INTERNATIONAL ATOMIC ENERGY AGENCY
UNITED NATIONS EDUCATIONAL, SCIENTIFIC AND CULTURAL ORGANIZATION

INTERNATIONAL CENTRE FOR THEORETICAL PHYSICS
34100 TRIESTE (ITALY) - P.O.B. 586 - MIRAMARE - STRADA COSTIERA 11 - TELEPHONES: 224281/2/3/4/5/6
CABLE: CENTRATOM - TELEX 460392-1

SMR/90 - 18

COLLEGE ON MICROPROCESSORS:

TECHNOLOGY AND APPLICATIONS IN PHYSICS

7 September - 2 October 1981

AN INTRODUCTION TO DISTRIBUTED COMPUTING - II

R.W. DOBINSON

EP Division
CERN
1211 Geneva 23
Switzerland

4. SHARED MEMORY SYSTEMS

I want now to discuss two types of shared memory system: shared disc and multiprocessors.

Shared disc

An example of such an arrangement is shown in Fig. 20. Two NOVA minicomputers can have access to the same discs. There are two practical problems to be solved. First, since the discs can be accessed from either computer the controller hardware must prevent simultaneous use of a drive, i.e. you cannot have a situation where both machines are trying to move the disc head at the same time. Secondly, in very many applications the machines are both wanting to use the same files; there is a "shared data base". This requires some precautions to avoid the shared files being corrupted. Take, for example, a situation where the shared files contain reservation information for airline flights. In response to a request to book two seats on a particular flight computer A reads back a record containing data about seat availability. It sees there are just two seats left and allocates them; the flight is full. Meanwhile, computer B also receives a request to book two seats on this flight. It reads the flight information and also finds two free seats which it allocates. Computer A has not yet up-dated the shared file with the information that it has allocated these same two seats! Result, the same seats are reserved twice, chaos, the passengers are extremely angry. To prevent this situation an interlock facility is required so that only one computer at a time is allowed to have access rights to change a shared file. This is done in the case of the two NOVAs by the IPB (inter processor bus), connecting the two computers, over which agreements are made as to who has control of a shared file at any time.

Why are shared disc systems used?

- for reliability: if one computer breaks down there is still one (or even perhaps more) to carry on
- for availability: if maintenance is required on one machine the other(s) can work normally
- to spread the computing load over several processors, to increase flexibility, through-put and response time

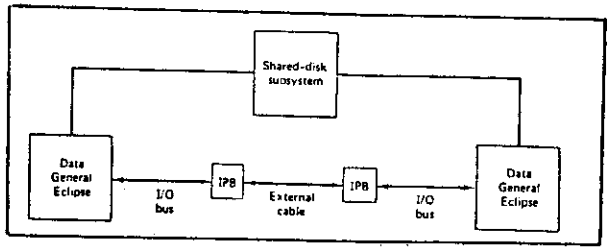


Fig 20
Shared disc

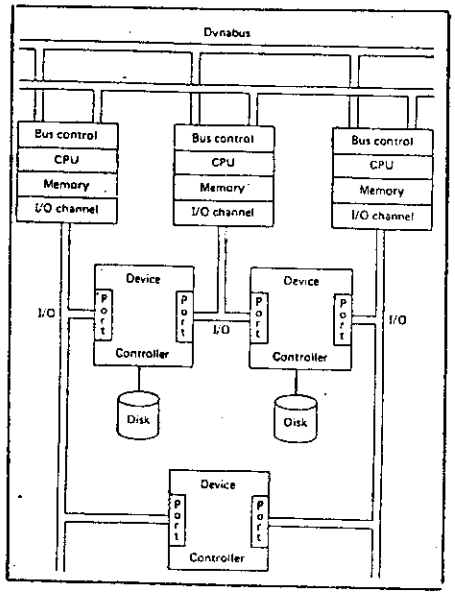


Fig 21
The TANDEM highly reliable computer system

Figure 21 shows a shared disc system made by Tandem Computers Inc. Such an arrangement offers very good reliability and availability by having three processors driving multiple device controllers.

What are the typical areas of application for shared discs? In banking, in stock control, in airline reservation schemes and in computer systems for emergency services. Note shared disc systems share secondary memory (e.g., magnetic disc), not primary memory (e.g., core or MOS) which contains programs and data accessible directly by the CPU. Shared disc systems are not, therefore, true multiprocessors.

Multiprocessors

Now let us turn to some examples of multiprocessor systems: systems in which processors share primary memory. The motivation for building multi-processors is basically similar to that for shared discs:

- through-put
- flexibility
- availability
- reliability

Figure 22 shows a multiprocessor system based on two microprocessors. µP1 handles input and output for four serial I/O channels. Typically it services interrupts on a character by character basis; when a record has been accumulated it is placed in a common memory where it can be read by µP2. This common memory, which is accessible from either µP, we call a dual-port memory. µP1 therefore acts as a front-end or I/O processor for µP2 which, being unburdened by having to service large numbers of interrupts, is free for processing the data.

Figure 23 shows an application using two PDP11 minicomputers to perform data acquisition for a nuclear physics experiment. Computer A receives an interrupt telling it a measurement has been taken and that it must read the contents of the analogue-to-digital converters attached to its bus. It reads these values and stores them in its memory. After it has accumulated, say, 500 measurements it interrupts Computer B which reads the whole block of data from A's memory and processes it. This all looks very similar to the previous application but, in fact, the shared memory is accessed in a different way; our present example uses a bus window to link the two UNIBUS. The bus window

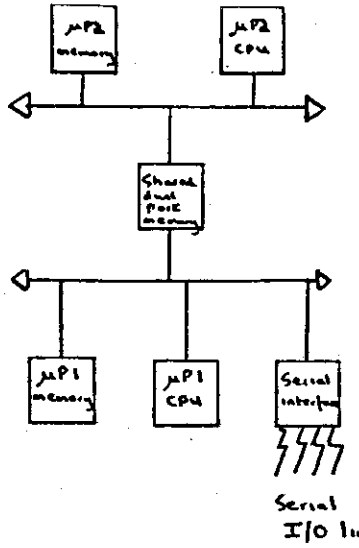


Fig 22
A dual port memory system

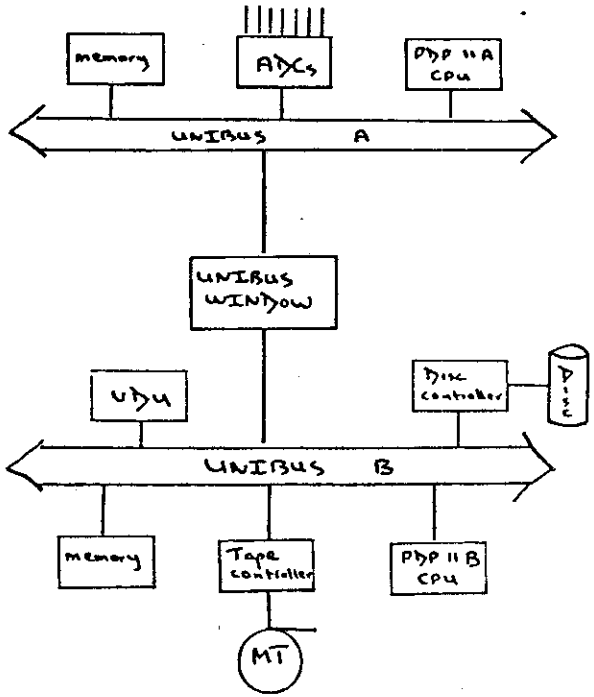


Fig 23 Two PDP 11 communicating via UNIBUS window

allows each machine to generate addresses on the other's UNIBUS as if it were its own. Each processor can therefore access the other's memory as if it were its own. A computer designates part of its address space to the other's bus. The bus window recognizes this block of addresses on its near side and will, after obtaining control of the far side bus, generate the appropriate bus signals on it. A shared memory window operation therefore involves both the UNIBUS on which the request was made, the originator bus, and the bus on which the memory access was performed, the so-called target bus. Notice I have included this example under the heading of shared memory systems, even though Anderson and Jensen's taxonomy puts it in a category of its own. Actually, I consider it a hybrid system: both shared memory and bus window!

So far I have described two multiprocessor applications where a dedicated processor performs input and output operations in order to leave a second processor more free for other work. Let me now turn to the question of whether one can build high performance computing systems using multiple microprocessors. Why is it attractive to consider doing this? Because the price-performance ratio of microprocessors is very attractive and over the past few years has improved more rapidly than for large scale computers.

Figure 24 shows a simplified layout of the PULSAR multiprocessor system. Sixteen DEC LSI-11s are attached to a common high performance bus. They have access to banks of common memory on another separate bus. Between the two buses is a cache memory to cut down the number of times main memory needs to be accessed. The idea of this project was to be able to run it like a single DEC minicomputer. Processors could be added or removed but programs would still be able to run without the user being aware of the number of processors being used or how they were interconnected.

The PULSAR is only one of several projects to build general purpose computing facilities from multiple micro or minicomputers. Other examples are the Cm* and Cmmp systems designed and built at Carnegie-Mellon University (see References). Despite these efforts I feel we are still some way off being able to use this type of multiprocessor in general purpose applications. The major problems currently facing such systems are, however, in my view not so much hardware as software. For example, how should tasks now executed on uniprocessors be decomposed to run on a set of smaller processors? Can compilers or specialized run-time systems be developed to do this decomposition or must the programmer do this decomposition explicitly?

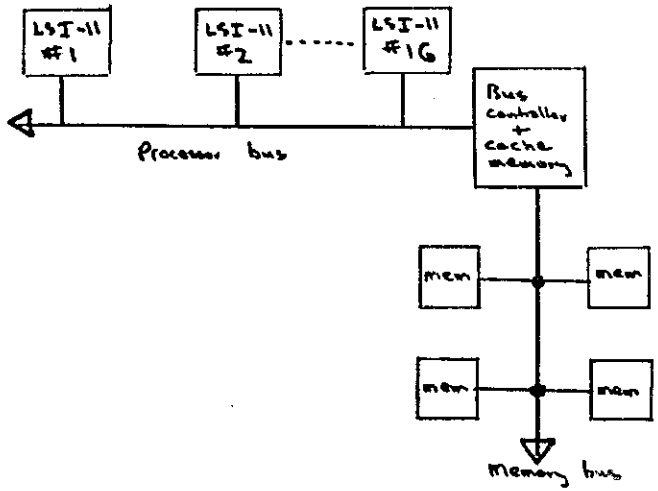


Fig 24 Simplified diagram of DEC PULSAR multiprocessor

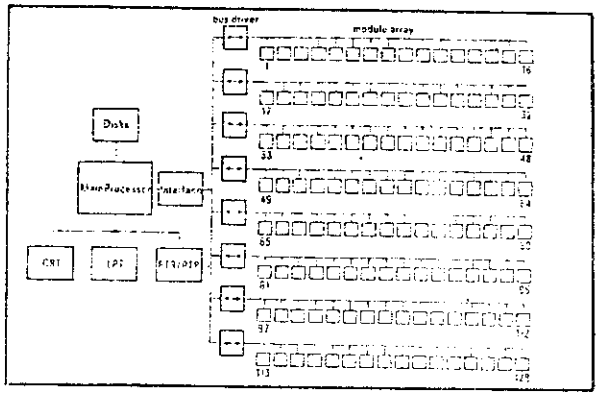


Fig 25 a multiprocessor system for weather forecasting

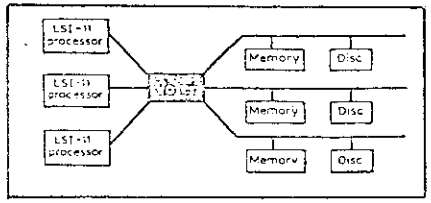


Fig 26 the C.vmp

Despite the difficulties of producing a general purpose facility, let me give you one nice example of another experimental multiprocessor system which has been used to solve a specific problem. This one has been applied to weather forecasting computations. Figure 25 shows the set-up used. It consists of a large number of identical modules, each consisting of a processor with its own program and data memory coupled via a communication memory to a bus system and controlled by a main processor. The main processor, the supervisor, starts off identical programs with different input data in all microprocessors. After all processors have completed their computation, the supervisor collects their results and arranges any interprocessor exchanges of data that are required via the communication memories. The process then continues iteratively until the final results are arrived at.

Before ending my discussion of multiprocessor systems I would like to give you an example of a highly fault-tolerant system, the C.vmp, see Fig. 26. The processors, memory and discs are all triplicated. The signals from each member of the triplet are compared and if they differ a majority vote is used to determine the correct signal. This type of arrangement could be used in situations where failures would put life at risk, for example, in aircraft control systems.

5. LOCAL AREA NETWORKS

Local area networks (LANs) are one of the most exciting and fast moving fields in computing today, in particular because they are finding applications in many facets of day to day life, for example in office automation.

What are LANs? How do they differ from the multiprocessor systems we have discussed previously?

- Geographically speaking, multiprocessor systems span distances from a few cm to ∿ 100 m. LANs connect processors and peripherals separated by a few metres to a few kms. See Fig. 27.

- Philosophically speaking, multiprocessor systems rarely, it ever, perform useful work if the interprocessor connections are removed. Networks such as LANS connect autonomous nodes, each node is capable of operating on its own, however its performance and capabilities are enhanced by the presence of the network.

- Multiprocessor interconnections are in most cases parallel, separate lines are used for address and data (sometimes the two are multiplexed), and for control and timing. A message in the Anderson and Jensen sense, is very often a single word. Messages in networks are usually sent serially as blocks of words (packets of information).

- Multiprocessors normally involve processors of the same family; networks typically connect different types of processor. The interconnections therefore need to be processor independent, and the exchange of information standardised.

- The longer distances covered by LANs mean it is more difficult to ensure that messages are not corrupted during their transmission (for example by electrical noise). A problem which risks to become more troublesome for long haul networks. The detection of errors, and the necessity to agree on the format of messages exchanged, leads to the establishment of rules, or protocols, between source and destination for the transmission of data.

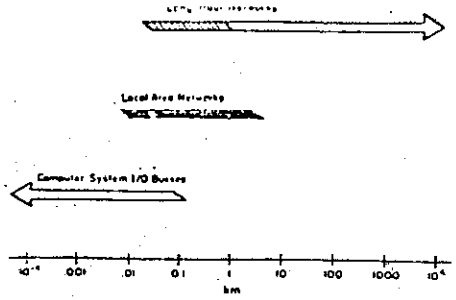
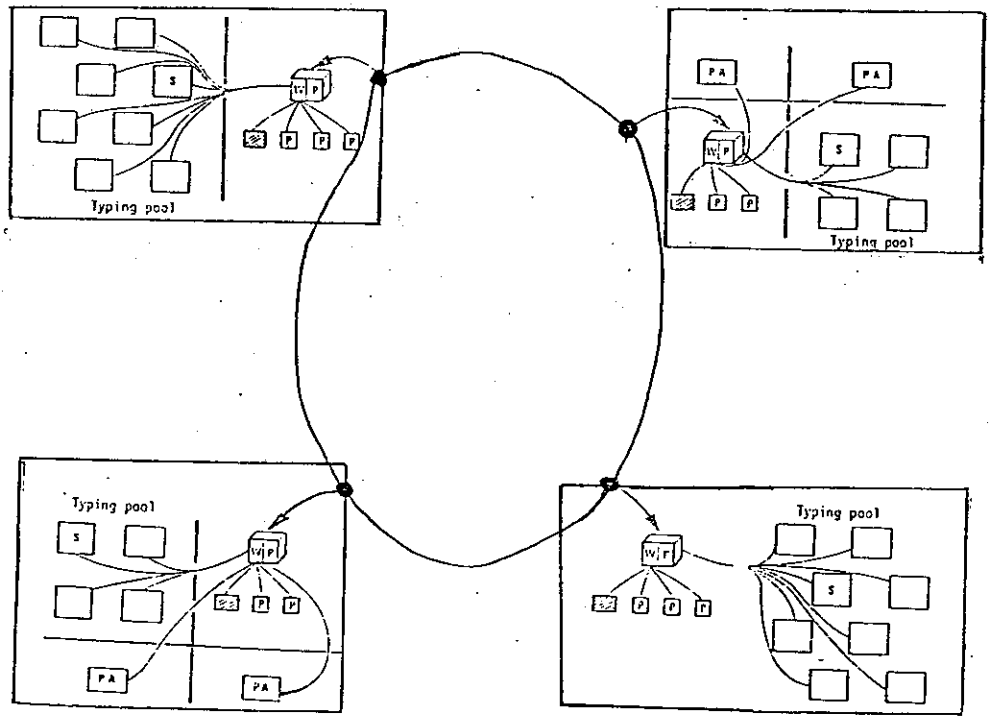


Fig 27 Geographical range of local area networks



Key:
- Printer control station
- WP Word processor
- S Supervisor
- PA Personal Assistant
- P Printers

Fig 28 LAN for office automation.

What are LANs used for ? Why have they become important ?

- Microprocessors are coming into use in more and more areas of application. Micros themselves are relatively cheap, however, they may need shared peripheral access, they may need to communicate with other micros, and with minis and mainframe computers. A cheap connection scheme is required if one is to avoid the cost of linking from dominating all else.

- A typical building, or cluster of buildings, may contain a large amount of computing equipment which needs to be linked together. In an office block for example there may be very many terminals and processors used to prepare letters, manuscripts and reports. Access to centrally stored records and documents is required, messages need to be distributed from department to department (this is called electronic mail), expensive high quality printers need to be shared. One of the dominant driving forces in LANs has been its usefulness, and the enormous potential market, in office automation (See Fig. 28).

- One other area of use for local area network is in connecting up peripherals and processors at large computer centres. The performance of this type of LAN is normally much higher than that required for office automation applications, 50 M bits/s compared to 5-10 M bits/s, consequently the price tag is much higher.

Which topographies are used in local area networks? Since LANs are intended to be cheap the simpler topographies are used

- star
- bus
- ring or loop

A star network eliminates the need for every network node to make a routing decision. All routing is centralised in one central switch. This leads to a simple structure in the other nodes. Such a topography is ideal for many secondary nodes communicating with one primary node e.g., a typical timesharing systems. If, however, communications are not mostly between a primary node and several secondary nodes, then reliability appears as a possible disadvantage of the star topography. An additional disadvantage of a

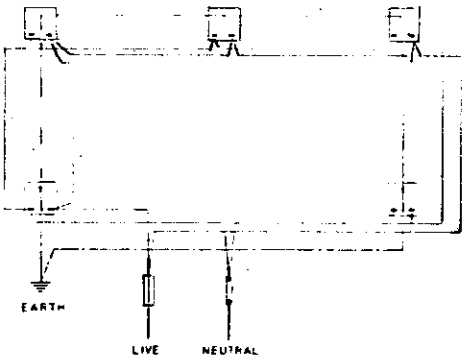


Fig 29 Domestic ring main system

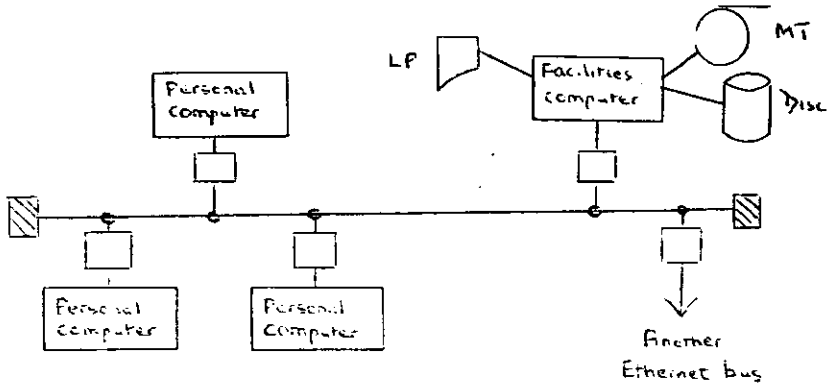


Fig 30 An Ethernet Local Area Network

star network is that every new processor or peripheral attached needs a new connection, whereas what is desirable is to be able to attach up new equipment by plugging into, or tapping off from, an existing communications medium. Take the analogy with the electricity supply in a house; when you want power you don't run a wire out from a centre place, you plug into a mains ring circuit system. See Fig 29. Bus and ring systems both allow a good approximation to this analogy, we shall concentrate our attention on them in the discussion which follows.

Ring and bus topographies attempt to eliminate the disadvantages of the central node of the star system without sacrificing too much inherent simplicity. Two practical examples will be given to illustrate these types of LAN:

- Ethernet, a bus system
- the ring topography used by Apollo Computer Inc in their DOMAIN system

Both these types of LAN introduce a problem of deciding which node may transmit at a given time.

Ethernet

A typical Ethernet configuration is shown in Fig. 30. A large number of nodes (or stations) are connected to a co-axial cable, the serial bus, of length up to 1 km. Each station monitors the bus passively. A station that wants to transmit first senses whether anyone else is putting out information on the bus (a carrier is present). If another transmission is in progress then it defers its transmission, if not a packet of information is broadcast serially. The packet contains the source and destination addresses as well as data. This packet is heard by all stations and is copied and stored by the node whose address corresponds to that specified as destination by the source. In some cases two or more stations, having simultaneously sensed there is no carrier present on the cable, may begin transmission at the same time, a collision occurs and information will be garbled and lost. This is taken care of by the transmitter listening whilst it is sending. If there is a difference between the transmitted data being put on the bus and the received data then a collision is detected and transmission is abandoned. Retransmission takes place after a random interval so colliding stations will not normally collide again. This method of organising multiple access to a shared bus is called carrier sensed multiple access with collision detector (CSMA-CD)

Ethernet has been used by the Xerox Corporation internally since the mid seventies. A consortium of Xerox, INTEL and Digital Equipment has been formed to produce and promote the system. There is a move to adopt Ethernet as an international standard in fact. What is even more significant is the plan to manufacture Ethernet integrated circuit chips, which it is claimed will be very inexpensive. It would make no sense in the future to spend $2000 connecting a $1000 terminal to a local area network. Xerox has recently announced an office equipment product, the Star system, which is based on Ethernet. The connection of equipment to the local area network is advertised to be a single wall plug labelled information outlet. So in the not too distance future you could have in your house a tap for water, a plug for electricity and another plug for computer services!

The Apollo DOMAIN System

The Apollo DOMAIN system (Distributed Operating Multi-Access Interactive Network) is shown in Fig 31. I shall use it as an example of a local area network with a ring topography, and also to illustrate a fashionable trend, that of personal computers.

Most people using computer systems share facilities with other users. They use timesharing systems to prepare programs, and then compile and run them by adding them to queues of jobs, from different people, that the computer must execute .A modern trend, encouraged by the cheapness and power of microdevices, is to provide single user computational nodes, typically:

- a powerful microprocessor
- a large memory (minimum 1/4 Mbyte, up to 1 Mbyte)
- a keyboard
- a very high quality display screen that can be used in an extremely flexible and sophisticated way.

Different computational nodes are linked together via a high performance local area network, in the case of Apollo a ring. Each user has the advantage of very good response for interactive work, for example computer aided design, engineering studies, text processing, as there is only one user per computer. The LAN allows sharing of peripherals, discs and printers at peripheral nodes, between many users, furthermore each user can access his own data or his neighbours with equal ease, and different nodes can talk back and forth to each other. DOMAIN is just one example of personal computers linked together by a LAN, another is the Three Rivers PERQ system, which incidently uses Ethernet. However now lets go on to see how a ring LAN works.

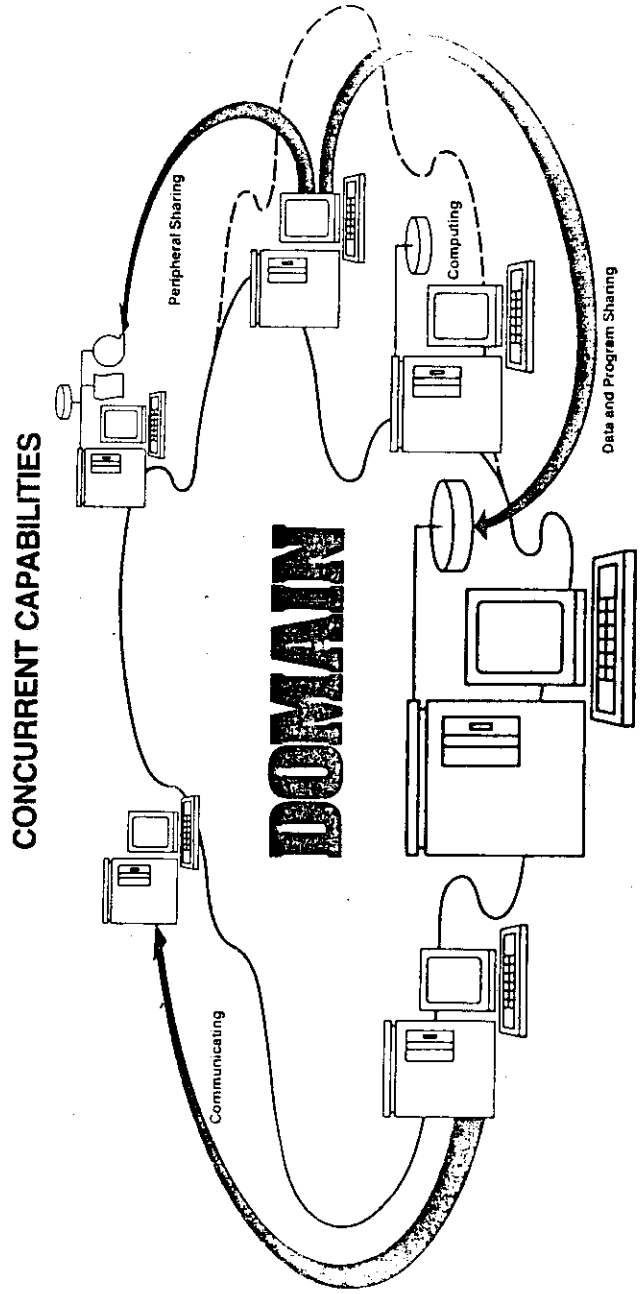


Fig 31 The Apollo Domain System

What is the mechanism for deciding which node may transmit ? The general idea is that permission to use the loop is passed sequential from node to node. A "control token" is passed around the DOMAIN loop. Any node upon receiving this token may remove it from the ring, send a message, and when this is finished pass on the token. Packets of information transmitted over the network include a source and destination address, the latter is recognised by the relevant receiver node which stores the packet. The receiver acknowledges it has seen the information sent to it by setting a special bit in the packet which continues on back to the transmitter. The packet is removed by the transmitter which also checks the acknowledgement bit. Variations of this method of operating a loop LAN are possible (see references).

A comparison between Ethernet (a bus) and DOMAIN (a ring).

A bus topography such as Ethernet is characterised by the fact that the bus is a purely passive medium each node when not transmitting is just listening. Failures of a node will therefore not tend to corrupt the bus. Ring topographies like DOMAIN require that each node be able to remove a message or pass it on. This requires an active repeater at each node which must be made very reliable, as its failure will disrupt the whole network. Typical techniques to ensure reliability include powering the repeaters from the ring, and providing a relay in every repeater that can mechanically provide a bypass it in the event of failure. Ring LANs do have, however, one advantage; an acknowlegement can be given by the destination for every packet transmitted in a convenient and natural way. Ethernet sources have no way of knowing immediately that a packet was taken by the destination. The destination must send back a separate acknowlegement message.