERMS USED IN THIS USPEC

The glossary only contains the terms which have been used in this USPEC. These and other terms are more fully explained in the British Computer Society publication, *A Glossary of Computing Terms.*

Access. The process of obtaining data from the storage device.

Analog-to-digital converter. A device used to interface various pieces of measuring equipment to a computer in which continuous (analog) measurements are converted to their (approximate) digital equivalents (see 3.5).

Applications software/package. A set of specialized programs and associated documentation to carry out a task (such as stock control or teaching a concept in physics) (see 5.4).

Architecture. The structure and interconnections of the component parts of the microcomputer (see 3.2).

Assembly language. A low-level programming language which is translated into machine code by an assembler (ie, a program usually provided by the manufacturer and incorporated into the microcomputer) (see 3.2).

Audio-cassette. A cassette, as manufactured for a standard domestic (sound) cassette recorder, which is used for storing data and programs (see 4.5.1).

Backing store. The means of storing programs and data which can be kept for later use (eg, audio cassettes and floppy discs) (see 4.5).

BASIC. (Beginners All-purpose Symbolic Instruction Code) the high-level language most commonly available with microcomputers and most widely used in education (see 5.5).

Batch processing. A technique in which computer processing does not begin until all the data and programs have been loaded into the machine (ie, 'batched').

Baud. The rate at which data is transferred (normally synonymous with the number of 'bits' per second) (see 4.8).

Bits. (Binary digit) one of the digits used in binary notation (ie, either 0 or 1).

Byte. A set (usually 8) of 'bits' often corresponding to a single character (see 4.2).

COBOL. (Common Business Oriented Language) the high-level language most commonly used in business and commerce (see 5.5).

Compiler. A program which translates a high-level language program into a computer's machine code. Each high-level language instruction is changed into several machine code instructions before the program is executed (see 5.1 and 5.3).

Computer assisted learning (CAL). The use of the computer to aid the pupil in his learning process (see 3.3).

Computer managed learning (CML). The use of the computer to organize or keep records of individualized learning, etc (see 3.4).

Daisy-wheel printer. A printer which produces well formed characters by impact of a rotating wheel which can be easily changed (see 4.6.3).

Data. Information coded in a form acceptable for input to, and processing by, a computer system.

Data logger. A device which can be set to take experimental readings at predetermined times (see 3.5).

Disc-operating system. The set of commands available to control the microcomputer and its associated discs (see 4.5.3).

Dot matrix printer. A printer which forms characters from rows of dots made by a vertical column of 7 or 9 needles which hit a typewriter ribbon to mark the paper (see 4.6.2).

Error message. An indication from a computer program that an error has been detected, and possibly where that error has occurred (see 5.3).

Extended BASIC. A version of BASIC containing extra features. There is no standard for Extended BASIC (see 5.5).

File. An organized collection of related records (see 5.5).

Fixed-shape graphics. Simple graphics such as hearts and diamonds (see 4.4.1).

Floppy disc. A non-rigid, lightweight magnetic disc which is used for storing data and programs for rapid access. There are two sizes, mini and standard (see 4.5.2).

FORTRAN. (FORmula TRANSlator) a high-level language which was designed specifically for scientific use (see 5.5).

Graphics. The ability of a computer to draw diagrams (shapes and lines) as opposed to merely printing text (see 4.4).

Hardware. The term used to mean the various pieces of electronic equipment which make up the microcomputer.

High-level (programming) language. A problem-oriented language which enables the user to communicate with the computer in near English (and thus to solve problems) (see 5.1 and 5.5).

High-resolution graphics. Graphics where the diagrams displayed have a resolution of more than 250 horizontal picture points (see 4.4.4).

Interface. The connection between two systems or parts of a system.

Internal memory. (See Memory).

Interpreter. A program which translates and executes anther program (written in a high-level language) one instruction at a time (see 5.1 and 5.3). o

K. Used to indicate the size of the immediate access store (ie, the store within the central processing unit); in computer technology its value is 1024 (2^{10}), eg, 8K of store is 8192 storage locations. It is derived from 'k' meaning 'kilo' (1000) (see 4.2).

Keyboard. A set of keys similar to that of an electric typewriter producing electrical signals (see 4.1.4).

Language. (See high- and low-level languages).

Light pen. A light sensitive pen which is used to interact with the computer via the screen of the visual display unit.

List/listing. A line-by-line sequence of program statements which comprise a computer program (see 5.4).

Low-level (programming) language. A machine-oriented language in which each program instruction is related to a single machine code instruction.

Low-resolution graphics. Graphics where the diagrams displayed have only limited resolution (ie, 80 horizontal picture points or fewer) (see 4.4.2).

Machine code (instruction). Instructions which make up the basic repertoire of the computer, and which can be executed without further translation (see 3.2).

Mainframe computer. A large computer installation.

Mark-sense card reader. An input device which reads by optical means cards that have been marked with a pencil in predetermined positions (see 4.7).

Medium-resolution graphics. Graphics where the diagrams displayed have a resolution of between 80 and 250 horizontal picture points (see 4.4.3).

Memory. Part of a computer where data and instructions are held or stored.

Microcomputer. A small computer based on the use of microelectronic 'chips'.

Microelectronic chip. A complex electronic circuit formed on a single silicon chip using large scale integration (LSI) techniques.

Microprocessor. A processor formed on a 'chip' (see 4.1.5) (see also Processor).

Operating system. An advanced form of control program which allows the computer to run without the need for continual operator intervention — it provides an overall means of communication between the user and the computer (see 5.1 and 5.2).

PASCAL. A high-level language which looks like being the first widely available structured programming language for microcomputers (see 5.5).

PILOT. (Programmed Inquiry Learning Or Teaching) a high-level language which has been specifically designed for computer assisted learning and facilitates the use of interactive branching programs (see 5.5).

Port. The point at which signals enter or leave the central computing system.

Printer. A machine which produces printed output (printout) from the computer (see 4.6).

Processor. The central control unit of the computer which also carries out the arithmetic functions.

Program. A complete set of statements to perform a specified task.

Programming language. A set of key words, and the syntax rules for combining these words, with which a user may instruct the computer to perform tasks.

Random access memory (RAM). Memory of a computer which is capable of holding the user's program and data. The content of this kind of memory is lost when the computer is switched off (see 4.2).

Read only memory (ROM). Memory whose content is fixed (ie, not lost when the machine is switched off, nor can it normally be changed by the user). The operating system is usually held in ROM so that it is available as soon as the computer is switched on (see 4.2).

Real-time on-line data capture, analysis and control. A system which is able to receive continuously changing data from outside sources, and which is able to process that data sufficiently rapidly to be capable of controlling or influencing the sources of data (eg, air-traffic control, airline bookings) (see 3.5).

Remote computer/terminal. The term used to describe any input/output device which is used to communicate with the computer from a remote site.

Software. A program or sets of instructions designed to perform a particular task.

Source listing. A source listing of a program is one given in the programming language in which the program was written, ie, a listing before translation (see 6.2).

Teleprinter. An input/output device which prints on ordinary paper (see 4.6.1).

Teletext. The generic name for the Ceefax and Oracle systems which enable text and simple graphics to be transmitted via the broadcast television network and displayed on the domestic television receiver (see 4.3.1).

Thermal printer. A printer which leaves characters formed in dots on specially heat-sensitized paper (see 4.6.4).

Text editor. A piece of software used to edit text stored in a computer. The text may be a computer program, an article or a letter (see 6.2).

Viewdata. The generic term for an interactive data storage and display system using the telephone network to enable access to central computers from the user's visual display unit (eg, the Post Office PRESTEL system) (see 4.3.1).

Visual display unit (VDU). A display device, incorporating a cathode ray tube (as used in a television receiver), on which information is displayed (see 4.3).

Word processor. Computer, keyboard, printer and software sold as a package to aid the preparation of text. It is capable of storing drafts so that later they can be edited and a fair copy can be produced without a complete retype (see 3.6).

APPENDIX: TYPICAL COST FIGURES, AS AT DATE OF PUBLICATION

System A	£ 200 (including display)
System B	£ 500
System C	£ 500 — £1000
System D	£2000 (mini floppy discs) £3000 (standard floppy discs)

Printers

Teletype	£1000
Small dot matrix	£ 250 ⟶
Large dot matrix	£ 550 ⟶
Daisy wheel	£1800
Thermal	£ 200 ⟶

Mini floppy discs	£2.50
Standard floppy discs	£5.00
Mark sense card readers	£600 ⟶
Visual display units	£500 ⟶
Monochrome 'monitors'	£100 ⟶