WIDE AREA TECHNIQUES

IN A LOCAL AREA NETWORK

A thesis presented by

MICHAEL MURPHY

in partial fulfilment of the requirements

for the degree of

Master of Science in Computer Science

at

Massey University

February 1986

ACKNOWLEDGEMENTS

The production of this thesis has been made possible by the co-operation and efforts of many people.

My sincere thanks go to my two supervisors, Mr Tony McGregor and Mr Paul Lyons, for their guidance during the development of MASSEYNET and for their helpful criticisms in the presentation of this thesis.

I would like to extend my special thanks to Mr John Anson for his encouragement and assistance, particularly during the implementation and testing stages of this research.

My thanks also go to the many members of the Computer Science Department who have willingly assisted in the testing of MASSEYNET and production of this thesis.

Finally, a very special thank you to my family for their support, patience and love during my two years in New Zealand. For this, I will always be grateful.

Michael Murphy

CONTENTS

LIST OF FIGURES

ABSTRACT

Packet switching networks have been developed on a large scale (national or even international) as a means for enabling inter-computer communication. They spread the responsibility for data handling amongst a number of "nodes".

In contrast, local area networks are restricted to a single site or organisation, and depend on the reliable operation of a central facility (sometimes a computer, sometimes a shared communication cable).

This thesis describes the design and successful implementation of MASSEYNET, a local area network embodying wide area techniques. The aim of the network is to provide a cheap system which will permit communication between devices located throughout Massey University, and permit individual terminals to interface to any one of the attached devices.

CHAPTER 1

INTRODUCTION

1.1 Local Area Networks

In the early years of computing, the long term vision was of a single large computer serving all of an organisation's computational needs. This model is rapidly becoming obsolete, making way for systems of smaller, interconnected computers. The development of such systems is motivated by the existence of a large number of mini and micro-computers, and their need to communicate with each other and share central resources. These systems are called local area networks because they enable communication between devices within a small geographical area (a diameter of not more than a few kilometres [Tane.1981]). The short distances involved have made it economically feasible for such networks to use a central, high-speed cable to carry all the data within that site. Devices attach to the cable via interface message processors (IMPs) and communicate by following a common set of procedures called the communication protocol. Local area networks are usually classified by their shape or topology because it governs how each device attaches to the network and how information is passed amongst them. Commonly used topologies include the star, loop, ring and bus configurations.

1.1.1 Star Networks

Star networks involve a direct connection between each network device and a centralised computer. The computer at the centre of the star is generally used as a processing resource but may also be used as an IMP to switch messages from one network line to another.

A common example is a minicomputer installation where terminals, printers, disk drives, and other peripherals are directly connected to the main computer (as shown in figure 1.1). The information exchange (communication protocol) is generally interrupt-driven but, in some cases the central processor may use round-robin polling to determine if each device has any information to send.

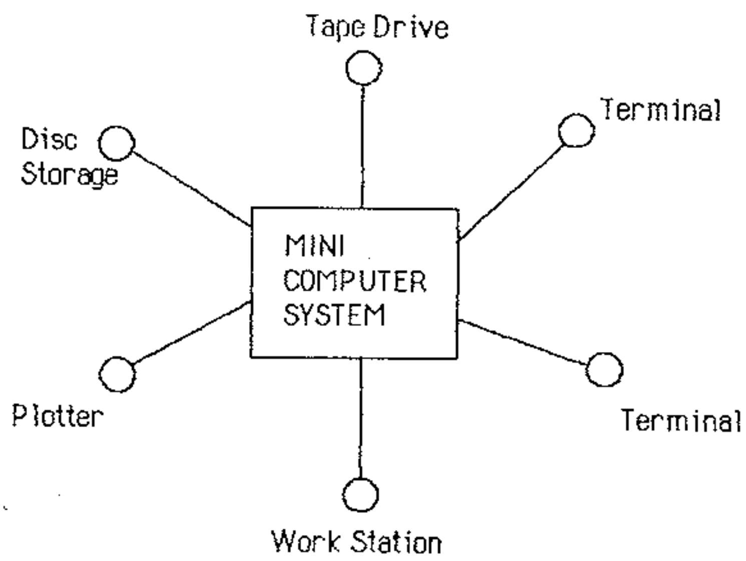

Figure 1.1 Star Network - Minicomputer Installation

The shared resource in star networks is the central computer not the transmission medium as with typical local area networks.

1.1.2 Loop Networks

A loop network consists of a controlling device positioned at some point in a loop of cable. Network devices are attached to the cable at various points and share it for message transmission as shown in figure 1.2 [Gee.1982]. There are two communication protocols commonly used in loop networks.

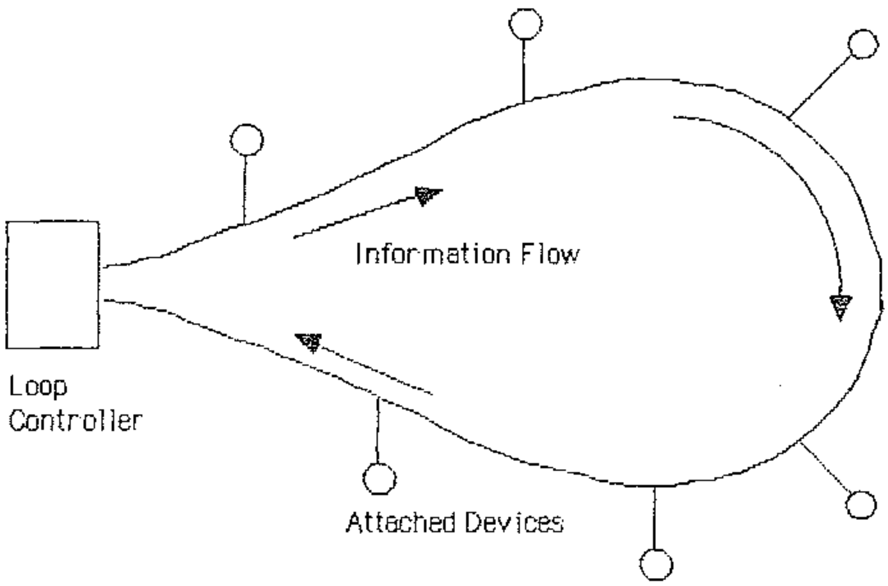

Figure 1.2 Loop Network

The first protocol requires the controlling device to poll each of the other devices by means of a specially addressed packet. A packet is a block of data which has a fixed structure and a fixed maximum size. The packet contains addressing information which specifies its destination, control information which includes the packet type, and a data field for transporting information between network users. If the polled device has data to send, it fills that packet with data and places the name of the required destination into the packet's address field. The controlling device is then responsible for ensuring the safe delivery of this data packet to the destination device. Normally, each of the network devices is polled in a strict, cyclic sequence (round-robin polling) so that each device gets equal opportunity to use the line.

The round-robin polling technique can be inefficient because channel capacity is wasted polling devices which have no data to send. The second protocol eliminates the inefficiencies of polling idle devices by having a single empty packet circulating around the loop available to all devices. Devices communicate as above by filling this empty packet with data and placing the required destination name in the packet's address field. The packet is passed around the loop to the destination device where it is emptied and placed back in the loop ready for re-use. With this latter technique, the controlling node must ensure that a single device does not "hog" the network resources.

1.1.3 Ring Networks

Ring networks consist of a series of point-to-point cables linking adjacent nodes (IMPs) in a circular arrangement (figure 1.3) [Gee.1982]. Point-to-point cables provide a direct link between the devices attached at either end but do not allow devices to tap onto the cable at any point in between. Each node in the ring has equal status and there is no controlling device as in loop networks (although a special node is sometimes used to monitor the ring traffic and recover from erroneous transmissions). The communication protocol is responsible for ensuring that each node gets its fair share of the available network capacity. Three common communication protocols, described below, are empty slot, token rings, and register insertion.

Empty Slot

The empty slot technique involves one or more packets circulating around the ring. These packets may be either "available" or "in use" as indicated by a flag in the packet header. When a node has data to transmit, it waits for an "available" packet to arrive, fills that packet with data, and inserts the address of the destination node into the

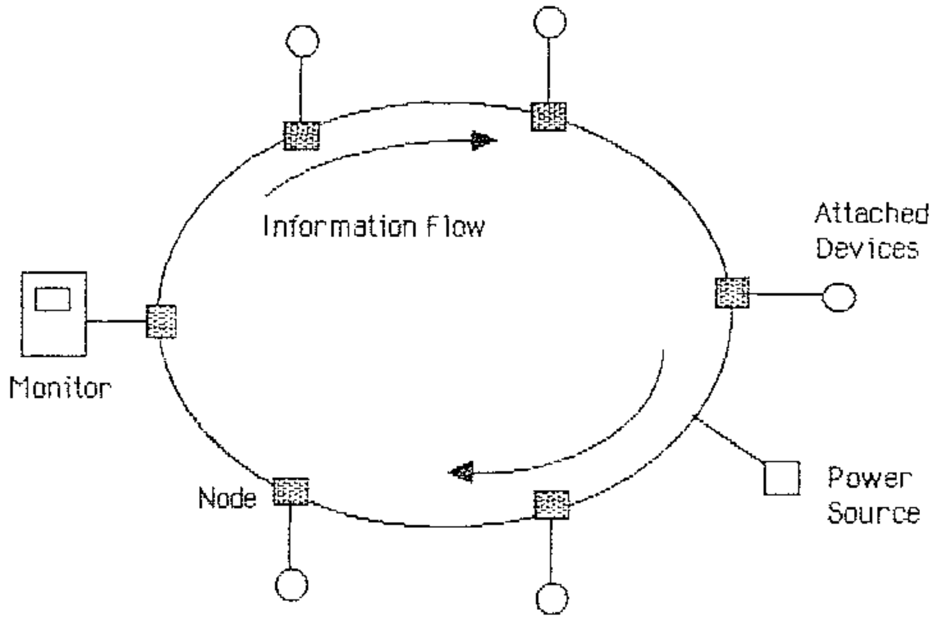

Figure 1.3 Ring Network

packet's address field. The packet is marked as being "in use" and passes around the ring from one node to the next until it reaches the destination. This node reads the data and marks the packet as being "received". The packet then circulates back to the originating node where it is marked as being "available" [Blea.1982]. Usually, the same node is not permitted to reuse the packet immediately, since this would enable a single node to "hog" the network resources. This communication sequence is illustrated in figure 1.4.

A monitor is often used in empty slot rings to spot and correct defective packets. This monitor is different from the controlling device used in loop networks because it does not control the communication protocol. Its primary purpose is to detect and correct corrupted packets but it may also be used to collect statistics on ring traffic.

The empty slot ring is often refered to as the Cambridge Ring because the technique was first developed at the Cambridge University Computer Laboratory [Blea.1982].

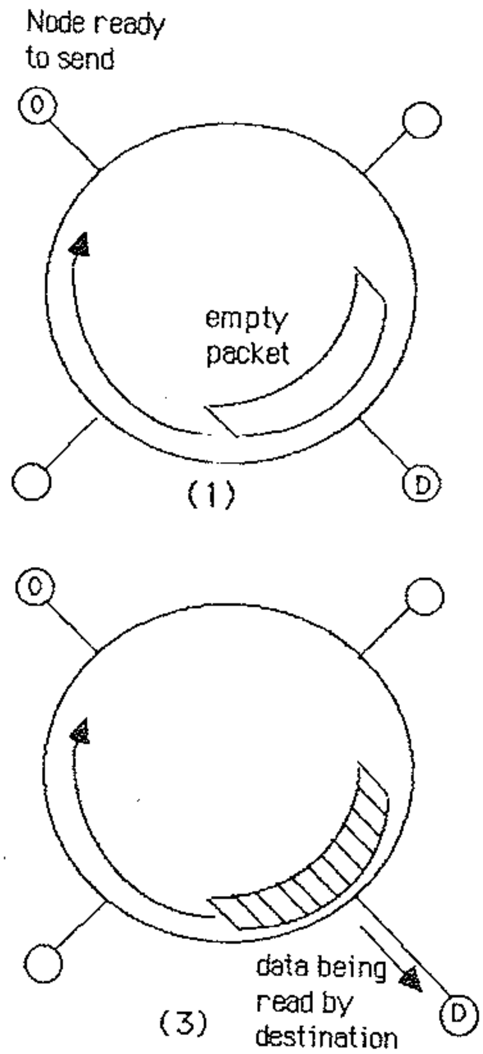

Figure 1.4 Operation Of An Empty Slot Ring

Token Passing

The token passing communication protocol is similar to the empty slot technique because a special bit pattern called a token circulates around the ring from one node to the next. Any node wishing to transmit must wait until the token is passed to it by the previous node in the ring. That node then removes the token temporarily and begins transmitting its own data. Having completed transmission or after using its maximum time allocation, the node places the token back into the ring and transmits it. The data received by the next node thus consists of one or more packets followed by the token. If that node also wishes to transmit, it must first of all relay the circulating packets to the next node, and then transmit as described for the previous node. The resulting situation after two nodes have each transmitted one packet of data is shown in figure 1.5(c). This string of messages makes a complete revolution of the ring. As each packet passes its destination, that node reads in the packet's data and acknowledges it as being received (by setting a flag in the packet's control field). When the packet returns to its origin node, it is removed from the ring. If no errors have occurred, the packet to be removed should be the first one in the returned string as shown in figure 1.5(e) [Gee.1982].

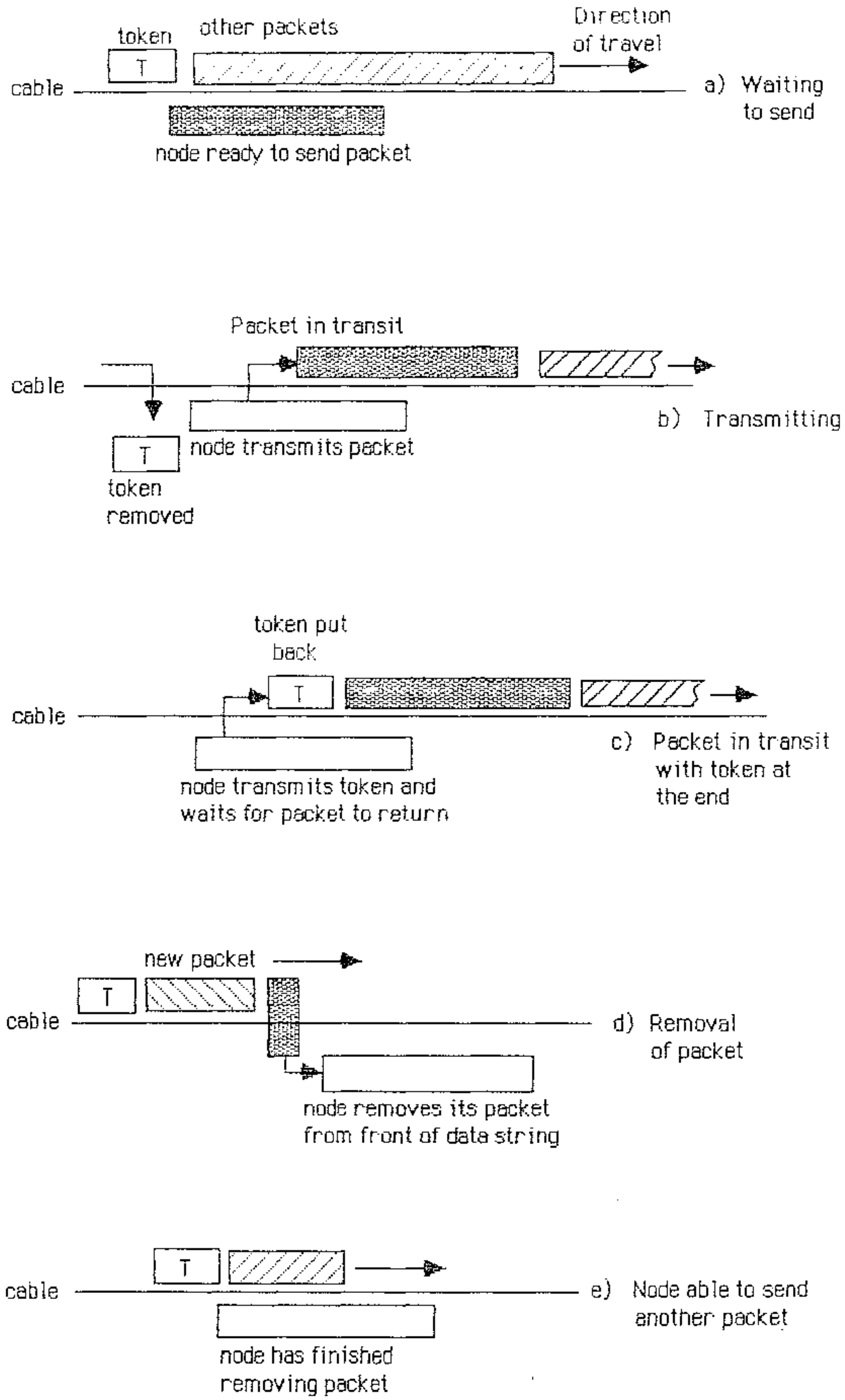

Figure 1.5 Token Passing

Register Insertion

The register insertion protocol operates with multiple packets on the ring. Each node has two registers (i.e. buffers) which can be switched into and out of the ring as required. The first buffer called the shift register, is used to store, check and forward information passing around the ring. If the check finds that the packet is destined for that node, then the packet is diverted to the associated device and removed from the ring. Otherwise, the packet continues its journey via the next node.

When a node has a packet of data to send, it stores it in the output buffer. (Packets may be variable in length up to the size of this buffer). At the end of each transit packet, the shift register checks to see if there is a packet awaiting output and if the number of empty slots in the shift register is at least as large as that output packet. If both these conditions hold, the output buffer is switched into the ring and its data is emptied. During this time, new input is stored in the shift register. The second condition guarantees that there will be enough room in the shift register to accomodate the maximum amount of input data arriving during the output process. At the end of output, the shift register is switched back into the ring and emptied as before. The ring interface which controls the switching between the shift register and the output register is illustrated in figure 1.6 [Tane.1981].

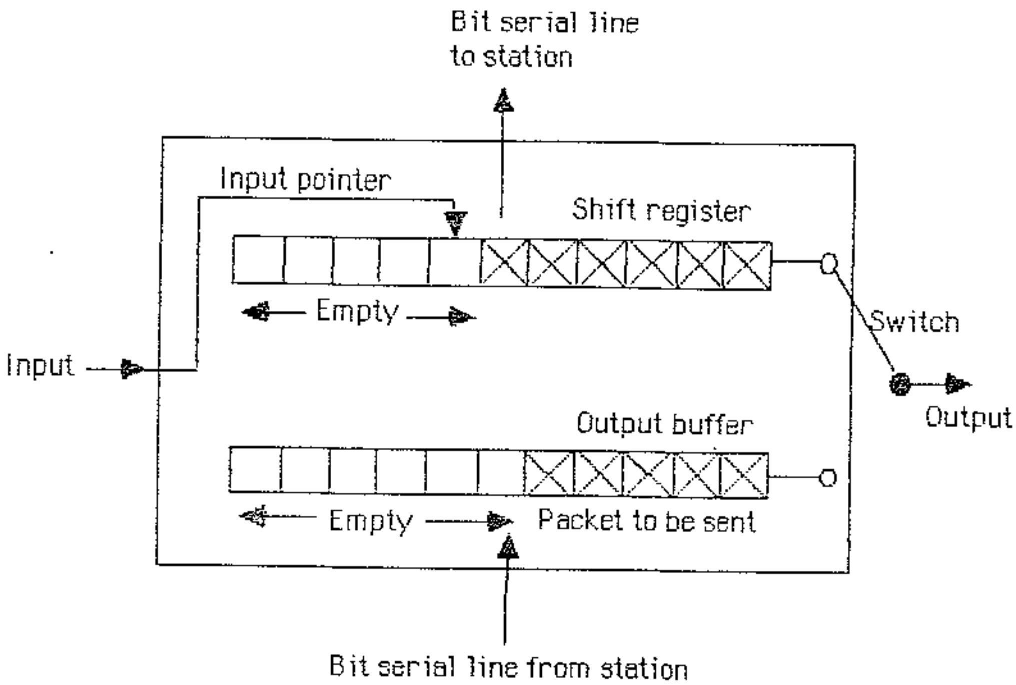

Figure 1.6 Register Insertion Ring Interface

1.1.4 Bus Networks

Bus networks consist of a central, linear channel which transports information between devices attached to it. (A channel is logically equivalent to a cable). Nearly all computer systems have a built-in bus network for connecting various components such as the processor, the memory, and peripheral controllers. Since many devices share the one transmission medium, an important issue is the allocation and sharing of that channel.

Ethernet

Ethernet is a well known bus network developed at the Palo Alto Research Centre of the Xerox Corporation in California during the 1970's [Gee.1982]. It was originally an experimental network but proved so successful that it has been installed in many business organisations and universities particularly in the United States. The main design objective of Ethernet focussed on the office environment which required a cheap, reliable network with the capacity for steady growth and easy handling of bursty data traffic.

Ethernet uses a high-speed co-axial cable as its central bus channel. Devices tap onto the co-axial cable through an IMP and communicate by "broadcasting" messages to all other IMPs on that channel (figure 1.7). Each IMP "listens" to the broadcasts extracting data for itself while ignoring everything else. There is no automatic acknowledgement of packet receipt. If this is required, it must be added explicitly by the system/program using the network.

The transmission protocol used by Ethernet is called Carrier Sense Multiple Access with Collision Detect (CSMA/CD). The "carrier sense" phrase is more easily understood as "listen before transmitting" while the "multiple access" phrase

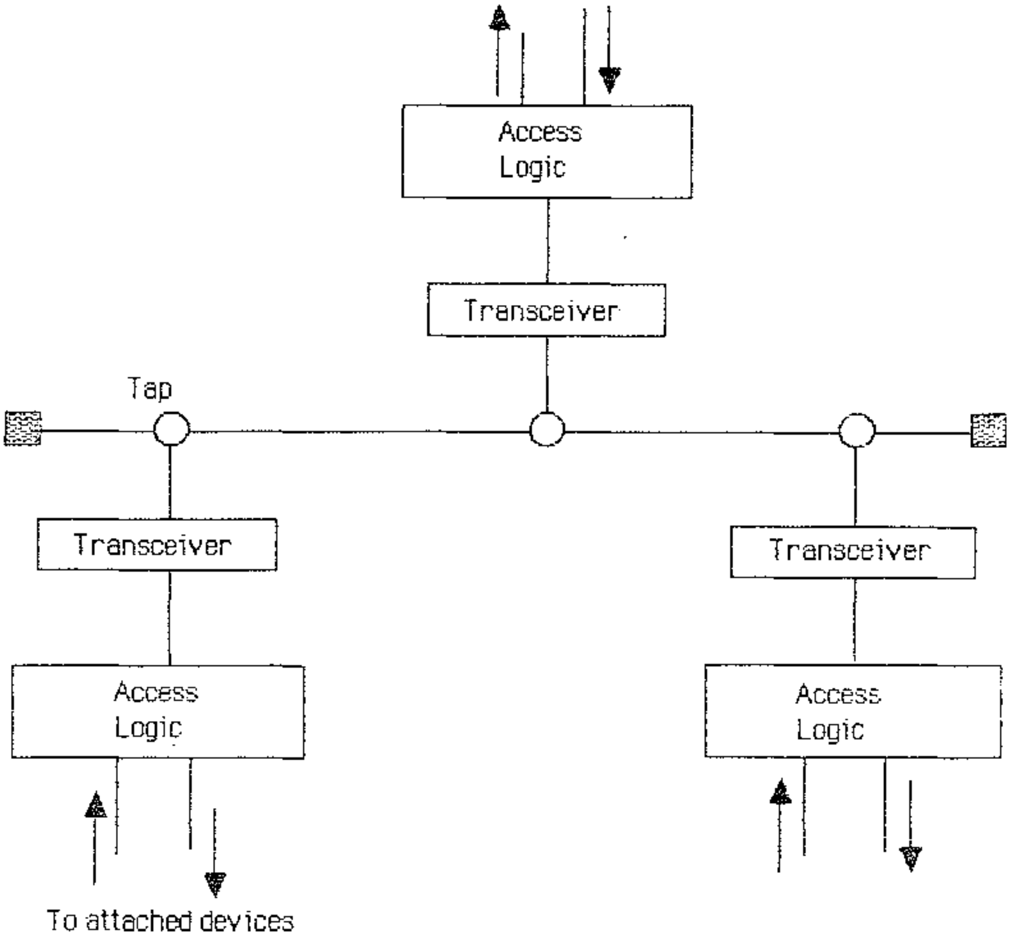

Figure 1.7 Ethernet Bus Configuration

simply means that there are several devices sharing the one communication medium. So, with only a CSMA network, any device wishing to transmit must first of all "listen" to determine if any other device is currently using the network. If so, the device must wait until the network is free. This "listen before transmitting" rule greatly reduces but does not eliminate the chance of colliding packets. Collisions will still occur if two or more IMPs transmit simultaneously.

The "collision detect" phrase can be reworded as "listen while transmitting"; i.e. once an IMP starts transmitting, it must continue listening to the channel. If it detects a collision, it must cease transmission immediately and wait a random period of time before trying to retransmit. Ceasing transmission immediately upon collision detection minimises the amount of channel bandwidth (channel capacity) which is wasted. Bass [Bass] suggests that effective bandwidths of 97% of the total bandwidth can be achieved with the CSMA/CD technique. In contrast, Petitpierre [Peti] states that "local area networks based upon a bus architecture, like Ethernet, suffer severe distance/throughput limitations related to the collision-detection mechanism." Unfortunately, Petitpierre does not support his claim with experimental data.