PROGRAMME INTERCHANGE IN THE DIGITAL DOMAIN - AN OVERVIEW

FRANCIS RUMSEY

University of Surrey
Guildford, U.K.

Digital audio programme material may be transferred from one system or location to another using a variety of means. Recording media such as disks and tapes may be physically transported, but there are compatibility problems between systems; real-time interfaces may be used to replace analogue interconnections for signal routing, and computer-style networks may be used for the transfer of audio in the form of data files. The relative merits of different approaches to programme interchange are examined with reference to operational requirements, and the various problems of data format and compatibility are examined.

0 INTRODUCTION

In the analogue domain, two-track audio recordings are typically made onto tapes. Such tapes, in the professional field, are normally quarter-inch mono or stereo recordings, playable on virtually any machine in the world, there being only minor issues of equalisation and speed to contend with to ensure replay compatibility. The Dolby A noise reduction system has also been widely accepted as a 'standard' for interchange. In the multitrack field the interchange format for analogue recordings has been 2 inch, 24 track tape, and although there are alternative 'budget' formats using narrower tape they are normally used in private studios where interchange is not an issue. Such a situation has only been reached after many early years in which such compatibility could not be taken for granted. Signal interconnections in the analogue domain are also relatively straightforward, since the audio signal is carried either on a balanced or unbalanced line, conforming to one of a selection of conventions concerning peak signal level. Even if the receiving device does not conform precisely to the conventions of the transmitter, the likelihood is that the programme will still be retrievable at the receiver, perhaps with slightly inferior quality.

The situation in the digital domain is considerably more complicated. Once an audio signal is converted into a data signal the issue of formatting becomes important. What does each bit mean, and what is its function? Unless a receiver is capable of interpreting the format of the transmitted data, or unless a replay device conforms to the same format as the original recorder, there will be no communication and no replay. Except in private operations where a facility for programme interchange is not needed, international standards become necessary, since without standards there is no guarantee that a digital signal or recording produced by one system will be accepted by another. The alternative to this is that one manufacturer's interchange format predominates through commercial pre-eminence, which is only an acceptable situation if customers are not limited to that manufacturer's hardware alone. (There are examples where more than one manufacturer has adopted the same format, either through licensing or otherwise.) As digital audio technology becomes more diverse there is a greater need for some common ground, to provide a point of interchange between the diverse formats, and this is true in the case of each of the three main methods for transferring audio data from one place to another.

Digital programme interchange can be split broadly into three areas, and it is absolutely vital to ensure that they are treated separately. Firstly there are recording formats and physical media. Secondly there are real-time point-to-point digital interfaces (the digital equivalent of the analogue signal cable); and thirdly there are computer networks, which will increasingly be used to carry audio data as the use of computer systems and peripherals for audio purposes grows. Each has its merits and each is appropriate in certain operational situations. In the following paper a summary will be given of the role of each in professional audio and broadcasting systems, with discussions of the compatibility issues to be addressed in each case. Individual topics will be covered in greater depth in the papers to follow.

1 RECORDING FORMATS AND PHYSICAL MEDIA

1.1 Dedicated audio formats

Digital recording products have been commercially available for over ten years, and during that time a number of formats have been introduced, some of which have already been replaced. The most common and widely used have been the EIAJ format (PCM-F1, PCM-701, etc.), the Sony CD mastering format (PCM-1610/1630), the DASH format, the ProDigi format, and the DAT format. Sony is now about to release a replacement for the U-matic CD mastering format, based around a custom optical disk. DASH and Prodigi have embraced both multitrack and two-track fields, whilst the others have addressed only the two-track field. The important point about all these systems is that they use dedicated audio recording formats, in that they convert the analogue audio input into a digital audio signal which is uniquely formatted with error correction data, channel coded and stored on a specific physical medium whose characteristics are clearly defined. The point should also be made that none of these formats is physically compatible or format-compatible with the others. DASH and Prodigi use open reel tapes, EIAJ used consumer video tapes, the CD mastering format uses U-matic video tapes, and DAT uses special cassettes. The new Sony format will use special optical disks not based on the ISO standard.

1.2 Audio data files

The situation is quite different when digital audio is stored as a computer file using one of the recent workstation products. In such systems, analogue audio is converted into the digital domain in much the same way as with the dedicated formats, but is not formatted in anything like the same way. In such systems the audio signal is treated much like any other computer data (such as text) when it comes to storage or transfer, and is stored as a raw binary data file on a standard computer peripheral such as a disk drive. The file may also contain other data such as markers or timing data, but the process of recording does not involve the formatting and channel coding of the data so as to incorporate error correction information or sync preambles. This is because the computer data store handles these operations as part of its normal operation, and in any case operates in a rather different manner.

The Winchester disk drives on which audio data files are often stored are pre-formatted, and part of this formatting process involves mapping out the 'bad blocks' on the disk where errors occur, preventing data from being written to such areas. Furthermore, computer data storage is expected to be error free in normal operation, and thus peripheral storage devices are constructed and formatted in such a way that effective error rates are extremely low in comparison to those encountered with dedicated audio formats. Because of this, and because read errors may often be rectified by re-reading the erroneous block, the data to be stored does not require further protection or formatting.

An audio recording in the form of a computer data file is a very different entity to an audio recording made on one of the dedicated systems described above. Firstly the file is a uniquely addressable entity which will normally have a name associated with it. It will either be retrievable in its entirety or, if a block is corrupted in some way, will be totally irretrievable in normal operation. Secondly, the data blocks relating to that file may be spread about in a discontiguous fashion. Thirdly, the file will have a known size, precisely to the byte. Fourthly, there will be a directory file, containing details of all the files on the disk and their locations.

Operational processes such as 'dropping in' to a recording take on a new meaning, since a 'dropped in' section will probably be recorded as a new file rather than overwriting part of the old one. A 'sound file' is not storage-medium specific - in other words it does not really matter what physical medium the file is stored on; the file may be transferred from a Winchester disk to an optical disk to a tape streamer and still exist unchanged as the original file. The vital point is to separate discussion of the file itself from the medium on which it is stored - a task which is impossible with dedicated audio formats. This becomes important when considering transfer of audio over networks, and is vital to understanding the difference between audio transfer over real-time interfaces and file transfer over a computer network.

1.3 Issues of compatibility

Because computer files and the media on which they are stored need not be related there are two main issues of compatibility to be addressed, whereas with dedicated audio formats there is only one. If an engineer wishes to replay a DAT tape then he or she must find a DAT machine, and there is then a strong likelihood of success in getting the tape to replay. On the other hand, if an engineer has an optical disk with some sound files stored on it, he or she must first find a disk drive capable of accepting and reading the disk, but this does not guarantee that the system controlling the disk drive will be able to interpret the format of the stored sound files. There is clearly a second level of compatibility in question here. First there is the physical medium compatibility and second there is the file compatibility. One does not follow directly from the other.

A further issue which may arise is the question of the interface used to connect the peripheral storage device to the host computer. The SCSI interface is commonly used for this purpose, since the majority of personal computer disk drives are already equipped with it, this being a fairly fast parallel interface capable of transferring data between up to seven peripherals. If a workstation uses SCSI to interface its storage devices it is possible for the user to choose from a wide range of these devices, provided that they meet the performance criteria for real-time audio purposes specified by the workstation manufacturer. SCSI is not a real-time audio interface such as those discussed in the next section, but is a general purpose data interface. There is considerable misunderstanding amongst users in the audio industry over this issue. One SCSI interface may not simply be connected to another with the hope of interchanging audio data directly, as with the AES3 audio interface. The SCSI interface simply carries raw binary data, whose meaning is determined by the filing structure in use by the systems concerned.

To date, most manufacturers of digital workstations have produced systems which use sealed Winchester drives as storage media, and thus the issue of disk compatibility has not really raised its head. Now that optical disks and tape streamers are being used more widely there is greater pressure for some degree of compatibility between systems in terms of the file format used, and to some extent in terms of the physical medium. Most manufacturers' sound file formats are incompatible with each other, although a format devised by Apple Computer (known as AIFF) is emerging as something of an interchange standard amongst a few systems. Led by the American company Avid, a number of manufacturers are now collaborating under the banner of the Open Media Framework (described in a subsequent paper) to define common interchange formats for audio and related files, known as OMF Interchange or OMFI [1]. Such an interchange file format is also a key to the success of networked file exchange between systems.

2 REAL-TIME AUDIO INTERFACES

The real-time audio interface is to the dedicated audio recording format what the computer network is to the sound file. Real-time audio interfaces are the digital audio equivalent of signal cables, down which digital audio signals for one or more channels are carried in real time from one point to another, possibly with some auxiliary information attached. A real-time audio interface uses a data format dedicated specifically to audio purposes, unlike a computer data network which is not really concerned with what is carried in its data packets. A recording transferred over a digital interface to a second machine may be copied 'perfectly' or cloned, and this process takes place in real time, requiring the operator to put the receiving device into record mode such that it simply stores the incoming stream of audio data. The auxiliary information may or may not be recorded (usually most of it is not).

Real-time interfaces are normally uni-directional, point-to-point connections, and should be distinguished from buses and computer networks which are often bidirectional and which carry data in a packet format. It is not possible to address one of a selection of individual devices to receive data from a real-time interface, and there is no handshaking process involved in the data transfer. Sources may be connected to destinations using a routing matrix, very much as with analogue signals (see Figure 1). Audio data is transmitted in an unbroken stream, and erroneous data may not be retransmitted - there is no mechanism for requesting its retransmission. The data rate of a real-time audio interface is directly related to the sampling rate, wordlength and number of channels of the audio data to be transmitted, thus ensuring that the interface is always capable of serving the specified number of channels. If a channel is unused for some reason its capacity is not normally available for assigning

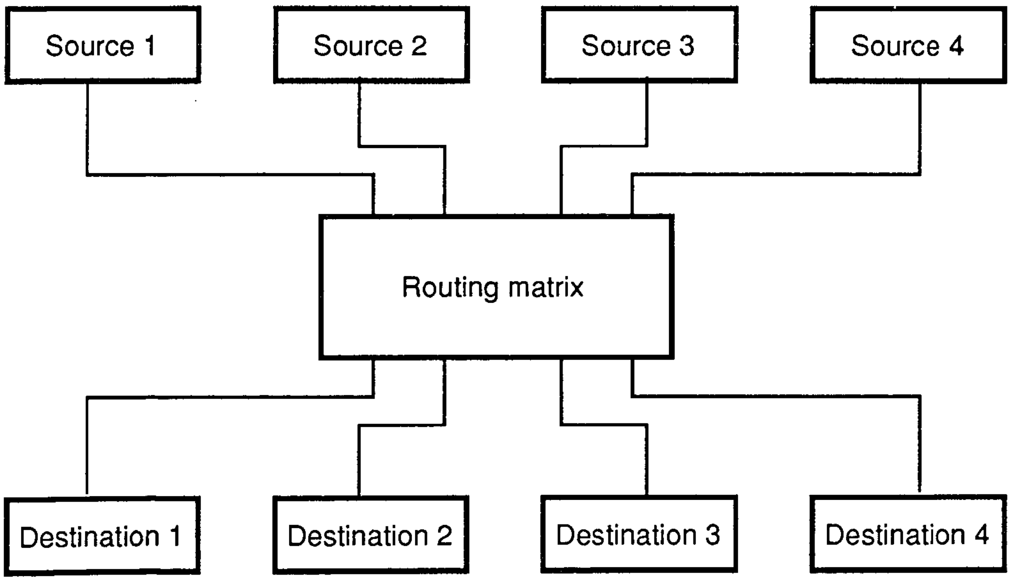

Figure 1 Routing of real-time audio interface signals using a matrix

to other purposes (such as higher speed transfer of another channel, for example).

Standards exist for both two-channel and multichannel interfaces in the form of AES3 [2], known as the AES/EBU interface, and AES10 [3], known as MADI - the Multichannel Audio Digital Interface - which carries 56 channels of audio data. There also exist a number of formats specific to certain manufacturers such as Sony, Yamaha and Mitsubishi which are incompatible with each other and with the international standards. For detailed coverage of these interfaces, their features and compatibility, the reader is referred to Rumsey and Watkinson [4].

The AES3 two-channel interface is self-clocking, and data is transferred synchronously with the sample rate unless buffering or sample rate conversion is employed. In large systems where most interconnections are digital, and which use such an interface, it is important for all devices to be locked to a common sample clock, and this is normally distributed in a similar manner to a video sync reference, using a central generator and a distribution matrix, perhaps with local slave generators in larger operations.

The AES10 multichannel interface (MADI) is designed for use with either coaxial cable or optical fibre, and, like the AES3 interface, is intended as a point-to-point interface for up to 56 channels. It would be used in such applications as linking a mixer to a tape recorder, or in carrying multiple channels of audio data between two studios situated fairly closely. (The coaxial version will only cover 50 m, whereas the proposed fibre interface would cover over 1 km.) Again programme transfer is always a real time process, and it is necessary for both receiver and transmitter to be locked to a common sample clock. MADI is the digital equivalent of the analogue multicore cable. Some broadcasting organisations have also experimented with using MADI for routing multichannel audio using optical fibre around a broadcast centre, multiplexing and demultiplexing channels at gateways.

Clearly, then, real-time interfaces such as those described above, are best suited to operational situations in which analogue signal cabling needs to be replaced by a digital equivalent, and where digital audio signals are to be routed from place to place within a studio environment so as to ensure dedicated 'on-air' signal feeds.

3 COMPUTER NETWORKS AND FILE TRANSFER

In contrast with the real-time audio interface, a computer network typically operates asynchronously, and data is transferred in the form of packets. One would expect a number of devices to be connected using a single network, and for an addressing structure to be used such that data might be transferred from a certain source to a certain destination (see Figure 2). Bus arbitration is used to determine the existence of network traffic, and to avoid conflicts in bus usage. The 'direction' of data flow is determined simply by the source and destination addresses. The speed of the network effectively determines the amount of traffic which may be present on the bus and the rate at which it will be transferred, this bearing little or no relationship to the audio sample rate, the number of channels or the wordlength. The format of data on the network is not audio specific, and the same network may carry text data, graphics data and spreadsheet data, for

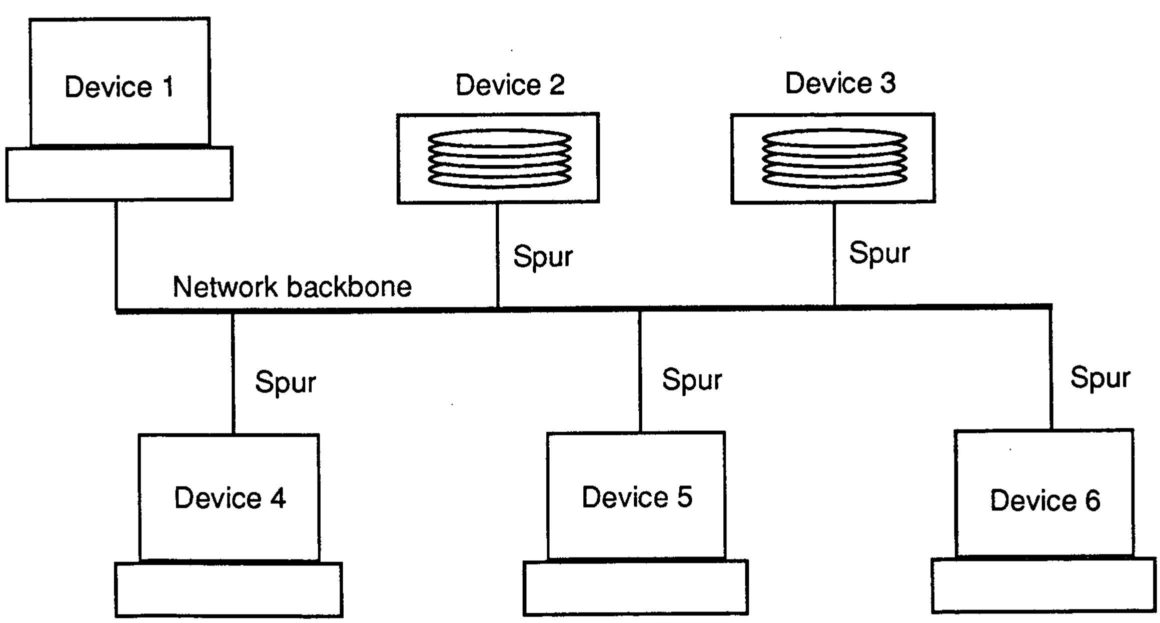

Figure 2 Typical serial network topology

example. This subject is covered in greater detail by Lawrence in a subsequent paper in this conference.

Since this paper's purpose is to examine issues of programme interchange and compatibility it is necessary to look at the operational situations in which such a network might be used instead of a real-time audio interface.

3.1 Applications in audio systems

The principal application of computer networks in audio systems is in the transfer of audio data files between workstations, or between subsections of a large system. When a file is transferred in this way, a byte-for-byte copy results at the destination, with the file name and any other header data intact. There are considerable advantages in being able to perform this operation at speeds in excess of real-time for operations in which real-time feeds of audio are not the aim. For example, in a news editing environment a user might wish to load up a news story file from a remote disk drive in order to incorporate it into a report, this being needed as fast as the system is capable of transferring it. Alternatively, the editor might need access to remotely stored files, such as sound files on another person's system, in order to work on them separately. In audio post-production for films or video there might be a central store of sound effects, accessible by everyone on the network, or it might be desired to pass on a completed portion of a project to the next stage in the post-production process (see the paper by Pope, later in this conference).

3.2 Network speed related to application

Simple desktop computer networks such as Apple's LocalTalk are very slow by audio standards and are not capable of transferring even a single channel sound file at real-time speeds, whereas the faster Ethernet is fast enough to carry audio data files slightly faster than real time, depending on network loading. Optical fibre networks such as FDDI are capable of very high transfer rates, and are the key to handling large numbers of sound file transfers simultaneously at high speed. Unlike a real-time audio interface, the speed of transfer of a sound file over a network depends on how much traffic is currently using it. If there is a lot of traffic then the file may be transferred more slowly than if the network is quiet (very much like motor traffic on roads). The file might be transferred erratically as traffic volume varies, with the file arriving at its destination in 'spurts'.

It would seem, therefore, that the computer network is not to be relied upon for transferring audio where an unbroken audio output is to be constructed from the data concerned. It would seem that the faster the network the more likely one would be able to transfer a file fast enough to reliably feed an unbroken audio output, but this should not be taken for granted. Even the highest speed networks can be filled up with traffic! The point is that if such a network is to be used for reliable real-time data transfer it must be possible to implement such an arbitration structure that once the data capacity for a certain number of real-time channels has been reserved between a source and destination it can be preserved for the duration of the transfer. This may seem unnecessarily careful until one considers the application of such an approach in broadcasting, as illustrated in Figure 3 [5].

Here incoming news stories are recorded as files on a central file server, they are edited at local workstations and released for transmission. If the edited files are to be transmitted directly from the central server, via a dedicated workstation, then it must be possible to ensure unbroken

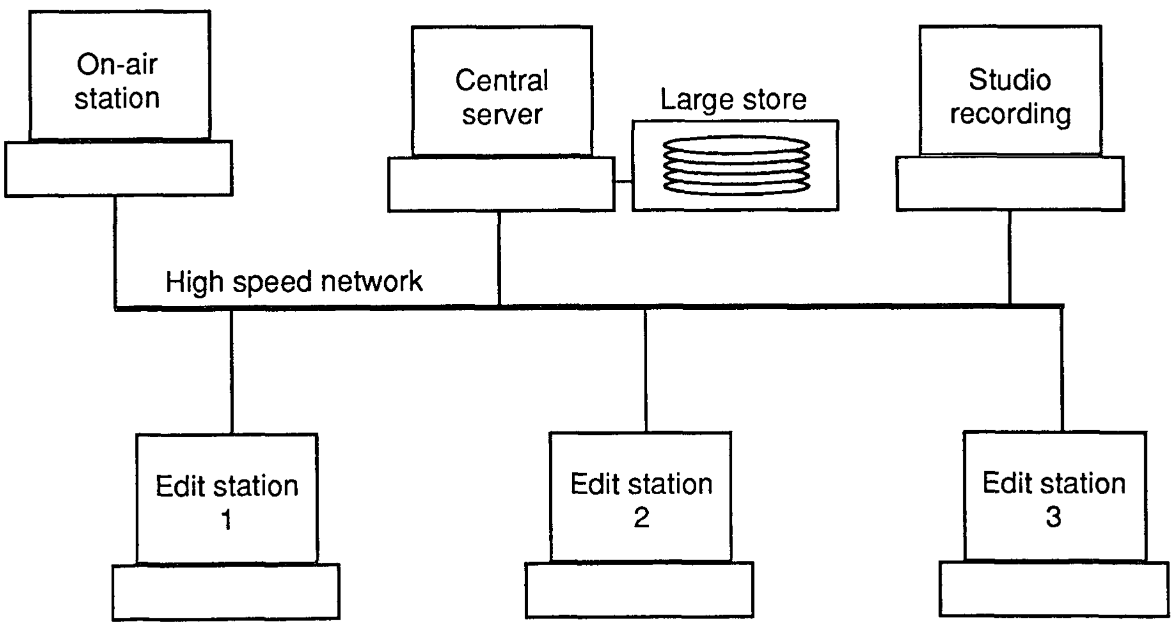

Figure 3 Conceptual diagram of a networked newsroom system

access to those files at an average rate adequate for real time transfer. The alternative is to transfer the edited story to a separate system dedicated to real-time play-out, but this would take time and somewhat negate the speed advantage of a networked news editing system. It is possible to arrange high speed computer networks to allow for this possibility, and upcoming network standards such as FDDI-2 will make it more straightforward by incorporating synchronous packets.

There is a difference between a networked system, where processing and storage are distributed, and a system based on the mainframe concept where one central high-speed computer is accessed by multiple users, as illustrated in Figure 4. Here the interfaces between local terminals and the mainframe are for control data traffic only, and would be unlikely to carry audio data. Such an approach is used in the BASYS DCART system for newsroom automation. In such a system the system software allocates storage and processing time to the various tasks in hand, including real-time play-out.

3.3 Compatibility Issues

The main compatibility issues when interconnecting audio workstations using a computer network are layered in nature. There is the physical network itself, either electrical or optical; then there is the network file transfer protocol; and there is the file format itself. There are parallels here with the compatibility of sound files stored on physical media, except that here the transfer medium is not a physical store but a network. The physical network standard (e.g. Ethernet) can be compared with the physical storage medium (e.g. 5.25" ISO optical disk); the network protocol (e.g. Ethertalk) can be loosely compared with the interface used to connect the physical peripheral to the host (e.g. SCSI); and the file format is still the file format!

Again, no matter what the physical medium and protocol used for transferring the file, the file format is the thing which determines whether the destination device will be able to make sense of the sound file which has been transferred. One could interconnect two workstations using Ethernet and transfer a file between them, but there is still no guarantee that the file could be interpreted by the receiver. We come back to the need for some agreement over file formats, and exactly the same criteria apply as discussed above with physical media interchange.

4 SUMMARY

In the preceding paper a number of different methods for interchanging audio data between systems have been outlined, and an attempt has been made to compare and contrast them, highlighting their important differences and the compatibility issues involved.

REFERENCES

[1] Avid (1993) *OMFI: Open Media Framework Interchange, Version 1.0.* April 1993. Avid Technology, Inc., Tewkesbury, MA., USA

[2] AES (1992) *AES3-1992. Serial transmission format for two-channel linearly represented digital audio data.*

[3] AES (1991) *AES10-1991. Serial multichannel audio interface (MADI).*

[4] Rumsey, F. and Watkinson J. (1993) *The Digital Interface Handbook.* Focal Press, Oxford

[5] Rumsey, F. (1992) Approaches to editing in a mixed disk and tape based environment. Presented at the *92nd AES Convention, Vienna, Austria, March 24-27.* Preprint 3277

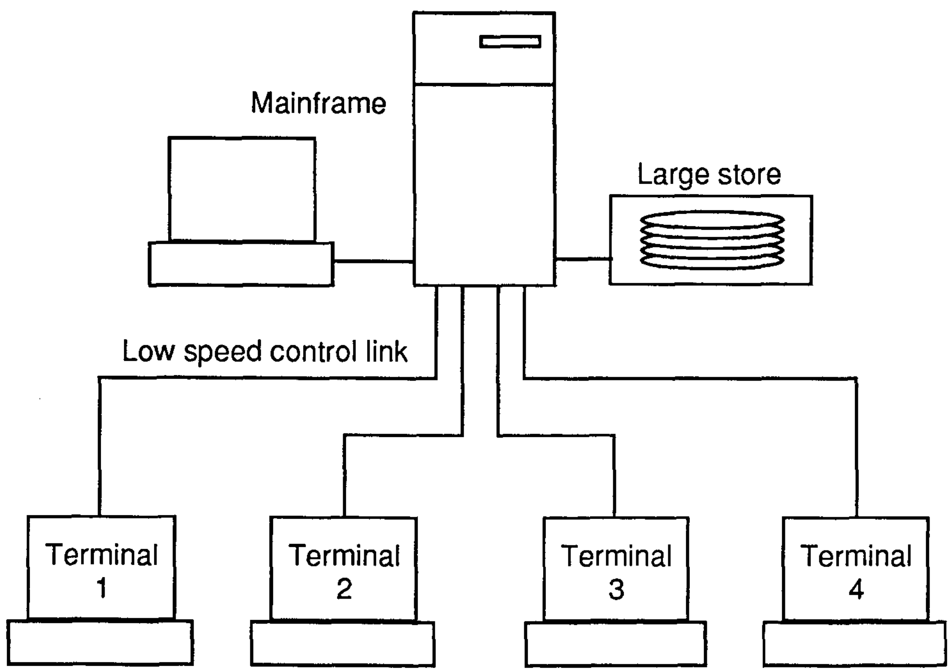

Figure 4 Central mainframe processor and terminals