SPACELAB SYSTEM ANALYSIS

A STUDY OF COMMUNICATIONS SYSTEMS FOR ADVANCED LAUNCH SYSTEMS

CONTRACT:
NAS8-36717

FINAL REPORT
PHASE 1

09/30/88

FRANK INGELS
JOHN OWENS
PRINCIPLE INVESTIGATORS

STEVEN DANIEL
TECHNICAL INVESTIGATOR

F. AHMAD
W. COUVILLION
INVESTIGATORS

ELECTRICAL ENGINEERING DEPT.
MISSISSIPPI STATE UNIVERSITY
601-325-3912

(NASA-CR-[illegible]) SPACELAB SYSTEM ANALYSIS: A STUDY OF COMMUNICATIONS SYSTEMS FOR ADVANCED LAUNCH SYSTEMS Interim Final Report, Jun. 1987 - Oct. 1988 (Mississippi State Univ.) 208 p CSCL 22B G3/15 — N89-14248 Unclas 0174698

SPACELAB SYSTEM ANALYSIS

A STUDY OF COMMUNICATIONS SYSTEMS FOR ADVANCED LAUNCH SYSTEMS

CONTRACT:
NAS8-36717

FINAL REPORT
PHASE 1

09/30/88

FRANK INGELS
JOHN OWENS
PRINCIPLE INVESTIGATORS

STEVEN DANIEL
TECHNICAL INVESTIGATOR

F. AHMAD
W. COUVILLION
INVESTIGATORS

ELECTRICAL ENGINEERING DEPT.
MISSISSIPPI STATE UNIVERSITY
601-325-3912

A STUDY OF COMMUNICATIONS SYSTEMS FOR ADVANCED LAUNCH SYSTEMS

Table of Contents

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

Table of Contents (continued)

LIST OF FIGURES

LIST OF FIGURES (CONTINUED)

LIST OF FIGURES (CONTINUED)

LIST OF TABLES

1 Introduction

With the advances made in electronics and computing it has become necessary to reevaluate the internal avionics and communications systems of launch vehicles. In the past a central flight computer has been responsible for collecting all sensor data, performing all data manipulation, and controlling actuators. This was a result of the high cost of computing hardware and it's large size. As microprocessor technology has evolved it has become both feasible and desirable to off-load some of the routine computational task from the main flight computer. This may be accomplished by the replacement of sensors and controllers by smart sensors and smart controllers. These smart systems would be based on a microprocessor such as the MC68000 and could perform some of the data computation tasks usually assigned to the flight controller. This architecture has the advantage of off-loading the computations from the main flight computer at the cost of requiring more data transfers and increasing the complexity of ground checkout. However, distributed processing does provide some increased reliability in that it tends to reduce single point failure opportunities.

1.1 Background

Past launch vehicles such as the Centaur have typically used a command/response system for communications. These systems have usually used a subset of the MIL-STD-1553B local area network protocol. This system has adequately served the needs of the vehicles because the required data rates were approximately 1 Kbit/second with peaks to 1 Mbit/second. As advanced technology and additional systems are added to the avionics system this protocol will become inadequate.

1.2 Purpose

The purpose of this project is to study the application of local area network technology to the communications problems presented by the introduction of new sensors and distributed computing systems in advanced launch vehicles. The introduction of these systems will require changes in the basic philosophy of the communications system. The application of distributed computing will require that the distributed processors may take command of the communications system in order to collect data and to inform the main flight controller of the final results of computations. This will involve a change from the command/response system presently used to either a token passing or contention system so that applicable results may be transferred to the flight controller when they are available rather than waiting for polling. The probabilistic nature of distributed computing system communication would make the efficient application of a scheduled communication system very difficult. There are however many constraints that must be met by the communications system of a launch vehicle. These include reliability, fault tolerance, guarantees of data delivery, maximum allowable data latency, ability to withstand severe environmental conditions, and the ability to gracefully survive dynamic configuration changes caused by the jettisoning of a spent stage in a multistage launch vehicle.

1.3 Objectives, Conditions and Scope of the Study

1.3.1 Conditions

The objective of this study is to determine a suitable local area network architecture/hardware/protocol system for advanced launch vehicles. The goal of this study is to determine a system that would be suitable for several generations of advance launch systems as evolving sensor and distributed processing technology is applied to these systems. In order to provide a system that may be used with evolving technology a direction of evolution for advanced launch vehicles must be assumed. It is assumed that a multistage launch vehicle will be required for heavy lift systems for at least the next ten years. For the purpose of the study three stages will be assumed. The evolution and application of smart sensor technology will first require an increase in the total amount of data transferred between sensor systems and the main flight computer and then a decrease in traffic as intelligent sensors and distributed processing are employed. The increase in traffic will be caused by additional data, such as data trends being sent to the flight controller along with the sensor data. However, as confidence and expertise in smart sensors and smart systems grows the flight controller will only be informed of changes and state of health while the subsystem controller will be left to control the subsystem. This will lead to a reduction in the amount of traffic on the main bus. The requirements of reliability, guaranteed data delivery and data latency will remain unchanged throughout the program. This is also hold true for the requirement of being able to withstand severe environmental conditions.

Present communication systems for launch vehicles operate at an average load of 100 to 400 kilobits/second with bursts of up to 1 megabit/second. These data rate characteristics have been used for the Shuttle, OTV, ROI, RFLY and other missions[BOEI87]. The data rates from these missions were used by Boeing in an Air Force study to determine the data requirements for next generation vehicles. Their study concluded that a 22.4 megabit/second local area network would be required to service these vehicles (This data was taken from missions that did not have docking radar or RF communications on the avionics bus). These data rates are however, increasing and the following is an estimate by Mississippi State University of the data rates that may be expected for 1995 to 2000 missions.

1.3.1.1 Vehicle Node Communications Requirements

The communications load for the ALS study is determined to fall within the parameters listed below. Typical nodes are presented with an additional station, RADAR, added to include future development where the data from complex radars may be put on the avionics bus. The nodes are:

NODE 1, SENSORS

Assume 100 sensors/stage and three stages.

Assume 100 samples/second and a 32 bit sample.

3 x 100 x 100 x 32 = 962 kilobits/second (Kbps)

Assume 6% overhead for transmission

Maximum data rate is 962 Kbps x 1.06 = 1.02 Mbps

NODE 2, ENGINE CONTROLLER/ SEQUENCER

Maximum data rate 1 Mbps maximum with overhead

NODE 3, POWER

100 parameters at 10 samples/second with 32 bits/sample

32 Kbps/second

6 % overhead

Maximum data rate 34 Kbps/second

NODE 4, TRANSPONDER

100 bytes/transmission 10 transmissions/second

Maximum data rate 8 Kbps

NODE 5, ENGINE HEALTH MONITOR

Assume ABACS type monitoring system

with only 10 of the 16 stations active at a time

Maximum data rate High estimate of 160 Kbps with overhead

This is all the data flow of the internal test bus which would only be put on the main system to be sent to the archives.

NODE 6, CONTROL COMPUTER

Maximum data rate 1 Mbps with overhead

NODE 7, SIMULATOR

Used to replace another station on the bus or provide input data for simulation

Maximum data rate Worst case 1 Mbps with overhead

NODE 8, ETC

This node is used to connect auxiliary services to the facility and will usually be used as a destination only. When used as a generating station a maximum offered load of 10 Mbps will be used for modeling purposes.

At this point the maximum load is 14.222 Mbps which could be serviced by many local area networks. The addition of a high data rate station such as a radar station will be considered next. This station will be designated node 9.

NODE 9, RADAR UNIT

CASE 1

1 operating radar with 60 - 100 updates/second

5 bytes/parameter and 10 parameters

100 byte packet with overhead in updates

Maximum data rate 100 bytes x 100 = 10000 Bps = 80 Kbps

This radar could easily be added to a local area network serving nodes one through eight.

CASE 2:

The raw radar information from a 50 kilo-pulses/second radar where the start time, stop time, and five samples are quantized would require approximately 100 bytes/pulse.

Maximum data rate 50 K x 100 = 5 MBps = 40 Mbps

This could be serviced by a high speed LAN but would be best served by a high speed point to point link.

CASE 3

The raw information provided by a Doppler radar can require an I/O capacity of up to 40 MBps = 320 Mbps for the transfers out of the MIS processor.

This data rate could not be served by a present day LAN. A point to point connection would be required.

Mississippi State's estimated data rates without RF/radar nodes compare favorably with those presented in the Boeing study. It will therefore be assumed for this study that a local area network that can service a 25 megabit load will be required for future launch vehicles. This data rate severely limits the number of existing local area networks that are applicable.

1.3.1.2 Ground Checkout

Ground checkout has been a driving force in data bus requirements for several of the most recent vehicles such as the shuttle and OMV. The use of advanced sensor systems with embedded self-test will reduce the data bus requirements for ground checkout. However, the application of these systems and their interface with a smart ground checkout system must be studied. This is an appropriate task for the MAST facility. A study of the proper mix of embedded test and ground checkout could be undertaken using the various subsystems tied to the MAST facility. Different mixes of embedded test and ground checkout could be conveniently made so that performance data and requirements for these systems could be obtained.

A problem added by the use of a central local area network that services many subsystem networks is the inability to test the subsystem directly. This is a driver for self-test of subsystems. The local area network of a subsystem must also be tested and this can be accomplished by a monitor node, such as a Lanalyzer, for the subsystem. Many commercially available local area network monitors and test programs are available. These programs could be placed in one of the onboard computer systems that would have the responsibility for testing the local area network. This system would be required to insure that each station could access the communcations channel as well as receive messages from other stations.

An additional problem will be the testing of the protocol's error recovery systems. The monitoring station should introduce faults into the system such as lost tokens for token access systems or collisions for contention systems. These features may require that the monitor/checkout terminal of the network have additional hardware to perform these tests.

1.3.1.3 Launch Vehicle Environment

The launch vehicle environment requires equipment able to withstand both high and low temperatures, changes from atmospheric pressure to vacuum, vibration and the ability to withstand high G forces. This will usually require the addition of hardware to most systems so that they may operate reliably in this severe environment. The requirements for fault tolerance will also require modifications to most hardware.

1.3.1.4 Evolution of Intelligent Sensor Technology and How it Affects the Communications System

As the evolving technology of the smart sensor and distributed computing fields is applied to advanced launch systems a new analysis of data and command communications within the launch vehicle will be required. In the past a central flight computer has taken data from various sensors for pressure, temperature, flow rate, position and velocity and transmitted commands to actuators for thrust vectoring and flow control. This communication system has naturally taken the form of a star architecture. With the addition of smart sensors that will perform trend analysis and additional functions to off-load computing from the main flight computer this architecture will not be required to change. However, the addition of distributed processing and intelligent sensors that may act as bus masters will require a change in the basic philosophy of the main flight computer serving as the sole bus master.

Several steps will be required in the addition of this new technology to launch vehicles. These steps fall into three general cases:

1) The first case deals with the sensors and flight computers used in today's launch vehicles. In this case sensors only provide data words to the flight computer and the computer provides commands to the actuators. A possible advantage of applying local area network technology to this type system is that, with a bus based LAN, configuration changes could be easily accomplished. This could prove beneficial to a reusable system whose launch configuration changes between uses.

2) The second case arises from the addition of smart sensors that off-load some computing task from the main flight computer. This may be accomplished by the sensor performing trend analysis, state of health self-checks, and the self-recognition of proximity to or crossing of critical limits. These tasks, which are presently performed by the flight computer, could be accomplished by the addition of a microprocessor to the sensor system. This has the benefit of off-loading computations from the main flight computer at the disadvantage of increasing the amount of data that must be communicated between the sensor and the main flight computer due to the addition of trend information to the data.

3) The third case arises when intelligent sensors and distributed computing are applied to the launch vehicle. This will entail a major change in the on-board avionics communications system. The use of distributed processing may be accomplished in several ways. The first would be to have sub-computers that simply take tasks assigned to them by the main flight computer and return their results to this computer or they may be assigned responsibility for certain sensors and control tasks and only be required to communicate with the main flight computer for unusual problems. In this case the main flight computer would only serve as a flight status monitor until an unusual problem occurred that would require its intervention. Distributed computing could also be used in various subsystems such as an engine subsystem controller. The subsystem could be sent a message to change throttle settings and the processor within the subsystem would compute the control settings and take control of the actuators to accomplish the changes, off-loading this task from the central flight controller. The use of these types of distributed processing and intelligent sensors would reduce the amount of communication that will be required in the system as compared to case two but will also require a change from command/response to a distributed access control for the communications system to obtain the benefits of intelligent sensors and distributed processing.

1.3.2 Application of Local Area Network Communications Systems to an Advanced Launch Vehicle

The communications system requirements of the evolving launch systems can be met by the application of local area network technology. A local area network that provides reliable performance with guaranteed data delivery and bounds on data latency can be used. The advantages possible in this system would be: more timely access to time critical data than in a polling or command/response system and more effective utilization of distributed processing. Possible problems would be the reduction in the transfer of critical information due to the overloading of the communications path by routine data in unusual conditions and the loss of command/control of the system by the main flight computer if another system takes control of the communications system in a critical situation. Any local area network applied to launch vehicles must address these problems.

1.3.3 Determination of a Suitable Protocol/Architecture/Hardware to Implement the Local Area Network Communications System for an Advanced Launch Vehicle

The determination of a suitable local area network for an advanced launch vehicle may be broken into three interrelated tasks. These tasks are the determination of appropriate rules of communication, the determination of an appropriate architecture and the determination of the hardware needed to implement the protocol on the given architecture. The task of determining appropriate rules of communications depends on the following:

The amount of traffic to be handled.

The differing levels of priority that stations or packets may have.

Requirements such as maximum data latency, guaranteed access time, etc...

The determination of an appropriate architecture requires knowledge of physical layout of the stations to be placed on the network and reliability requirements for the network.

The task of determining appropriate hardware is often the hardest task. Many protocols can be designed that meet the communications requirements for a system but can not be implemented. Another aspect of this problem is gambling on the development of hardware for leading edge protocols in the development stage. For space applications it would be desirable to have a system that is well proven with a long service record.

1.4 Significance

This study will provide a document that can be used in development of an avionics communications system for future launch vehicles. Suitable local area networks for evolving advanced launch systems will be determined. The goal of this study is to provide general outlines for the application of local area networks to the three cases presented in section 1.3.1.2. The actual design of these systems will be dependent on project specific parameters that will require additional evaluation.

2 Network Architectures

In communication networks, topology describes the method in which the nodes of the network are physically interconnected, it is a function of the communication links and switching elements in the network. The path taken by the data going from one node to another in the network is determined by its topology. In general, communication networks topologies are of the following types :

2.1 Ring

The ring topology is illustrated in Figure 2.1. In this figure the boxes represent the nodes of the communication network and the lines which connect the boxes are the links of the network. If we define a communication network in which one node participates in all available links of the network as a centralized network, then it is obvious from Figure 2.1 ring topology is a decentralized one, moreover, it is a closed loop. Since every node in a network with this kind of topology has an unbuffered repeater and utilizes two links only, no routing decision is required and data circulation is in one direction, i.e. either clockwise or counter-clockwise. When a node needs to send information to another node in the network, it will transmit the information in packets. Among other information, each packet contains the address of the receiving node. The packets are transmitted one at a time and each packet is circulated bit by bit through the repeaters in the network. When the receiving node identifies its address in a packet it copies the information into its buffer as the packet passes by. One characteristic of the ring topology is the determination of which node can transmit at any given time, this determination is achieved through access control mechanisms (protocols). Such mechanisms are discussed in Part Four of this report. [STAL84]

2.2 Bus

The bus topology is depicted in Figure 2.2. In this communication network topology no routing decisions are required. All nodes are attached to a linear transmission medium

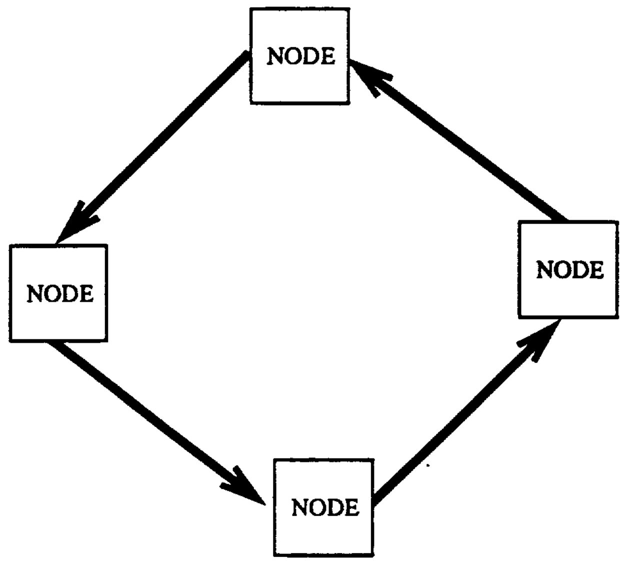


FIGURE 2.1 RING TOPOLOGY

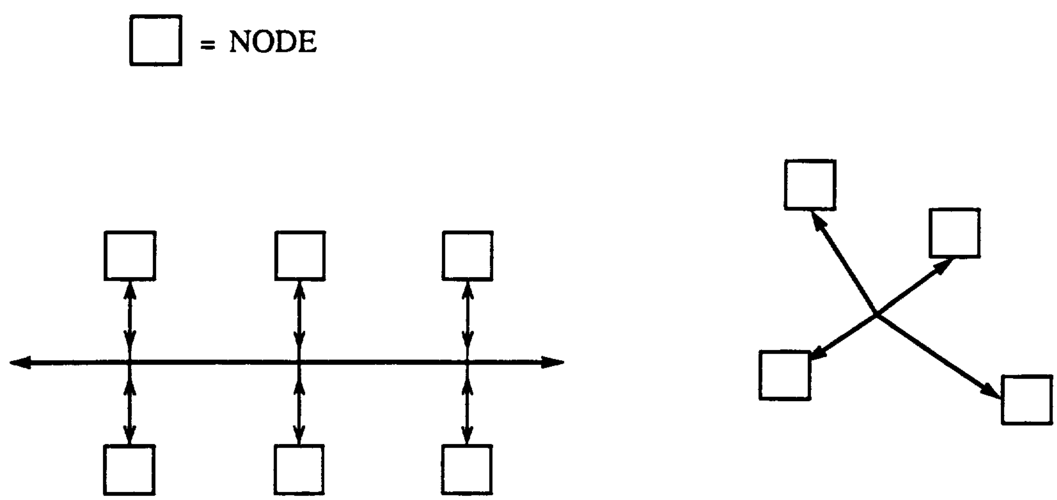


FIGURE 2.2 TWO WAYS OF REPRESENTING BUS TOPOLOGY

(i.e. the bus) via suitable hardware interfacing. A message transmitted from any node in the network will flow in both directions on the bus. The designated node identifies its address and will receive the message. Since the message is transmitted through the bus and all the nodes are attached directly to the bus, reception of the message is accomplished by every node in the network if needed. Another representation for this topology is shown in the same figure, it is clear that all the network's nodes share a common point of connection. In bus topology only one node can transmit at any given time hence a control mechanism is needed. Such mechanism is presented in part four of this report. Tree topology is a generalization of bus topology. [STAL84]

2.3 Star

Figure 2.3 shows a star topology for a communication network, it is a centralized network. Usually the central node (CN) has complex switching capabilities, and part of its task is the following : In the network, when node A wishes to communicate with node B it will first ask for permission from the central node. It will supply to the central node, among other information, node B's address. The central node executes the required steps to set the circuit, once the circuit is set the information between the two nodes is exchanged as if the two nodes were connected via a dedicated point-to-point link. [STAL84]

2.4 Hybrid

We define a hybrid topology as a topology containing more than one type of the topologies mentioned previously . Hybrid topology can be of two kinds:

Mesh Topology : Figure 2.4 shows an example of this kind of topology, if the dotted line is removed from the mesh the resulting network will have a ring topology. This type of network is called a "multi-nodal", "distributed", or "fully interconnected "network also. In this topology every node has a dedicated point-to-point link to every other node in the network. The controlling mechanism which determines the manner in which any two nodes can communi-

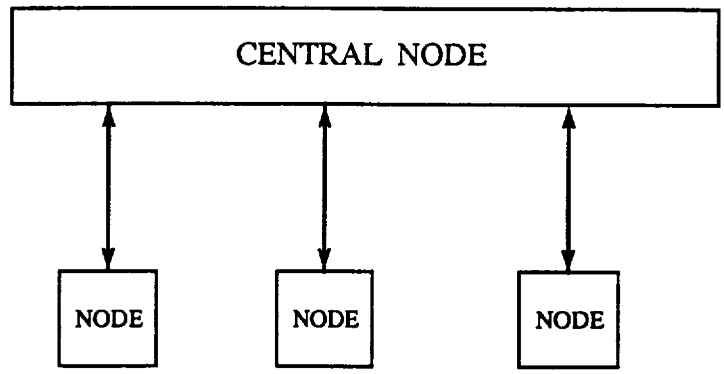


FIGURE 2.3 STAR TOPOLOGY

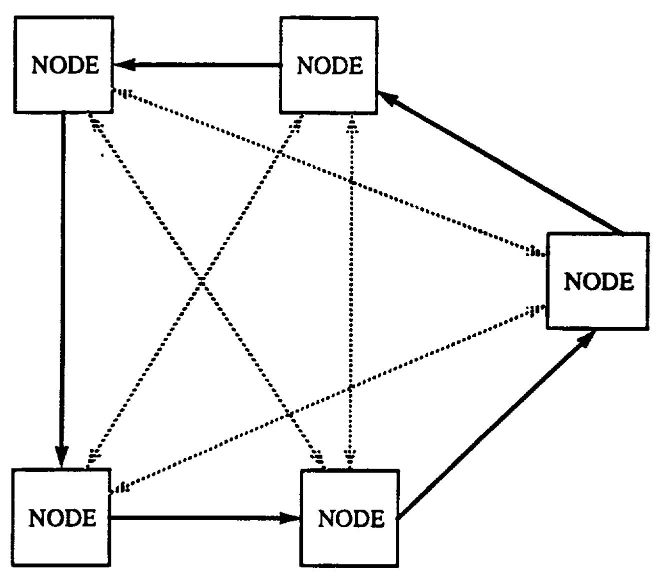


FIGURE 2.4 MESH TOPOLOGY

cate will be presented in part four of this report. It is important to mention that for N nodes, the mesh topology needs N(N-1) links and every node requires (N-1) input/output ports. In general this is the case with every point-to-point communication link.

Multi-mesh Topology : An example of this type of topology is illustrated in Figure 2.5. Some of the nodes of this topology have the ability to interface with more than one type of topology, this of course will introduce some complexity to their input/output ports, such nodes are called Bridges and Gateways. The controlling mechanism of this type of topology is discussed in part four of this report. A brief description on Bridges and Gateways is presented in the following section. [SHER85]

2.5 Bridges and Gateways

In a hybrid topology, the interconnection of sub-networks that exhibits the same interface techniques and protocols for medium access (i.e. homogeneous sub-networks) is accomplished through Bridges, see Figure 2.5. The structure of a Bridge is shown in Figure 2.6. The Bridge receives a message from a node in sub-network A and buffers the information while waiting for the opportunity to transmit the information to a node in sub-network B, the bridge uses its packet buffer to perform this task. In addition packet buffers help during cross-bridge traffic peaks. This is a time during which the traffic offered by one sub-network exceeds the available capacity of the other. The duty of the control filter in the bridge structure is to decide from which sub-network the bridge should " pull off " and buffer a transmitted message until it will have the opportunity to transmit it to the other sub-network. On the other hand, if the sub-networks in a hybrid topology exhibits different interface techniques and protocols for medium access a protocol convertor device is used, such device is called a Gateway, see Figure 2.5. For example in packet switched interfaced systems a Network Interface Unit (NIU) is considered as a gateway, this device performs the following functions:

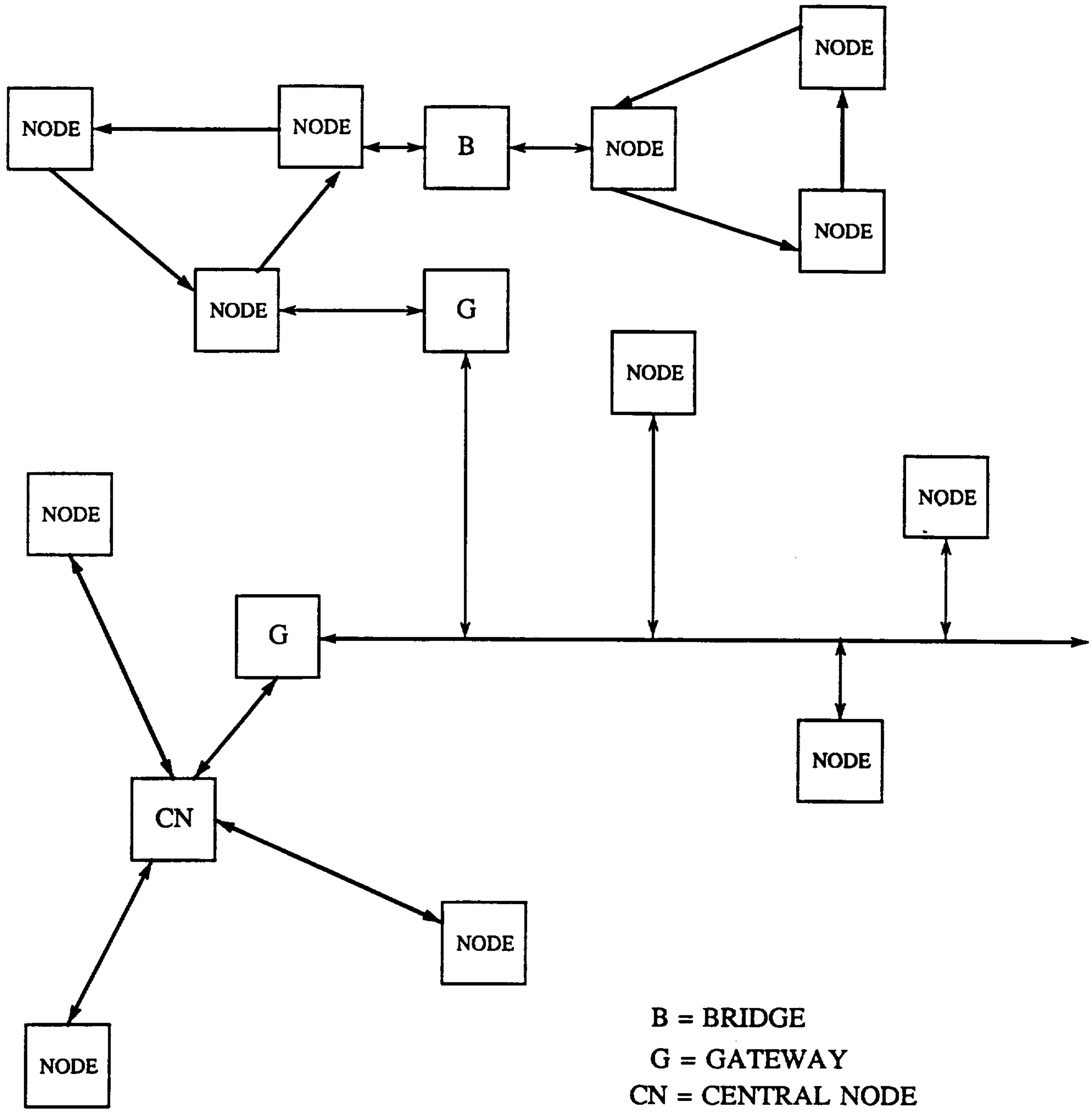


FIGURE 2.5 MULTI-MESH TOPOLOGY

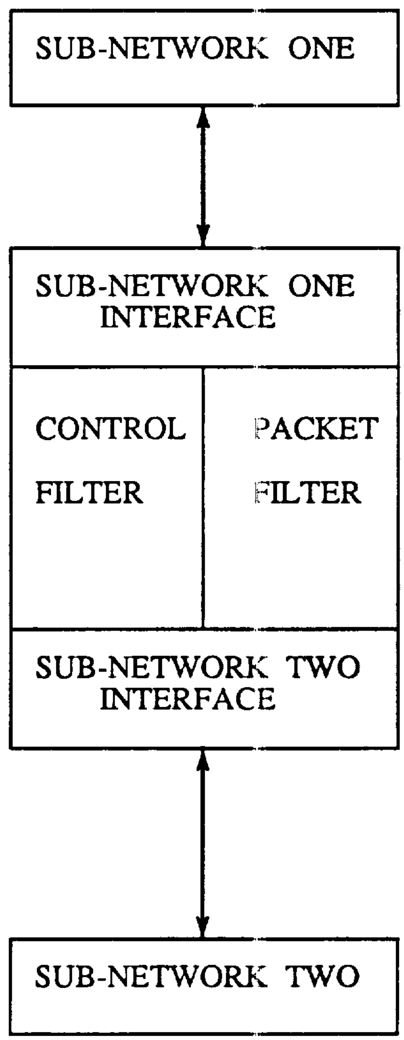


FIGURE 2.6 BRIDGE STRUCTURE

a- Accept data from attached device

b- Buffer the data until medium access is achieved

c- Transmit data in addressed packets

d- Scan each packet on medium for its own address

e- Read packet into buffer

f- Transmit data to attached device at the proper data rate

The architecture of this device is shown in Figure 2.7. [STAL84]

2.6 Recommended ALS Architecture

Many parameters will affect the design and implementation of the ALS architecture. Some of these parameters are:

(a)- The degree of advancement in the sensors (a sensor is assumed to be composed of a remote terminal and a sensing device) and actuators of the system.

(b)- The importance of the sensors and actuators services at any given time. The needed information from a sensor and the function of an actuator at a given time may range from unimportant up to extremely important.

(c)- The physical spacing between a group of sensors and the importance of the availability of their information at a given node. Such spacing may affect the homogeneity of the arrival time of the information.

(d)- Different stages in the system may require different architectures. This is because different stages are designed to accomplish different tasks.

(e)- Since the complete launch process is a multi-phase process, it is far from simple. This could justify the implementation and utilization of a hybrid system.

(f)- It is extremely important to have the complete information reaching a decision making node as soon as possible from a given sensor. Accordingly the " action taking " part of the system needs to receive such decisions at the precise moment. Hence optimal timing and routing is required.

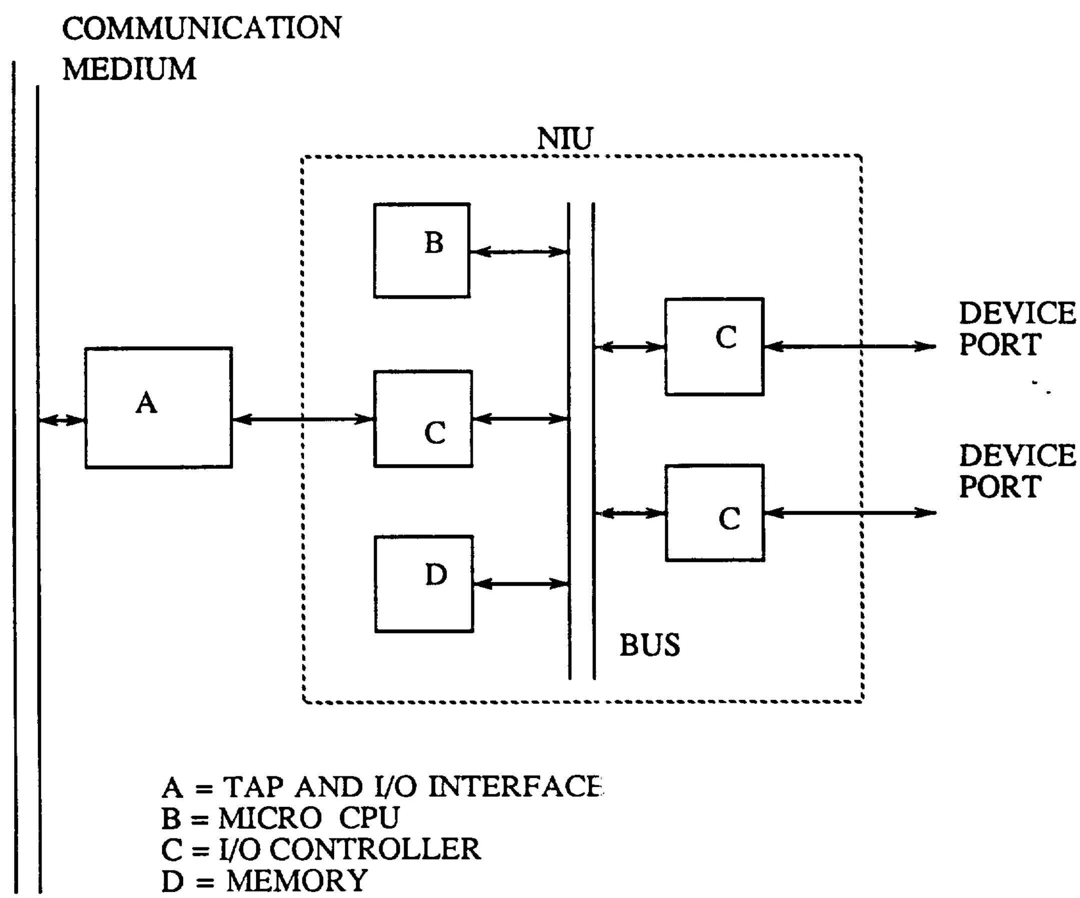


FIGURE 2.7 NETWORK INTERFACE UNIT AS AN EXAMPLE OF A GATEWAY

(g)- Physical breaks in the system's topology are of two kinds, anticipated and accidental.

2.6.1 Justification by Pros and Cons

In light of what has been mentioned, different types of architectures may be considered for implementation in a future launch system. First, information on the general advantages and disadvantages of some of the typical topologies is of benefit. Some of the most important advantages in using the ring topology are the utilization of the point-to-point communication link, and the regeneration of the message at every node which helps in minimizing transmission error and maximizing the total distance covered by the network. In ring topology token latency and repeater delays increase as the number of nodes increases. This results in decreasing the efficiency of the communications system. Decreasing transmission speed and/or average packet size will degrade this topology's performance. Selection of transmission media affects ring topology characteristics, for example, when optical fiber links are used between the repeaters, very high throughput is achieved. However, a break in one link of the ring will result in a fatal crash.

One phenomena of ring topology is that when a message is injected in the ring it will continue circulating until it is completely attenuated or until is removed. This can introduce echoes in the network unless messages are at some point stripped from the ring.

A bus or Tree topology will allow multiple nodes to share the same path, however only one node can transmit at any given time. In this kind of topology no switches or repeaters are needed. A linear token passing multiplex data bus has been proposed [SAE AS 4074.1 VERSION 4.0 JANUARY 25,1988 TASK GROUP COPY proposal]. This bus topology is advertised as providing high reliability, high bandwidth, and low latency characteristics.

A star topology is completely dependent on the central node's abilities and operation levels . The required complexity in a reliable central node must be considered as a disadvantage for this kind of topology. Currently, this topology is used in launch vehicle local area networks.

An implemented protocol on a ring topology may furnish and support reliable connectivity after a physical break in the ring. On the other hand a physical break in a bus topology may disable the bus completely or partially. In the star topology a physical break in a link will only disable the corresponding node. Among the three major topologies (i.e. bus, star, ring) the bus topology has the greatest flexibility of the removal or the addition of a node with minimum labor. Primarily three conceivable cases need to be consider.

2.6.1.1 Case 1

In the first case the majority of the ALS network nodes except the flight controller (the flight controller is assumed to be composed of a computer and a command terminal) are considered primitive. A star topology in which the flight controller is the central node will give the flight controller an exclusive right to monitor, dictate and prioritize any act of communication taking place in the network. In this case it is the duty of the flight controller to keep an optimal process of communication with all parties of interest at any given time. Above that, it needs to take the responsibility of the house keeping process, such as the re-configuration of the network system in case of a loss of a link or node. In general the flight controller must be armed with very sophisticated switching, amplifying, and signal processing circuitry. This of course will demand a very advanced flight controller. A decentralized network topology on the other hand, will result in a poor communication environment since this will deny the flight controller its role. Also, the technical status of the sensors and the actuators in the system does not encourage the idea of utilizing a bus, tree or ring topology. Hence, a communication/command/control domain with the flight controller being the only intelligent node will perform at its best with a star topology. Present launch vehicles communication network use the MIL-STD-1553 bus system. In this system the flight controller acts as a central node and it is the only node in the system which can initiate an act of communication with any other node in the system.

In addition to the network suggested above and present launch networks, a LAN system with three MIL-STD-1553 buses all connected to the flight controller and each serving a different stage of a three stage launch vehicle is a possibility. In this LAN system, communication between stages is assumed to be a rarely required communication.

2.6.1.2 Case 2

It is a known fact that the ALS is a multi-stage system. If the use of semi-skilled sensors and actuators is to be considered then the ALS network may be partitioned in to sub-networks. A possible system configuration under these circumstances is suggested in Figure 2.8 where each sub-network utilizes a star topology, and each central node (CN) of such sub-networks is interfaced to the system's backbone network via a bridge. The backbone network of the ALS is shown to be a bus. The flight controller is on the bus where it performs its duties such as monitoring, sending commands, and receiving information. The star topology keeps the flow of the information into and out from any stage of the ALS at its optimum while the bus topology used for the backbone provides the flight controller with a means to communicate with any node in the network at any given time. The number of the stages is assumed to decrease in time, as at time t_0 for example stage 0 may knowingly be physically separated from the whole system and at time t_1 stage 1 may be separated and so on. Taking into account what has been stated above and the topology suggested for this case the flight controller's burden is smaller as compared with case one. This is because the central nodes in each sub-network are assumed to have a degree of intelligence in addition to what the corresponding nodes have, hence information and command manipulation will be taken care of partially at the corresponding sub-networks without the intervention of the flight controller, unless there is an emergency. It is possible to replace the backbone's bus topology with a ring topology. If this is done, then the flight controller will no longer have a direct path of communication with any node in the network.

2.6.1.3 Case 3

The ultimate advancement in sensor and actuator technology in the future will offer the ALS a network composed of skilled devices. A completely decentralized communication network topology may then be used. Such a topology is shown in Figure 2.9 where the LAN in every stage in the system has a ring topology, the sub-networks are interfaced through Bridges/Gateways and the flight controller's activity is diminished. Generally, in this case the flight controller's duty is to monitor the whole system performance and conceivably perform system diagnostics. On the other hand each sub-network in the system will take the responsibility of performing optimal command/communication/control among their nodes and themselves. It is clear that the information is routed between all parties of the network, and at the mean time the flight controller is no longer holding the privilege of decision taking nor is it issuing all commands. Hence the communication network has to be a high speed, high bandwidth decentralized system. According to today's technology ring topology is in the lead in high speed, high bandwidth traffic phenomena.

2.6.2 Conclusions

Since in case one the flight controller has full authority in the ALS communication/command/control environment, a centralized topology is recommended. A star topology will be suitable for this case. On the other hand in case two and three the flight controller's roll is gradually changing. It is changing in a way that it is not the only node with intelligence capabilities. Therefore a decentralized topology is recommended for both case two and three. Case two will utilize a hybrid topology in which the backbone part insures a solid communication environment between the flight controller and the rest of the network. The topology of this part is suggested to be a bus for reasons discussed in case two. Case three's network system will have N ring topology sub-networks. The speed and ease of the flow of the information in all parts of the system will be insured via this topology.

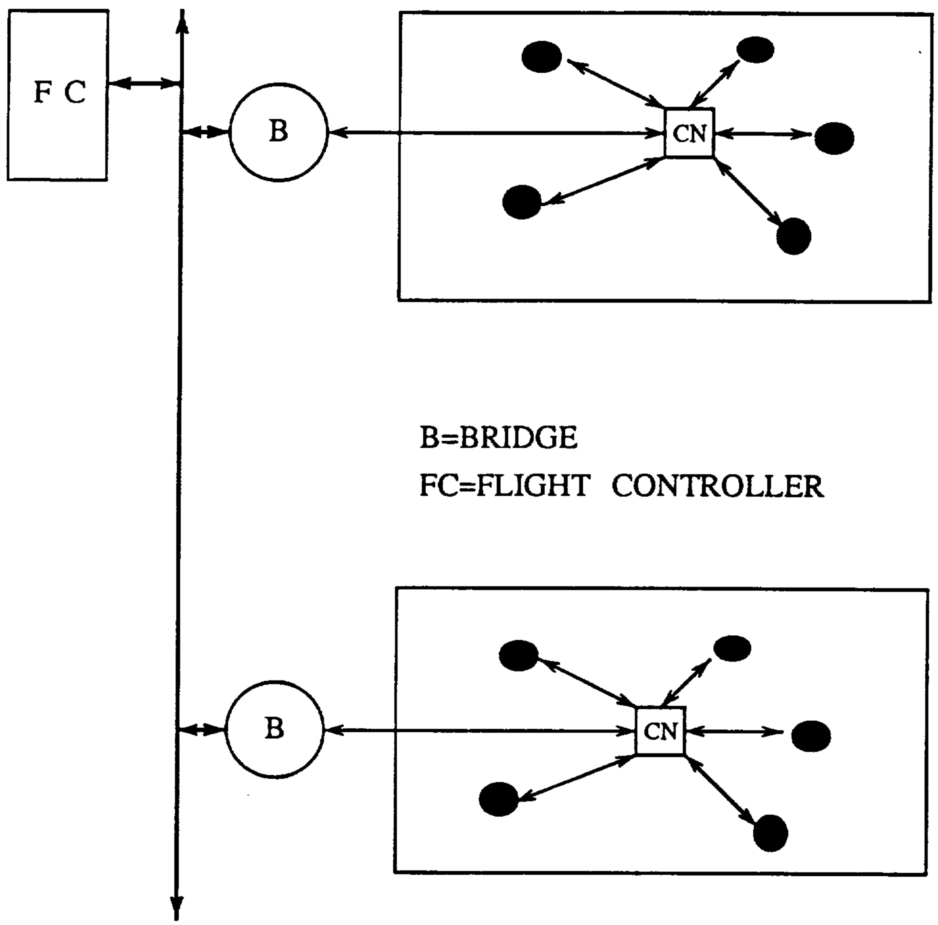


FIGURE 2.8 A SUGGESTED TOPOLOGY FOR CASE TWO. THE SYSTEM IS ASSUMED TO BE OF TWO STAGES.

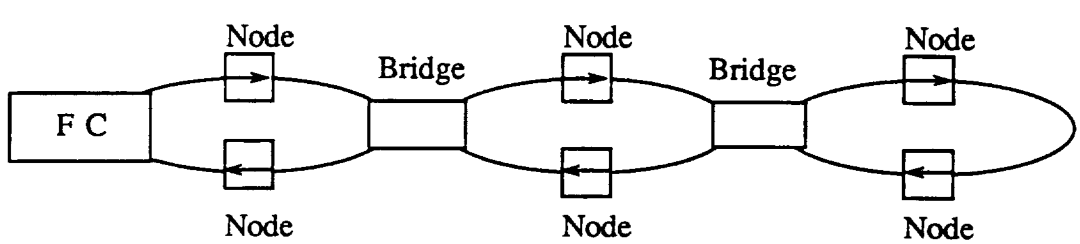


FIGURE 2.9 A SUGGESTED TOPOLOGY FOR CASE THREE. THE SYSTEM IS ASSUMED TO BE OF THREE STAGES.

3 Hardware

3.1 Transmission medium

3.1.1 Twisted pair

In the twisted pair medium conductivity is established through two wires of copper or through the copper of steel coated copper. The two wires are twisted to minimize electromagnetic interference. Analog or digital signals can be transmitted through this medium and amplifiers/ repeaters are used for transmission continuation. Voice is commonly transmitted through this medium, the twisted pair has a capacity of 24 voice channels using a bandwidth of up to 268 KHz. When using this medium for digital transmission, modems (modulators /demodulators) are used and the aggregate data rate is a function of the speed at which the modems operates. This medium could be used in point-to-point and multi-point communication systems. Using twisted pair instead of coaxial cable transmission medium, for example, will result in lowering the system's price at the cost of degrading the system's performance. This medium is not immune to noise and shielding is required, doubling the shielding will reduce the effect of the EMP (Electromagnetic Pulse) energy on its characteristics [STAL85]. As an example, in today's technology a system of twisted pair LAN pushes data at 4 Mbps for up to 8500 feet. This system is available from Corvus System, Inc.

3.1.2 Coaxial Cable

The coaxial medium is made of two concentric conductors, hence outer and inner conductors. The outer conductor is a hollow cylinder which can be either solid or braided, and the inner conductor can be either solid or stranded. Regularly spaced insulating rings or solid dielectric material is used to hold the inner conductor in its position. The unique configuration of this medium permits it to operate over a wide rage of frequencies, and it is usually identified by its characteristic impedance, for example 50 ohm cable or 75 ohm cable etc.... The 50 ohm coaxial cable is used almost exclusively for digital transmission, various modulation schemes have been used for this type of transmission. These modulation schemes include ASK (Amplitude Shift Keying), FSK (Frequency Shift Keying), and PSK (Phase Shift Keying) techniques. The use of FDM (Frequency Division Multiplexing) techniques will allow for the transmission of a large number of channels through this medium. Point-to point and multi-point connection could be performed using coaxial cables. It's noise immunity depends on the application and implementation adapted in the system design.[KRAU84 / SHER85]

3.1.3 Fiber Optics

The fiber optic medium is fabricated using two different compositions of glass. One of the compositions has relatively high index of refraction and is used to form the core of the fiber; the core is surrounded by the second composition which has lower index of refraction in relation to the first composition; the second composition is called the cladding portion of the fiber. The means by which the light is propagated through the optical fiber are:

-Total internal reflection.

-Internal refraction.

-Internal guiding.

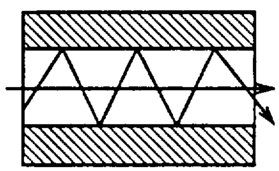
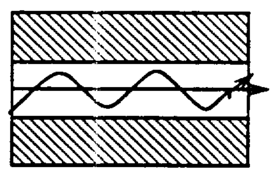
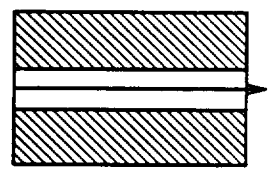
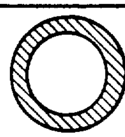
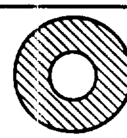
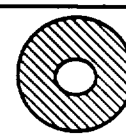
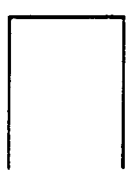
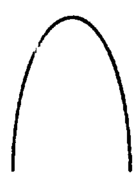
Most of the light is propagated through the core of the fiber and the cladding is used to reduce the scattering loss resulting from dielectric discontinuity at the core surface. The cladding also adds mechanical strength to the fiber body; and it protects the core from the absorbing surface contaminations with which it might come in contact. Extra protection is accomplished through encapsulating the fiber in an elastic plastic buffer coating. Basically there are three types of optical fiber:

(a)- Multi-mode step-index fiber

(b)- Multi-mode Graded-index fiber.

(c)- Single-mode step-index fiber.

The three types of optical fiber are illustrated in Table 3.1. In this table the propagation mechanism, geometry, and the refractive index profile are shown [DALY84]. A data rate

TABLE 3.1 COMPARISON BETWEEN FIBER OPTIC TYPES			
Fiber Type	**Multimode Step-Index Fiber**	**Multimode Graded-Index Fiber**	**Single-Mode Step-Index Fiber**
Propagation Mechanism	**Reflection** 	**Refraction** 	**Guiding** 
Geometry			
Refractive Index Profile			

distance product for a system using an LED optical source and emitting in the 800-900 nanometer region is about 150 Mbps.Km, while the same product may reach a value of 2500 Mbps.Km if a laser is used as a source, the value of this product will even get higher, up to 25 Gbps.Km, if the laser is a InGaAsP laser. Optical fiber medium is immune to electromagnetic and noise interferences[KEISER]. Table 3.2 shows a brief comparison between the three major fiber optic types using bandwidth, splicing difficulty, and cost among other parameters.[AKER87]

3.1.4 Transmission Medium Utilization in LANs

In today's technology different networks use different transmission media. Among other qualities, maximum distance coverage and price are two elements that need to be considered when designing a LAN. Table 3.3 illustrates this information for typical networks.

3.2 Connectors and Chips

3.2.1 Connectors

Connectors must be used only to insure the continuity of energy flow, they should never be used to support any part of the system or the transmission media . It is significant to understand that connectors are one source in the system from which energy leaks, hence care must be taken in selecting the type of the connectors and accordingly their utilization. Popular commercially available connectors are:[AMP82]

-Electric Pin and Socket Connectors

-Electric Printed Circuit Board Connectors

-Electric Coaxial Connectors

-Electric Ribbon and Flat Cable Connectors

-Electric Network / Premises Interconnections

-Optical Fiber Connections

-Optical and Electrical Directional Couplers

TABLE 3.2 COMPARISON BETWEEN FIBER TYPES PARAMETERS			
FIBER TYPE / PARAMETER	SINGLE-MODE FIBER	GRADED-INDEX MULTIMOUDE FIBER	STEP-INDEX MULTIMODE FIBER
SOURCE	REQUIRES LASER	LASER / LED	LASER / LED
BANDWIDTH	VERY VERY LARGE > 3 GHz.Km	VERY LARGE 200 MHz TO 3 GHz.Km	LARGE. < 200 MHz.Km
SPLICING	VERY DIFFICULT DUE TO SMALL CORE	DIFFICULT BUT DOABLE	DIFFICULT BUT DOABLE
EXAMPLE OF APPLICATION	SUBMARINE CABLE SYSTEM	TELEPHONE TRUNK BETWEEN CENTRAL OFFICES	DATA LINKS
COST	LESS EXPENSIVE	MOST EXPENSIVE	LEAST EXPENSIVE

TABLE 3.3 NETWORK DISTANCES AND COST			
CABLE	NETWORK	AVERAGE TRANSMISSION DISTANCE	COST
Fiber Optic	Ethernet, ISDN, FDDI, Token Ring	3,000 ft.	High
Coaxial Cable	Ethernet	Bus Length of 600 ft.	Moderate to High
Shielded Twisted Pair	Token Ring	300 ft., from work station to wiring closet	Moderate
Shielded Twisted Pair	Ethernet	600 ft., from work station to wiring closet to work station	Moderate
Shielded Twisted Pair	ISDN	Usually greater than 300 ft.	Moderate
Twinaxial	Ethernet	300 ft.	Moderate
Unshielde Twisted Pair	Ethernet, Token Ring ISDN	100 ft.	Low

The optical connetions and the optical and electrical directional couplers are disscussed in the remainder of this section.

3.2.1.1 Optical Fiber Connections

Interconnection in optical fiber medium occurs at the optical source, at the photodetectors, at intermediate points within a fiber cable where two fibers are joined, and at intermediate points in a link where two cables are connected. If the connection made is a permanent bond then it is called a Splice, and if it is a demountable joint then it is called a Connector. The following is a brief description of both techniques.

Splicing Technique : Three splicing techniques are popular, they are:

1- Fusion Splices : The fusion splice is made by thermally bonding together prepared fiber ends. In this method the ends of the fibers are first pre-aligned and butted together, this is done either in a grooved fiber holder or under a microscope with micromanipulators. The butt joint is then heated with an electric arc or a laser pulse so that the fiber ends are momentarily melted and, hence, bonded together. This technique can produce very low splice losses (e.g. 0.1 - 0.2 dB. [KEIS83 / JONE88]

2- V-groove Splice : In this technique the prepared fiber ends are first butted together in a V-shaped groove. The ends are then bonded together with an adhesive or are held in place by means of a cover plate. The V-shaped channel could be either a grooved silicon, plastic, ceramic, or metal substrate. The splice loss in this method depends strongly on the fiber size and the position of the core relative to the center of the fiber (eccentricity).[KEIS83 / JONE88 / AMP82]

3- Elastic-tube Splice : In this type of splice a unique device that automatically performs lateral, longitudinal, and angular alignment is used. It splices multimode fiber with losses in the range 0.1 to 0.2 dB, but much less equipment and skill are needed in comparison to the Fusion Splices. The mechanism of this technique is basically a tube made of an elastic material, and a wide range of fiber diameters can be inserted into it. The fibers that need to be spliced do not have to be equal in diameter, which is a very good feature of this technique.

Popular Commercial Splices:

1- Square Tube : The square-tube splice is an alignment mechanism using a V-groove to achieve two points of contact. The fibers are installed into a relatively large square tube filled with epoxy. Maintaining a slight bend on the fibers as they are pushed into the tube forces their ends into a V-groove formed by the corners of the tube. The fibers are butted together and an index-matching epoxy eliminates the effects of Fresnel reflections. The splice works well with fibers nearly the same diameter. [KEIS83 / JONE88]

2- Three-Rod : One example of a three-rod splice uses three "dumbbell" -shaped rods enclosed in a collar. The collar is slightly raised in the center. The fibers fit easily between the rods. When press rings are forced onto the raised portion of the collar the rods press inward to hold the fibers. The splice is a primary alignment mechanism using the rigid rods as the first layer. The resiliency of the collar and the inward bend of the rods permit compensation for differences in fiber diameters. Here the splice loss is between 0.16 dB and 0.25 dB. [KEIS83 / JONE88 / AMP82]

Connecting Technique : A wide variety of optical fiber connectors based on different principles of operation are available. Some of the principal goals of a connector design are to have the following characteristics [KEIS83] :

- Low coupling losses even after numerous connects and disconnects
- Interchangeability with connectors of the same type
- Ease of connection
- Simple and low-cost construction
- Reliability of connection
- Low sensitivity to environmental conditions such as temperature, dust, moisture, and G-forces.

Popular Commercial Connectors:

- Watch jewel ferrule connectors
- Groove- or channel- based connectors
- Concentric sleeve connectors
- Molded connectors
- Expanded beam (lensed) connectors

3.2.1.2 Directional Coupler

Directional coupler for optical medium : To build an optical directional coupler two dielectric waveguides are brought into close proximity over a fixed distance L. The distance between them must be small enough so that each waveguide lies within the evanesent wave (the wave of constant energy density propagating in one of two adjacent electromagnetic media parallel to the interface) of the other. Normally this type of directional coupler has two inputports and two output ports, in some applications, however, only two or three of the four I/O ports are used. The two waveguides can be cylindrical optical fibers or slab waveguides. Typical coupling coefficient (K) in an optical directional coupler is 700. Given the input power to the directional coupler (P_i), the value of (K) is used to calculate the output power (P_o) as:

$$P_o = P_i \sin^2 (K L) \text{ [YARI85]}$$

Directional coupler for coaxial medium : A directional coupler is used in coaxial cables to combine or divide RF energy provided that the corresponding cables maintain the same value of the characteristic impedances. Coaxial directional coupler has three ports :

-An input port

-An output port

-A tap port

The performance of a coaxial coupler is described by its insertion loss, tap loss, isolation, anddirectivity [KRAU84]. Well designed coaxial directional couplers have a directivity of only 30 to 35 dB [LIAO85]

3.2.2 Chipsets

3.2.2.1 The Supernet Family for FDDI [Advanced Micro Devices(AMD)]

This family is composed of five chips namely Am79C81, Am79C82, Am79C83, Am7984, and Am7985. The interconnect block diagram of this family is shown in Figure 3.1, the distinctive characteristics of this family of chips are :

a- Compliant with the proposed ANSI X3T9.5 (Fiber Distributed Data Interface, FDDI specification)

- 100 Mbps data rate
- Fiber optic transmission media
- Ring topology
- Timed token passing protocol

b- CRC generator / checker

c- Diagnostics features

- Multiple loopback modes for run time diagnostics
- Accumulates network management status information

d- Supports Master and Slave system interfaces

e- Complete memory management

- Supports 256 Kbytes of local frame buffer memory
- Link list transmit frame structure
- Supports up to 200 Mbps dual port memory access

The block diagram of CMOS RAM Buffer Controller (RBC) Am79C81 chip is shown in Figure 3.2, its distinctive characteristics are;

a- Total memory buffer management

-16 bit address bus supports 64 Kwords (32 bits wide) with the Am79C82 Data Path Controller (DPC)

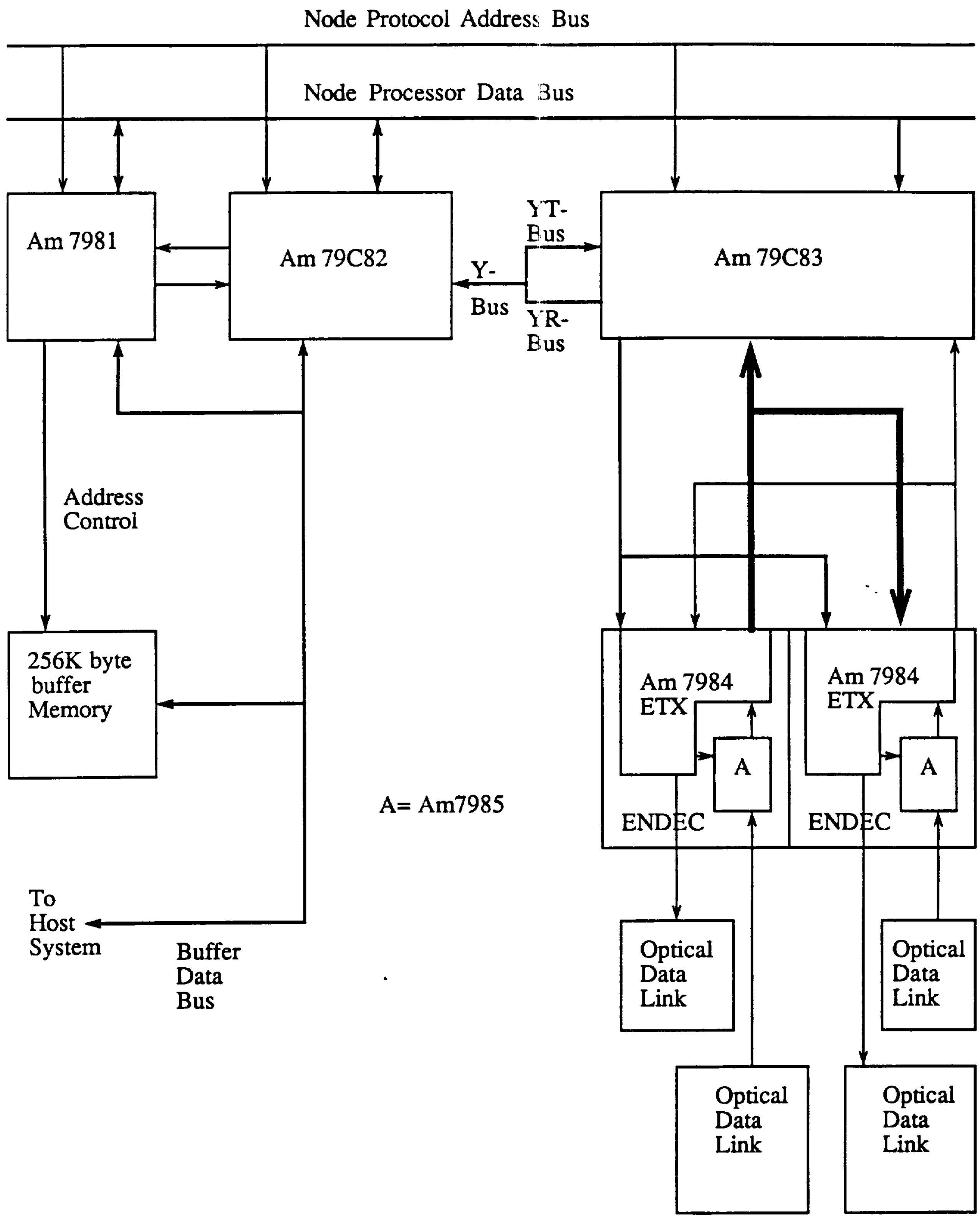


FIGURE 3.1 BLOCK DIAGRAM OF SUPERNET FAMILY

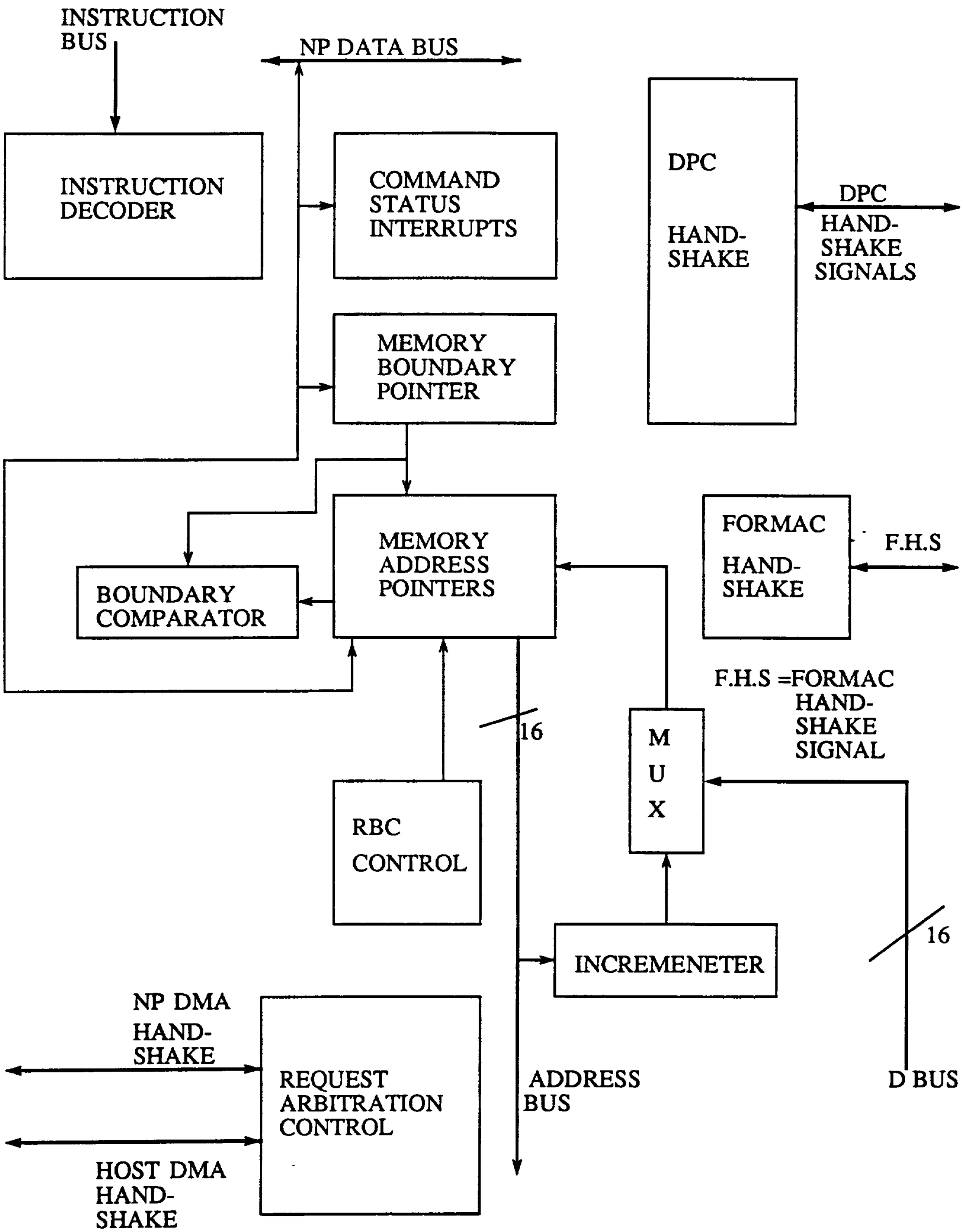


FIGURE 3.2 BLOCK DIAGRAM OF Am 79C81 CHIP [AMD]

- Programmable registers and pointers

- Memory full and empty notification

- DMA arbitration between the Data Path Controller (DPC), Node Processor (NP), and Host

b- Supports transmit link list addressing

c- 12.5 MHz byte clock

d- TTL compatible I/O

e- Single +5 V power supply

f- 145 lead pin grid array package

The block diagram of CMOS Data Path Controller (DPC) Am79C82 chip is shown in Figure 3.3, its distinctive characteristics are :

a- Preforms reception and transmission of frames

b- Byte (8 + 1 bits) to word (32 + 4 bits) conversions

c- Reports error status

d- Performs parity check and generation

e- 12.5 MHz byte clock

f- 145 lead pin grid array package

g- Single +5 V power supply

The block diagram of Fiber Optic Ring Media Access Controller (FORMAC) Am79C83 chip is shown in Figure 3.4, its distinctive characteristics are :

a- Implements Media Access Control (MAC) layer protocol for the ANSI X3T9.5 standard (Fiber Distributed Data Interface, FDDI)

b- Perform frame reception, transmission, repetition, and removal

c- Error detection capability

- Cyclic redundancy checking and generation

Note: To detect serious FDDI ring faults, the network nodes continuously transmit beacon frames, yielding to upstream nodes. If there is a physical break in the FDDI ring , all nodes

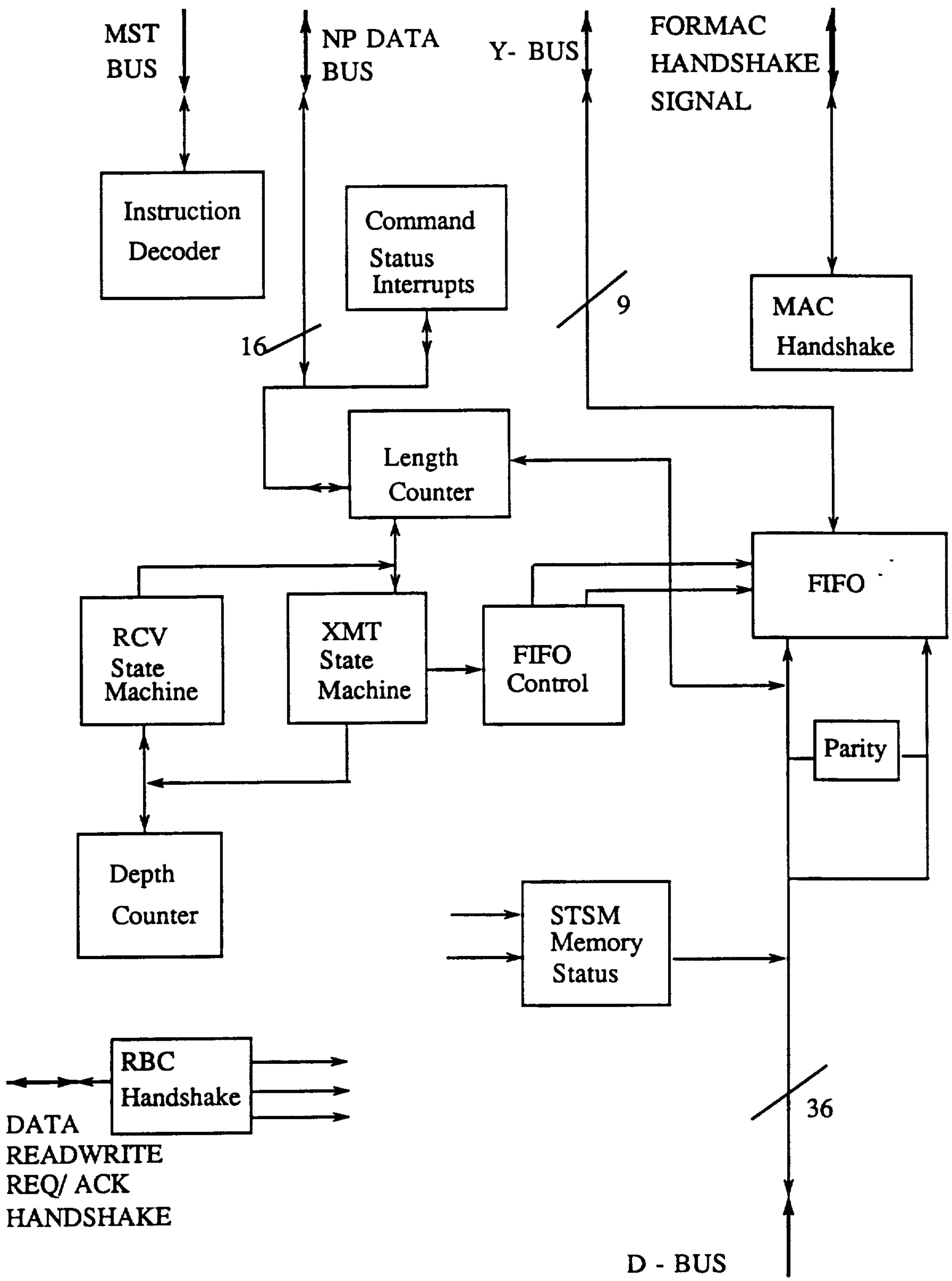


FIGURE 3.3 BLOCK DIAGRAM OF Am79C82 CHIP [AMD]

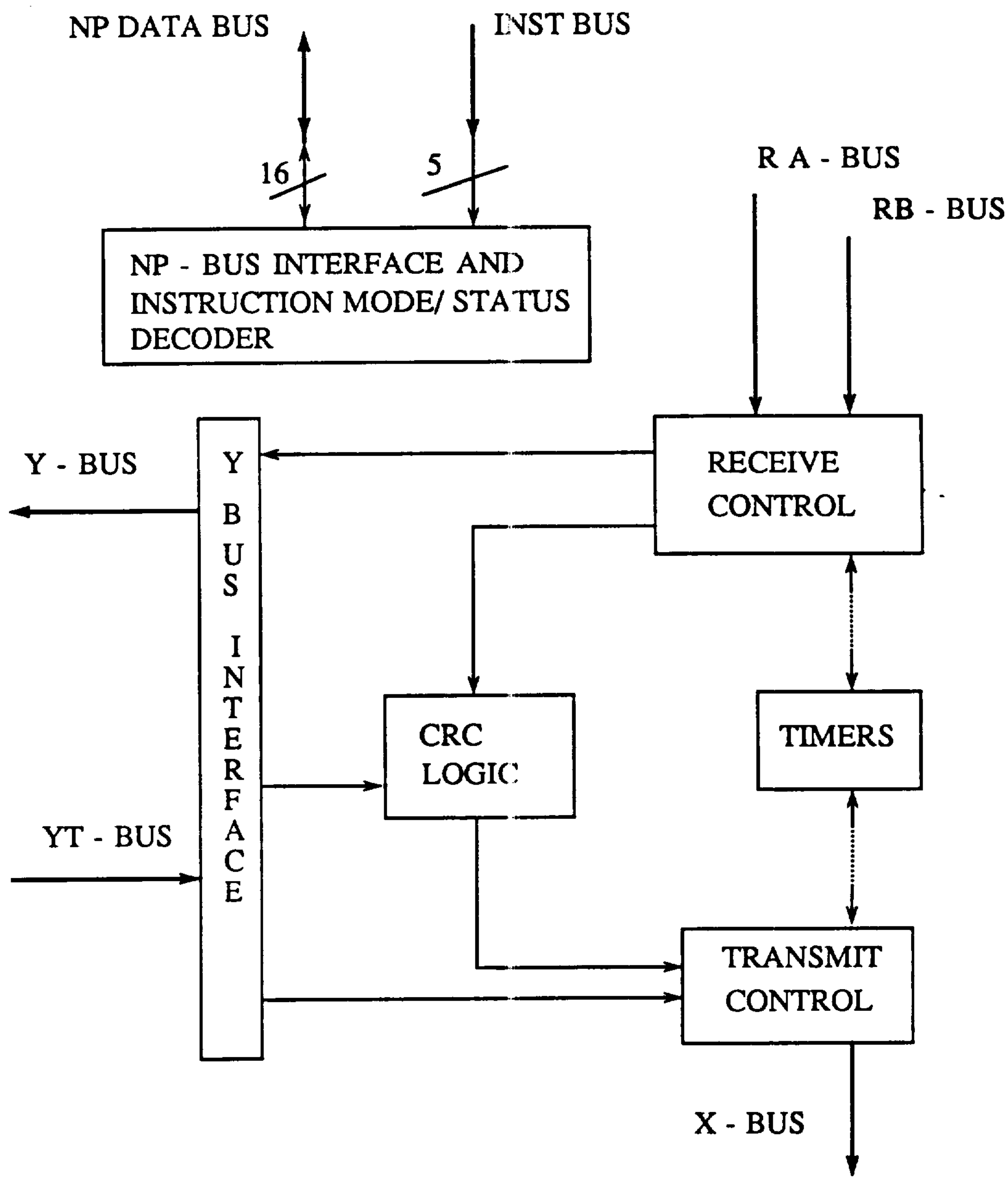


FIGURE 3.4 BLOCK DIAGRAM OF Am79C83 CHIP [AMD]

will receive a continuous stream of beacon frames from the node immediately downstream from the fault and ring recovery procedures would follow. [08/16/88 TELEPHONE CONVERSATION WITH MS. AMY CHANG FROM AMD INC.].

- Token claiming and beacon modes

d- Diagnostics Features

- Four loopback modes
- Status bit collection

e- Token management

f- Supports data rates up to 100 Mbps

g- Single +5 V power supply

The block diagram of ENDEC Transmitter (ETX) Am7984 chip is shown in Figure 3.5, its distinctive characteristics are :

a- Implements 4B /5B encoding as specified by the ANSI X3T9.5 (Fiber Distributed Data Interface, FDDI) standard

b- 100 Mbps, 125 Mbaud serial output

c- Byte clock and nibble clock output

d- Selectable loopback and repeat modes

e- Line state decoder

f- Repeat filter

g- Single + 5 V power supply

h- 84 pin PLCC and LCC packages

The block diagram of ENDEC Receiver (ERX) Am7985 chip is shown in Figure 3.6, its distinctive characteristics are :

a- implements 4B / 5B decoding as required by the ANSI X3T9.6 (Fiber Distributed Data Interface, FDDI) standard

b- 100 Mbps, 125 Mbaud serial input

c- Clock recovery

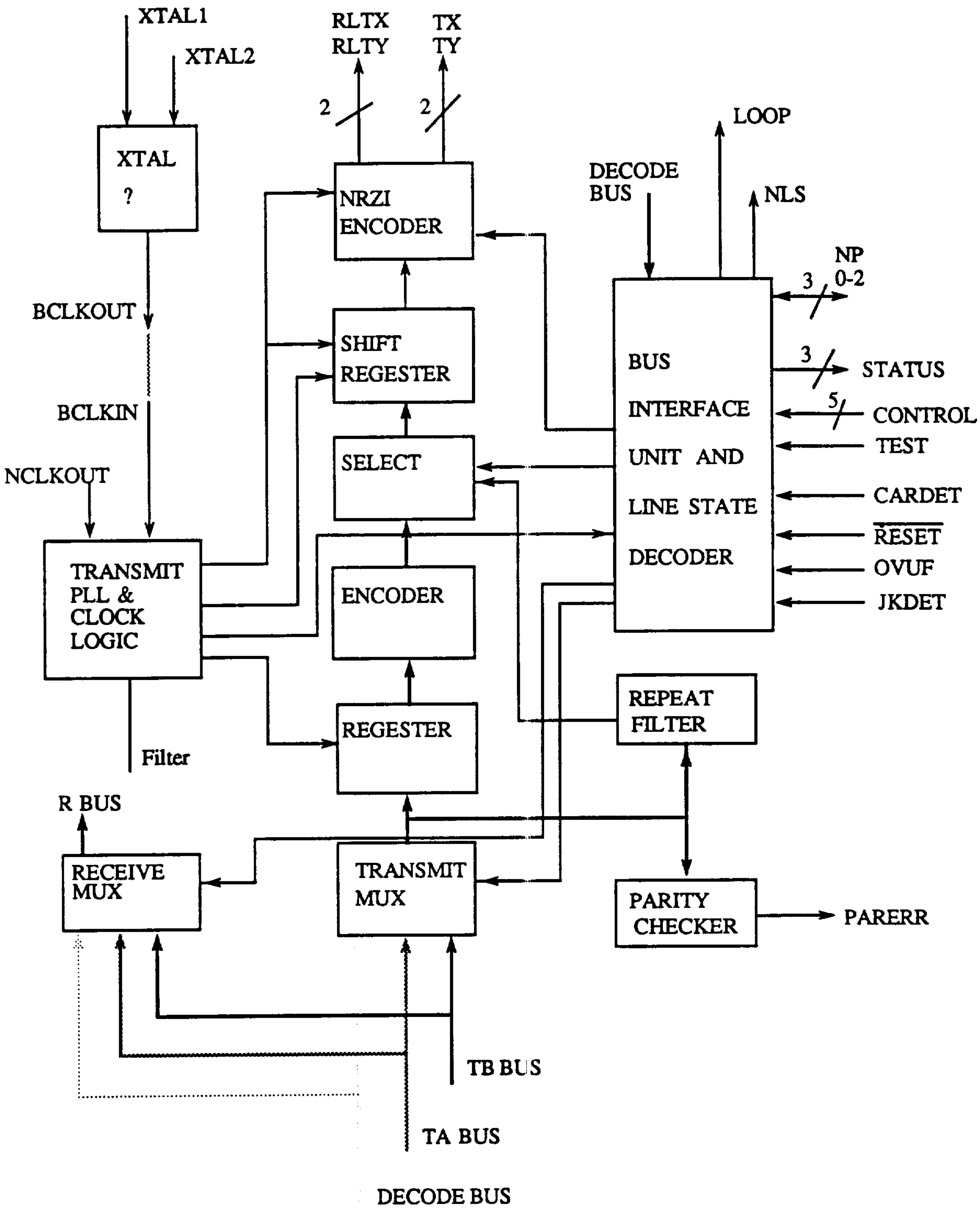


FIGURE 3.5 BLOCK DIAGRAM OF Am7984 CHIP [AMD]

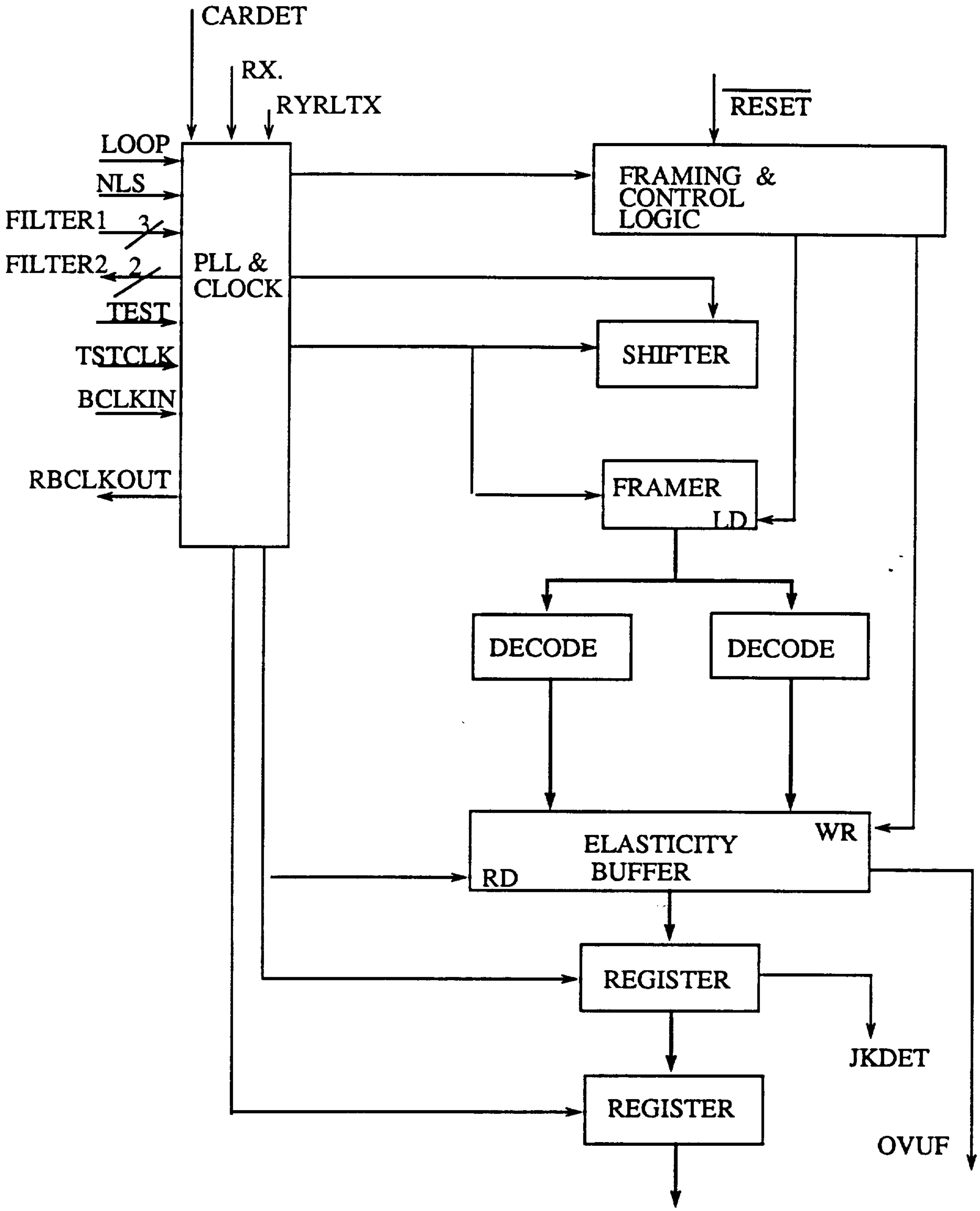


FIGURE 3.6 BLOCK DIAGRAM OF Am7985 CHIP [AMD]

d- Decodes data with up to 3.0 nanoseconds jitter

e- Selectable loopback mode

f- Internal elasticity buffer to compensate for clock mismatch

g- Single +5 V power supply

h- 44 pin PLCC and LCC packages

3.2.2.2 Local Network Controller (LENT) R68802 chip [Rockwell]

The R68802 Local Network Controller implements the IEEE 802.3 CSMA / CD standard. It supports Ethernet (10BASE5), Cheapernet (10BASE2), and StarLAN (1BASE5) implementations of this standard. The basic function of the LNET is to execute the CSMA / CD algorithm, perform parallel-to-serial and serial-to-parallel conversions for data streams up to 10 Mbps, and assemble and disassemble the packet format. In addition, the LNET provides an 8-bit or 16-bit processor interface, the required DMA interfaces, and the proper interface to the Manchester Code Converter (MCC) used to connect the LNET to an IEEE 802.3 defined Media Attachment Unit (MAU). The block diagram of this controller is shown in Figure 3.7, and its main features are :

a- Meets the IEEE 802.3 specifications for local networks (e.g., Ethernet, Cheapernet and StarLAN)

b- Serial data rates as high as 10 Mbps

c- Compatible with a variety of 8- or 16- bit processors and DMA controllers

d- Interfaces to a variety of manchester code converters

e- Programmable interframe wait times for smaller topologies and lower data rates

f- CSMA / CD algorithm :

- Wait before transmit
- Jam on collision
- Binary exponential backoff

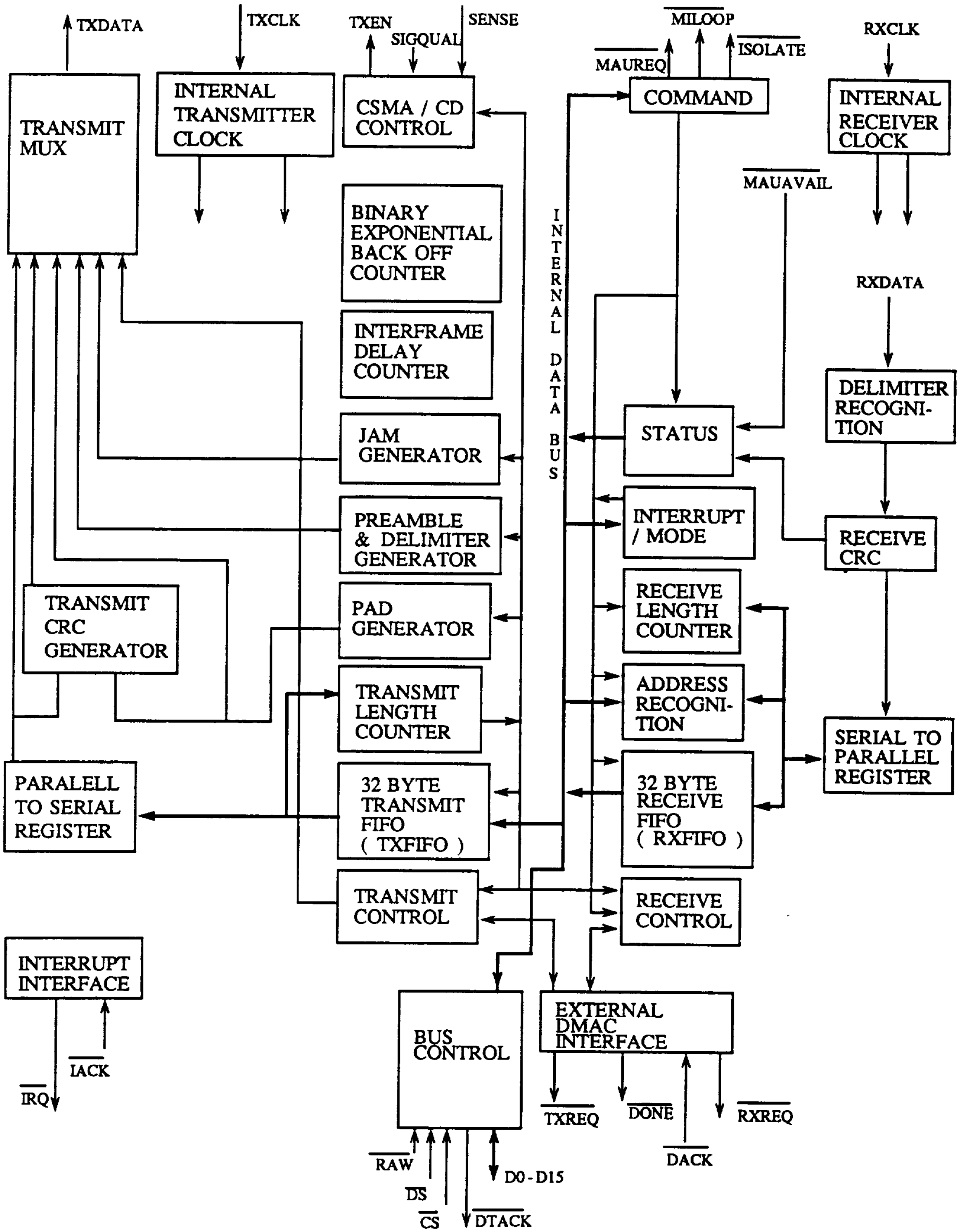


FIGURE 3.7 LNET BLOCK DIAGRAM [ROCKWELL]

g- Programmable 2- or 6- byte address recognition

h- Supports loopback self-test

i- Extensive network management capabilities

j- Programmable disable on reception

k- Programmable collision handing minimizing CPU intervention

l- 32 bit CRC generation and reception

m- Broadband applications

n- 32 byte FIFO on both transmitter and receiver

o- TTL compatible I/O

p- 40 pin DIP

q- Single +5 V power supply

3.2.2.3 Ethernet Serial Interface 82501 [Intel]

The distinctive features of this chip are the folowing;

a- It is compatible with IEEE 802.3 / Ethernet and Cheapernet Specifications.

b- 10 Mbs operation.

c- Replaces 8 to 12 MSI components

d- Manchester Encoding / Decoding and Receive Clock Recovery.

e- 10 MHz Transmit Clock Generator.

f- Driving / Receiving IEEE 802.3 Transceiver Cable.

g- Fail-Safe Watchdog Timer Circuit to prevent continuous Transmissions.

h- Diagnostic Loopback for Fault Detection and Isolation

i- Directly Interfaces to the Intel's 82586 LAN coprocessor.

3.2.2.4 Ethernet Transceiver 82C502 [Intel]

The distinctive features of this chip are the following;

a- Conforms to IEEE 802.3, Ethernet Rev.2, and Cheapernet standard as

- Anti- jabber function

- Receiver based collision detection.

- Signal Quality Error (heartbeat) test.

- Supports redundant jabber timer.

b- Requires minimum boardspace

- On chip voltage refrence.

- 16 pin DIP.

c- No external adjustments required.

d- Reliable CHMOS technology.

3.2.2.5 Local Area Network Coprocessor 82586 [Intel]

The distinctive features of this chip are the following

a- Performs Complete CSMA / CD Data Link Functions without CPU Overhead

- High level command interface

b- Supports Established and Emerging LAN Standareds

- IEEE 802.3 / Ethernet

- IEEE 802.3 / Cheapernet

- IBM PC Network (2 Mbps Broadband)

- 1 Mbps Networks

d- On Chip Memory Management

- Automatic buffer changing saves memory

- Reclaim of buffers after receipt of bad frames

- Save bad frames

e- Interfaces to 8 - bit and 16 - bit Microprocessors

f- Supports Minimum Component Systems

- Shared bus configuration

- No TTL interface to iAPX 186 and 188 microprocessors

g- Supports High Performance Systems

- Bus master, with on - chip DMA
- 4 MBytes/second bus bandwidth
- Compatible with dual port memory
- Back to back frame reception at 10 Mbps

h- Network Diagnostics

- Frame CRC error tally
- Frame alignment error tally
- Location of cable opens / shorts
- Collision tally

i- Self Test Diagnostics

- Internal loopback
- External loopback
- Internal register dump
- Backoff timer check

3.2.2.6 Single Chip LAN controller 82588 : High Integration Mode / 82588 - 5 : High Speed Mode [Intel]

The distictive features of these chips are the following

a- Integrates ISO Layers 1 and 2

- CSMA / CD Data Link Controller
- On chip Manchester, NRZI Encoding / Decoding
- On chip Logic based Collision Detect and Carrier Sense

b- Supports Emerging IEEE 802.3 standards

- 2 Mbps Broadband
- 1 Mbps Baseband

d- High level command intrface offloads the CPU

e- Efficient memory use via Multiple Buffer Reception

f- User Configurable

- Up to 2 Mbps Bit rates with on chip Encoder / Decoder (High Integration Mode)
- Up to 5 Mbps with External Encoder / Decoder (High Speed Mode)

g- No TTL Glue required with iAPX 186 and 188 microprocessors

h- Network Management and Diagnostics

- Short or Open Circuit localization
- Station Diagnostics (External loopback)
- Self test Diagnostics, Internal loopback, User readable register

3.3 Other Considerations

Depending on the transmission medium and topology the implementation of other hardware items needs to be considered thoroughly. Such items are discussed below.

3.3.1 Repeaters

The repeater is used to boost the energy level at different locations in the utilized network. It is a device that amplifies and repeats its input, the repeater has the ability to "clean" its input before sending it out to the rest of the system. If amplification is performed in the repeater, then one amplifier is needed for each direction of signal propagation in a bidirectional system (total of two amplifiers). Only one amplifier is needed for unidirectional system. Unfortunately, in addition to the desired signal, the repeater amplifies and re-transmits noise and collisions. A failure in a certain repeater will result in a "break" in the topology utilizing it, such a break could be fatal if the topology in use was a ring topology, for example.

3.3.2 Fiber Optic Couplers

An optical coupler is used in a different way. It may be used to combine light from two fibers into one, or split light from one fiber into two. Also couplers may be used to combine different wavelengths for the purpose of transmission over a single fiber (multiplexing) or to separate wavelengths transmitted over a single fiber into individual fibers (demultiplexing). Coupling efficiency is one parameter that must be considered when dealing with coupling. Two types of couplers are of interest here:

Fiber optic Star coupler: A star coupler is a passive mixing element, the optical power from the input ports are mixed together and then divided equally among the output ports. Star couplers are of two kinds, see Figure 3.8:

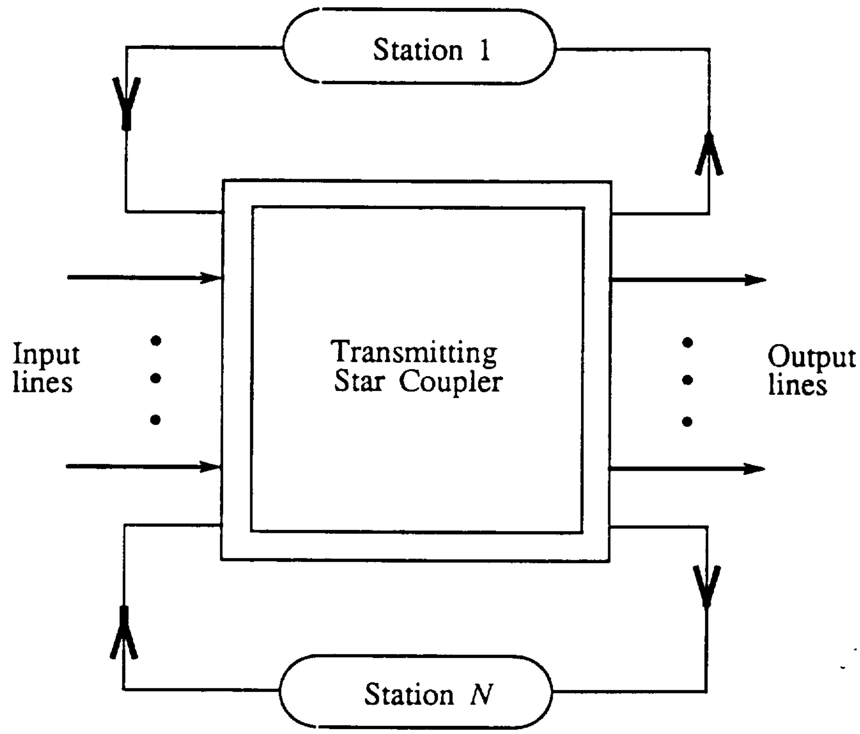
(i) Reflection.

(ii) Transmission.

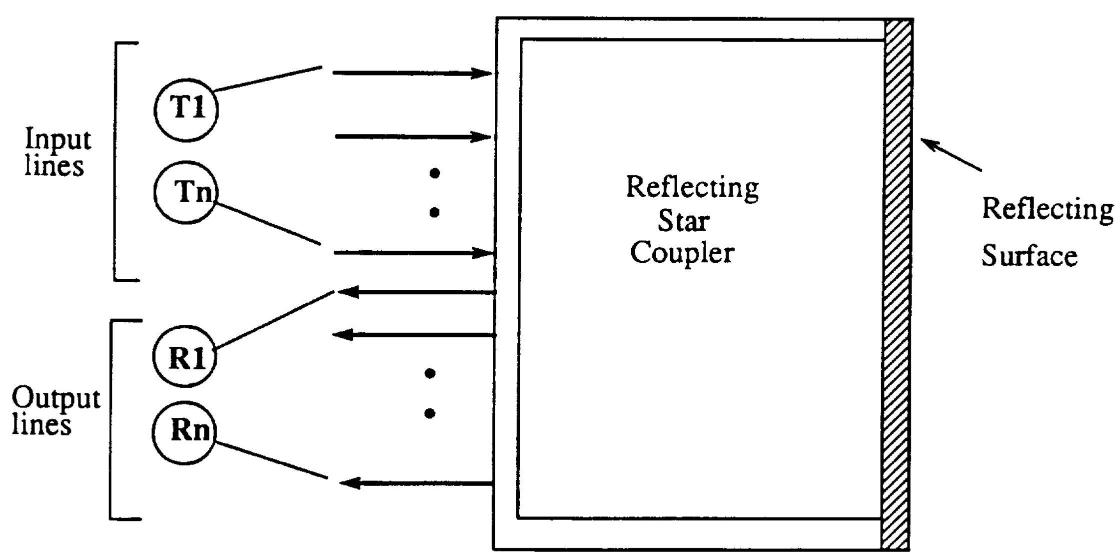
A portion of the light which enters the reflection star coupler is injected back into the input fiber, therefore for a given number of input and output ports the transmission star coupler is twice as efficient as the reflection star coupler. The reflection star coupler is more versatile, however, because the relative number of the input and output ports may be selected or varied after the device has been constructed. On the other hand transmission star coupler will have the number of the input and output ports fixed by initial design and fabrication.

Fiber optic T-coupler: A T-coupler could be passive or active . An active T- coupler shown in Figure 3.9. In this type of coupler the photodiode receiver converts the optical energy, flowing in the data bus, into an electric signal and the processing element can remove or copy part of the electric signal while the remainder of the energy is forwarded to the optical transmitter where the electric signal is reconverted to optical energy and re-transmitted on the optical bus.

The processing element has the ability of adding energy to the main signal also. This kind of coupler can easily be constructed by using photodiodes and light sources. A passive T- coupler is shown in Figure 3.10. This kind of coupler is used to remove a portion of the



(a)



(b)

FIGURE 3.8 OPTICAL STAR COUPLERS
(a) TRANSMITTING
(b) REFLECTING

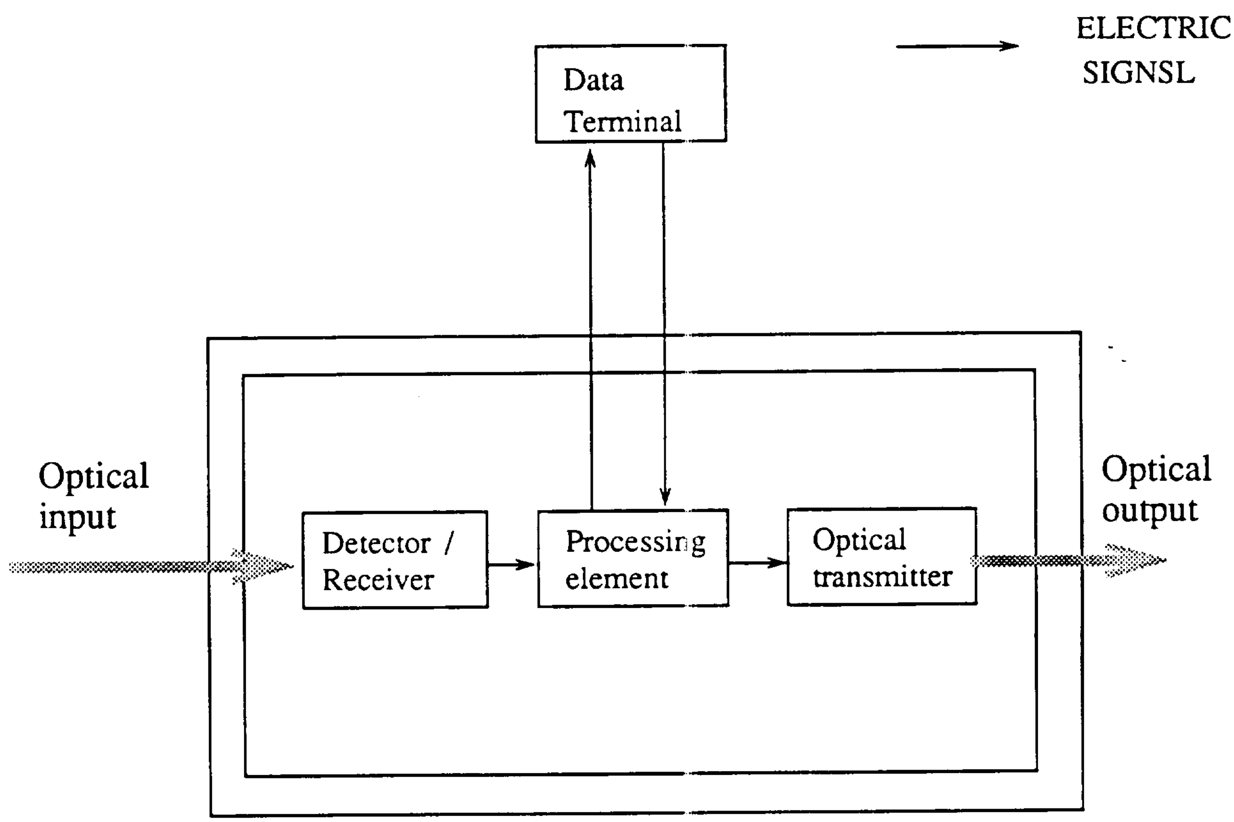


FIGURE 3.9 OPTICAL ACTIVE T-COUPLER

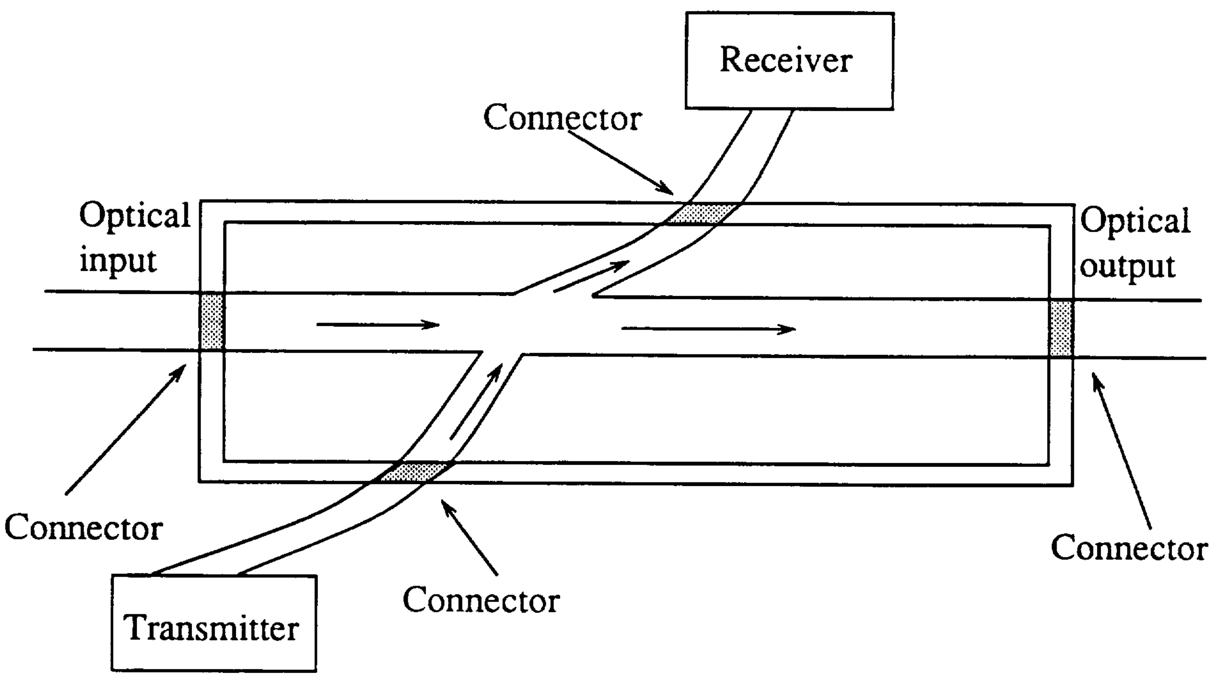


FIGURE 3.10 OPTICAL PASSIVE T-COUPLER

optical energy from the optical fiber bus, or on the other hand to inject additional light energy onto the optical bus. This process will affect the power budget of the network. Due to this fact the number of terminals between taps is limited to a small number (e.g. less than or equal to ten terminals). [KEIS83 / YARI85]

3.4 Recommendations

Analog optical fiber systems are generally used for short unrepeated video links but most optical fiber applications are in digital transmission with simple on-off modulation. The distamce between repeaters is based on the wavelength being used . For example, as shown in Figure 3.11, the maximum distance between repeaters is limited by the signal attenuation at low data rates and by the dispersion at high data rates[JONE88]. The basic attenuation mechanisms in a fiber are absorption, scattering, and radiation losses of the optical energy. On the other hand dispersion, which is the spreading in a pulse as a function of wavelength and it is measured in nanosecond/Km/nanometer. In present technology losses from tapping a fiber optic medium will severely limit the number of nodes a system can utilize and accommodate. However, methods of extracting energy from an optical fiber do not always cause such limitations, for example the method suggested by Shelby, Levenson, and Perlmutter of IBM [SPEC88] which is shown in Figure 3.12. In this technique, a krypton-ion laser directs light at two red wavelengths, 647 and 676 nanometers down the same optical medium. When light is intense the silica fiber behaves in a non-linear fashion and its index of refraction varies with the amplitude of the light. This variation modifies the phase of a probe beam passing through the fiber medium, thus the amplitude fluctuations due to both the signal and noise on the signal beam modulate the phase of the probe beam and vice versa. By measuring the phase fluctuations of the probe beam with a phase shifting cavity the amplitude of the signal beam can be deduced to an accuracy better than that allowed by conventional methods. Present fiber taps are more difficult to make and are more expensive than taps for coaxial cables. This is a technological problem which could possibly be over-

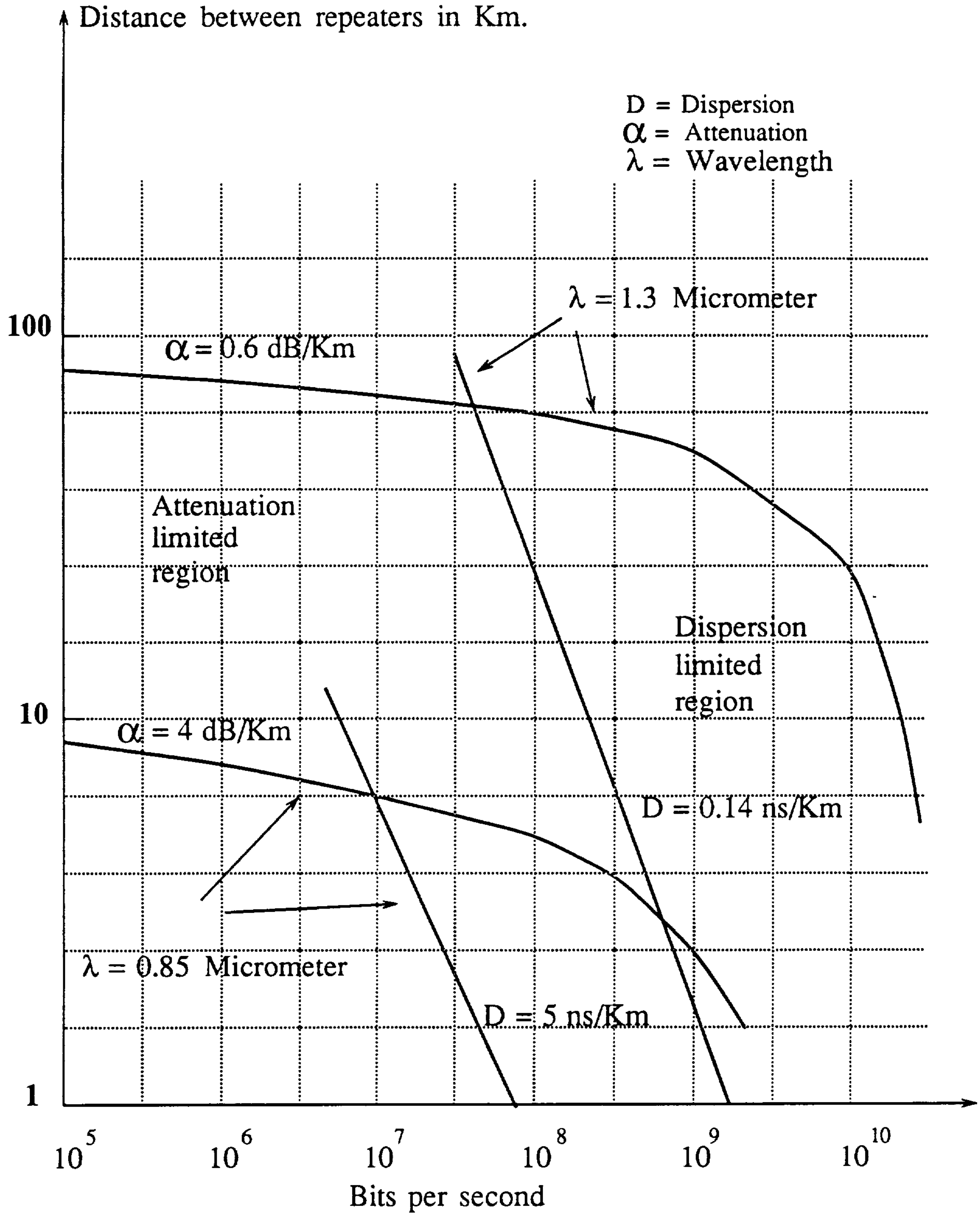


FIGURE 3.11 DISTANCE BETWEEN REPEATERS IN OPTICAL SYSTEM

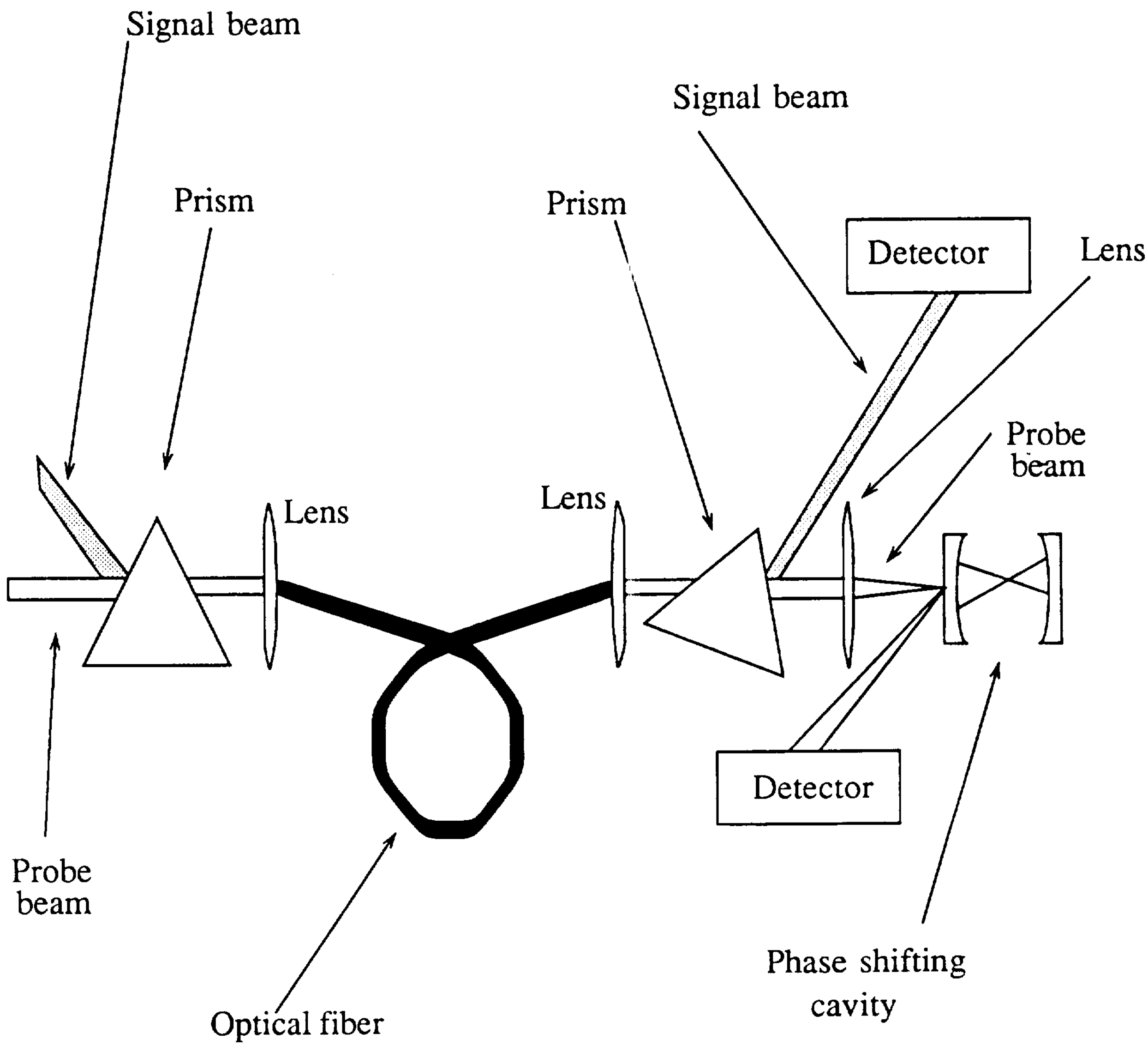


FIGURE 3.12 SHELBY, LEVENSON, AND PERLMUTTER METHOD

come. The availability of high data rates and high bandwidth in fiber optic medium in comparison to that of the coaxial medium makes the medium attractive for many applications, see Table 3.4. In this table a comparison is made between different networks taking in to consideration the data rate and the medium, indeed many standard LANs have implemented and utilized optical fiber medium, because this medium is superior in its data rate and its EMP and noise immunity. Carefully taking the state-of-the-art in to consideration, it is possible to reach a decision to implement and utilize optical fibers in this part of the system topology. This is because the number of immediate nodes interfaced with the backbone is limited, and every point-to-point link in the system topology can be of optical fiber. In general for bidirectional networks or sub-networks two links of fiber are needed and one link of fiber is needed for unidirectional networks or sub-networks. Optical fibers support a large bandwidth-distance product, they occupy smaller physical space as compared with other media. Near a wavelength of 1.55 micrometer optical fiber suffers from a 0.2 dB/Km loss. On the other hand coaxial cable medium is a versatile transmission medium. Digital and analog signaling techniques are possible in this medium and the maximum rate of data transmission in this medium is stated in the literature as being 50 Mbps. The maximum range covered using this medium is stated to be 10's of Km, while the practical number of nodes this medium can accommodate is stated to be in the 1000's when using a 75 ohm coaxial cable and digital signaling or analog signaling with FDM. [DALY84 / KEIS83 / GEOR82 / AMP82].

3.4.1 Justification by Pros and Cons

In Sections 2.6.1.1, 2.6.1.2, and 2.6.1.3 three different topologies were suggested for three cases, according to these topologies the corresponding transmission medium should be selected. If the suggested topology was a bus topology, then both twisted pair wire and coaxial cables are appropriate as a transmission medium. Optical fiber medium can be selected for bus topology if it is cost effective. Twisted pair wire medium, baseband coaxial cable medium, and optical fiber medium are suitable for ring or star topologies. The high data

TABLE 3.4 COMPARISON BETWEEN POPULAR LOCAL AREA NETWORKS		
NETWORK	DATA RATE	CABLING
ASCII	19.2 Kbps	Twisted Pair
IBM 5251	1.0 Mbps	Twin Ax.
PC NETWORK	2.0 Mbps	75 ohm Coax
IBM 3278/9	2.35 Mbps	92 ohm Coax
TOKEN RING	4.0 Mbps	Dual Twisted Pair Shielded Data Cable
IEEE 802.4 MAP (G M)	5/10 Mbps	75 ohm Coax or Fiber Optic
IEEE 802.3 ETHERNET	10 Mbps	50 ohm Coax
ADVANCED TOKEN RING	16 Mbps	Fiber Optic
IEEE 802.6	50 Mbps	Fiber Optic
ANSI FDDI	100 Mbps	Fiber Optic

rate that a coaxial cable or a fiber optic medium can handle may overwhelm the central node switching capabilities in a star topology using either coaxial cable or optical fiber as a transmission medium.

3.4.2 Conclusions

In order to decide which type of hardware to use many factors must be considered. Mainly the type of topology used, access type, data rate, maximum number of nodes, and the environment in which the hardware will be utilized are the main factors. Fiber optics supports the highest data rate and bandwidth as well as having excellent EMP and noise immunity and space/weight saving characteristics. Most of the commercially available optical cables are made to withstand severe environments. For instance, Young's modulus of fused silica glass and jacket material which are used to manufacture optical fiber cables is 65 Gpa and 20 - 500 Mpa respectively (Steel's Young modulus is $2x10^4$ MPa). In addition, at Bell laboratories, current research is going on to develop photonic switches which manipulates photons instead of electrons. Systems using photonic switches and optical fiber will result in Gbit systems in near future [COM87 / AT&T lab. Telepone Conversation 06/19/88].

One advantage gained from using coaxial cable instead of a twisted pair wire as a transmission medium is the outstanding bound on the maximum number of nodes that this medium can support. Coaxial cable can be utilized for baseband and broadband signaling, also taping energy from this medium does not generate hardware complexity as the case is for optical fiber medium. Twisted pair medium supports lower data rate and it's immunity to EMP and noise is weaker in comparison to both optical fiber and coaxial cable. Based on the chosen medium and corresponding protocol a set of chips must be chosen. This chip set must support the desired protocol for the selected medium.

Chapter 4 Protocols

4.1 Tutorial

"Protocol" is a term used for the set of ru.es and procedures governing communications in a network. While "protocol" can refer to the communications between two ISO layers, for this report, the term will be used primarily in regard to communications between two terminals, that is, communications taking place from the OSI network lay downward. Table 4.1.1 1z5] explains the OSI layers.

Although topologies for which each type of protocol is particularly suited may be given, this is not meant to imply it is used exclusively on those topologies.

4.1.1 Command/Response Protocols

As the name implies, in networks using a command/response protocol, terminals communicate only when they are "commanded" to by a controller.

Command/response protocols are particularly well suited to star topologies, although they often used with bus configurations, and even in some ring networks. An example of a command response protocol is MIL-STD-1553B, a protocol developed for the military.

4.1.2 Contention Protocols

In a contention protocol, terminals operate asynchronously, transmitting whenever they are ready. If one or more terminals begin transmission when another is using the transmission medium, a "contention" will occur, and all terminals involved will be unsuccessful in their transmissions.

Most contention protocols employ CSMA/CD (Carrier Sense Multiple Access with Collision Detection). "CSMA" implies that each terminal monitors the transmission medium and only tries to transmit when the medium is clear. Contention occurs only when one terminal initiates transmission, and before it transmission can propagate down the medium to be detected, one or more additional terminals begin transmitting. "CD" implies that the termi-

Table 4.1.1 The OSI Layer Network Model [STALL85]

Layer	Description
Application	Provides access to the OSI environment for users and also provides distributed information services
Presentation	Provides independence to the application processes from differences in data representation (syntax)
Session	Provides the control structure for communication between applications; establishes, manages, and terminates connections (sessions) between cooperating applications
Transport	Provides reliable, transparent transfer of data between end points; provides end-to-end error recovery and flow control
Network	Provides upper layers with independence from the data transmission and switching technologies used to connect systems; responsible for establishing, maintaining and terminating connections
Data Link	Provides for the reliable transfer of information across the physical link; sends blocks of data with the necessary synchronization, error control and flow control.
Physical	Concerned with transmission of unstructured bit stream over physical medium; deals with the mechanical, electrical, functional, and procedural characteristics to access the physical medium

nals detect the collision. Most CSMA/CD protocols require that, on detection of a collision, the terminals involved must wait a (usually random) time period before retransmitting.

Contention protocols are generally used for bus topologies. Some star networks also use contention. ALOHA and Ethernet are examples of contention protocols.

4.1.3 Time Division Multiplex Protocols

In time division multiplex protocols, each terminal is scheduled a time period when it can use the transmission medium, and it will transmit its data only then.

The most common type of time division multiplex protocol is the slotted ring. Used (obviously) on ring topologies, in slotted ring, one or more "slots" or "frames" travel about the ring, preceded by a header to indicate their passing. Each terminal is allotted a certain amount of time in the frame. When a frame is passing through a terminal, it reads any data addressed to it, and puts any data it must send into the frame.

Most time division multiplexing protocols are referred to as "virtual" or "implicit" token-passing, and so for the remainder of this report, time division multiplexing protocols will be considered a class of token-passing protocols.

4.1.4 Token Passing Protocols

In this type of protocol, use of the transmission medium is granted by possession of a "token". The token is usually a bit pattern (such as a long string of ones) rarely occurring in data. If a terminal possesses the token, when it has data to send, it removes the token from the network, and begins transmitting. When the terminal is finished transmitting, or when it receives an acknowledgment that its data was received at the proper address (depending on the particular token passing protocol), it places the token back onto the network.

In some token-passing networks, the token is not passed by means of a control message, but rather a set of conditions of the network and various internal timers. Each terminal monitors the network and its timers, and when the proper set of conditions arise, a terminal "knows" that it is in possession of the token. This is known as "implicit" or "virtual" token passing.

Token passing is used almost exclusively on ring topologies, with one or more tokens traveling around the ring, being passed from terminal to terminal. Some busses also use token passing, where the token is placed on the bus to be removed by a terminal ready to transmit. ProNet and FDDI are examples of a token passing protocols.

4.1.5 Hybrid Protocols

In order to take advantage of the features of two or more of the basic protocols described above, hybrid protocols were developed. For example, in a command/response protocol the controller may grant permission to transmit by use of a token, or a network may have a controller to decide which terminal gets to transmit in case of a contention.

In simulation and in practice, some of the more promising hybrids have been contention/token passing protocols. A hybrid of token passing and contention protocols was simulated by Gopal and Wong at the University of Waterloo. [GOPA84] In their hybrid, a token is passed logically from terminal to terminal. The token does not grant control of the transmission media, however. Terminals use a CSMA/CD type protocol, and the token comes into play only in case of a contention, in which case, if one of the terminals involved in the collision possesses the token, it can retransmit immediately while the others have to wait.

L-Expressnet is a contention/token-passing protocol that is commercially available. L-Expressnet has a time known as the "scheduling period" in which when in possession of an implicitly passed token, a terminal may transmit, guaranteed of no collision. The scheduling period is then followed by a "contention period" in which any terminal may transmit at risk of collision. Hyperchannel is another example of a hybrid protocol.

4.2 Command/Response Protocols

In this section, we will discuss the MIL-STD-1553B and MIL-STD-1773 command response protocols. Both were designed for use on bus topologies, although the MIL-STD-1773 is also particularly well suited for a star topology.

4.2.1 MIL-STD-1553B

The multiplex data bus system functions asynchronously in a command/response mode, and transmission occurs in a half-duplex manner, that is, data may travel in either direction on the bus, but not in both directions at the same time. A terminal called the bus controller controls all information transfer on the bus, and is the only terminal which may initiate a transmission. The information flow on the data bus is comprised of messages which are in turn, formed by three types of words: command, data, and status.

4.2.1.1 MIL-STD-1553B Word Types

4.2.1.1.1 The Command Word

The Command word is comprised of a sync waveform, a remote terminal address field, a transmit/receive (T/R) bit, a subaddress/mode field, a word count/mode code field, and a parity (P) bit, as shown in Figure 4.2.1. The command sync waveform is an invalid Manchester waveform at least three bit times wide. The sync waveform is positive in the first half of the field, and then negative for the rest of the field, as shown in Figure 4.2.1.

The next five bits of the command word after the sync waveform are the remote terminal (RT) address. Each RT has a unique address, but all RTs possessing the broadcast option may also be addressed by placing 31 (11111 binary) in the address field. Each remote terminal is responsible to respond when its unique address is transmitted as part of a command word on the data bus by the bus controller. Systems using the broadcast option may not use 31 as the unique address of a remote terminal.

After the terminal address comes the transmit/receive, or T/R bit, which indicates the

BIT TIMES

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20

COMMAND WORD

	REMOTE TERMINAL ADDRESS 5	T/R 1	SUBADDRESS/MODE 5	DATA WORD COUNT MODE CODE 5	P 1

DATA WORD

	DATA 16	P 1

STATUS WORD

	REMOTE TERMINAL ADDRESS 5	ME 1	I 1	SR 1	RESERVED 3	BC 1	BU 1	SF 1	DB 1	TF 1	P 1

ME: MESSAGE ERROR
I: INSTRUMENTATION
SR: SERVICE REQUEST
BC: BROADCAST COMMAND RECEIVED
BU: BUSY
SF: SUBSYSTEM FLAG
DB: DYNAMIC BUS CONTROL ACCEPTANCE
TF: TERMINAL FLAG

FIGURE 4.2.1 MIL-STD-1553B WORD FORMAT

action required of the RT. A logic high (1) instructs the RT is to receive, and a logic low (0) instructs the terminal to transmit.

The next five bits after the T/R indicate an RT subaddress or use of optional mode control. The subaddress/mode control values of 00000 and 11111 are reserved for optional mode control. The mode code control is used only to communicate with the multiplex bus related hardware and to assist in the management of information flow. It is not used to extract or feed data to a functional control subsystem. Optional subaddress/mode code of 00000 and 11111 imply that the contents of the word count field are to be decoded as a five bit mode command.

The next five bits following the subaddress/mode control are used to specify the number of data words to be either sent or received by the RT or the optional mode code as specified in the previous paragraph. A maximum of 32 data words may be transmitted or received in any one message block. The field contains the binary expression of the number of data words to be transmitted, unless that number is 32, in which case the field contains all zeroes.

The last bit is the parity check bit for the preceding 16 bits. Odd parity is used.

4.2.1.1.2 The Status Word

A status word is comprised of a sync waveform, an RT address, a message error bit, and instrumentation bit, a service request bit, three reserved bits, a broadcast command received bit, a busy bit, a subsystem flag bit, a dynamic bus control bit, a terminal flag bit, and a parity bit as shown in Figure 4.2.1.

The status word sync waveform is the same as the command word sync. The next five bits contain the address of the terminal transmitting the status word.

The ninth status word bit is used to indicate that one or more of the data words associated with the preceding receive command word from the bus controller has failed to pass the RT's validity test. Logic one indicates an error, and logic zero indicates a valid message.

The tenth status word bit is the instrumentation bit, and is always logic 0. This bit distinguishes the status word from a command word, whose tenth bit is always logic 1.

The eleventh status word bit is the service request bit, and its use is optional. When used, this bit indicates the need for the bus controller to take specific pre-defined actions relative to either the RT or associated subsystem. If one or more subsystems interfaced to a single RT requires service, the service request bit is set to logic 1, and a separate data word is needed to identify the specific requesting subsystem. The service request bit is intended to be used only to trigger data transfer operations which take place on an exceptional, rather than periodic basis. Logic 0 indicates no need for service. If this function is not used, the eleventh bit is set to 0.

Bits twelve through fourteen are not currently used, and are set to zero.

The fifteenth status word bit is used to indicate whether or not the preceding valid command word was a broadcast command. If the command word was a broadcast command, this bit is set to 1, and if not, it is set to 0. If the broadcast option is not used, this bit is always set to 0.

The sixteenth bit is the busy bit, and its use is optional. When used, this bit indicates that the RT or subsytem is unable to move data to or from the subsystem in compliance with the bus controller's command. A logic one indicates a busy condition and a logic zero indicates readiness. If the busy bit is set in response to a transmit command, the RT transmits its status word only. If this function is not implemented the bit is set to logic zero.

The seventeenth status word bit is the subsystem flag bit, and its use is also optional. When used, this bit flags a subsystem fault condition and alerts the bus controller to potentially invalid data. If one or more subsystems interfaced to a single RT detect a fault condition the subsystem flag bit is set to a logical 1 and a separate data word is needed to identify the specific reporting subsystem. If all subsystems are healthy, or this function is not being implemented, this bit is set to 0.

The last bit of the status word is used as a parity check bit for the preceding sixteen bits. Odd parity is used.

4.2.1.1.3 The Data Word

A data word is comprised of a sync waveform, data bits, and a parity bit as shown in Figure 4.2.1. The data sync waveform is the inverse of the command and status words, i.e., where they are positive, it is negative and where they are negative, it is positive. Note that if the bits preceding and following the sync are logic zero and logic one respectively, then the apparent width of the sync waveform will be increased to four bit times.

The sixteen bits following the sync are used for data transmission. The last bit in the data word is used for parity check over the preceding sixteen bits. Odd parity is used.

Data words are used to transmit information, control, and state data. They are distinguished from command and status words by the inverted three-bit sync pattern.

4.2.1.2 Message Transfers

The messages transmitted on the data bus are in the formats of Figures 4.2.2 and 4.2.3., in Manchester code. The bus controller provides an intermessage gap from 4.0 to 12.0 µseconds between messages. This time period, shown as T on Figure 4.2.4, is measured from the mid-bit zero-crossing of the last bit of the preceding message to the zero-crossing of the next command word sync. A remote terminal must respond to a valid command within this time period. Different message formats transmitted on the bus are explained below.

4.2.1.2.1 Bus Controller to RT Transfers

For transfers between the bus controller to a remote terminal, the bus controller issues a receive command followed by the specified number of data words. The command and data words are transmitted contiguously, that is, with no gaps between the words. The RT, after validating the message, transmits a status word back to the controller.

4.2.1.2.2 RT to Bus Controller Transfers

For remote terminal to bus controller transfers, the bus controller issues a transmit

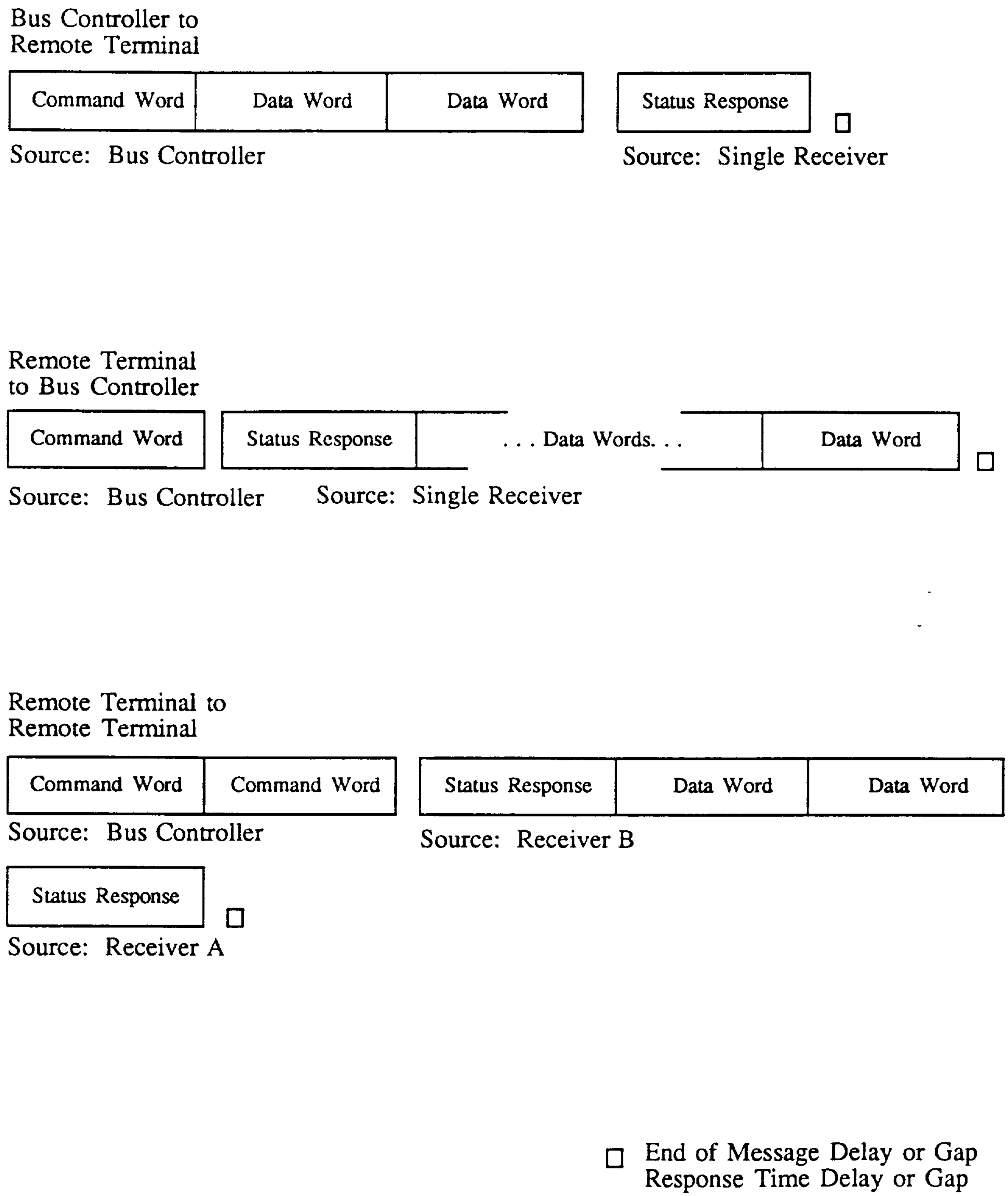


FIGURE 4.2.2 MIL-STD-1553 RECEIVER DATA MESSAGE FORMAT

BUS CONTROLLER TO REMOTE TERMINALS

COMMAND WORD	DATA WORD	DATA WORD	DATA WORD	□

SOURCE: BUS CONTROLLER
ADDRESS: 11111
SUBADDRESS: UNIQUE = 11111 OR 00000
WORD COUNT: 1-32
T/R: RECEIVE

REMOTE TERMINAL TO REMOTE TERMINALS

COMMAND WORD	COMMAND WORD	STATUS RESPONSE	DATA WORD	DATA WORD

SOURCE: BUS CONTROLLER
ADDRESS: 11111
SUBADDRESS: UNIQUE = 11111 OR 00000
WORD COUNT: 1-32
T/R: RECEIVE

SOURCE: RECEIVER A

SOURCE: BUS CONTROLLER
ADDRESS: UNIQUE A
SUBADDRESS: UNIQUE = 11111 OR 00000
WORD COUNT: 1-32
T/R: TRANSMIT

□ END OF MESSAGE DELAY OR GAP
RESPONSE TIME DELAY OR GAP

FIGURE 4.2.3 MIL-STD-1553B MULTIPLE RECEIVER DATA MESSAGE FORMAT

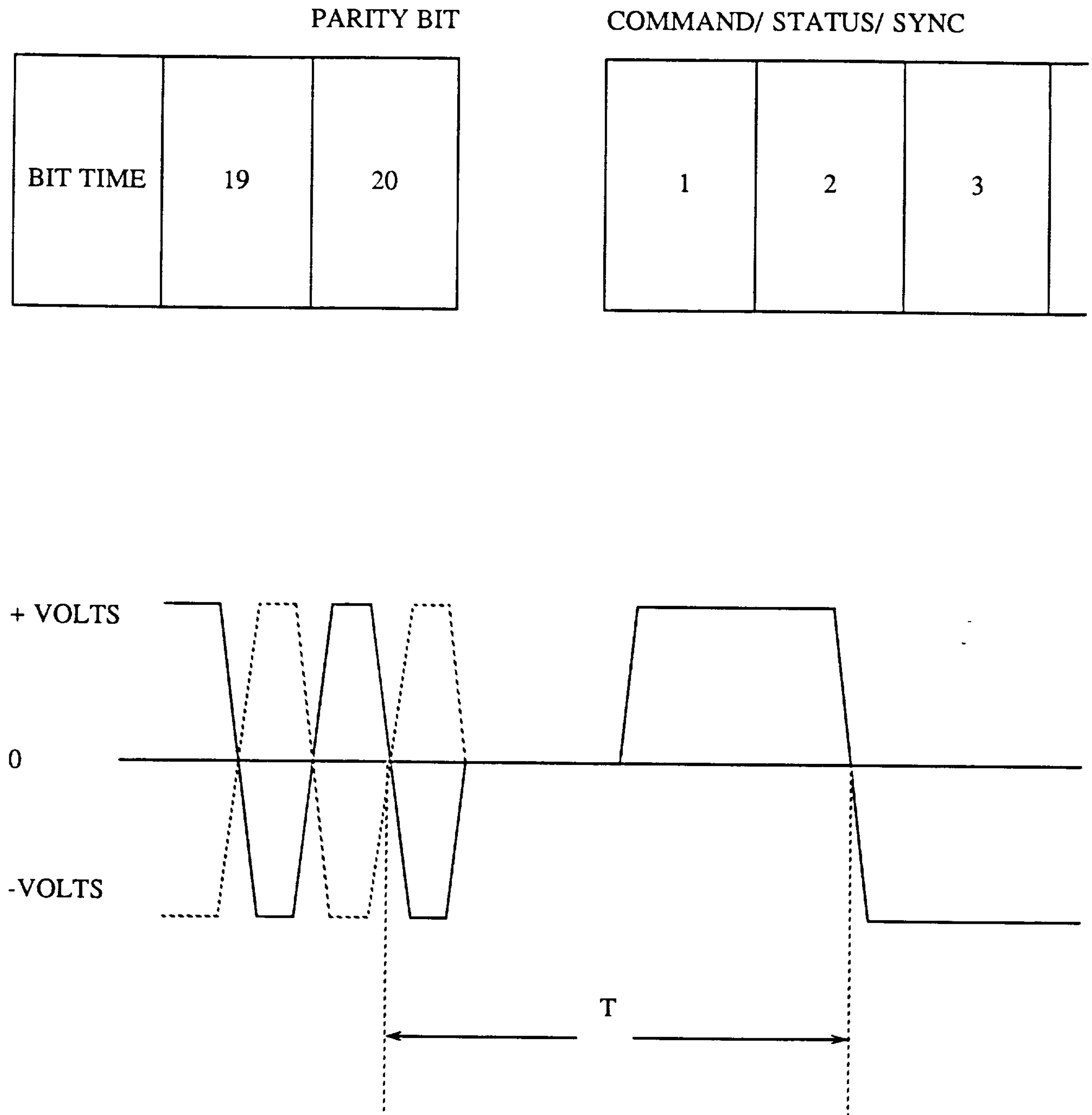


FIGURE 4.2.4 MIL-STD-1553B INTERMESSAGE GAP AND RESPONSE TIME

command to the RT, which after validating the command word, transmits a status word back to the bus controller, followed by the specified number of data words. The status and data words are also transmitted contiguously.

4.2.1.2.3 RT to RT Transfers

For remote terminal to remote terminal data transfers, the bus controller issues a receive command to RT A, followed contiguously by a transmit command to RT B. RT B, after verifying the command word, transmits a status word followed by the specified number of data words with no gaps in between. After receiving the data from RT B, RT A transmits a status word within a specified time period.

4.2.1.2.4 Bus Controller Broadcasts

When the bus controller must "broadcast" a message to more than one remote terminal, it issues a receive command word with 11111 in the RT address field followed by the specified number of data words. The command and data words are transmitted contiguously. The RTs with the broadcast option, after validating the message from the bus controller, set the broadcast command received bit in the status word, but do not transmit the status word.

4.2.1.2.5 RT Broadcasts

When a remote terminal has a message to broadcast to other remote terminals, the bus controller issues a receive command word with 11111 in the RT address field followed by a transmit command to RT A using the RT's address. RT A then specifies the number of data words. The status and data words are transmitted with no gap. Except for RT A, all terminals with a broadcast option set the broadcast received bit in the status word, but do not transmit the status word.

4.2.2 MIL-STD-1773

MIL-STD-1773 is a fiber optic version of MIL-STD-1553B. MIL-STD-1773 is designed so that, if implemented in systems using MIL-STD-1553B, no major modifications of the users will be required. The timing as well as the frame and word formats are the same as described above for MIL-STD-1553B.

Because of the simplex nature of optical fiber (as opposed to the duplex nature of twisted pair), it is necessary for there to be two seperate fibers, one for each direction, as opposed to the single twisted pair cable of MIL-STD-1553B. Because fiber is so much lighter, as well as offering other advantages as described above, this is not a problem, and in fact offers alternative architectures to the simple star or bus used by MIL-STD-1553B, as shown in Figures 4.2.5 and 4.2.6. [RELI83]

4.3 Contention Protocols

In this section, we will discuss Ethernet, the most prevalently used contention protocol. The Ethernet original baseband version was designed, developed, and patented by Xerox and was publicly announced in 1979. Since then, a cooperative effort by Digital Equipment Corporation, Intel, and Xerox has produced an updated Ethernet which is considered the standard for cable-based LANs because it is very close to the IEEE 802 CSMA/CD standard. The Carrier Sense Multiple Access with Collision Detection (CSMA/CD control technique) is the more publicized method for bus/tree topologies and is compatible with the IEEE 802 standard.

The Ethernet is basically a multi-access, packet-switched communications channel which is managed by the control technique CSMA/CD for carrying digital data among locally distributed computing systems. A primary goal of the Ethernet specification is compatibility. Ethernet was in fact the first to accomplish this by allowing devices from different manufacturers to communicate directly with one another.

Using the CSMA/CD control technique, each station attached to the bus must con-

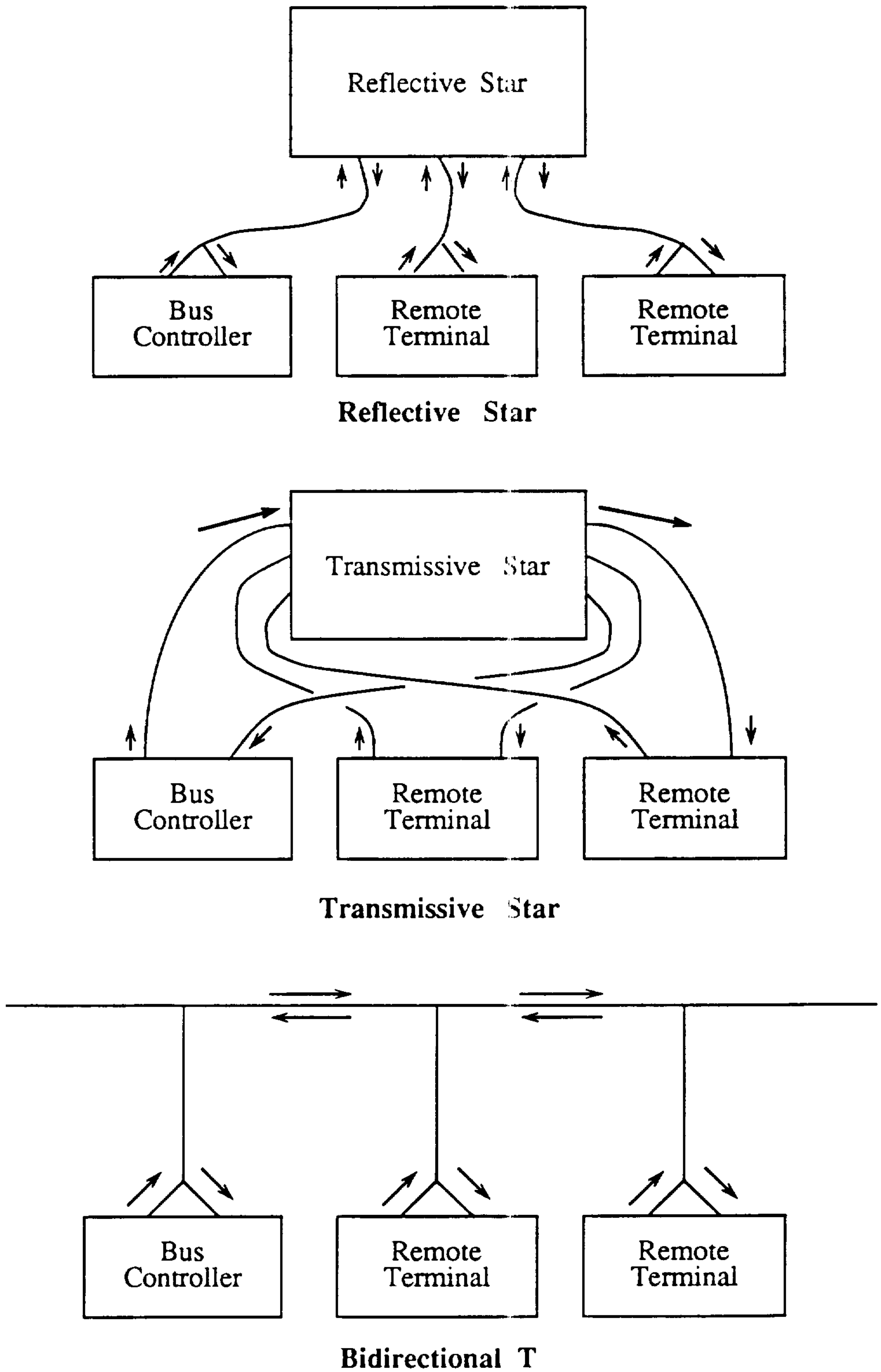
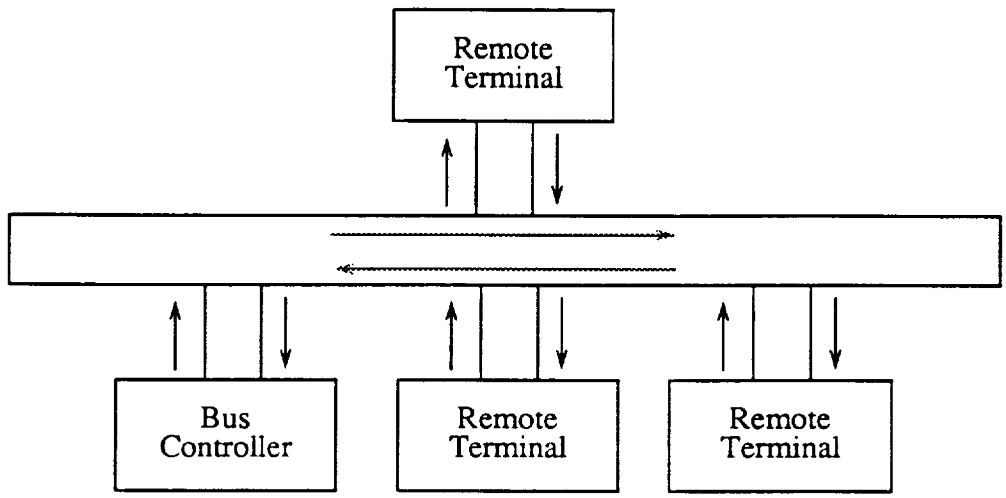
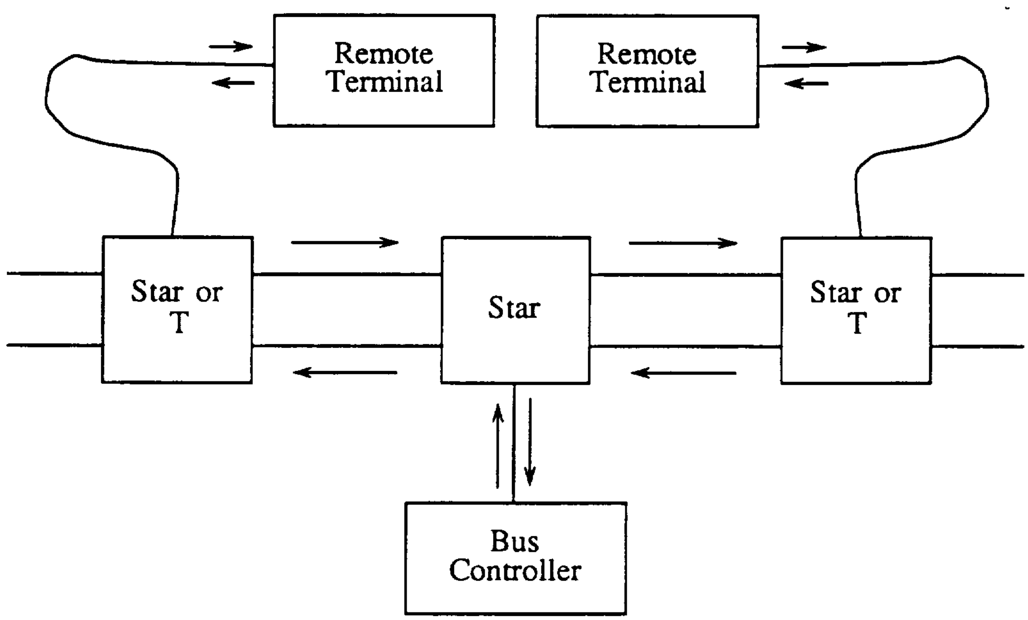


FIGURE 4.2.5: MIL-STD-1773 CONFIGURATIONS [RELI83]



Unidirectional Hybrid



Bidirectional Hybrid

FIGURE 4.2.6: MIL-STD-1773 CONFIGURATIONS [RELI83]

tend with other stations to access the bus. There is no central controller which allocates access to the channel. Each station must listen, i.e., use carrier sense to detect whether the bus is free. A station must wait or defer its transmission until the bus is quiet if another station is transmitting. After gaining access to the bus, the transmitting station continues to monitor the medium to detect colliding transmissions on the bus. This is called "listen while talk" and refers to collision detection.

Each station on the common channel must be able to transmit and receive packets with the packet format and spacing as shown in Figure 4.3.1. A packet is made up of various bytes with the last bit of each byte transmitted first, and the preamble beginning a transmission. A packet may not exceed 1526 bytes or be less than 72 bytes. Included in each of these numbers is: 8 bytes for the preamble, 14 bytes for the header, the data bytes, and 4 bytes for the cyclic redundancy check, or CRC. The following defines each field of the frame:

1) Preamble: 64 bits of alternating 1s and 0s, and ending with two consecutive 1s. The preamble is used by the receiver to establish bit synchronization and then to locate the first bit of the frame.

2) Destination Address: 48 bits specifying the station or stations which are to receive the packet. The packet may go to one station, a group of stations, or broadcast to all. This is determined by the first bit: 0 - one destination, and 1 - multiple stations. If all 48 bits are set to 1, then packet is broadcast to all stations.

3) Source Address: 48 bits specifying the station which is transmitting the packet.

4) Type Field: 16 bits identifying the type of higher level protocol (protocols above the network layer) associated with the packet. Used to interpret the data field.

5) Data Field: 46 to 1500 bytes of data or pad characters. A minimum combination of 46 bytes is required to ensure that the frame will be distinguishable from a collision fragment.

6) CRC - Packet Check Sequence: 32 bits containing a cyclic redundancy check (CRC) generated by treating all preceding bits of the packet from the first bit of the destina-

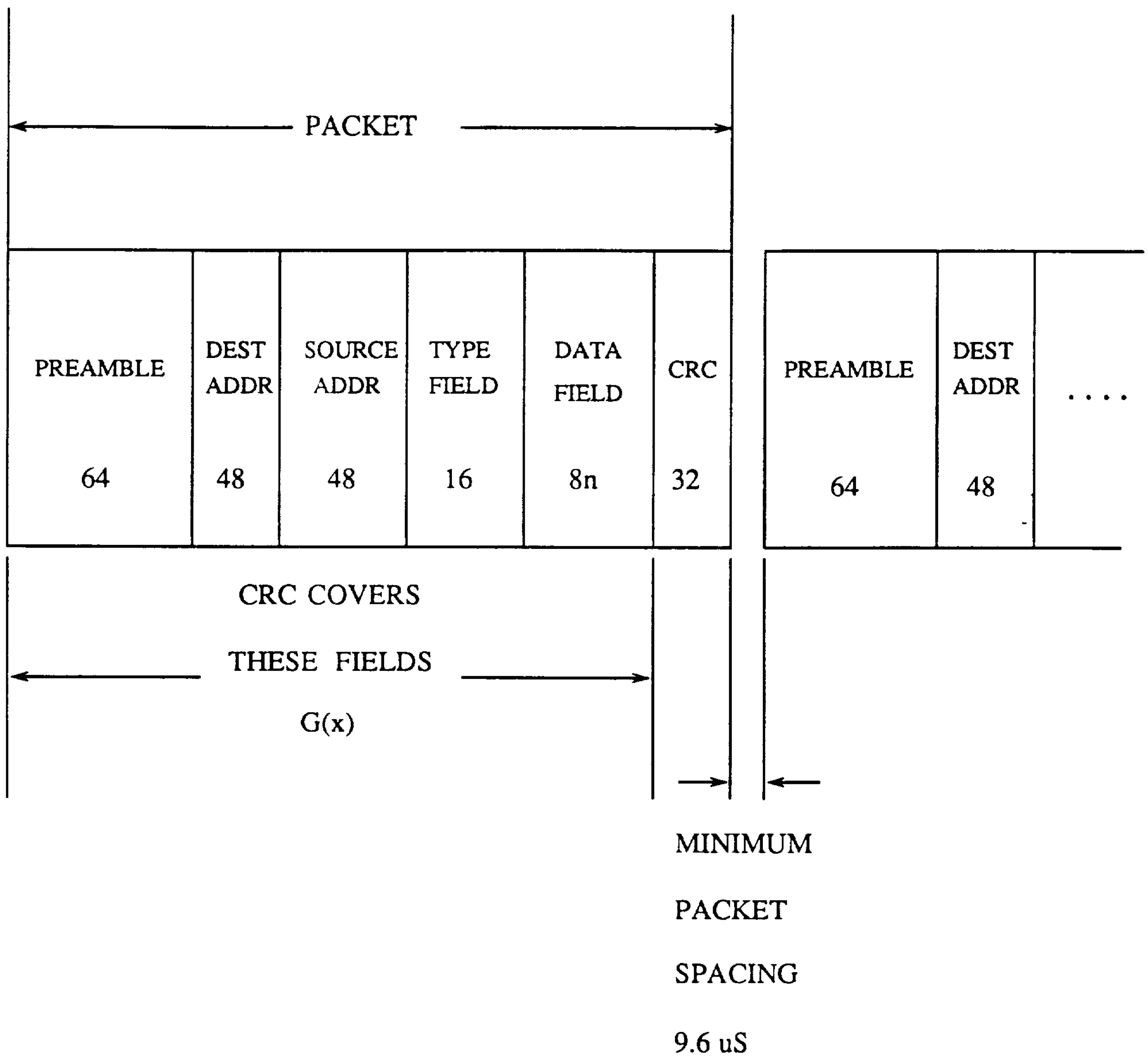


FIGURE 4.3.1 ETHERNET PACKET STRUCTURE

tion field back as terms of a polynomial, dividing them by the generator polynomial:

$$G(x) = x^{32} + x^{26} + x^{22} + x^{16} + x^{12} + x^{11} + x^{10} + x^{8} + x^{7} + x^{5} + x^{4} + + x^{2} + x + 1$$

and then taking the remainder (the inversion of which will be the CRC) by means of a linear feedback shift register, initially set to all 1s. As a packet comes in, the a shift register in the receiver performs the same operation. After receiving a good packet, the receiver's shift register contains 11000111000001001101110101111011 ($x^{31} + x^{30} + x^{26} + \ldots + x + 1$).

The Ethernet has an enforced waiting time on the bus of 9.6 μseconds, that is, 9.6 μseconds is the minimum amount of time which must elapse after the end of a transmission before another may begin. For one bit to travel from one end of the longest bus length allowed to the other (the round-trip propagation delay time) requires 51.2 μseconds. If any station receives a packet or bit sequence shorter than 72 bytes long, the information is discarded and considered a collision fragment.

When a terminal experiences a collision, it must wait a period of time known as the "backoff" period before it mat retransmit. The backoff period in Ethernet is determined by an algorithm known as "binary exponential backoff", which can be expressed by [MARA80]:

$$BP = R \times T_s, \quad 0 < R \leq 2^k - 1$$

where

BP = backoff period

R = a random number

k = the smaller of 8, or the number of collisions experienced during the current transmission attempt

T_s = slot time, a time slightly greater than the round-trip propagation delay

When attempting a transmission, an Ethernet terminal doubles the backoff range (the range of possible values of R) for eight contentions, until it is two hundred and fifty-six times the slot time, and then leaves the backoff range at this value for the next eight contentions. If a terminal experiences sixteen contentions in trying to transmit a packet, it abandons the attempt and the packet is lost.

4.4 Token Passing Protocols

In this section, we will discuss several token passing protocols in detail.

4.4.1 The ProNet Protocol

ProNet was developed by Proteon Inc. for use on the model p1200 Multibus LAN. The information in this section was taken from the "Operation and Maintenance Manual for the ProNet Model p1200 Multibus Local Network System". ProNet operates on a classic ring or "star-shaped" ring, in which terminals are connected to the ring through a "wire center", which in the case of a terminal failure, can make the necessary connections to bypass that terminal, leaving the ring intact. These two configurations are shown as Figure 4.4.1 . For additional reliability, each ProNet ring is actually two counter-rotating rings, i.e., one ring in which data flows clockwise, and another in which data flows counterclockwise. If one ring should fail, communication can then take place on the other ring in a procedure known as "switch-back". If both rings should break at the same place for some reason, for example, a physical break or a terminal failure, the terminals on either side of the break can go into "loop-back", that is, connect the counter-rotating rings into one large ring, bypassing the break. Counter-rotating rings switch- and loop-back are illustrated in Figure 4.4.2. [PROT86]

4.4.1.1 ProNet Control Word and Message Formats

The control word and packet formats are shown as Figures 4.4.3 and 4.4.4, respectively. The token consists of the "flag" (a 0 followed by a string of seven 1s), and two additional 1s. If a terminal has no data to send, upon receiving the token, it passes, or "repeats" it to the next terminal.

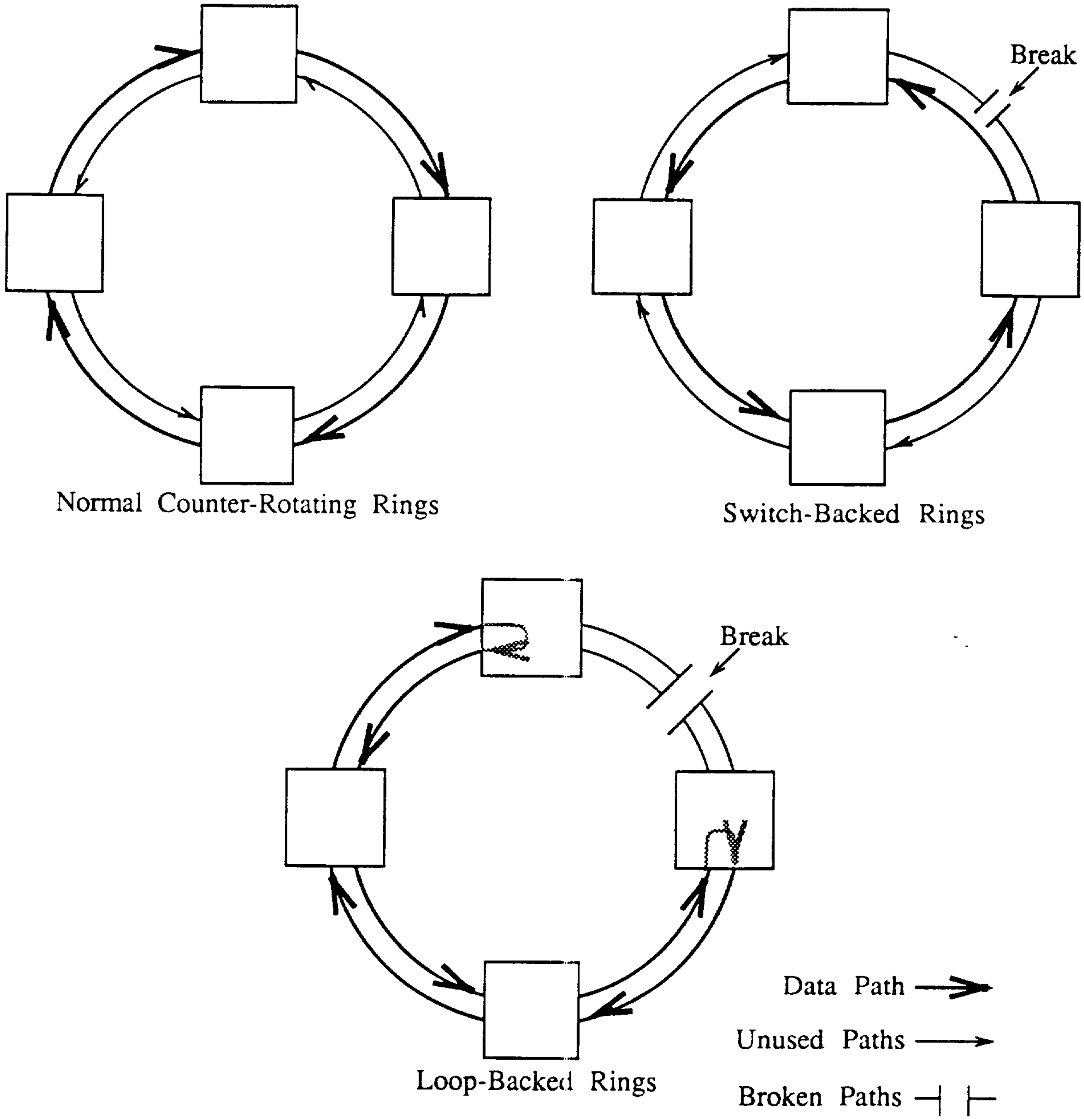
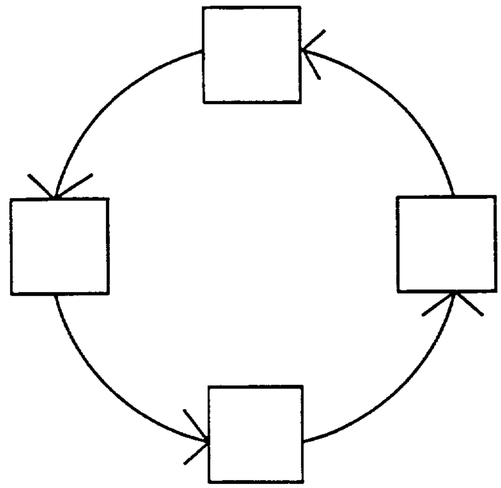
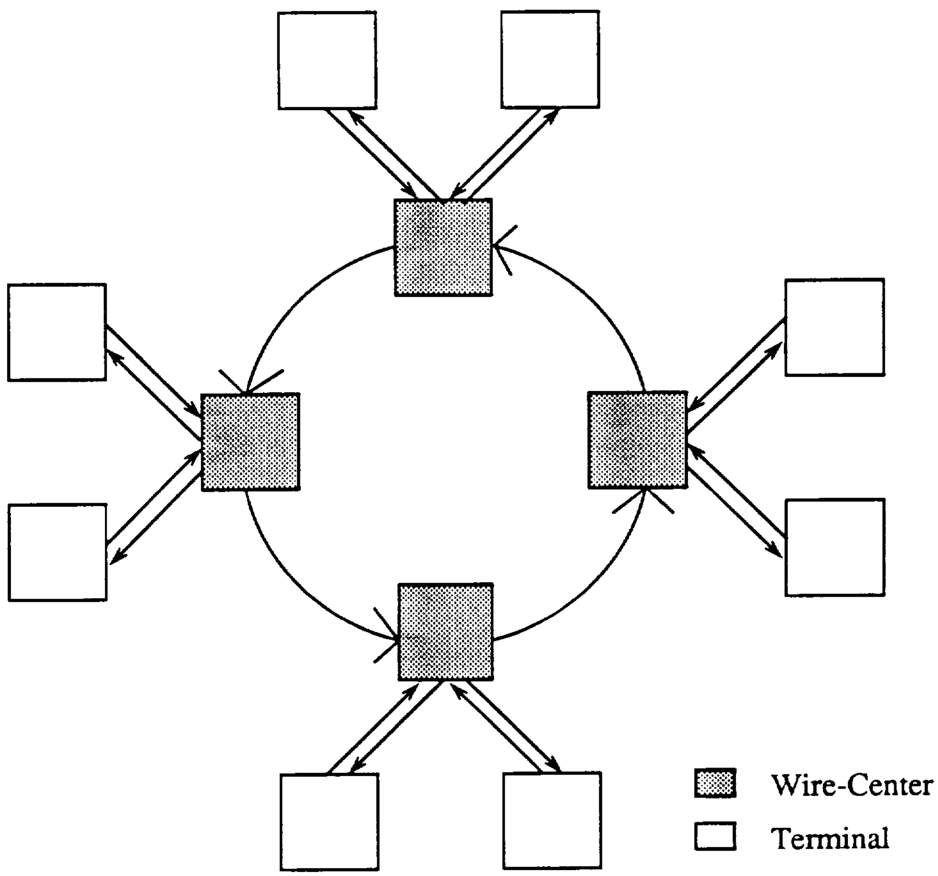


FIGURE 4.4.1 ProNet RING CONFIGURATIONS [PROT86]


"Classic" Ring



"Star-shaped" Ring

FIGURE 4.4.2 ProNet RING CONFIGURATIONS [PROT86]

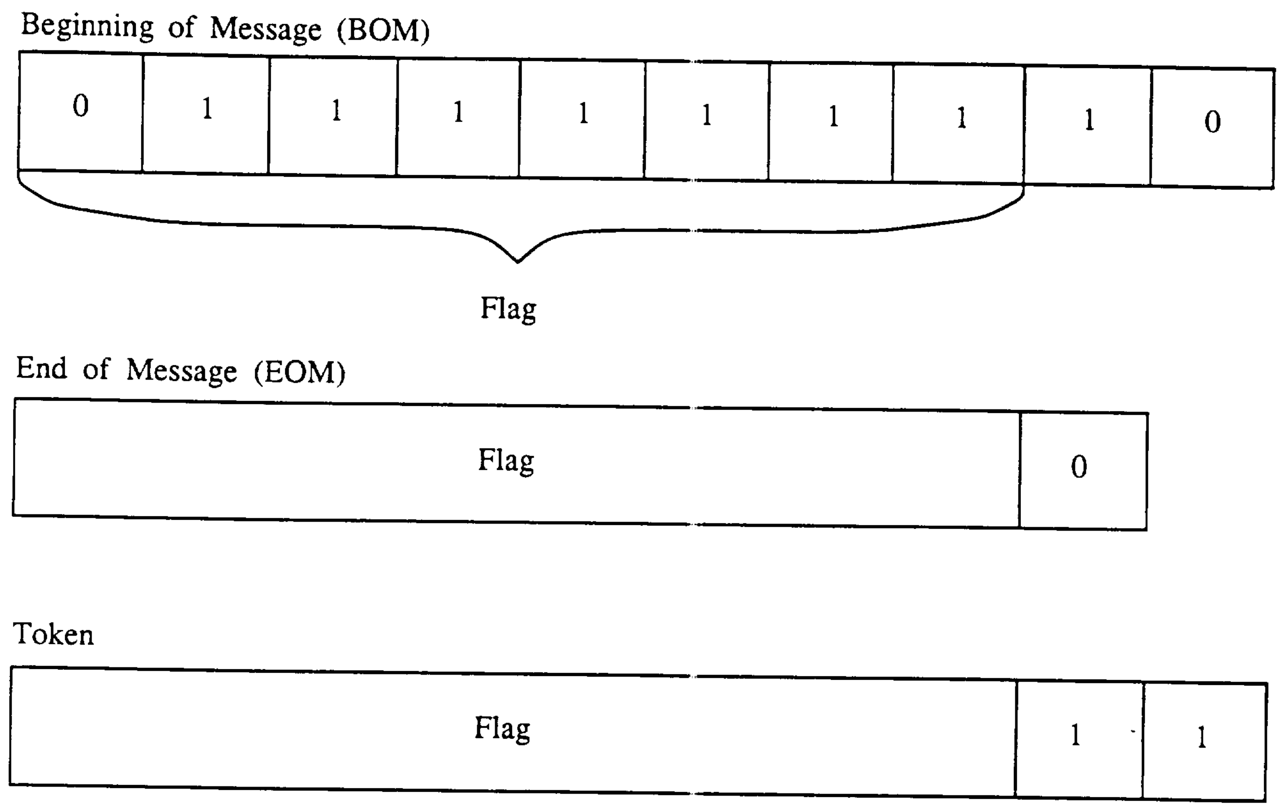


FIGURE 4.4.3 ProNet CONTROL CHARACTER FORMAT

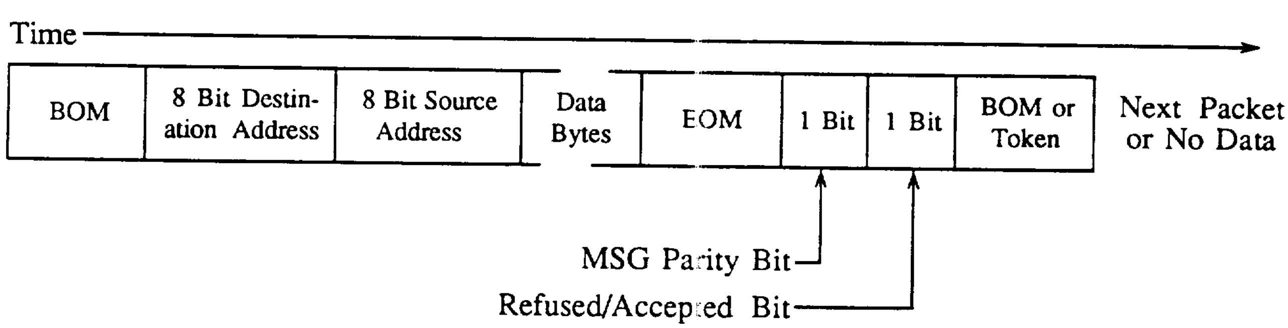


FIGURE 4.4.4 ProNet PACKET FORMAT [PROT84]

If a terminal does have data to be transmitted, it changes the last bit of the token to 0, thus making it a Beginning of Message (BOM) character. It then adds eight bit source and destination address bytes, its data or "message" (up to 2044 eight-bit bytes), an End of Message (EOM) character consisting of the flag and an additional 0, a parity check bit, a message refused/accepted status bit set initially to 1, and finally, a control character indicating the end of the packet, either a BOM of another packet, or the token. The packet format is shown as Figure 4.4.4. [PROT84]

In order to assure that no control character occurs in the message data, ProNet employs "bit stuffing". While creating a packet for transmission, if a terminal detects a stream of six consecutive 1s, it "stuffs" a 0 into the data behind them, insuring that the seven 1s of the flag will never occur in the data message. This stuffed 0 will be removed by the addressed terminal as it "copies" the data from the ring into a buffer for its own use.

When a terminal recognizes its address as the destination address in a data packet, it copies the message part of the packet into a buffer for its own use, and then sends the packet back onto the ring, resetting the message refused/accepted status bit to 0. If a terminal is for some reason unable to copy data addressed to it, it repeats the packet along the ring, leaving the message refused/accepted bit 1. When a terminal detects a data packet which it had previously transmitted, it removes it from the ring, leaving only the token or the BOM of the next message. Depending on the application, the terminal may monitor the message refused/accepted bit, and retransmit its data (if necessary) upon again receiving the token.

In order for the ring to recover from errors such as the loss of a token or packet, each terminal is equipped with three hardware timers: the token timer, the flag timer, and the message lost timer. The token and flag timers track the amount of time between the detection of a token or the flag respectively. If either of these counts exceed a set amount, the terminal will "re-initialize" the ring by generating and repeating a token. If two or more terminals try to re-initialize the ring within 500 μseconds of each other, a collision will occur, and the ring will still be without a token. Upon the detection of a collision, the token and flag timers will be reset, leaving it for another terminal to re-initialize the ring. Proteon Inc. feels that collisions during re-initialization are very unlikely.

The message lost timer serves a different function. After a terminal has transmitted a packet, it repeats no other data along the ring until it detects its own packet, indicating that the packet has made it successfully around the ring. (This is to remove "noise" from the ring.) The message lost timer begins counting when a packet is transmitted, and resets when that same packet is again detected. If the message lost timer exceeds a set count (determined by the maximum amount of time required for a packet to entirely circle the ring), it is assumed that the packet is lost, and the terminal resumes repeating all data that comes into it. Depending on the application, the terminal may or may not attempt to retransmit the lost packet upon receiving a token. [PROT84]

4.4.2 The Fiber Distributed Data Interface Protocol

The Fiber Distributed Data interface (FDDI) is a proposed standard for a 100 Mbit/second ring network. FDDI terminals may be connected directly to the ring, in a "classic" ring configuration, or through "concentrators" (devices similar to ProNet wire-centers) in a "star-shaped" ring. Unlike the ProNet wire-center, however, FDDI concentrators may be addressed and treated as independent terminals as well as simply connecting other terminals to the ring and thus may serve as gateways linking FDDI rings. In such linked rings, the token is passed asthough all terminals were on one, large ring.

In addition to concentrators, FDDI terminals have an optical bypass that maintains the integrity of the ring even if that terminal should fail, as shown in Figure 5.4.5 [FDDI88]. Up to three adjacent terminals may be bypassed without harming the ring. For additional reliability, FDDI maintains counter-rotating rings, and may employ switch-back or loop-back similar to ProNet.

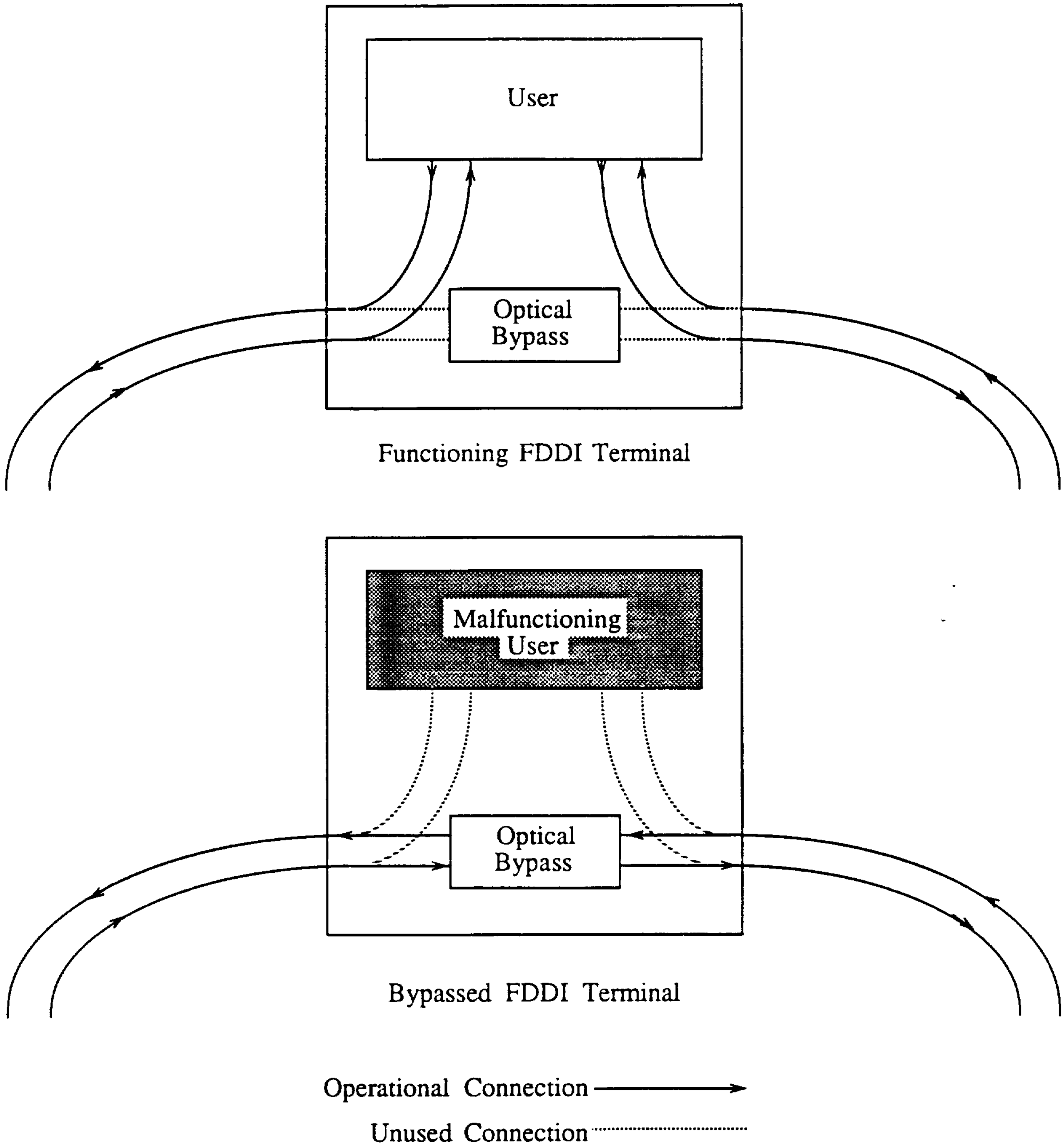


FIGURE 4.4.5 FDDI BYPASS OPERATION [FDDI88]

C-2

4.4.2.1 FDDI Data Formats

This section will deal with the formats of frames and tokens used by FDDI. FDDI words are made of five-bit "symbols" generated by a four-out-of-five code. Sixteen of the thirty-two member FDDI symbol set are reserved for encoding data, each symbol representing four binary bits. The rest of the symbols are used for various functions such as: three symbols used as starting and ending delimiters, two used as control indicators, and three are used as line-state signaling and are recognized by the physical layer.

The symbols are grouped together in fields and the fields in turn make up frames or tokens, as shown in Figure 4.4.6. The fields are described in greater detail below:

a) The Preamble (PA): This field consists of IDLE line-state symbols and serves as a maximum frequency signal used for establishing and maintaining clock synchronization. A PA field must precede every transmission.

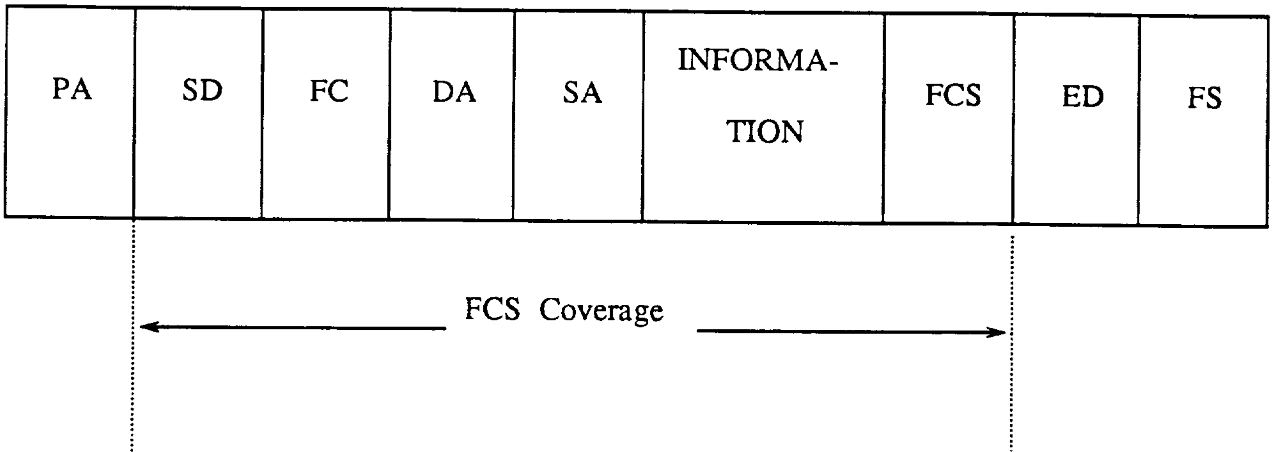
b) The Starting Delimiter (SD): The SD, field consists of a sequence of two delimiter symbols. The SD field establishes the symbol boundaries for the content that follows.

c) The Frame Control (FC): The FC field contains information about the frame, and indicates to the receiving terminal if the frame is synchronous or asynchronous, the address field length, and the frame type (logical link control or station management). The FC field may indicate that the frame containing it is unformatted, and that terminals should simply repeat the frame down the ring without checking for their address or the frame's validity

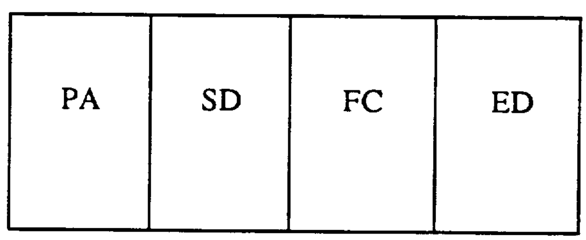
d) Address Fields: The Source Address (SA) field contains the address of the terminal from which the frame originated and the Destination Address (DA) field contains the address of the terminal for which the frame is intended. Both may be either 4 or 12 symbols wide, depending on the number of terminals on the ring. The DA field may contain either the address of a specific terminal, or a sequence indicating the message is to be received by several terminals.

e) The Frame Check Sequence (FCS): The FCS field performs the same function as the Ethernet CRC word, i.e., a cyclic redundancy check. It is generated by the ANSI stan-

Frame Format



Token Format



PA = Preamble (16 or more symbols)
SD = Starting Delimiter (2 symbols)
FC = Frame Control (2 symbols)
SA = Source Address (4 or 12 symbols)
DA = Destination Address (4 or 12 symbols)
FCS = Frame Check Sequence (8 symbols)
ED = Ending Delimiter (1 or 2 symbols)
FS = Frame Status (3 symbols)

FIGURE 4.4.6 FDDI FRAME AND TOKEN FORMATS [ROSS86]

dard polynomial. The fields covered by the FCS consist only of data symbols. No other field contains data symbols.

f) The Ending Delimiter (ED): The ED field consists of one delimiter sequence and indicates the end of information in a frame and the end of a token.

g) Frame Status (FS): This field is used by terminals to indicate that they have recognized the frame as being addressed to them, copied its data, or detected and error in the frame. The terminal that transmitted the frame will check the FS field to see if the destination terminal received the frame and/or copied its data.

Any terminal detecting an error in a frame while repeating it will change flags in the FS field. Thus the FS field may be used by receiving terminals to determine the validity of the frame. [ROSS86]

4.4.2.2 Timed Token Rotation

The timed token rotation method of FDDI access may best be described by describing the initialization procedure as well as the functions of various timers in the FDDI terminals.

4.4.2.2.1 Initialization

The initialization period is used for establishing the target token rotation time (TTRT). Each terminal calculates the maximum amount of time that the token may take to completely circle the ring and yet is still fast enough to support all of that terminals synchronous traffic needs. The shortest of these times becomes the TTRT.

During initialization period, the percentage of bandwidth each terminal may use for transmission of its frames is allocated by assigning each terminal a percentage of the total TTRT, or bandwidth, in which it may transmit upon capturing the token. This percentage of TTRT is known as the terminal's operational target token rotation time (T_Opr). Frames transmitted during a terminals allocated time are referred to as synchronous data. A terminal may also transmit additional frames, if the traffic on the ring is light enough. These frames are referred to as asynchronous data. Assignment of bandwidth is application dependent, and may be changed during the operation of the ring. The sum of all terminals assigned bandwidth must not exceed 100%. If 100% of the bandwidth is assigned, there can be no asynchronous transmission. [JOHN87]

4.4.2.2.2 Timing

Each terminal is equipped with several timers and counters that tell it when it may capture the token and transmit its data. These include the token ring timer (TRT), the token holding timer (THT) and the late counter (Late_C). Their functions are described below.

The token rotation timer determines when a terminal may capture the token and transmit its synchronous data. The TRT is initialized to T_Opr and is decremented with every pulse of an internal clock. If the TRT expires, the late counter is incremented and the TRT is reinitialized to T_Opr.

If a token arrives before the TRT has reached zero, i.e. Late_C = 0, the current value of the TRT is placed in the token holding timer, and the TRT is reinitialized to T_Opr. The value placed in the THT thus represents the amount of time left over in the previous token cycle from its required minimum, T_Opr, that is, the amount of unused bandwidth. A station may transmit asynchronous data if THT has not expired and Late_C = 0. The THT is enabled only during the transmission of asynchronous data, as opposed to the TRT which is always enabled.

Expiration of the TRT indicates that traffic on the ring is heavy, which is why asynchronous transmission is allowed only when Late_C = 0. When a token arrives and Late_C is not 0, the late counter is reset, but the TRT is not reinitialized to T_Opr. [JOHN87]

4.4.2.3 Restricted Token Mode

Restricted Token Mode is an optional feature of FDDI used for extended communication "bursts" between two terminals. The only difference between restricted token mode and the operation described above is that the two terminals involved in the burst are the only terminals allowed asynchronous transmission. [JOHN87]

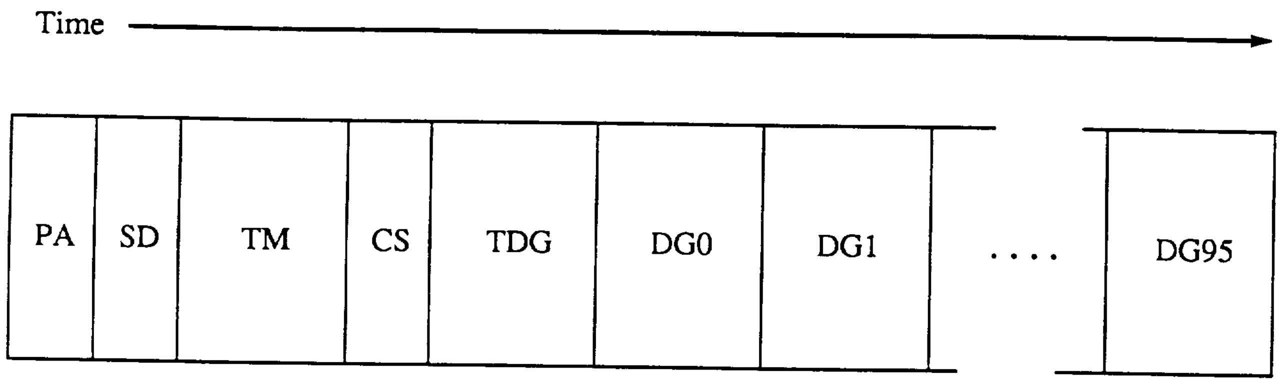
4.4.2.4 Virtual Circuit Switching

Another optional feature of FDDI is virtual circuit switching. FDDI systems featuring virtual circuit switching are often referred to as FDDI-II.

FDDI-II systems are initialized in the token mode, and if a terminal requires a virtual circuit connection, it vies for the position of cycle master. When a station has won the right to be cycle master, it imposes cycles on the network at an 8 kHz rate, i.e., one cycle every 125 μseconds. The cycle master may find it necessary to induce latency periods in order to maintain an integral number of cycles on the ring. The cycle format is shown in Figure 4.4.7.

A cycle and a frame both start with a preamble and a starting delimiter. In a cycle, however, the starting delimiter is followed by the isochronous channel temperature, which consists of 16 symbols, one for each possible isochronous channel. Each symbol indicates whether its corresponding channel is an isochronous channel or is free for use by the token channel. If the Nth symbol of the channel temperature indicates that its corresponding channel is isochronous, then the Nth byte of each of the 96 programmable data groups belongs to that channel. Only the cycle master may assign isochronous channels.

The dedicated token data group, along with all bytes in the programmable data groups not assigned to an isochronous channel make up a "super-channel" known as the token channel. Tokens and frames within this channel are the same as in non-FDDI-II systems, except that in place of the starting delimiter (which is reserved to indicate the beginning of the cycle), there are two line-state symbols, halt and idle. In the cycle format, the halt/idle combination is used exclusively as an alternative to the starting delimiter of frames in the token channel. [ROSS86]



PA = Preamble (1 symbol, nominal)
SD = Starting Delimiter (2 symbols)
TM = Iso-Synchronous Channel Temperature
(16 symbols)
CS = Cycle Sequence (1 octet)
TDG= Dedicated Token Data Groups
(16 octets)
DG0...DG95 = Programmable Data Groups
(16 octets each)

FIGURE 4.4.7 FDDI-II CYCLE FORMAT [ROSS86]

4.4.3 SAE AE-9B High Speed Token Passing Data Bus for Avionics Applications (AE-9B)

The SAE AE-9B High Speed Token Passing Data Bus for Avionics Applications, hereby referred to as AE-9B, is a proposed standard for an explicit token passing bus network for either fiber optic or wire media being developed by the AE-9B Linear Implementation Task Group , a subcommittee of the Society of Automotive Engineers. As of this writing, no one has developed hardware to support AE-9B, but we nonetheless include it in this report to demonstrate an example of explicit token passing on a bus network. Also, other standards proposed by the Society of Automotive Engineers, such as MIL-STD-1553B, have become quite common, and it is likely that hardware to support AE-9B will developed in the near future.

The terminals of AE-9B are physically arranged on a bus network, but logically, the token is passed from one terminal to another so that the network timing is very similar to a token passing ring, as shown in Figure 4.4.8. Unlike token passing rings, however, the AE-9B token must be specifically addressed to a terminal. Each terminal passes the token to the terminal with the next highest physical address, until it reaches the terminal with the highest physical address, which passes the token to the terminal with the lowest physical address. This is known as a "token cycle". AE-9B is designed to accommodate up to 128 terminals. [MEYE86]

4.4.3.1 Bus Initialization

When the bus is first brought into operation, no terminal possesses the token and the network must be initialized using the "claim token" procedure. Upon being powered up, each terminal monitors the bus for any activity. If a terminal sees no activity on the bus, it transmits a "claim token frame". If at the end of this transmission, there is still no activity on the bus, the terminal issues a token to its successor on the token path. The claim token frame is proportional to the length of the terminal's physical address, so that during initialization, the

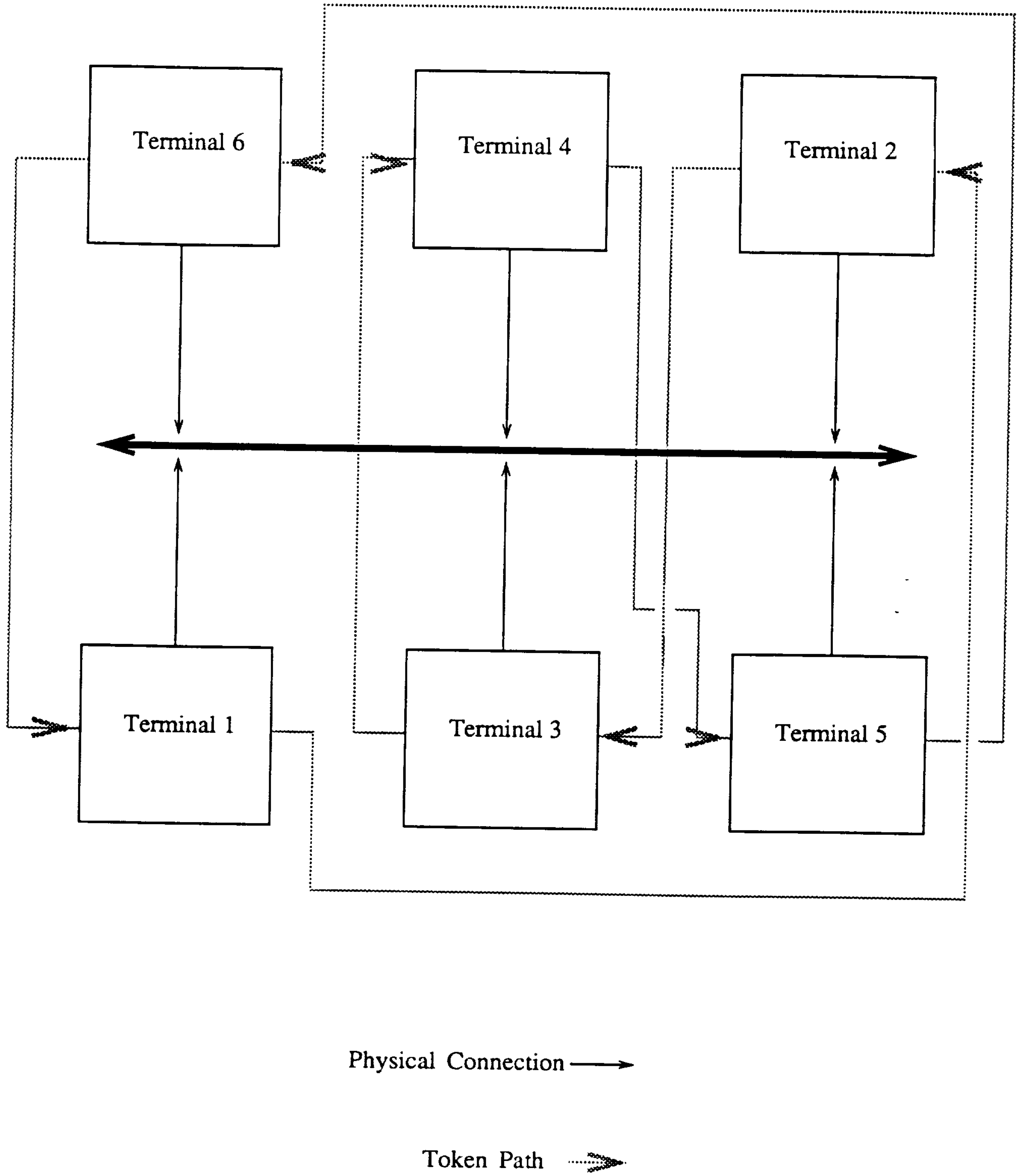


FIGURE 4.4.8 TOKEN PASSING PATH EXAMPLE [MEYE86]

token should be issued by the terminal with the highest physical address to the terminal with the lowest physical address. Each terminal will then "hunt" for a successor using the station insertion procedure described below.

4.4.3.2 Removal of Terminals from the Token Path

After a terminal issues a token to its successor, it monitors the bus for any activity from that successor. If it sees none, it issues the token again, and again monitors the bus for any activity. If a terminal fails to see any activity on the bus after two attempts to issue the token, it assumes that its successor has failed. The terminal issuing the token increments its successor address, and repeats the procedure described above until it detects a successful token pass, that is, upon passing a token, a terminal sees activity from its successor.

It must be pointed out that a terminal will still attempt to pass the token to its successor on every token cycle, even if that successor failed to receive the token on the previous pass. Thus failed terminals on a network can cause the bus to be tied up with the "successor hunt" procedure described above increasing "overhead" on the network, which in turn increases data latency, and reduces throughput.

Fortunately, if a terminal realizes that it will soon fail, or knows that it will not be needing the token for some time, there is a way for it to remove itself from the token path. To do this, a terminal issues an "exit token" to its successor. An exit token conveys control of the bus, just like a regular token, but in addition, it alerts the predecessor of the terminal issuing the exit token to increment its successor address. Thus, by issuing an exit token, a terminal effectively removes itself from the token path, eliminating the need for its predecessor to "hunt" for it during the next token cycle.

4.4.3.3 Insertion of Terminals onto the Token Path

To bring terminals on the network, each terminal periodically attempts to pass the token to all the terminals (if any) whose physical addresses lay between its own and its current successor address using the same "successor hunt" procedure described in the previous sections. If one of these terminals responds to the token pass, the terminal issuing the token changes its successor address to the address of the terminal that responded, thus effectively bringing that terminal onto the token path. The length of time between when terminals check to see if there are other terminals to be inserted into the token path is a parameter set by the user.

4.4.3.4 Timing and Prioritization

In order that a terminal may know how much traffic is on the network, it is equipped with four timers, the Token Holding Timer (THT), and three Token Rotation Timers (TRT). All timers are reset to user defined values when a station receives the token and begin counting down to zero. If their is a lot of traffic during a token cycle, it will take much longer for the token to complete the cycle, and terminals can detect this by the expiration of one or more of their timers.

AE-9B allows for four message priority levels for each terminal. Each timer corresponds to a different priority level. The user can define these priorities by defining the reset values of the THT and the TRTs. Highest priority messages may only be transmitted if, upon receiving the token, its THT has not expired, the highest message may be transmitted. If the THT has expired, there has been a great deal of traffic on the bus, and upon receiving the token, the terminal simply resets its timers and passes the token to its successor. Thus, by setting the reset value of a terminal's THT to a large value, the user grants that terminal greater access to the bus than a terminal with a smaller reset values for their THTs.

A similar procedure is followed for messages of lower priorities. A message of a given priority may only be transmitted only if its corresponding TRT has not expired. Thus the larger the reset value for a TRT, the higher the priority of its corresponding messages. Obviously, messages corresponding to a TRT may not be made a higher priority than the messages corresponding to the THT, since when the THT has expired, the terminal will not be granted access to the network, no matter what the values of its TRTs.

4.4.3.5 Terminal Management Functions

AE-9B allows for 512 "sub-addresses" for each terminal, so that terminals may give commands or instructions to one another. The user may program subaddresses for his own commands in additions to the built in commands, or "functions", described below.

Mode Control Command/Status Response: This command allows a terminal to force another terminal to perform one of the following functions:

1. Terminal Hardware Rest
2. Terminal Enable/Disable
3. Execute Built-In Test (BIT)
4. Enable/Disable Bus Loop-Back Tests
5. Report Status
6. Report Traffic Statistics
7. Enable/Disable Global Time "Master mode
8. Report Time

Load/Report Configuration: This command allows the user to define his own functions, or to set the user defined parameters of station insertion period, THT and TRT time out factors and bus length. This command can also be used to set a terminals physical address using what is known as the "message filter" function.

Test Messages Report: This command is used to test the integrity of the data path. Data words in a Test Messages Report (TMR) command are stored by the receiving terminal, an retransmitted upon again receiving a TMR command.

Built-In Test Functions (BIT): AE-9B provides several types of built in testing, such as internal message loop-back testing to test the host/bus interface unit integrity.

Time Synchronization Report: Each AE-9B terminal contains a 48 bit real-time clock for system function synchronization, or to provide a "time tag" to messages indicating the "staleness" of data. One terminal is designated the "time master", and periodically synchronizes all the other terminals clocks to its own using the time synchronization report command. If the time master fails to synchronize the network twice in a row, another terminal assumes the role of time master. The terminals that may function as time masters are assigned by the user. [MEYE86]

4.5 Hybrid Protocols

Hybrid protocols are not as prevalent as "pure" protocols, although a few have been developed and are commercially available and many more have been simulated. Hybrid protocols have shown excellent performance. This section will discuss SEAFAC, a command/response-token passing hybrid; a token passing-CSMA/CD hybrid; HYPERchannel, a virtual token passing-CSMA hybrid; and L-Expressnet, another virtual token passing-CSMA hybrid.

4.5.1 The SEAFAC Protocol

This protocol, hereby referred to as SEAFAC, was developed and simulated by Harold Alber and Wayne Thomas at the Systems Engineering Avionics Facility at Wright-Patterson Air Force Base to accommodate the avionics requirement of the next generation fighter and bomber aircraft. The information in this section was taken from their unpublished white-paper, "A Dual Channel High Speed Fiber Optics Multiplex Data Bus System".

SEAFAC is a hybrid of token passing and command response protocols. Usually, SEAFAC operates under a token-passing scheme. Upon receiving the token, if a terminal has no information to send, it sends a token message addressed to the next terminal, giving it control. However, should a terminal have information to send and is in possession of the token, it will send either a data or control message, using one of the formats shown in Figure 4.5.1. A control message sends only control and status information to the other terminals, while a data message sends this information along with from 1 to 256 data words. Both data

START SYNC	CMD WORD	TIME TAG WORD	1-256 DATA WORDS	CRC WORD	TOKEN WORD	END SYNC

DATA MESSAGE

START SYNC	CMD WORD	TIME TAG WORD	CRC WORD	TOKEN WORD	END SYNC

CONTROL MESSAGE

START SYNC	TOKEN SYNC	END SYNC

TOKEN MESSAGE

FIGURE 4.5.1 SEAFAC MESSAGE FORMATS [ALBE 86]

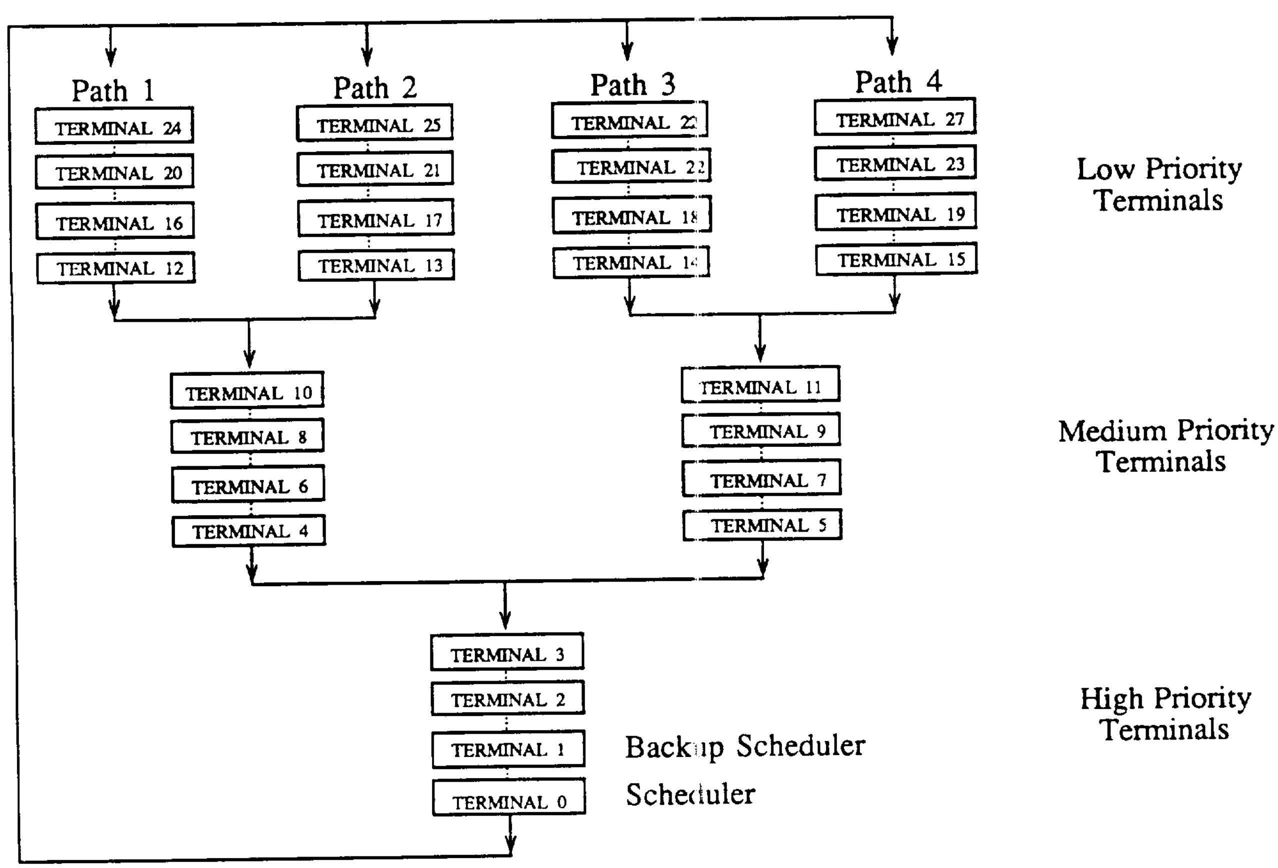


FIGURE 4.5.2 MULTIPLE PATH TOKEN PASSING EXAMPLE [ALBE86]

and control messages contain a token word addressed to the next terminal, giving it control.

If a terminal receives the token, and transmits no messages within 15.0 μseconds (the worst case propagation delay in a 1000m network), the scheduler terminal assumes control of the network, and determines to which terminal to send a token message. This command response type of behavior is also utilized when the network first begins operation.

In order to alleviate data latency problems inherent in token passing protocols, SEAFAC recognizes that some terminals just aren't as important as others, and prioritizes them such that important terminals will receive the token more often. The terminals are grouped into "paths", an example of which is shown in Figure 4.5.2. The scheduler passes the token from path 1 on the first pass, path 2 on the second pass, and so on. In the four-path example of Figure 4.5.2, high priority terminals are on all four paths, the medium priority terminals are on only two paths, and the low priority terminals each are only on one path. Thus, high priority terminals will see the token on every pass, while medium priority terminals will see the token only on two out of four passes, and low priority terminals will see the token only once every four passes.

It must be pointed out that the "paths" are not physical connections. This allows the priorities and paths of the terminals to be assigned dynamically by the scheduler. For example, a terminal important when the network first begins operating, yet whose importance diminishes as time passes can be prioritized accordingly.

Both the data and control messages contain a sixteen-bit command (CMD) word. The first bit of this word indicates whether or not the message is global, i.e., intended for a class of terminals as opposed to a specific terminal. If it is set to 1, the message is global, and the next three bits indicate to which of up to eight classes of terminal the message is intended. If, on the other hand, the message is intended for a specific terminal, the first bit of the CMD word is set to 0, and the next seven bits are used for the specific address of the one of up to two-hundred and fifty six possible terminals.

The eight bits following the address bits are used for "sub-addresses", which are essentially commands to the addressed terminal. The scheduler may use the sub-addresses to initialize terminals, initiate self-tests, request status information, load the token-passing address, synchronize the terminal, load the memory of the terminal, or request data from the terminal.

Terminals other than the scheduler may use the sub-addresses to transmit self-test and status information by putting their own address in the seven-bit address field of the CMD word. This allows for any terminal to know the status of any other in the network. The sub-addresses may also be used for hand-shaking messages such as request data, acknowledge data receipt, or request next block of data. The last also requires a terminal making the request to place in the address field of the CMD word the address of the terminal providing the data block.

The time-tag word of data and control messages tells the terminal or terminals being addressed the "staleness" of the message. All terminals have internal clocks periodically synchronized by the scheduler. When a terminal sends a message, it places the time at which the message originates into the time-tag word, so that the addressed terminal or terminals may know to within 20.48 µseconds how old the message is. If, for some applications, terminals must know the age of the message more precisely, an additional sixteen-bit time-tag word may be added, allowing the terminal to know the age of the data to within 0.01 µseconds.

Data words are sixteen-bit bytes, and a data message may contain up to two-hundred and fifty-six of them. The content and ordering of data words within the message is application dependent, but is the same for all messages with the same message ID.

The Cyclic Redundancy Check (CRC) word is a sixteen-bit word used for detecting errors in the message. The CRC word may be generated in the same manner as the Ethernet CRC word described above, except that

$$G(x) = x^{16} + x^{15} + x^2 + 1$$

is used as the generator polynomial. This is a very powerful method of error detection and is

Table 4.5.1 SEAFAC Data Efficiency [ALBE86]

Total Words	Data Words	Message Length Bits	Message Length Time (μseconds)	Efficiency Data Bits/Total Bits
1	0	22	0.44	0.0%
4	0	70	1.40	0.0%
5	1	86	1.72	18.6%
6	2	102	2.04	31.4%
8	4	134	2.68	47.8%
12	8	198	3.96	64.6%
20	16	326	6.52	78.5%
36	32	582	11.64	88.0%
68	64	1094	21.88	93.6%
132	128	2118	42.36	97.0%
260	256	4166	83.32	98.3%
516	512	8262	165.24	99.2%
1028	1024	16454	329.08	99.6%
2052	2048	32838	656.76	99.8%
4100	4096	65606	1312.12	99.99%

much more efficient than parity check bits in data words since it does not require a word-count field. Using a CRC word allows the following types of errors to be detected: all odd numbers of error bits, all single-burst errors of sixteen bits or less, 99.9969% of seventeen bit burst errors, and 99.9984% of single-burst errors of eighteen or more bits.

Table 4.1, shows the "data efficiency", or the ratio of data words to "overhead" in a data message. Since the overhead is limited to the fixed-length Start Sync, CMD, Time-Tag, and CRC words no matter how many data words, the longer the data message, the more efficient the system is. A price is paid, however, in the amount of time required to send a message. For data messages containing two-hundred and fifty-six data words (the maximum allowed by SEAFAC), data efficiency is 98.3%.

The Token Word is always the last word in a message, and indicates the next terminal that may transmit. If a terminal sees its address in the Token Word, it may transmit so long as the address parity is correct, the token word is composed of valid Manchester coded bits, and the token word is followed by an End Sync. The last eight bits of the token word provide status information for the terminal originating the message in which the token word is contained. This information can be used by the scheduler to determine the overall health of the system and/or which terminals need servicing. The status bits in the token word may be used by other terminals to decide whether to accept or reject the message based on the health of the terminal addressing them.

The Start Sync word is used to synchronize a terminal's front-end decoder, with synchronization maintained by the Manchester coded bits of the message. It also initializes a modulo 16 bit-counter and a cyclic redundancy check register that performs the same CRC operation used by the originating terminal to generate the CRC word. The End Sync words stops the operation of both the counter and the register. If, either the modulo 16 bit-counter does not contain the value 6, or the contents CRC register do not match the CRC word, the message is considered invalid and rejected. [ALBE86]

4.5.2 Hybrid Token-CSMA/CD Protocol

The Token-CSMA/CD hybrid protocol (hereby referred to as Token-CSMA/CD) described in this section was developed and simulated by P.M. Gopal and J.W. Wong at the University of Waterloo in Ontario, Canada. It is for use exclusively on bus topologies. While, there are no commercially developed protocols based on this hybrid, it has shown significant benefits over conventional CSMA/CD or token passing in simulation and merits discussion.

In order to use Token-CSMA/CD, terminals must be synchronized. Time is divided into units called "slots", each slot representing the time for a bit to travel completely up and down the bus medium. Terminals may only initiate transmission at the beginning of a slot. For best performance, data packets and backoff periods should be multiples of the slot-time, with the backoff period lasting as long or longer than the packet length

The token in Token-CSMA/CD is not a control message. Each of the M terminals in the network has an individual identity number from 0 to M-1. In order to know when it is possession of the token, each individually numbered terminal monitors the bus channel and increments and modulo M counter each time the channel undergoes a transition from busy to idle. When the contents of the counter are the same as the terminal's identity number, it is in possession of the token. For example, on a network of five terminals, terminal 0 will possess the token after 0, 5, 10, . . . channel transitions from busy to idle, terminal 1 after 1, 6, 11, . . . transitions, and so on.

The behavior of a Token-CSMA/CD is exactly like that of conventional CSMA/CD, except in the event of a collision involving a terminal in possession of the token. In this event, those terminals involved in the collision without the token go into their "backoff" period, waiting to retransmit. Upon detection of a collision, for a period of one slot-time, the terminal in possession of the token keeps the channel busy, but without transmitting its data. At the end of this delay, any other terminal attempting to send data would have detected a collision, and have entered its backoff period, thus insuring that the channel is free. The terminal with the token then sends its data, guaranteed of a successful transmission.

Packet delays for low through-puts are about the same for Token-CSMA/CD and conventional CSMA/CD, and for a small range of through-puts, conventional CSMA/CD is actually faster. For high through-puts in which many stations are involved in a collision, however, the effect of the token is seen, and Token-CSMA/CD has much lower packet delays than conventional CSMA/CD. [GOPA84]

4.5.3 The HYPERchannel protocol

The HYPERchannel protocol (hereby referred to as "HYPERchannel") is, like Ethernet, a CSMA protocol, that is each terminal "listens" to the medium, in this case a bus, and transmits only when the bus is free. Unlike Ethernet, HYPERchannel does not employ collision detection. Instead, it uses message acknowledgments, several delay types, and prioritized terminals to ensure that only very short messages will experience a collision.

4.5.3.1 HYPERchannel Delay Sequence

As stated above, in order to avoid collisions, after the bus becomes idle, HYPERchannel employs a sequence of delays, during which only certain terminals may transmit. That sequence of delays is as follows:

a) Fixed Delay: During this time, the terminal which received the previous transmission sends a response frame (described below) to the terminal from which it received the transmission. All other terminals experience a delay, whose length is described by the equation:

$$\text{Fixed Delay} = 4 \text{ nseconds x (trunk length in feet)} + 2.08\ \mu\text{seconds}$$

which is slightly greater than twice the amount of time it takes for a signal to propagate the entire length of the bus and back.

b) N-Delay: In HYPERchannel, each terminal is assigned a unique priority used in determining the amount of time it must wait after the fixed delay before it may transmit, or its N-Delay. The N-Delay of each terminal may be expressed by the following equations:

$$\text{N-Delay}(K) = \text{N-Delay}(K-1) + 4 \text{ nseconds} \times d + 1.6\ \mu\text{seconds},\ K = 1,2,\ldots,L$$
$$\text{N-Delay}(1) = 4.8\ \mu\text{seconds}$$

where

K = priority index of the terminal
L = the number of terminal on the bus
d = the distance in feet from the terminal of priority K-1 to the terminal of priority K

Thus, each terminal is guaranteed a period of 1.6 µseconds in which it may initiate a transmission guaranteed to be without collision. It is important to note that terminals with low priority indexes are able to transmit more often than terminals with high priority indexes. The times in which terminals are guaranteed collision-free transmission are collectively referred to as the scheduling period.

c) End Delay: After the scheduling period, each terminal must wait an additional time period before it may begin transmission. During this time, they listen to the bus medium to ensure that the terminal with the lowest priority, that is, the terminal with priority index, L, has not begun a transmission. This listening period is referred to as the end delay, and its length for each terminal may be expressed by the equation:

$$\text{End Delay}(K) = \text{N-Delay}(L) + 4 \text{ nseconds} \times d' + 1.6\ \mu\text{seconds},\ K=1,2,\ldots,L-1$$

where L and K are defined the same as for N-Delay, but d' is defined as the distance in feet from the terminal of priority index K to the terminal farthest from it.

After the end delay comes the contention period, where any terminal may transmit if it senses that the bus is idle. This is the only time when collisions can occur. HYPERchannel terminals do not detect collisions, but instead rely on an acknowledgment from the terminal to which their transmission was addressed during the fixed delay period. If this acknowledgment does not arrive, they know their transmission was unsuccessful.

4.5.3.2 The Wait Flip-Flop

To prevent high-priority terminals from dominating the bus medium, each terminal is equipped with a wait flip-flop that is set when a terminal completes a transmission, and cleared at the end of the beginning of the contention period. A terminal may not transmit when its wait flip-flop is set. The wait flip-flop of any individual terminal may be disabled if necessary, depending on its application.

4.5.3.3 HYPERchannel Frames and Sequences

The smallest unit of data transmitted by a HYPERchannel terminal is called the frame. There are three types of frames: transmission, response, and message-and-data frames. The sequences of frames for transmission are shown in Figures 5.5.4 and 5.5.5. The function of each is described below:

a) Transmission Frames: Transmission frames are used for "handshaking", i.e., the exchange of control and status information between two terminals. The terminology used in this report often differs from that used by Network Systems Corporation, the manufacturers of HYPERchannel systems. When this is the case, the equivalent Network Systems Corporation term will be placed in parentheses after our term. The transmission frames are listed and described below:

1) Request Status, RS (Copy Registers): The request status frame is used by a terminal to see if a terminal to which it wishes to transmit is capable of receiving the transmission. In frame sequences, the request status frame captures the bus for the transmission of subsequent frames.

2) End Message Proper, EMP (Clear Flag 8): This frame is sent by a transmitting terminal to indicate to the receiver that it has completed a message proper frame (to be described below). Flag 8 in the receiver is set when it is waiting for a message proper, and cleared by the transmitting terminal when the complete message proper has been transmitted.

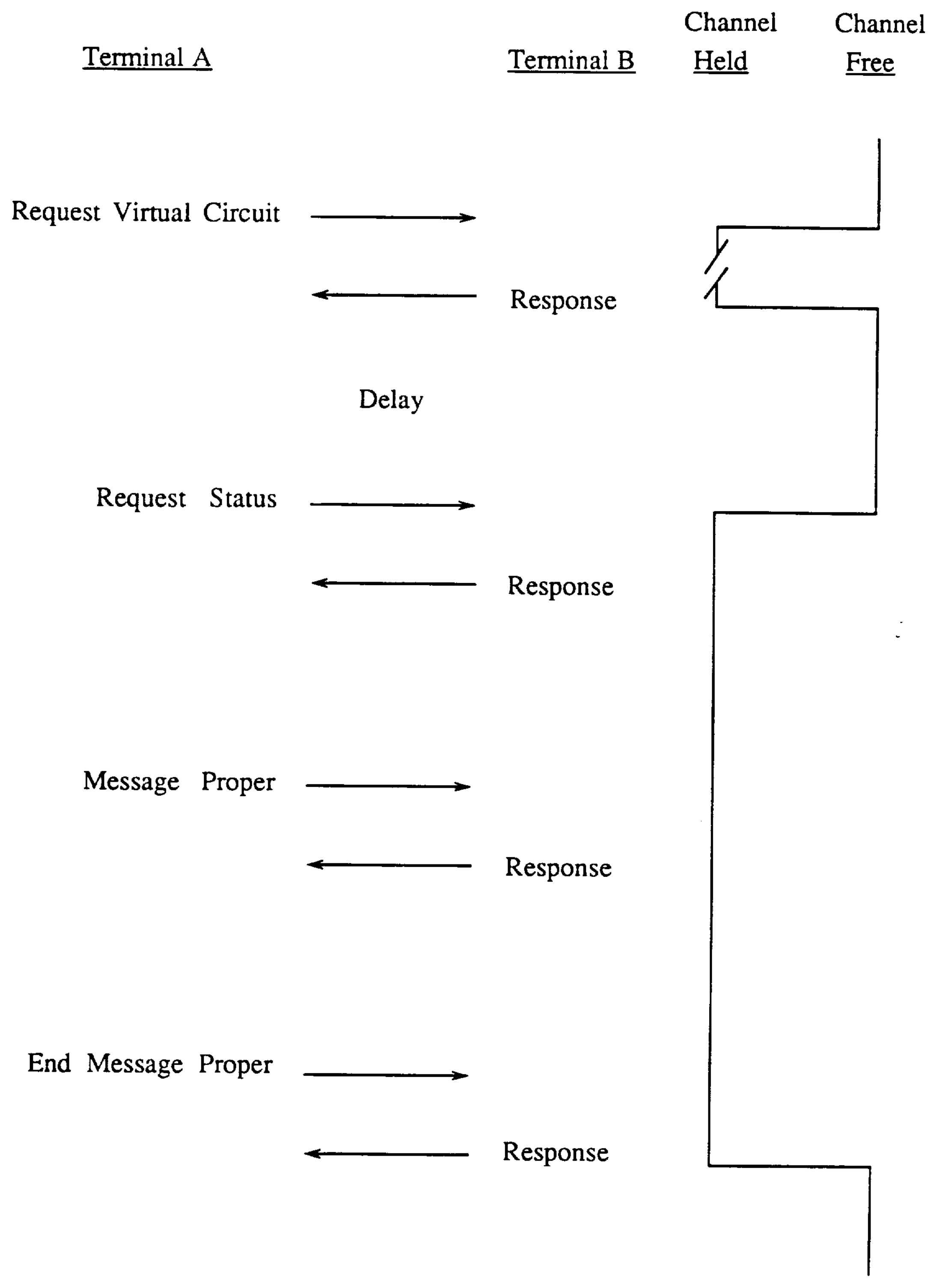


FIGURE 4.5.3 TIMING FOR TRANSFER OF MESSAGE-ONLY SEQUENCE FROM TERMINAL A TO TERMINAL B [FRAN84]

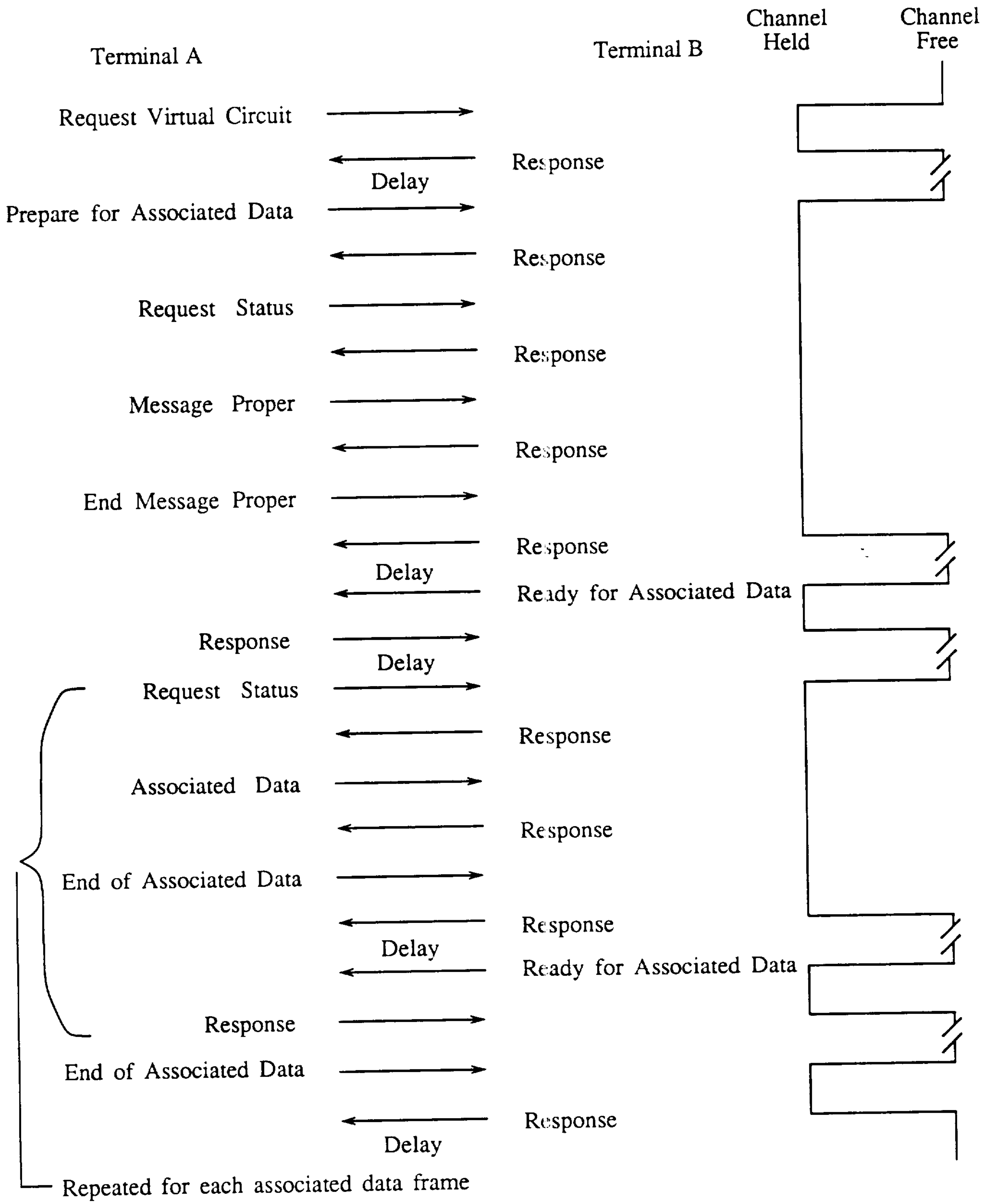


FIGURE 5.5.4 TIMING FOR TRANSFER OF MESSAGE-WITH DATA SEQUENCE FROM TERMINAL A TO TERMINAL B [FRAN84]

3) Prepare for Associated Data, PAD (Set Flag A): This frame alerts the receiving terminal that after the message proper, associated data frames (to be explained below) will follow.

4) Ready for Associated Data, RAD (Clear Flag 9): After a terminal receives a PAD frame, it must prepare buffer space in order to receive the associated data. When it has done this, it sends a RAD frame to the terminal which sent it the PAD. A terminal will not transmit associated data frames to their destination terminal until the destination terminal sends it a RAD.

5) End of Associated Data, EAD (Clear Flag A): This frame is transmitted by a terminal to indicate to the receiver that there will be no more associated data frames.

6) Request Virtual Circuit, RVC (Set Reserve Flag): This frame is sent by a terminal when it desires a virtual circuit connection (to be described later) with the receiving terminal.

b) Response Frames: Response frames are transmitted only during the fixed delay to acknowledge the reception of a frame. A response frame may contain status information if it is used to acknowledge a request status frame.

c) Message and Data Frames: There are two types of message and data frames, message proper frames, which contain up to 64 bytes of data; and associated data frames, which may contain up to 2 kilobytes of data.

The transmission of data from one terminal to another requires the exchange of several frames known as a frame sequence. Terminals do not control the bus for the entire durations of the frame sequence, but rather relinquish control at various times, as shown by Figures 4.5.3 and 4.5.4. There are two types of frame sequences: message only sequences, in which only the data contained in a single message proper is exchanged; and message with data sequences, in which associated data frames follow a message proper.

In order for two terminals to exchange a message with data sequence, they must establish a virtual circuit. The sending terminal, from which the data originates, reserves itself to only communicate with the receiving terminal, to which the data is destined. It then sends a request virtual circuit frame to the receiving terminal. If that terminal is capable of receiving a data sequence, it will reserve itself to communicate only with the sending terminal. When the two terminals are reserved only to communicate with each other, they said to be in a virtual circuit connection.

If the receiving terminal is unable to make a virtual circuit connection, the sending terminal will wait for a delay period which can be expressed by the equation

$$\text{Retry Delay (k)} = 2^{(k-1) \text{ modulo } 7} \times 1 \text{ µsecond}, \; k=1,2,\ldots,RC$$

where k is the number of the attempt, and RC is a terminal parameter known as the Retry Count. Thus, after failing to establish the virtual circuit connection once, the sending terminal must wait 1 µsecond before sending another RVC frame. If that attempt fails, it must then wait 2 µseconds, and then 4, and so on up to 128 µseconds, after which the cycle repeats itself until the number of attempts to establish a virtual circuit has exceeded the retry count. If this happens, the sending terminal will no longer be reserved for communication only with the receiving terminal. It is important to note that the retry delays are fixed times that double with each retry, and not random time periods chosen from an interval that doubles in size for each retry, as in the binary exponential backoff algorithm as used in Ethernet. [FRAN84]

4.5.4 L-Expressnet

L-Expressnet was developed for a bus network known as Campus Net (C-NET) by the Consiglio Nazionale delle Ricerche of Italy. It is similar to the Expressnet protocol developed at Stanford, and even more similar to a protocol known as BID for bidirectional buses. The information for this section was taken from "L-Expressnet: The Communication Subnetwork for the C-NET Project", by Flaminio Borgonovo, et al. [BORG85]

In L-Expressnet, the token is passed virtually, rather than through the reception of an actual message or control signal. Each terminal is equipped with several timers (whose functions will be explained later) that indicate when a terminal is possession of the token. In order for the L-Expressnet protocol to work properly, all terminals on the bus must have a unique index that reflects the terminal's spatial position on the bus. For example, the terminals may be indexed from left to right, each terminal having a higher index than the one on its left.

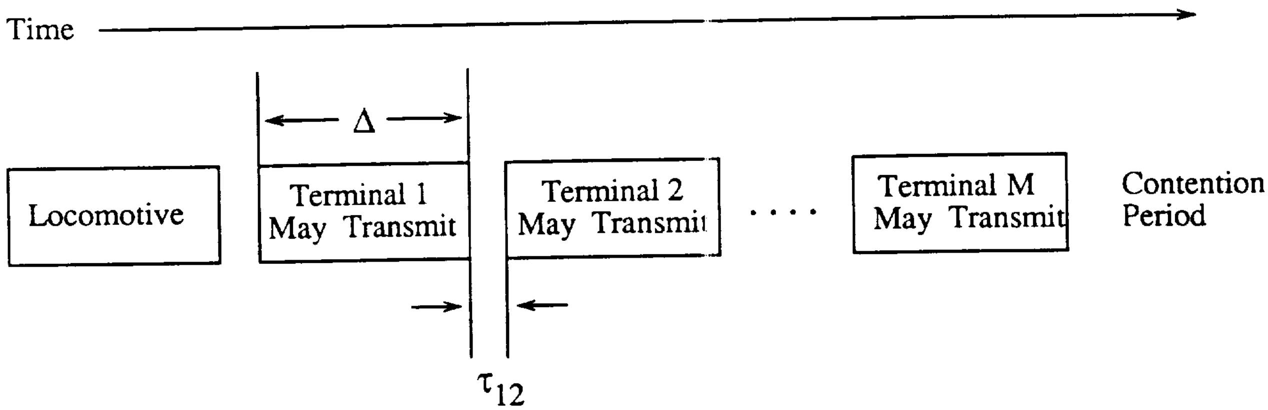
At the beginning of a cycle, or "train", as it is called, all of the timers on the enabled terminals on the bus are reset and begin counting upon the detection of a signal known as the "locomotive". The locomotive need not be a string of ones and/or zeroes. It may be simply a burst of the carrier signal, or the first 0 to 1 transition after the end of the previous train. Sample trains are shown in Figure 4.5.5.

A terminal of index i knows it is in possession of the token when a counter, CR1, reaches the value:

$$CR1(i) = (i - 1) \times \Delta$$

where Δ is a length time greater than the reaction time, i.e. the time it takes for a terminal to detect and act upon a carrier transition on the bus. After a terminals CR1 counter has reached the value described above there is a time period of length Δ in which it may transmit a packet guaranteed of no collision. Although a terminal may or may not have data packets to transmit when it is in possession of the token, there will be a time of length Δ before another terminal may transmit. If a terminal does transmit when in possession of the token, the end of its transmission marks the end of the train.

After a train has traversed the bus, a new locomotive must be generated by the lowest indexed enabled terminal on the network. To know whether or not it should generate a new locomotive, each terminal is equipped with another counter, CR2, that also begins counting upon the detection of the locomotive. A terminal may generate a locomotive if its CR2 reaches a value that reflects the amount of time required for a train to traverse the network ($M * \Delta$, where M represents the number of terminals on the bus), plus the amount of time it would have taken for a train generated by a terminal of lower index to reach it. That is, a terminal of index i may generate a locomotive if :



Δ = Time in which Terminal May Transmit Guaranteed of No Collision

τ_{12} = Propagation Delay from Terminal 1 to Terminal 2

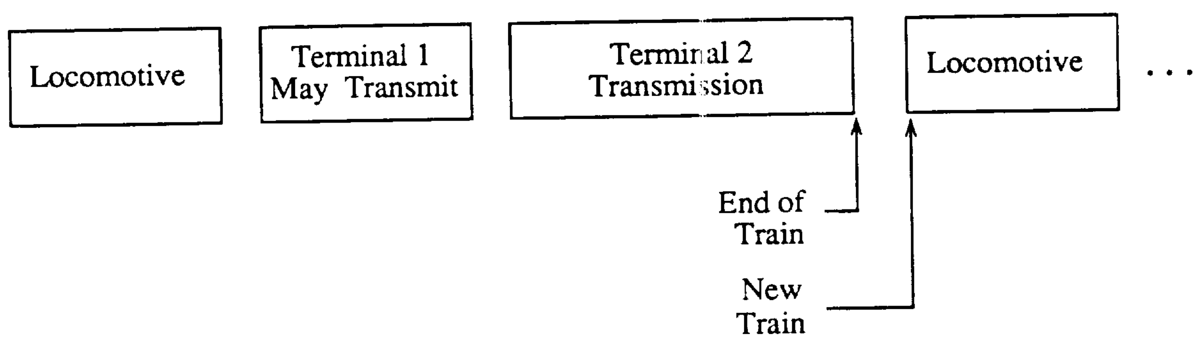


FIGURE 4.5.5 SAMPLE L-EXPRESSNET TRAINS

$$CR2(i) = M \times \Delta + 2 \times \tau + 2 \times (i-1) * \theta$$

where τ is the time it takes for a signal to make traverse the length of the bus and θ is a time such that

$$\theta > \max \left| \tau_{ij} / (i-j) \right|$$

where τ_{ij} represents the propagation time from terminal i to terminal j.

When the network first begins operation, the CR1 and CR2 counters are at zero, and will remain so until the terminal detects a locomotive, as described above. However, unless the network is somehow initialized, no locomotive will ever be generated, and therefore each terminal is equipped with a third counter, CR3 that begins counting when the terminal is first powered up, but reset by the detection of the locomotive. When a terminals CR3 counter reaches a value given by

$$CR3 = M \times \Delta + 4 \times \tau + 2 \times (M-1) \times \theta$$

it knows that the network is not initialized, and may generate a locomotive. The first locomotive is known as the "pilot". [BORG85]

4.6 Dismissals of Protocols by Cons

In this section, we will discuss why a class of protocols, or a particular protocol is unsuitable for use in the ALS system.

4.6.1 Contention Protocols

Contention protocols are too slow and too unreliable in their data delivery for use on the ALS. When few terminals need to use the medium, there are few collisions, and messages are very likely to arrive at their destinations very quickly. On the other hand, when many terminals need to use the medium, there are many collisions, and therefore messages may take a long time to reach their destinations, as their originating terminal sits out its backoff periods. Since most contention protocols will only attempt a finite number of transmission for each message, a great deal of data may be lost during high use periods.

In the ALS system using smart or intelligent sensors, the time when the ALS is in trouble is the time when the greatest number of sensors and controllers will need to communicate, and the time when contention protocols give their worst performance.

4.6.2 Implicit Token Passing

Aside from being few commercially available systems using time-multiplexing protocols, their synchronization requirements make them unsuitable for use on the ALS. If terminals are dependent on an external clock, then many additional connections must be made to it. If terminals each have internal clocks, these must all be synchronized to each other, which is very difficult. Should a terminal's internal clock lose synchronization, it may try to transmit a message while another terminal is using the medium. In order to avoid this problem, large delays must be inserted between transmissions to alleviate minor synchronization problems. This detracts from network efficiency and therefore greatly limits the bandwidth of time-multiplexing protocols.

4.7 Recommended ALS Protocols

In this section, we will explain why we feel that the MIL-STD-1553B is suitable for the current ALS system. We will also discuss why we feel that NASA may wish to change to a token passing or token passing hybrid to take advantages of developments in sensor technologies, particularly "smart" and/or "intelligent" sensors.

4.7.1 MIL-STD-1553B

MIL-STD-1553B is the communications network being used on present launch systems, and therefore has the advantage of being proven. Because it has already been used for a number of applications, it is cheap and readily available. It is also very reliable. Any terminal on a MIL-STD-1553B bus, with the exception of the controller, may break down, and still leave the network intact. The following sections will discuss MIL-STD-1553B's application to current and future technologies, as described in section 1.3.1.2 of this report.

4.7.1.1 Present Sensor Technology

Obviously, a MIL-STD-1553B local area network can support present sensor technology. It may be desirable, however, to reconfigure the busses for greater efficiency and reliability. For example, it would be desirable to modify MIL-STD-1553B for use on fiber-optic cable, rather than the current twisted-pair medium. Fiber optic cable is lighter, and because FDDI is being developed for the space station, space-qualified fiber will soon be available. Also, the use of optical fiber would eliminate echo problems thus allowing multiple stages to be connected on a single bus, rather than the current configuration of a separate bus for each stage. This would make triple redundancy much cheaper to achieve.

4.7.1.2 MIL-STD-1553B Support of "Smart" Sensors

Should the ALS be implemented with smart sensor technology, multiple MIL-STD-1553B busses for each stage should be able to handle the increased communications. This is similar to existing avionics systems, such as the F-16, which uses two MIL-STD-1553B busses for each wing. If, in order to perform trend analysis, certain sensor systems must communicate with each other, those systems should be placed on the same bus. Otherwise, their communications must be sent through the flight controller.

It is our opinion that this would not be the optimum system for use on an ALS using smart sensors. Multiple busses means multiple bus controllers, all of which must be monitored by the flight controller, greatly increasing its complexity. Also, if "intelligent" sensors should be developed, this configuration would not support them well, for reasons described below.

4.7.1.3 MIL-STD-1553B Support of "Intelligent" Sensors

MIL-STD-1553B, being a command-response protocol, would not easily lend itself to the distributed processing and control system mandated by use of intelligent sensors. Should an intelligent sensor system determine that an action must be taken, it must wait for permission from the bus controller to give the proper instructions to actuators to perform the action. If the action is urgent, permission may come too late.

4.7.2 FDDI

FDDI's primary advantage is speed. The 100 Mbit/second bandwidth of FDDI should be fast enough to support all three levels of sensor technology as described below. The synchronous and asynchronous frame types, as well as FDDI-I and FDDI-II systems also make it quite versatile and adaptable to a variety of control schemes.

Although currently FDDI hardware is quite expensive, FDDI (SAFENET) is being developed for the Navy and the space station, so the price should soon come down. Also, since it is being developed for the space station, space-hardened versions of FDDI hardware will soon be available. [COHN88/AWST88]

4.7.2.1 FDDI Support of "Dumb" Sensors

Again, the advantage here is speed. The present system of a flight controller demanding data from dumb sensors could be supported especially well by FDDI-II systems. The flight controller terminal could act as the cycle master, using one cycle to instruct which of up to 16 sensor systems should transmit their data and on which isochronous channel they should transmit it. The next cycle could then pass through the ring, collecting the data and carrying it back to the flight controller, which would then use a cycle to send instructions to various mechanisms based on that data.

4.7.2.2 FDDI Support for "Smart" Sensors

A smart sensor system would be almost the same as that for the dumb sensors, except that the flight controller should communicate with fewer than 16 sensors or mechanisms at a time, thus leaving open a large token channel that smart sensors could use to report results of self checks and trends in their data.

4.7.2.3 FDDI Support of "Intelligent" Sensors

In systems using intelligent sensors with distributed control, a non-FDDI-II system would be preferable. FDDI's synchronous/asynchronous transmission features make it well suited for distributed control. The synchronous transmission could be reserved for normal operation of the ALS, with asynchronous transmission still available should problems arise requiring more communication than usual. Since all terminals are on the same ring, all sensors could easily communicate with each other, as well as to any mechanisms they may need to send instructions to. This ease of communication between any two terminals on the ring also makes FDDI well-suited for distributed control.

4.7.3 The SEAFAC Protocol

While SEAFAC currently exists only on paper, with no available hardware to support it, we nonetheless include it in this report to illustrate the desirability of an explicit token-passing bus. Should NASA choose to develop its own protocol for the ALS, we believe that it should develop an explicit token passing bus or a hybrid such as SEAFAC.

The SEAFAC protocol has excellent features for the use with all three (dumb, smart and intelligent) sensors, as well as being suited for star and bus networks, the most fault-tolerant topologies. SEAFAC's ability to dynamically prioritize terminals is particularly suited for a multi-stage launch system. For example, the terminals representing the third stage of a three-stage launch vehicle will be of low priority during the initial phases of a launch, but will increase in priority as the launch progresses. The SEAFAC scheduler has the ability to place them "low" on a path, giving them low priority, and move them up as they become more important. As stages drop off, their terminal can be removed from paths by the scheduler, saving it the time of having to check and realize that they are no longer present before moving on to the next terminals.

4.7.3.1 SEAFAC Support of "Dumb" Sensors

Being a high-speed token passing protocol, SEAFAC is well suited for the case of "dumb" sensors that produce only data, unable to categorize it as to its importance. The SEAFAC scheduler can act as a flight controller, or follow the instruction of another terminal acting as flight controller, and prioritize the sensors as to the importance of their data as described above. The scheduler's ability to send instructions to either specific terminals or broadcast to several terminals via the subaddress fields makes it well suited for the command/response set-up used by current dumb sensors acting as slaves to a flight controller.

4.7.3.2 SEAFAC Support of "Smart" Sensors

Should "smart" sensors capable of trend analysis and self-checks be developed, SEAFAC could still be used, since it has provisions for any terminal, upon receiving the token, to transmit a state-of-health message. Also, the ability to dynamically prioritize terminals proves useful. For example, if a terminal's self check shows that it may soon prove unreliable, the scheduler can place it low on a path, and not waste channel time by giving it the token very often. Also, if a terminal spots an important trend in its data, say data indicating an impending component failure, the flight controller may want to watch that terminal more carefully than usual in order that it may take necessary precautions. The scheduler would then put that terminal high on a path to insure that it receive the token often so that it may we monitored closely.

4.7.3.3 SEAFAC Support of "Intelligent" Sensors

SEAFAC is also well suited for "intelligent" sensors capable of making decisions and requiring an ability to send instructions to other components. SEAFAC is designed so that any terminal may know the state of health or may transmit or receive data from any other terminal on the network. Also, using the pre-defined sub-addresses, any terminal on the network may send instructions to any other terminal, and thus SEAFAC could support a distributed control system mandated by the use of intelligent sensors.

4.7.4 SAE AE-9B High Speed Token Passing Bus

The AE-9B token passing bus system, like SEAFAC, has no commercially available software to support it. Unlike SEAFAC, however, it is likely that manufacturers will soon develop such hardware, whereas with SEAFAC, NASA would have to initiate development.

It has been demonstrated that a centralized system such as SEAFAC offers higher throughput for a given data latency that for a "distributed" token passing bus such as AE-9B [SPIE86]. However, AE-9B does offer the advantages of a token passing protocol, along with the reliability and simplicity advantages of the bus architecture, and as stated above, there will probably soon be commercially available hardware to support it.

4.7.4.1 AE-9B Support of "Dumb" Sensors

For the current level of sensor technology, AE-9B offers the advantage of speed, and several user defined sub-addresses that a central flight controller could use to give commands to other systems on the bus.

Also, terminals representing sensor systems providing more important data may be given higher priority than others by giving them large THT reset values. Like SEAFAC, these priorities may be changed during the course of operation.

4.7.4.2 AE-9B Support of "Smart" Sensors

Terminals capable of self-checks can take advantage of exit token feature. A terminal that detects impending failure would simply issue an exit token, thus removing itself from the token path.

Terminals serving sensor systems capable of performing trend checks may wish to stay off of the token path until a trend worth reporting to the flight controller is spotted. That terminal would then accept the token during the next terminal insertion period (as described in Section 4.4.3.3).

4.7.4.3 AE-9B Support of "Intelligent" Sensors

Again, the advantage is the numerous sub-addresses available for user defined functions. This easily lends itself to a distributed control system mandated by the use of intelligent sensors. For example, upon receiving the token, terminals representing an intelligent sensor system that has determined that an action must be taken may use the sub-addresses to send instructions to an actuator. The sub-addresses may also be used to order other sensor systems to send data so that a decision may be made.

4.7.5 Recommendations

The MIL-STD-1553B command response bus should be sufficient to handle current and smart sensor technologies. It is proven, cheap and readily available. We do, however, recommend that the MIL-STD-1553B be upgraded to a MIL-STD-1773 system. MIL-STD-1773 uses the same access protocols as MIL-STD-1553B, but its hardware has been designed for use on optical fiber medium, rather than shielded twisted pair. Thus MIL-STD-1773 has all the advantages of MIL-STD-1553B, with the additional advantages of optical fiber, light-weight and lack of echo.

If an high speed system that can be implemented as soon as possible is desired, then FDDI is available. It is also designed for use on a light-weight fiber optic medium. It is very reliable in terms of data latency and guarantee of delivery, and its virtual circuit capabilities and synchronous/asynchronous make it attractive for use with present day, as well as future sensor technologies. Its ability to dynamically assign bandwidth makes it well suited for multi-stage launch systems.

If NASA is willing to develop the necessary hardware, SEAFAC is the protocol best suited for the advanced launch system. It's features allow it to support both centralized and distributed control systems. It is designed for use on the fault-tolerant and versatile star or bus topologies realized by a light-weight fiber-optic medium. Its ability to dynamically prioritize terminals makes it particularly well suited for multi-stage launch systems.

5.0 Perfomance Analysis

Inorder to determine the applicability of a given protocol to a system the protocol's performance must be studied. One of the most important performance factors is average delay. In the following sections offer load versus delay is plotted for several protocols. The data points for these plots were obtained through computer simulation of the protocols and from various studies of protocol performance.

5.1 Performance Analysis of Ethernet

The results in this section shown in Figures 5.2 and 5.3 were taken from the paper "A Simplified Discrete Event Simulation Mode for an IEEE 802.3 Local Area Network" by Sharon K. Heatley of the National Bureau of Standards [HEAT86]. They are the results of a computer simulation of the IEEE 802.3 protocol standard, which has the same timing features as Ethernet. A typical simulation timeline is shown in Figure 6.1. The protocol was modeled using the following rules:

1: The arrival of packets at each terminal is a Poisson distributed random process

2: The propagation delay between any two stations is constant. This would be the case for an Ethernet network on which the terminals are equally spaced along the propagation medium.

3: After a collision, all terminals involved go into a back-off period, the length of which is determined by the binary exponential backoff algorithm.

and the following parameters:

1. Slot time=51.2 μseconds
2. Interframe gap (I)=9.6 μseconds
3. Jam size (J)=3.2 μseconds
4. Maximum propagation delay=25.6 μseconds
5. 20% of packets 1024 bytes, 80% of packets 64 bytes

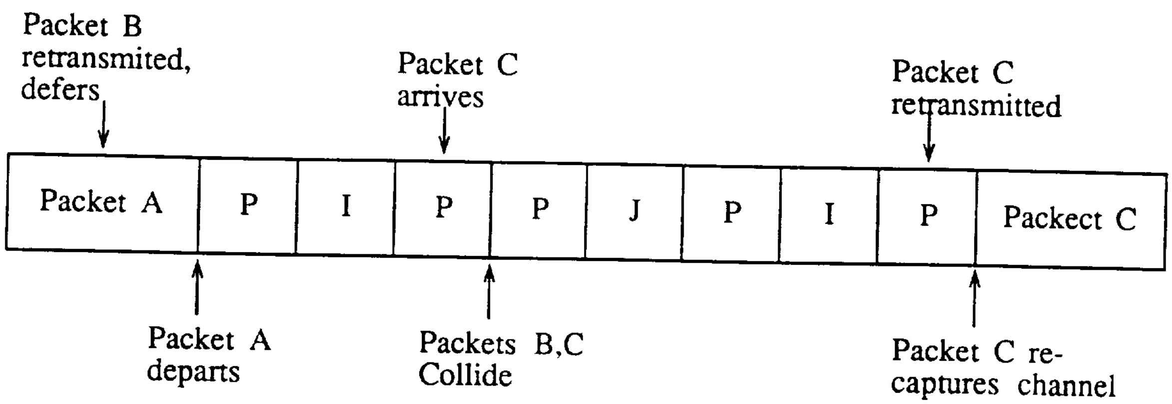


FIGURE 5.1 TYPICAL TIMELINE FOR SIMULATION MODEL [HEAT86]

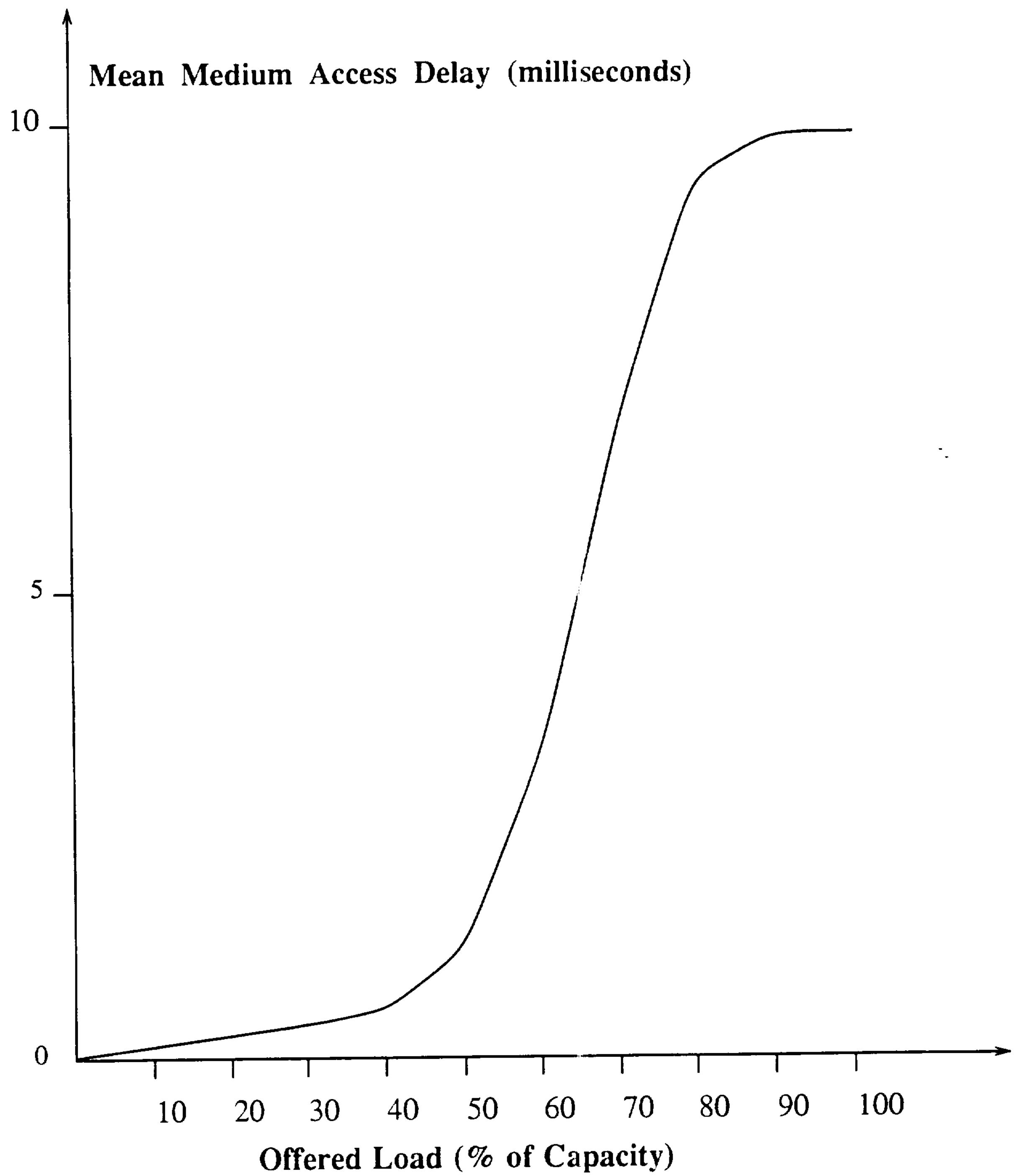


FIGURE 5.2 MEAN MEDIUM ACCESS FRAME DELAY VS. OFFERED LOAD FOR ETHERNET [HEAT86]

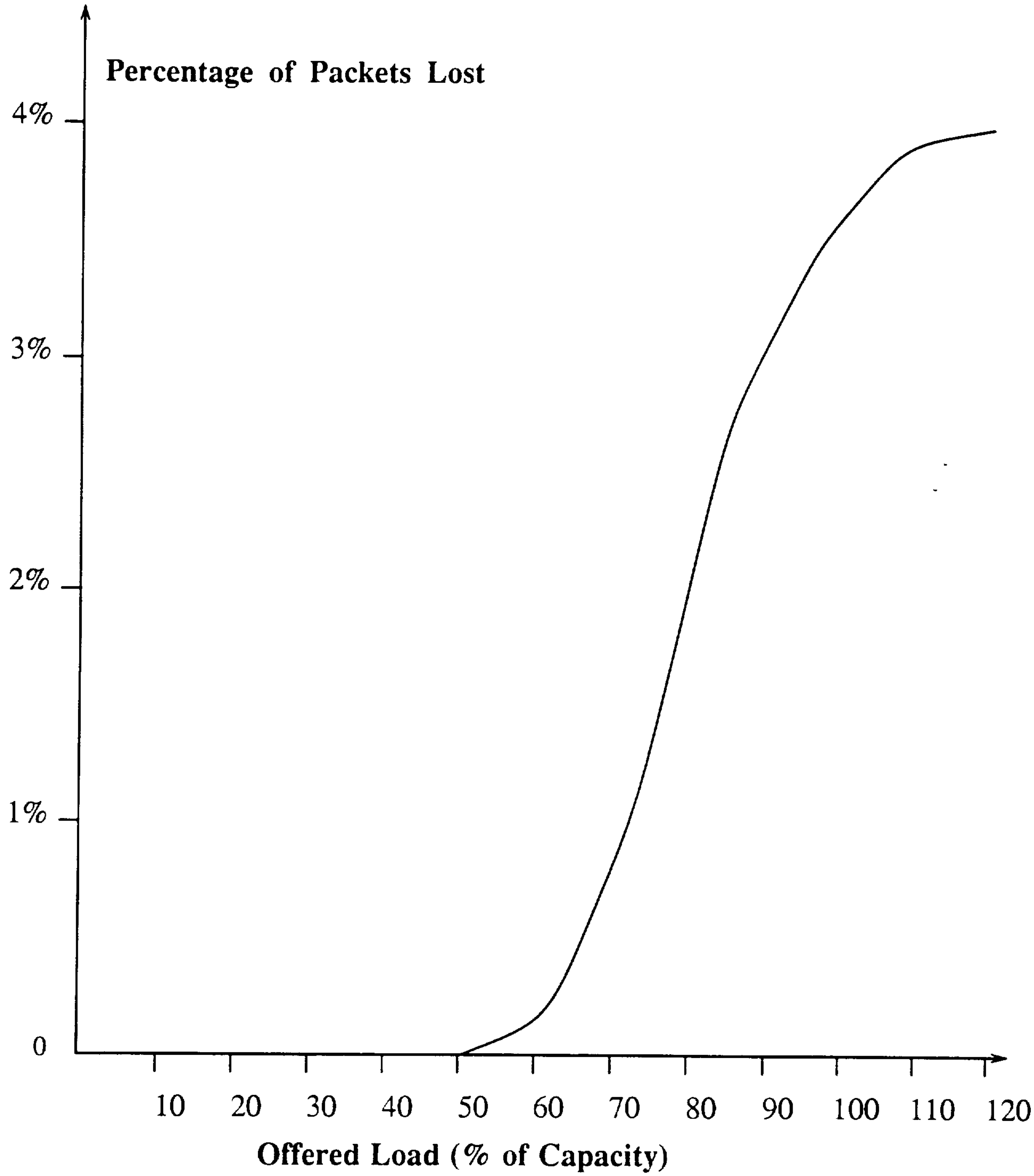


FIGURE 5.3 PERCENTAGE OF LOST PACKETS VS. OFFERED LOAD FOR ETHERNET [HEAT86]

5.2 Performance Analysis of HYPERchannel

This section gives the results of a HYPERchannel simulation presented in the paper "Measurement and Analysis of HYPERchannel Networks" by William R. Franta, and John R. Heath. A detailed description of the HYPERchannel protocol can be found in section 4.

The HYPERchannel network simulated was composed of six terminals connected to a 1000 foot bus. Three of the terminals were designated "senders", and served only as data sources. The remaining three terminals were designated "receivers", and served only to collect data. The "data" was generated by a thirty-two bit random number generator. Each node was provided with a seperate "seed" for the random number generator in order to avoid the repeated transmission of identical frames.

Several simulations of over 110,000 frame sequence transfers yielded results within 2% of each other. Figures 5.4 and 5.5 show the average delay normalized to "transmission period units", plotted against the percentage of 50 Mbits/second of the offered load. Figures 5.6 and 5.7 show the throughput, i.e., the trunk utilization versous the offered load.

It was also observed that the effect of the wait flip-flop was not what the designers of HYPERchannel expected. Instead of preventing the higher priorities from "hogging" the channel, it has reversed the priorities with every frame sequence. [FRAN84]

5.3 Performance Analysis Of FDDI

An in depth analysis of a system using a ring topology, FDDI protocol and 100 Mbit/sec rate is presented by Webster and Johnson [JOHN85]. The analysis is conducted through simulation, the rules which govern the system simulation are as follows:

1- Ring Structure: A dual-redundant ring structure (A&B) is modeled and the data is transmitted in the opposite direction on ring A referenced to ring B.

2- Station Count: The simulation can model an unlimited number of stations.

3- Distance Between Stations: The simulation can model a variable and unlimited physical distance between the stations.

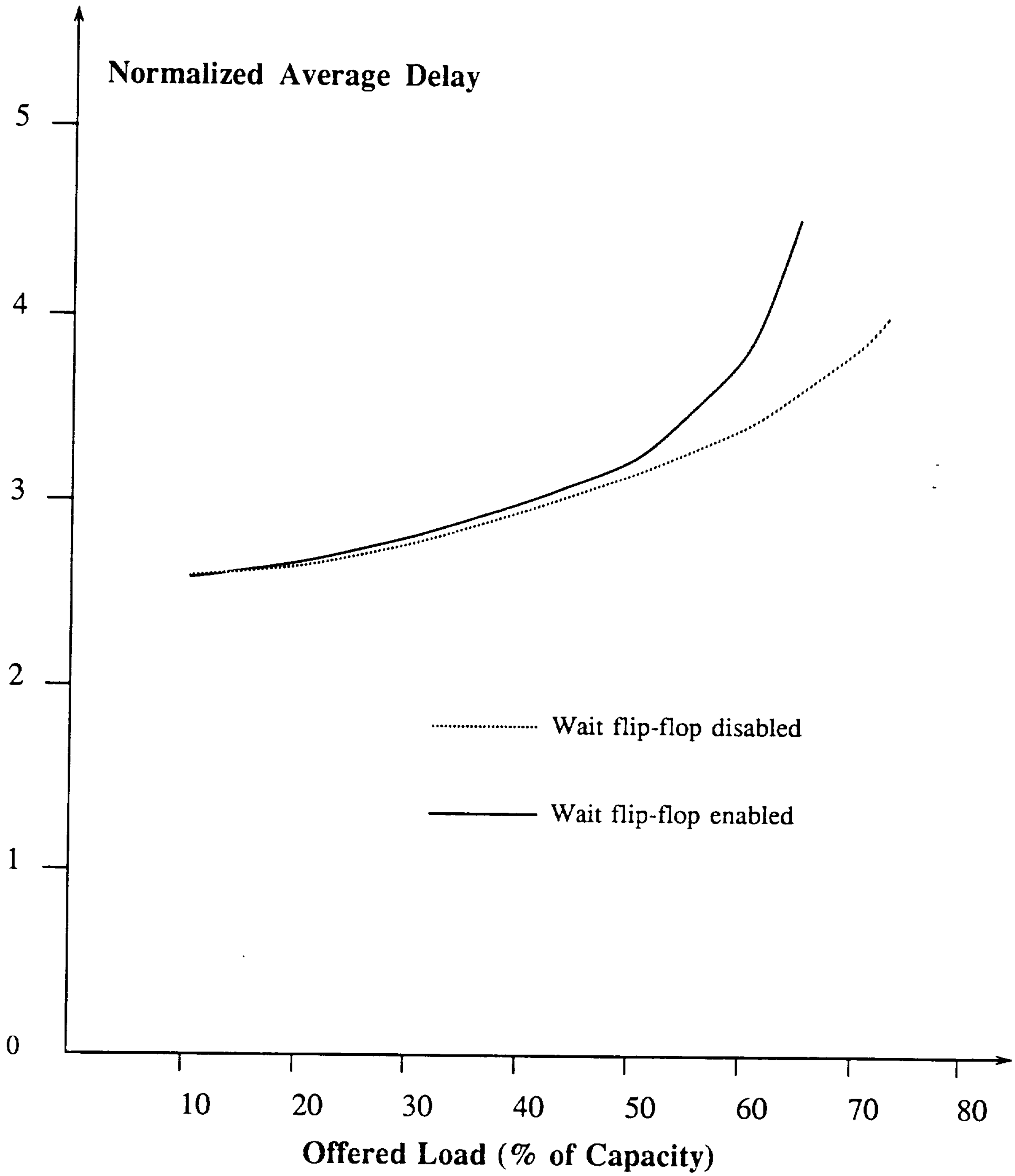


FIGURE 5.4 AVERAGE DELAY FOR 64-BYTE MESSAGE-ONLY SEQUENCES VS. OFFERED LOAD FOR HYPERCHANNEL

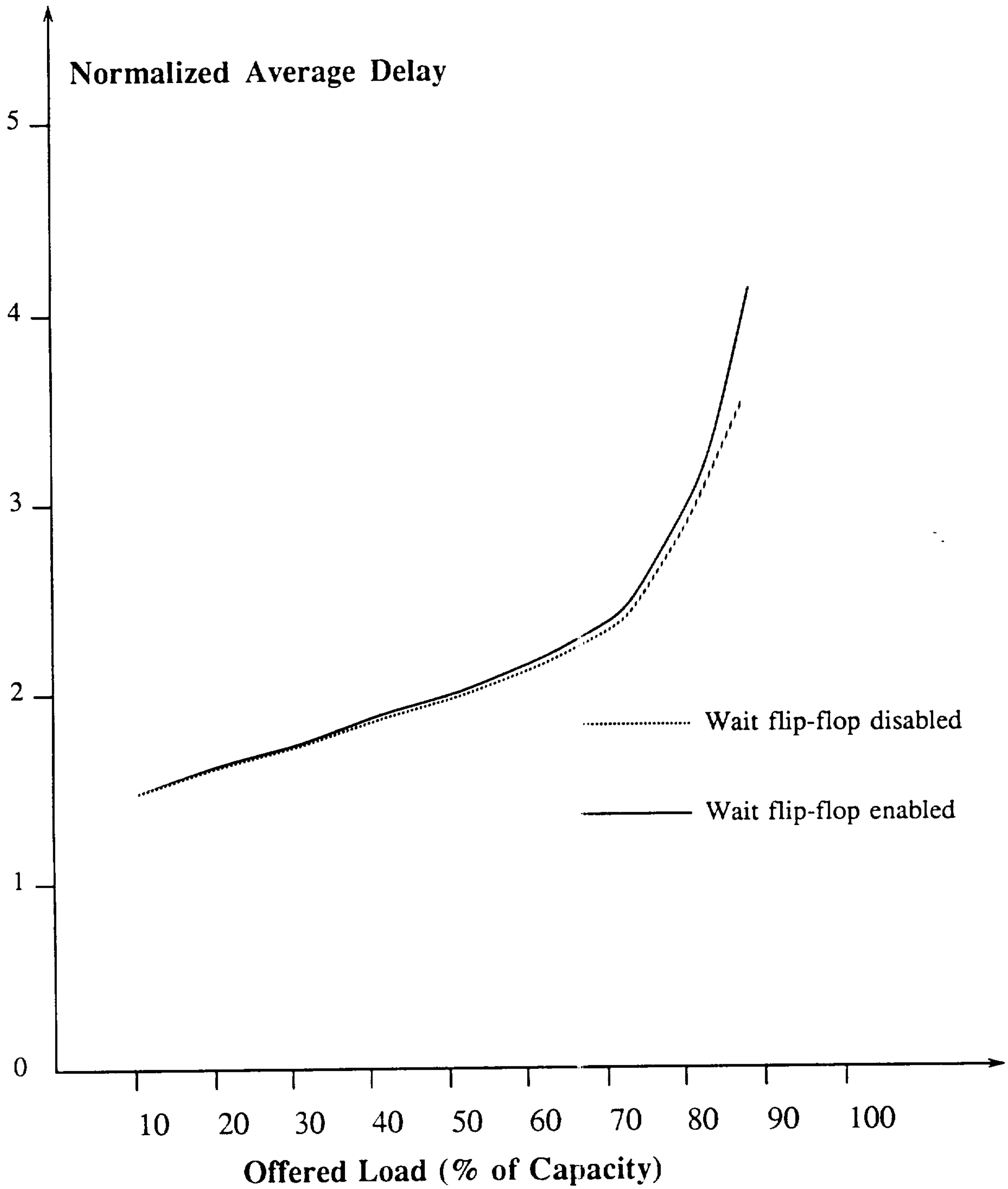


FIGURE 5.5 MEAN MEDIUM ACCESS FRAME DELAY VS. OFFERED LOAD FOR HYPERCHANNEL

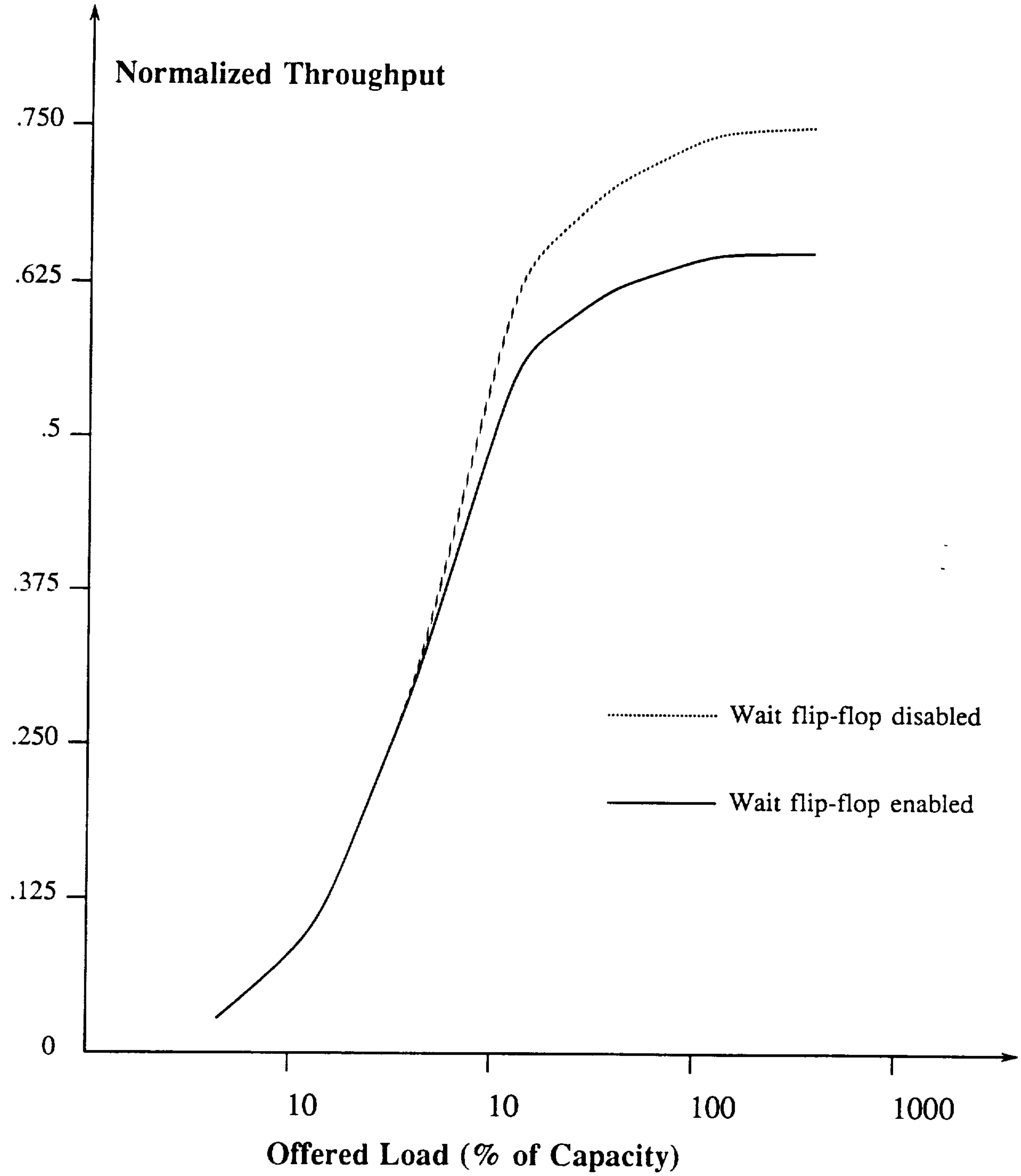


FIGURE 5.6 64-BYTE MESSAGE-ONLY SEQUENCE THROUGHPUT VS. OFFERED LOAD FOR HYPERCHANNEL

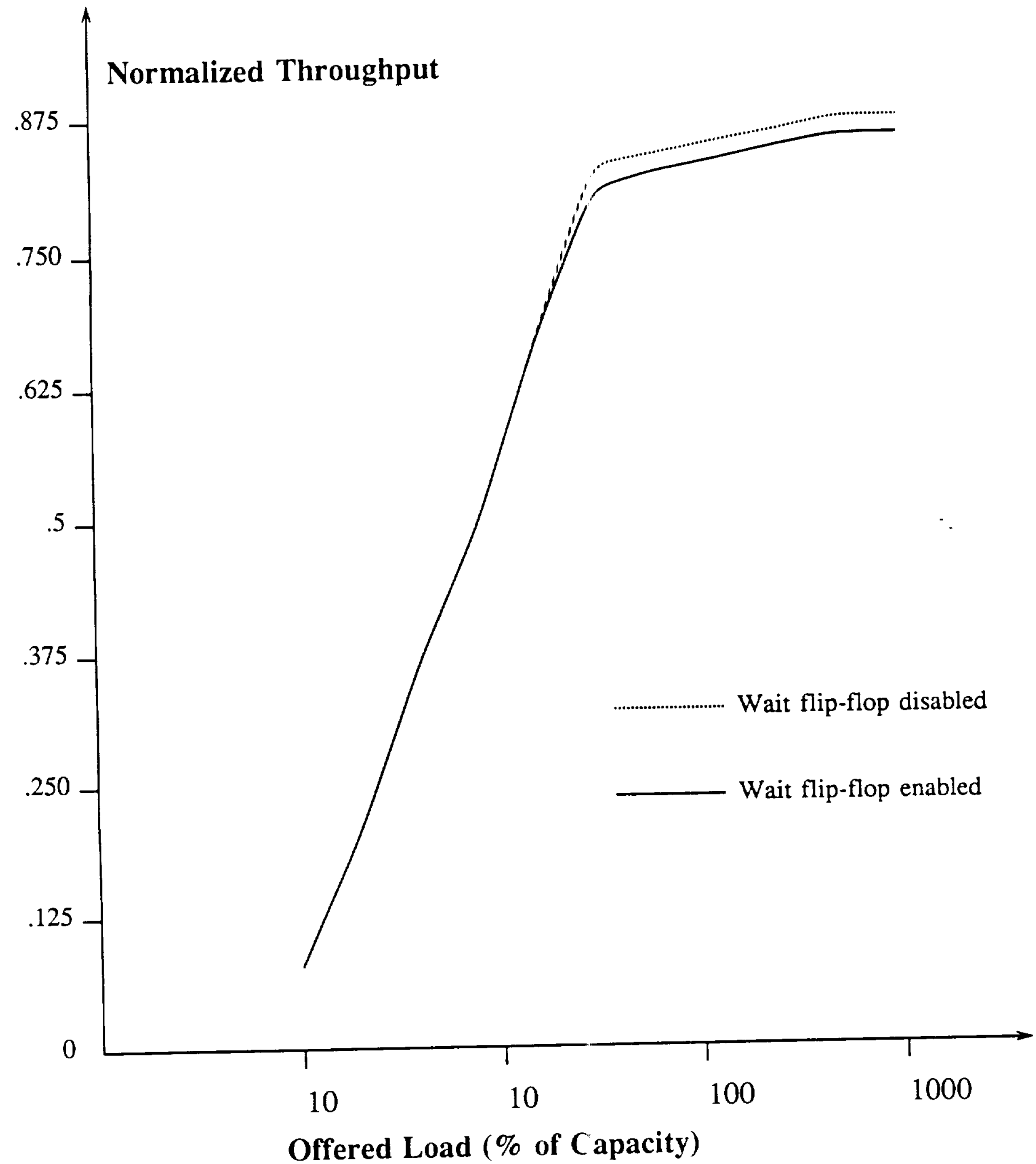


FIGURE 5.7 64-BYTE MESSAGE PROPER WITH 4 KBYTE DATA SEQUENCES THROUGHPUT VS. OFFERED LOAD FOR HYPERCHANNEL

4- Transmission media: In the simulation, data is transmitted between the stations on optical fiber cable.

5- Data Transmission Rate: The data rate is taken to be a selectable variable.

6- Relation Of Stations To Each Other: Each station communicates with one uplink station and one downlink station. In addition each station is capable of communicating on either of two redundant rings.

7- Elasticity Buffer: An elasticity buffer is present in each station. This buffer is used to maintain bit synchronization.

8- Frame Type: Synchronous and asynchronous frame types are included in the simulation model.

9- Free Token: In the simulation model the free token consists of 11 octets. An octet is eight serial bits.

10- Frame Structure: In the simulation model a frame structure consists of 20 octets. The Start of Frame Sequence, Frame Control Field, Address Fields, Frame Check Sequence, and End Of Frame Sequence are present in the frame. An information field consisting of no more than 4488 octets is present in the frame also.

11- Address Recognition: In the simulation each station is provided with a unique address. Each station is capable of recognizing its own address when present in the destination address field of a frame.

12- Frame Copied Indicator: This indicator is set by the simulator to indicate the frame was copied into the addressed station.

13- Error Detected Indicator: This indicator is set by the simulator to indicater a detected transmission error.

14- Valid Transmission Timer: This timer is needed if fault conditions are to be injected in the simulator.

15- Target Token Rotation Time: The (TTRT) is set in the simulator through negotiation as part of ring initialization.

16- Token Rotation Timer: The (TRT) is used in the simulator for ring scheduling and serious ring problem detection. Each station has its own (TRT).

17- Token Holding Timer: The (THT) controls the transmission of asynchronous frames. Each station has its own (THT).

18- Synchronous Timer: The synchronous timer is used by the simulator to control the transmission of synchronous frames. Each station has its own synchronous timer.

19- Capture Of Token: According to a set of rules, a station that has frames queued for transmission captures a token.

20- Token Passing: In the model the token is passed after all queued frames at token holding station have been transmitted.

21- Frame Removal: The station which transmits, in the simulation model, a frame is responsible for its removal from the ring.

22- Transmission Queue: The number of frames which may be contained in this queue is supplied by the user to the simulator.

23- Receive Buffer: A receive buffer is used in the model to copy frames from physical layer to the link layer. This buffer is contained in the link layer.

24- Frame Retransmission: The network layer in the simulation model has the ability to reschedule transmission of a refused frame. A refused frame is a frame returned to the sending station with its Frame Copied bit not set or with the Error Detected bit set.

25- Data Buffering: The network layer in the simulator provides buffering for both transmission and reception of frames and messages.

26- Received Message Buffer Space: In the simulation model, a user specified number of octets of storage exist at each station.

27- Transmit Message Buffer Space: A user defined number of octets of buffer space will exist at the network layer of the simulation model to store messages for transmission which originated in the load layer of the model.

28- Long Transmit Messages: In order to have a successful transmission, a message which is longer than the maximum information field length will be broken down into frames before transmission.

29- Receive Messages: Messages which have been completely loaded into the message buffer space will be passed to the load layer in the simulation model.

30- Message Acknowledgment: Message acknowledgment is an option provided by the model. It is a receipt-of-message acknowledgment which is sent from the receiving station to the message-transmitting station.

31- Frame Rejection: In the simulation model a receiving station can reject all frames from a transmitted message until space becomes available in its physical buffer space.

32- Ring Recovery: Ring recovery is modeled in the simulation by a short delay of time.

33- Message Generation: The load layer of the simulation model is responsible for message generation. The load layer, for example, is responsible for the generation of message type, length, destination, inter-arrival time and priority.

34- Load Types: At each station, in the model, the load is modeled as three distinct sub-loads. The first sub-load consists of short, control-type messages or load-level acknowledgments. The second sub-load consists of long, data-type synchronous messages. The third sub-load consists of long, data-type asynchronous messages.

35- Message Delivery To Load Layer: The transportation of a message from the network layer to the load layer of the model starts after a complete message has been copied into the network layer's receive message buffer.

EXAMPLE ONE :

Simulation results on the performance of FDDI are presented in a paper by Johnson [JOHN88]. The ring configuration used to obtain the results is presented in Table 5.1. The system is considered to be homogeneous and the traffic is taken to be asynchronous. Each node in the simulated network is assumed to generate frames at the same specified mean arrival rate. The Interarrival times for frames at each node is assumed to be exponentially

TABLE 5.1 RING CONFIGURATION FDDI EXAMPLE ONE	
PARAMETER	VALUE
Number of Nodes	20
Distance between Nodes	30 meters
T_Opr	40 milliseconds
Header Size	4000 bytes
Frame Size	40 bytes

distributed. In a system using FDDI, a timed-token-rotation protocol, the ring initialization process includes a negotiation between all the stations in the system. As a result of this negotiation a value for the target token rotation time (TTRT) is determined. (TTRT) specifies the expected token rotation time in the network. Each station requests a value that is fast enough to support its synchronous traffic needs. The shortest requested time is assigned to (T_Opr). The value of (T_Opr) specifies the expected token-rotation time and it is a well defined ring parameter. The main results from the simulation model for this example are the following:

<u>Average Frame Delay :</u>

The delay measured in the simulation is the time from generation of the frame at the source node to reception of the frame at the destination node. Figure 5.8 shows the average frame delay versus offered load.

<u>Channel Utilization :</u>

The utilization of the channel as function of offered load is presented in Figure 5.9. Utilization increases linearly as a function of the offered load until the network is saturated. For an offered load of 99.9% or more the utilization function becomes parallel to the X- axis.

<u>Queue Lengths :</u>

As soon as the frames are generated at a given node they are placed into the transmission queue, they remain there until they are transmitted on the channel. For our example, the number of frames in queue vs. offered load is shown in Figure 5.10. In this figure both average and maximum queue lengths as a function of offered load are presented. The maximum value given is the maximum over all the nodes, and since in our example the network is assumed to be homogeneous the average number of frames in the transmission queue at the individual nodes are all considered to be approximately the same.

From Figure 5.8 and Figure 5.10, at an offered load of 98% the average frame delay is about 5000 microseconds and the maximum number of frames in queue is 7. This information suggests that when the offered load is as high as 98% the ring is able to service all the traffic

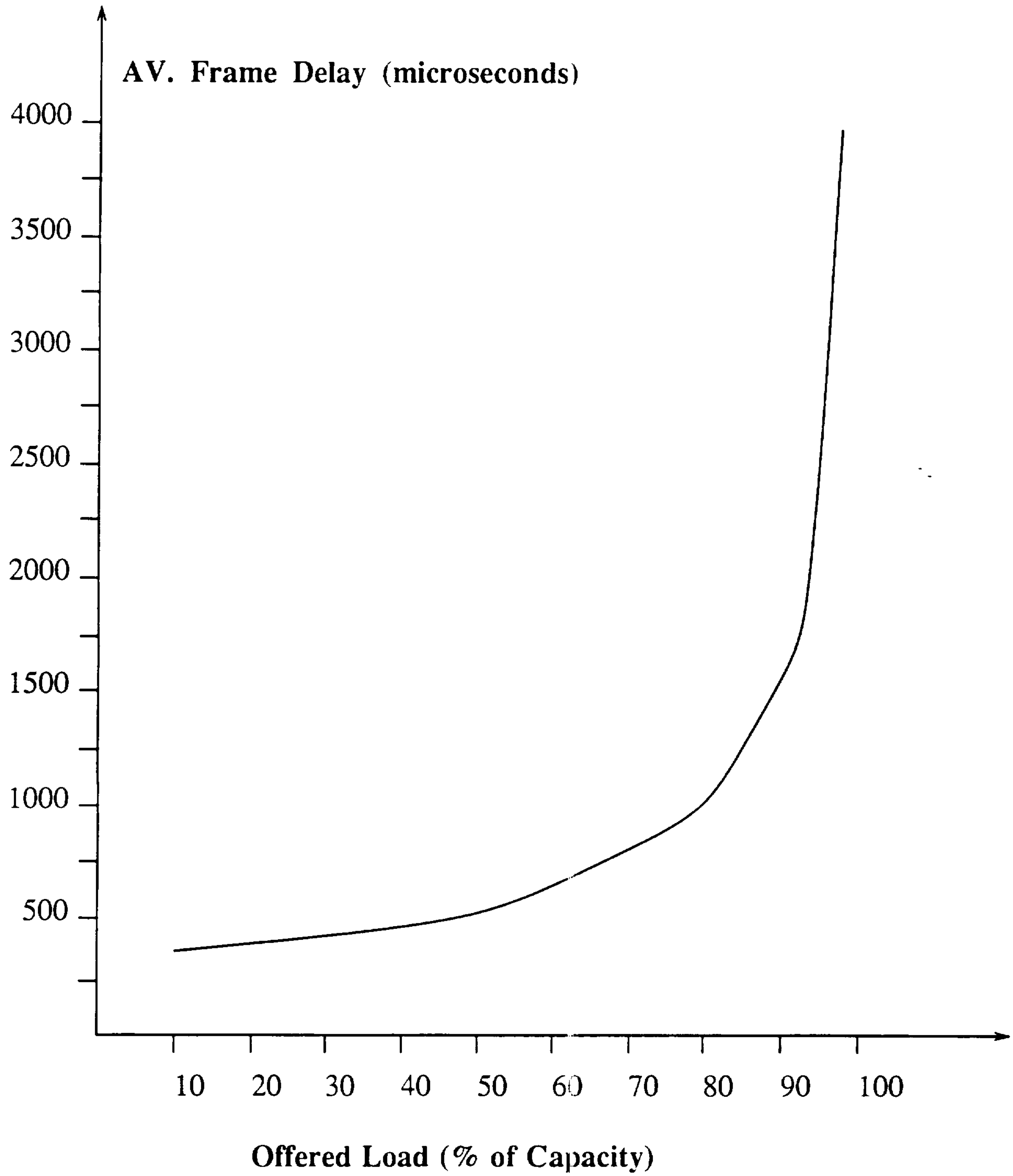


FIGURE 5.8 AVERAGE FRAME DELAY VS. OFFERED LOAD FOR FDDI

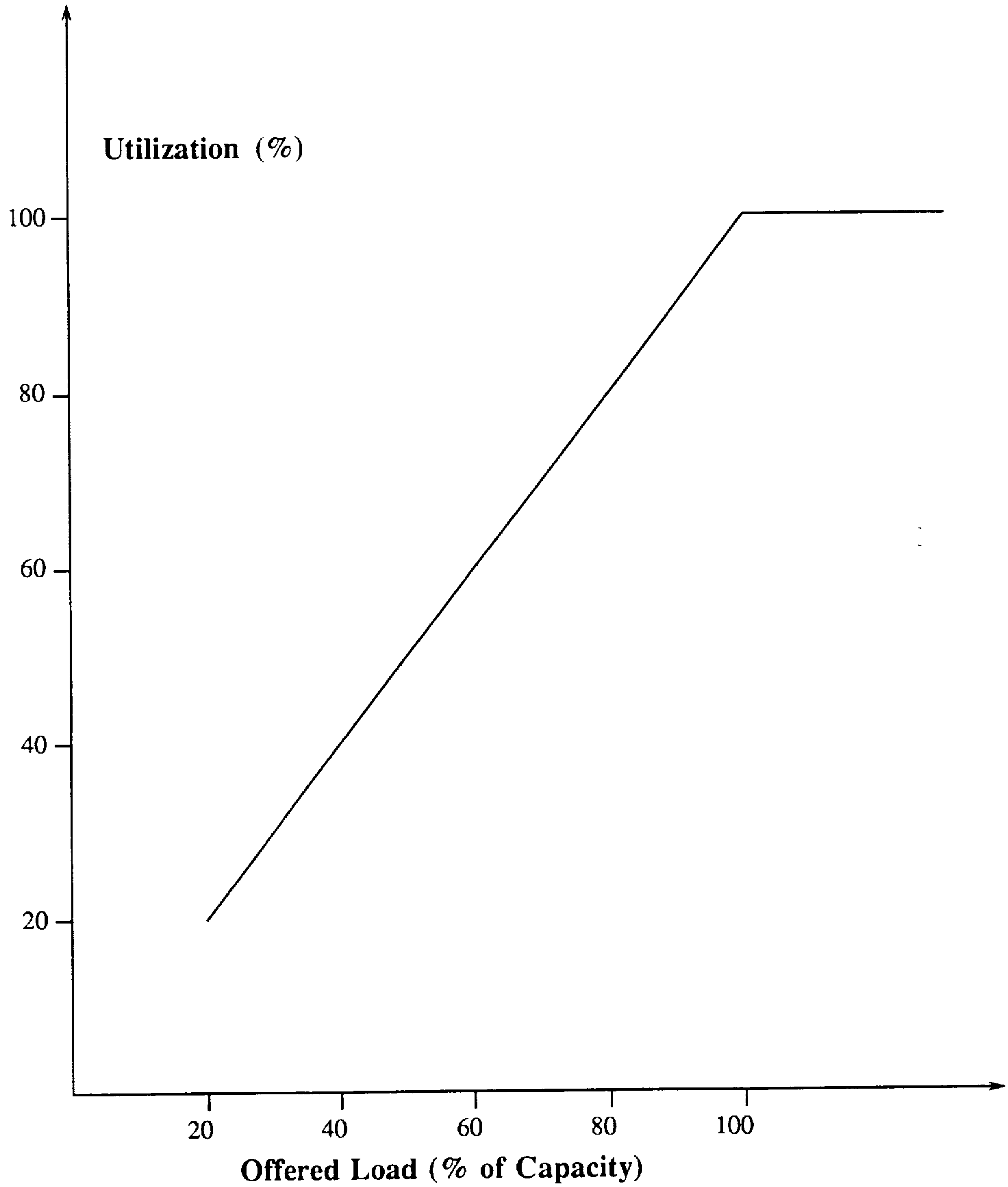


FIGURE 5.9 UTILIZATION VS. OFFERED LOAD
FOR FDDI

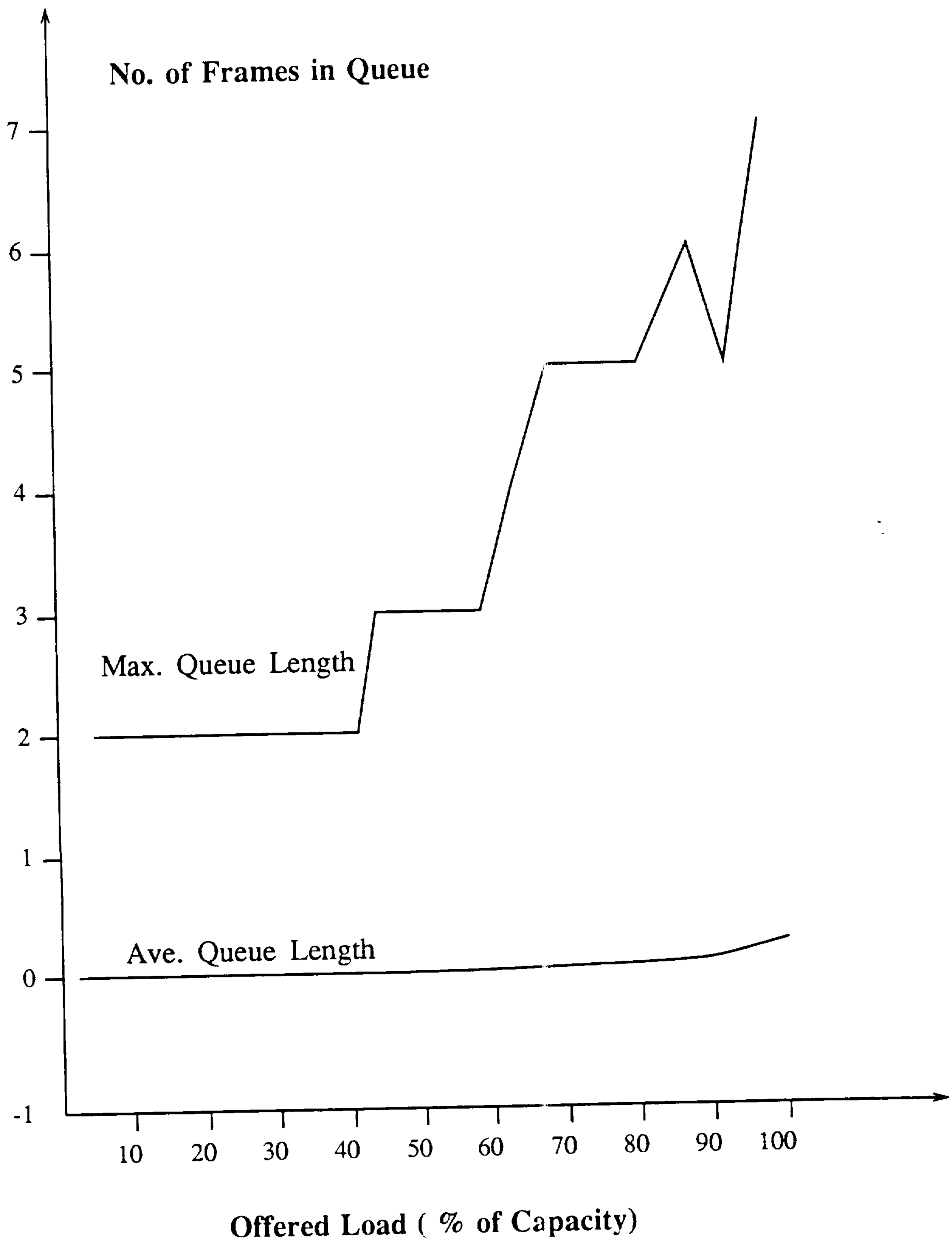


FIGURE 5.10 QUEUE LENGTH VS. OFFERED LOAD FOR FDDI

satisfactorily.

Average Token-Rotation Time :

For this example the time required for the token to rotate around an empty ring is 15 microseconds(propagation delay). It has been proven that the maximum token-rotation time is 2x(T_Opr) [SEVI87], also it has been proven that the average token-rotation time is less than or equal to (T_Opr) [Sevick & Johnson]. Figures 5.11a and 5.11b show the average token-rotation time as a function of the offered load. It can be seen from the mentioned figures that the average token-rotation time approaches the value of (T_Opr) [in this example (T_Opr) = 40 microseconds] only when the offered load exceeds the capacity of the ring.

Waiting Period For A Usable Token :

The amount of time a node must wait to be serviced when it has one or more frames queued for transmission is another measure of FDDI responsiveness. For the example considered, Table 5.2 illustrates both average and maximum values of waiting time for a range of offered load.

In the next example synchronous traffic is taken in consideration. The example and the result of the analysis are taken from a paper by Marjory J. Johnson [JOHN88].

EXAMPLE TWO :

In this second example a ring configuration, presented in Table 5.3, is simulated using the simulation model presented by Webster and Johnson [JOHN85]. In this configuration the asynchronous nodes generate asynchronous traffic only and the synchronous nodes generate synchronous traffic only. In this example, a synchronous node generates one frame every 6750 microseconds. On the other hand the interval between consecutive asynchronous frames generated at different asynchronous nodes is staggered. It is desired that any given frame at a given synchronous node should be transmitted before the queueing of the next frame at the same node for transmission. Each synchronous node is allocated synchronous bandwidth to transmit exactly one synchronous frame each time it receives the token, this

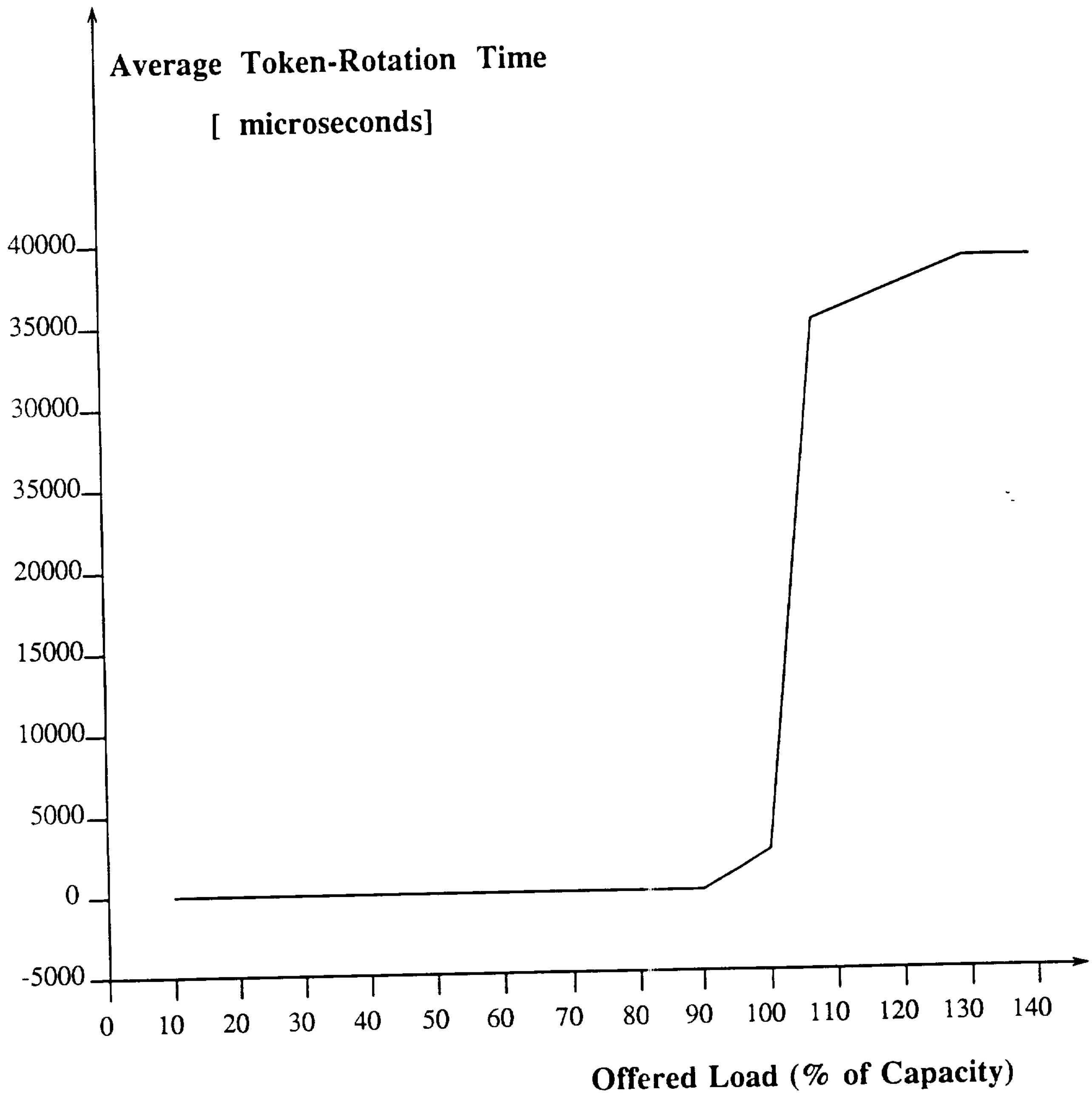


FIGURE 5.11a AVERAGE TOKEN- ROTATION TIME VS. OFFERED LOAD FOR FDDI.
[COMPLETE GRAPH]

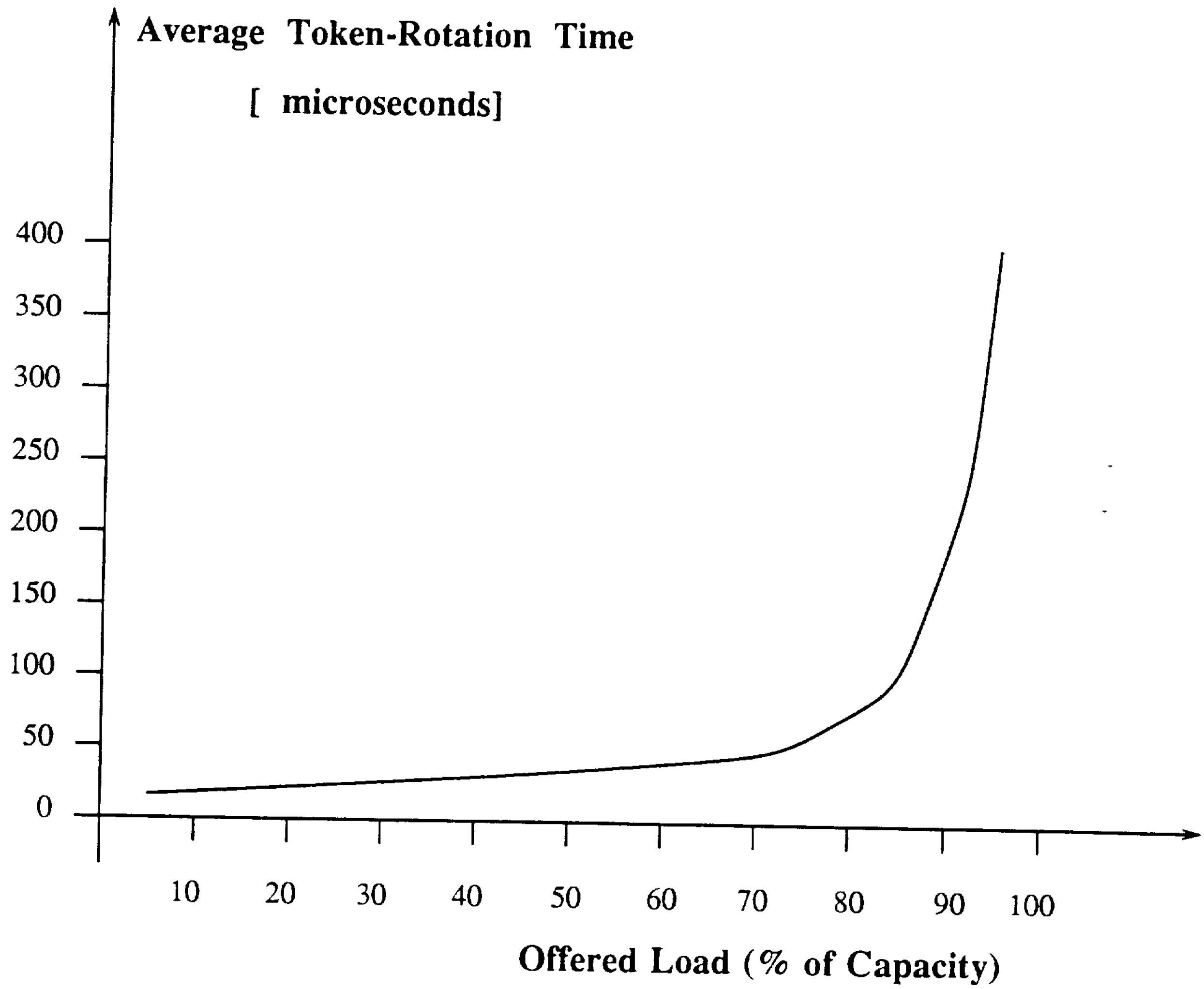


FIGURE 5.11b AVERAGE TOKEN- ROTATION TIME VS. OFFERED LOAD. [ENLARGEMENT OF BOTTOM PORTION OF FIGURE 4a GRAPH]

TABLE 5.2 FDDI WAIT FOR USABLE TOKEN		
Offered Load (% of capacity)	Average Wait (microseconds)	Maximum Wait (microseconds)
10	30	509
20	47	968
30	76	1328
40	133	1913
50	151	2189
60	309	5723
70	421	4800
80	650	6097
90	1367	9469
95	2348	13244
105	30246	38493

TABLE 5.3 RING CONFIGURATION FOR FDDI EXAMPLE TWO	
PARAMETER	VALUE
Number of Synchronous Nodes	15
Number of Asynchronous Nodes	5
Interarrival Time Between Synchronous Frames	6750 μs
Distance Between Nodes	30 meters
T_Opr	6750 μs
Length of Synchronous Access Time Interval	6750 μs
Synchronous Bandwidth Allocation	75%

condition is the reason why 75% of the value of (T_Opr) is allocated for the total synchronous bandwidth.

Synchronous Service :

During the simulation, the synchronous load was increased until the total offered load was 95%. The ring performance was satisfactory for this range of offered loads. This is because even at this high level of offered load all synchronous frame delays are less than 6750 microseconds. On the other hand when the asynchronous load was increased so that a total offered load of 120% the resulting ring performance was not satisfactory. Approximately 3.2% of the synchronous frames experienced delays that exceeded 6750 microseconds. Figure 5.12 illustrates a histogram of synchronous frame delays for node 12 . Delays greater than 6750 microseconds are shown to occur in clusters, this type of delay for one frame will cause frames to back up in the transmission queue. Since in this example the ring configuration allows only one synchronous frame to be transmitted during each token rotation, since it has been mentioned the token-rotation time in a saturated ring approaches (T_Opr), and since an additional frame is added to the queue every (T_Opr) microseconds, it may take several token rotations before the queue becomes empty again. From Figure 5.12, node 12 of this example experiences five clusters of excessive delays. These clusters can be eliminated by purging a synchronous frame pending transmission when a new synchronous frame becomes queued for transmission at this node. As a result of this purging technique only five synchronous frames will be lost from node 12, the rest of the frames will experience delays within the acceptable bound.

EXAMPLE THREE :

In this example a ring configuration of 20 homogeneous nodes transmitting asynchronous frames only at an offered load exceeding the ring capacity (saturation) is simulated. The " Fairness of Access for Asynchronous Traffic " is the only outcome of concern from this experiment. The result is shown in Figure 5.13 in this Figure a histogram of the number

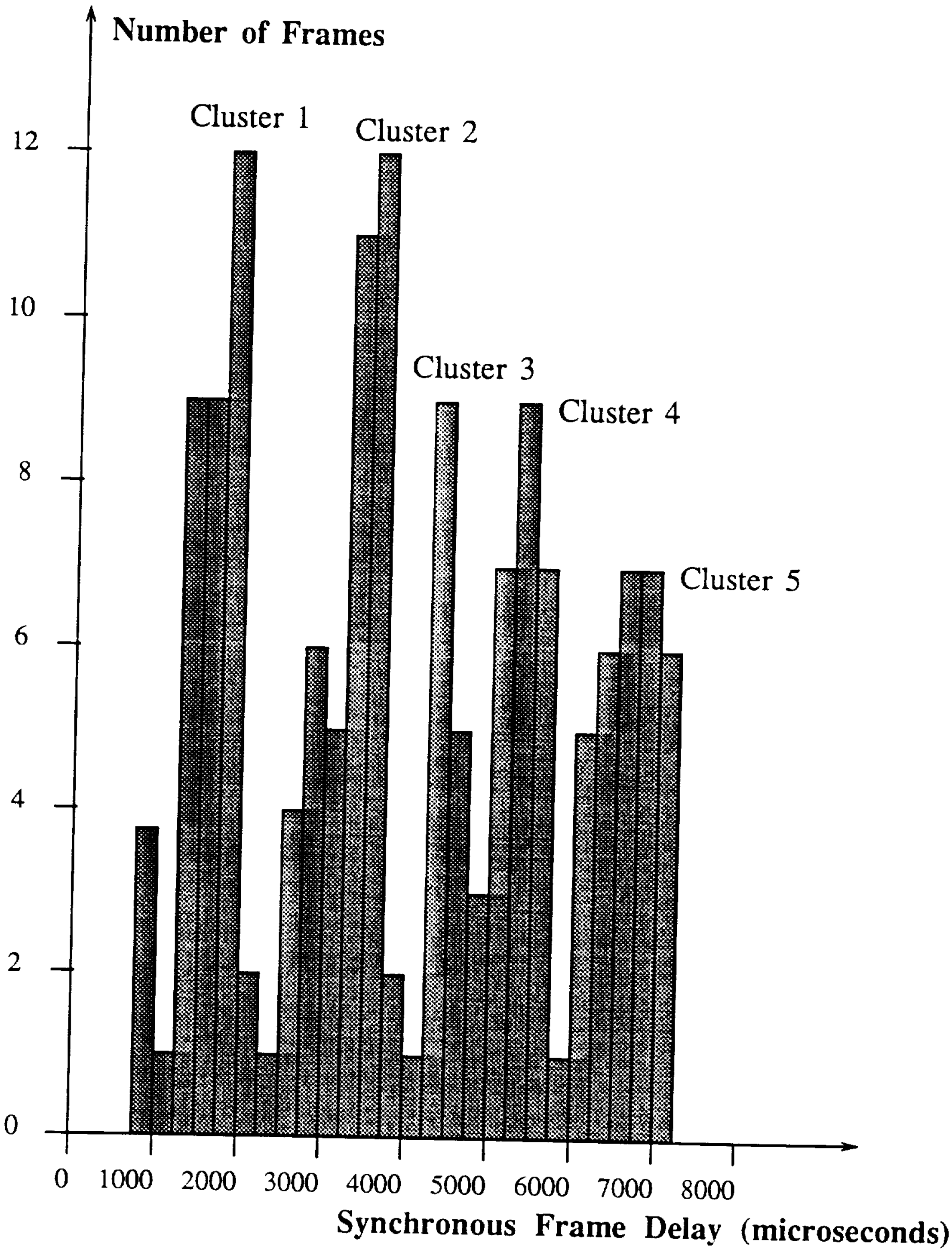


FIGURE 5.12 FREQUENCY DISTRIBUTION OF SYNCHRONOUS FRAME DELAYS FOR NODE 12

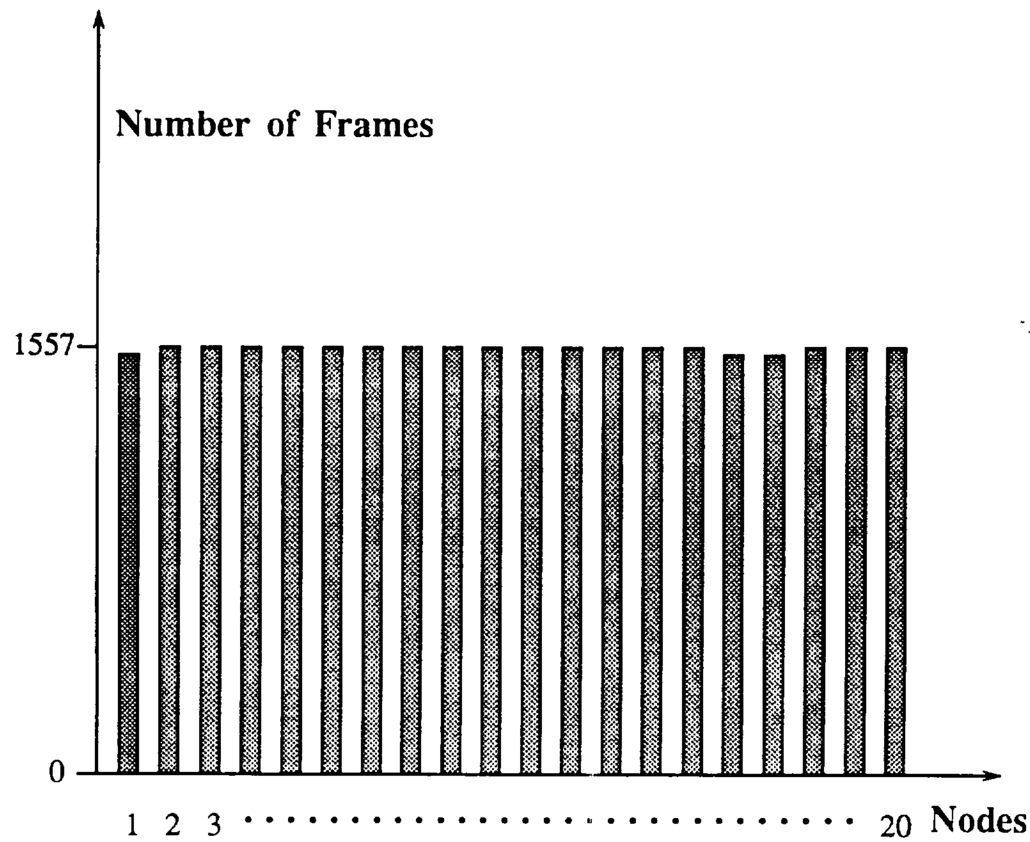


FIGURE 5.13 NUMBER OF ASYNCHRONOUS FRAMES TRANSMITTED BY INDIVIDUAL NODES (HOMOGENEOUS RING)

of frames transmitted over a period of 10 seconds by each node in the ring is presented. The largest number of frames transmitted by a single node was 1557 and the smallest number of frames transmitted by a single node was 1530. The ring operation in this example is time division multiple access (TDMA), with a six-frame time slot for each node during each token rotation. This represents the most efficient channel utilization in a saturated ring.[JOHN88]

EXAMPLE FOUR :

In Marjory Johnson's " Proof That Timing Requirements Of The FDDI Ring Protocol Are Satisfied. IEEE Trans. on Com. Vol. 35 No. 6, June 1987 " a realistic situation is presented. It is given in this presentation that the propagation time for fiber optic media is 5085 ns/Km, the latency per physical connection is 600 ns, the token transmission time is 0.00088 ms, and the maximum transmitter idle time after token capture is 0.0035 ms. Using the information given in Table 5.1 for this example also, the timing values are as follows:

1- Total propagation delay around the ring = (5085 x 20 x 30 / 1000) ns

2- Total latency due physical connections = (600 x 20) ns

3- Maximum overhead due to token transmission time = (0.00088 x 20) ms

4- Maximum overhead due to transmitter idle time after token capture = (0.0035 x 20) ms

5.4 Comparative Results from Analytic and Simulation studies of CSMA/CD & Ring Protocols

1- Comparison between the delay characteristics of the token ring, slotted ring, the register insertion, and CSMA/CD protocols : [Liu, M. T.; Hilal, W.; and Groomes, B. H. " Performance Evaluation of Channel Access Protocols for Local Computer Networks." *Proceedings, COMPCON 82 Fall*, 1982.]

1- Conditions under which the comparison was conducted:

- Normalized transmission media = 0.005
- Register insertion ring packet are removed by the destination station.

- Slotted ring and Token ring packets are removed by the source.

2- Results are shown in Figure (1).

3- The following can be concluded from the results:

- At light loads, the token ring suffers greater delay than CSMA/CD.

- At heavy loads the token ring protocol suffers less delay than CSMA/CD. At heavy loads the token ring's delay appears to be stable.

- Token ring delay characteristics are better than that of slotted ring at a given offer load.

- Slotted ring has the poorer performance. The reason for this may be that the overhead occupies a great portion of the used small slots and/or the time needed to pass the empty slots around the ring is significant. The passing of the empty slot in this topology is used to guarantee fair bandwidth in the system.

2- <u>Comparison between the delay characteristics of the token ring and CSMA/CD protocols :</u> [Okada H.; Yamamoto T.; Nomura Y.; Nakanishi Y. "Comparative Evaluation Of Token Ring And CSMA/CD Medium Access Control Protocols In LAN Configurations." IEEE 1984.]

1- Conditions under which the comparison was made:

- Channel Capacity = 5 Mbps.,

- Number of nodes used = 50.

- Maximum distance covered = 1.0 Km.

- Packet length = 1000 bits.

- Repeat delay at each node = 8 bits.

- Token header length = 24 bits.

- Maximum length of contention to control = 7. (CSMA/CD binary exponential back-off)

2- Results are shown in Figure (2).

3- The following can be concluded from the results:

- At light throughput values, CSMA/ CD protocol experiences less delay than the token

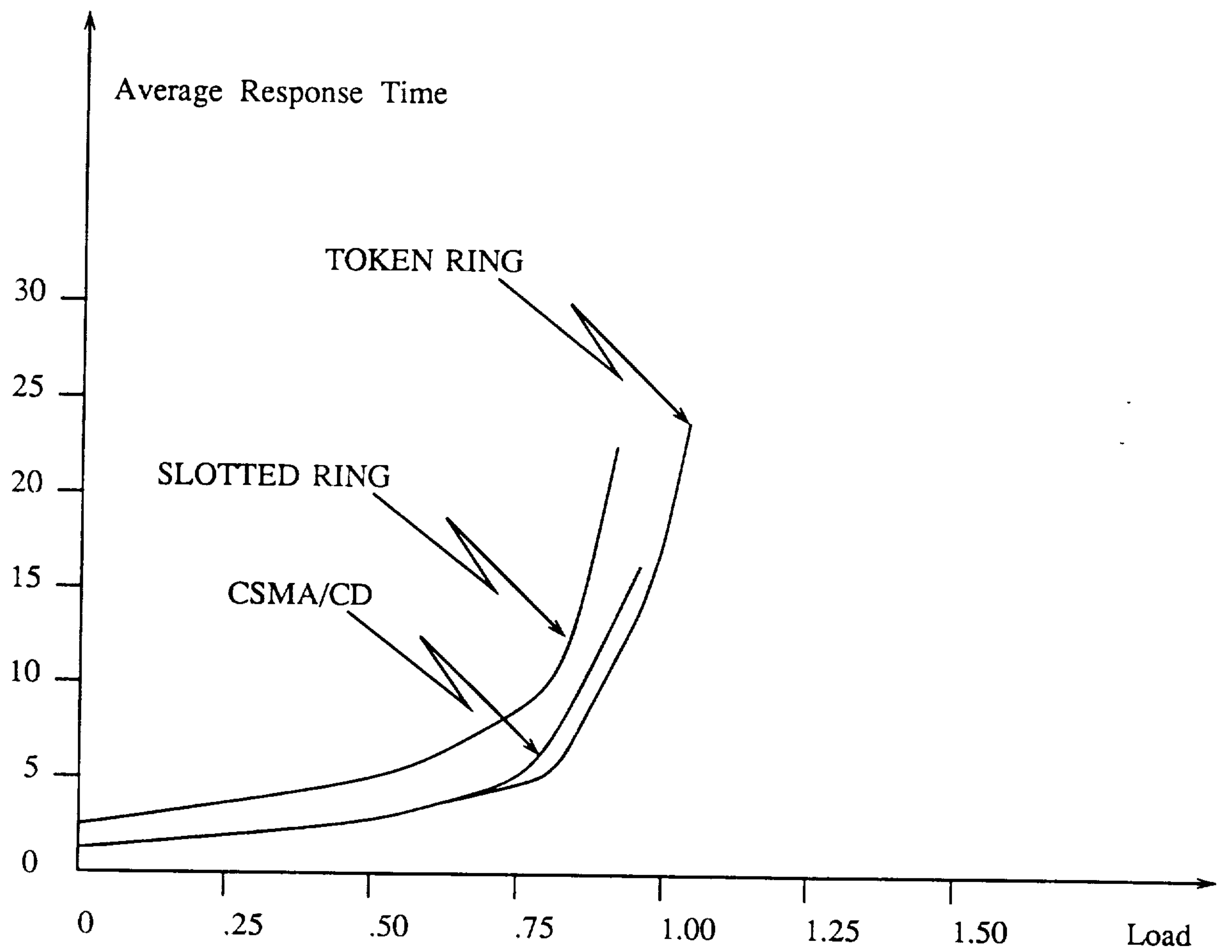


FIGURE 5.14 AVERAGE RESPONSE TIME COMPARISON

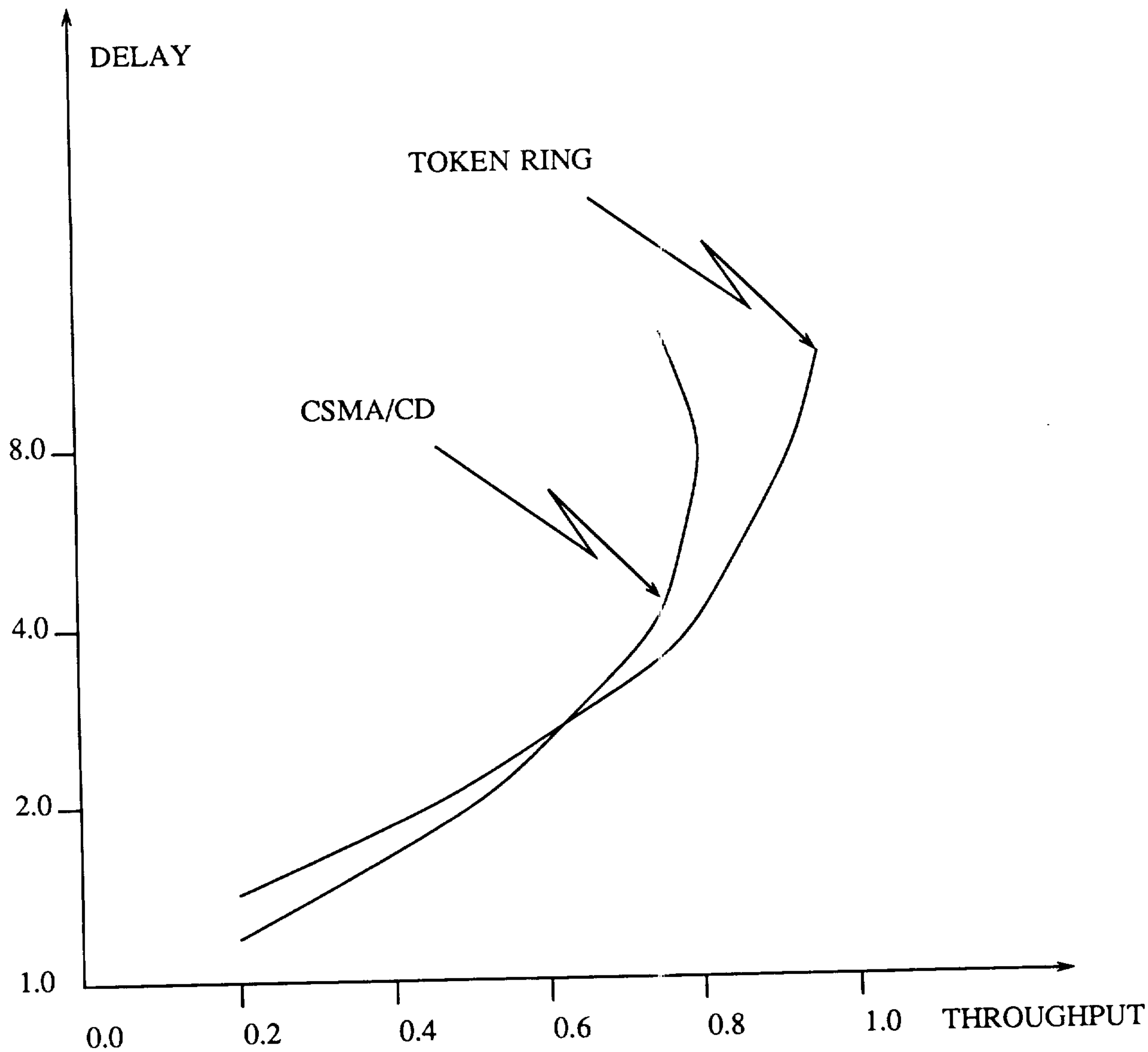


FIGURE 5.15 THROUGHPUT-DELAY CHARACTERISTICS COMPARISON.

ring protocol.

- At heavy throughput values, the token ring protocol experiences less delay than the CSMA/CD protocol.

- For Values of 0.5 - 0.8 normalized throughput, the rate of change in the token ring protocol delay is less than that of the CSMA/CD protocol.

5.5 Performance Comparisons for ALS Parameters

One of the first parameters to be considered in designing a communications system is the total offered load. This value should be commpared with the channel capacity C of applicable protocols. A simple method to find the upper bound for the average number of bits per second a node in a K node homogeneous system can output with a given protocol is described below.

Given K nodes and S samples per second, let N be the number of bits per sample that a node outputs,. Therefore the bound on N can be calculated using the following relation:

$$N \text{ (bit/sample)} \times S \text{ (sample/sec)} \times K \text{ (number of nodes)} < C \text{ bit/sec}$$

For an example assume a 100 megabit FDDI system with 10 nodes each producing N bits when sampled and all sampled 100 times a second. Therefore the bound on N can be calculated using the following relation:

$$N \text{ (bit/sample)} \times 100 \text{(sample/sec)} \times 10 < 10^{8} \text{ bit/sec}$$

or $N < 100 \times 10^3$ bit/sample.

Therefore for a successful transmission in a system using a 100 Mbit/sec FDDI protocol the output of a node must not exceed 1 Mbit/sample. Table 5.4 shows a related example using a ring configuration of 10 nodes.

Using this type of analysis the allowable node traffic for a ten node system is presented for MIL-STD-1553B, Ethernet and Hyperchannel in Table 5.4.

TABLE 5.4 : ACCEPTABLE NODE OUTPUT FOR HOMOGENEOUS 10 NODE SYSTEM

EACH NODE AVERAGE OUTPUT IN BITS/SAMPLE (100 SAMPLES/SEC)	A 1 Mbit/sec MIL-STD-1553B WILL SUPPORT	A 10 Mbit/sec ETHERNET WILL SUPPORT	A 100 Mbit/sec FDDI WILL SUPPORT
0.16 K	**YES**	**YES**	**YES**
1.6 K	**NO**	**YES**	**YES**
16 K	**NO**	**NO**	**YES**
160 K	**NO**	**NO**	**NO**
1.6 M	**NO**	**NO**	**NO**

Using this type of analysis and the information presented previously Table 5.5 is constructed to indicate the appropriate protocols and average delay for offered loads of 100 kilobits/sec (present vehicles), 22.4 megabits/second (Boeing Air Force study) and 55 megabits/second (Mississippi State University's proposed future vehicle load study with case 2 radar). The delays are not given exactly because the delays are very sensitive to average packet size and for the two rings the delays also depend on the number of stations and distance between stations.

TABLE 5.5 : AVERAGE DELAY FOR LOADS OF INTEREST						
OFFERED LOAD / PROTOCOL	PRESENT AVIONICS LOADS 100 Kbit/sec		AF/Boeing Estimates NO RF/RADAR INPUTS 23 Mbit/sec		MSU Estimates Assumes RF and CASE 2 RADAR INPUTS 55 Mbit/sec	
	WMCC*	Delay	WMCC	Delay	WMCC	Delay
MIL-STD-1553B	YES	N/A	NO		NO	
ETHERNET	YES	< 1 ms	NO		NO	
HYPERCHANNEL	YES	< 1 ms	YES	< 1 ms	NO	
PRONET - 80	YES	< 1 ms	YES	< 1 ms	YES	< 1 ms
FDDI	YES	< 1 ms	YES	< 1 ms	YES	< 1 ms

* WMCC Within Maximum Channel Capacity

From the table it can be seen that all of the protocols we have studied are applicable to the present generation of vehicles in terms of delay. As avionics data rates grow a change to protocols able to serve these higher rates. The products currently available to serve these high rates are token passing rings such as the Pronet - 80 and FDDI.

6 ALS Recommendations

As shown in section 1 future local area networks for launch vehicles will be required to service missions with data rates from 25 to 55 megabits/second. In order to keep costs down these systems should use commercially available standard parts wherever possible. Several high speed protocols presented by the Society of Automotive Engineers would be applicable to this system. These protocols are the SAE-AE/9B HSRB and SAE-AE/9B LTPB. These protocols were developed for initial implementation at 50 megabit/second with the ability to increase speed as faster hardware becomes available. These protocols suffer from a lack of available hardware at this time but both Boeing and Lockheed are working on implementations. Other protocols that are applicable are Pronet-80 and the FDDI protocol. This FDDI protocol is a 100 megabit/second token passing ring that has been selected by NASA for use on the space station and by the Navy for use on ships and at shore facilities. This protocol is supported in hardware by Martin-Marrieta and Honeywell. Many other companies are beginning to support this protocol. Greg Chesson of Protocol Engines Inc/Silicon Graphics Inc. is developing Express Transfer Protocol (XTP) to efficiently use the FDDI system. This lightweight ptotocol will be implemented in hardware and should allow an effective trhoughput of 80 megabits/second on a 100 megabit/second FDDI system. This protocol is being developed with military applications in mind and should soon be available.

As shown in section 2 a bus architecture would be most suitable for a launch vehicle with a separate LAN serving each stage. The MIL-STD-1553B protocol should be able to service the next several launch vehicles with a possible move to MIL-STD-1773 for higher data rates. When these protocols' abilities are exceeded then SEAFAC or SAE-AE/9B would be the next protocols to consider. If these protocols still do not have hardware available when their performance is required then a change to a high speed token ring such as FDDI will be necessary. This protocol is already well supported and should be thoroughly tested for reliability by the time it is needed.

Appendix A Local Area Network Comparison

A.1 Comparison Tables

In the following pages comparisons are made between many commercially available LANs. Such comparisons are made in access type, data rate, maximum number of nodes, and the maximum length of the LAN considered.[Data sources]

PRECEDING PAGE BLANK NOT FILMED

PAGE 156 INTENTIONALLY BLANK

TABLE A.1 LANs COMPARISON					
COMPANY & LAN NAME	Token Passing	CSMA / CD	CSMA	CSMA / CA	Other
AMECOM Government / Military UBITS (Universal Bus Information Transfer System)		X			
APOLLO COMPUTER DOMAIN Distributed Opreating Multi-Access Network	X				
APPLE COMPUTER AppleNet		X			
APPLITEK UniLINK	X	X			
CODEX 4000 Series LAN		X			
COMPLEX SYSTEM XLAN			X		
COMPUTER AUTOMATION COMMERCIAL SYSTEMS DIVISION SyFAnet		X			
CONCORD DATA SYSTEMS Token / Net	X				
CONTEL INFORMATION SYSTEMS ConTelNet		X			
CORVUS SYSTEM Corvus Omninet				X	

TABLE A.1 CONTINUE					
COMPANY & LAN NAME	Token Passing	CSMA / CD	CSMA	CSMA / CA	Other
CR COMPUTER SYSTEMS X-Net					X
DATA GENERAL ZODIAC Network Bus	X				
DATAPOINT ARCnet					X
DAVONG SYSTEMS MultiLink	X				
THE DESTEK GROUP DESNET					X
DEVELCON ELECTRONIC Develnet					
DIGITAL EQUIPMENT DEC Ethernet/dataway		X			
FOX RESEARCH 10-NET					X
GATEWAY COMM. G/NET					X
GENERAL TELENET ETHERCOM				X	
GOULD MODWAY	X				
HARRIS HNET Campus / Work Group	X				

TABLE A.1 CONTINUE					
COMPANY & LAN NAME	Token Passing	CSMA / CD	CSMA	CSMA / CA	Other
IDEAS IDEAS LAN		X			
INTECOM LANmark					X
INTERACTIVE SYSTEM / 3M ALAN VEDIODATA LAN /I	X	X			
INTERPHASE CORP. LCN 5180	X				
INTERSIL SYSTEMS GEnet		X			
MICOM SYSTEMS INSTANET					X
NCR CORP. Mirlan		X			
NESTAR SYSTEMS PLAN 20 / 30 / 40	X				
NETWORK SYSTEMS HYPERbus HYPERchannel		X			
NOVELL, INC. NETWARE / S					X
ORCHID TECH. PCNET		X			
PERCOM DATA CORP. Precomnet	X				
PRAGMATORNICS TIENET		X			

TABLE A.1 CONTINUE					
COMPANY & LAN NAME	Token Passing	CSMA / CD	CSMA	CSMA / CA	Other
PRIME COMPUTER RINGNET	X				
PROLINK CORP. PROloop					X
PROTEON, INC. ProNET	X				
RACAL-MILGO planet	X				
SCIENTIFIC DATA SDSNET		X			
SIECOR CORP. Fiberlan-Net 10		X			
STANDARD DATA Disc-less Network		X			
STRATUS COMPUTER StrataLink		X			
SYTEK, INC. LocalNet		X			
TECMAR, INC. ComNet		X			
UNGERMANN-BASS Net / One		X Baseband	X Broad-band		
WANG LAB. WangNet		X			
WESTERN DIGITAL NetSource / PC-LAN	X				
XEROX CORP. Ethernet		X			

TABLE A.1 CONTINUE					
COMPANY & LAN NAME	Token Passing	CSMA / CD	CSMA	CSMA / CA	Other
XYPLEX, INC. The XYPLEX System		X			
NBI, INC.		X			
MARTIN MARIETTA	X FDDI				
HONEYWELL	X FDDI				

TABLE A.2 LANs COMPARISON				
COMPANY & LAN NAME	BIT RATE	MEDIUM	CONNEC-TIONS	MAX. DISTANCE
AMECOM Government / Military UBITS (Universal Bus Information Transfer System)	160 Mbps	Twisted Pair & Optical Fiber	16000	1000 ft.
APOLLO COMPUTER DOMAIN Distributed Opreating Multi-Access Network	12 Mbps	Coaxial Cable	Several 100's	1 Km. Between Nodes
APPLE COMPUTER AppleNet	1 Mbps		128	2000 ft.
APPLITEK UniLINK	10 Mbps	Optical Fiber & Coaxial Cable	1000-4000	2.5-30 Km
CODEX 4000 Series LAN	10 Mbps	Coaxial Cable	238	500 meters
COMPLEX SYSTEM XLAN	1 Mbps	Twisted Pair		10000 ft.
COMPUTER AUTOMATION COMMERCIAL SYSTEMS DIVISION SyFAnet	3 Mbps	Coaxial Cable	64	3000 ft.
CONCORD DATA SYSTEMS Token / Net	5 Mbps	Coaxial Cable	1000	25 miles
CONTEL INFORMATION SYSTEMS ConTelNet	2 Mbps 10 Mbps	Coaxial Cable	Unlimited	5 miles
CORVUS SYSTEM Corvus Omninet	1 Mbps	Twisted Pair	64	4000 ft.

TABLE A.2 CONTINUE				
COMPANY & LAN NAME	BIT RATE	MEDIUM	CONNEC-TION	MAX. DISTANCE
CR COMPUTER SYSTEMS X-Net	14.746 Mbps	Twisted Pair	255 sites each 2032 nodes	2.5 miles
DATA GENERAL ZODIAC Network Bus	2 Mbps	Coaxial Cable	32	1 mile
DATAPOINT ARCnet	2.5 Mbps	Coaxial Cable	255	4 miles
DAVONG SYSTEMS MultiLink	2.5 Mbps	Coaxial Cable	255	20000 ft.
THE DESTEK GROUP DESNET	2 Mbps	Coaxial Cable Optical Fiber	> 350	2 Km.
DEVELCON ELECTRONIC Develnet	24 Mbps	Twisted Pair	240 lines	
DIGITAL EQUIPMENT DEC Ethernet/dataway	10 Mbps	Coaxial Cable Twisted Pair	1024	2.8 Km.
FOX RESEARCH 10-NET	1 Mbps	Twisted Pair	32	2000 ft.
GATEWAY COMM. G/NET	1.43 Mbps	Coaxial Cable	255	7000 ft.
GENERAL TELENET ETHERCOM	10 Mbps	Coaxial Cable	1000	
GOULD MODWAY	1.544 Mbps	Coaxial Cable	256	15000 ft.
HARRIS HNET Campus / Work Group	10 Mbps / 1 Mbps	Coaxial Cable	254 Per Channel (campus)	5000 ft. for work group

TABLE A.2 CONTINUE				
COMPANY & LAN NAME	BIT RATE	MEDIUM	CONNEC-TIONS	MAX. DISTANCE
IDEAS IDEAS LAN	1.544 Mbps	Coaxial Cable		Function of topology
INTECOM LANmark	10 Mbps	Twisted Pair	8192 devices	10 miles
INTERACTIVE SYSTEM / 3M ALAN VIDEODATA LAN /I	5 Mbps 2.5 Mbps	Coaxial Cable	2000 per channel	
INTERPHASE CORP. LCN 5180	2 Mbps	Twisted Pair	255	30000 ft.
INTERSIL SYSTEMS GEnet	1Mbps	Coaxial Cable	2000 per channel	
MICOM SYSTEMS INSTANET	1.544 Mbps	Twisted Pair		1 mile
NCR CORP. Mirlan	1 Mbps			
NESTAR SYSTEMS PLAN 20 / 30 / 40	2.5 Mbps	Coaxial Cable	255	4 miles
NETWORK SYSTEMS HYPERbus HYPERchannel [H.C]	10 Mbps 50 Mbps [H.C]	Coaxial Cable & Fiber Optic	Unlimited	5000 ft. 10000 ft. [H.C]
NOVELL, INC. NETWARE / S	12 Mbps	Twisted Pair & Coaxial Cable	65	3000 ft.
ORCHID TECH. PCNET	1 Mbps	Coaxial Cable	64000	7000 ft.
PERCOM DATA CORP. Precomnet	1 Mbps	Twisted Pair	254	10000 ft.
PRAGMATORNICS TIENET	1 Mbps	Coaxial Cable	200	5 miles

TABLE A.2 CONTINUE				
COMPANY & LAN NAME	BIT RATE	MEDIUM	CONNEC-TIONS	MAX. DISTANCE
PRIME COMPUTER RINGNET	10 Mbps	Twisted Pair	247	750 ft. be-tween nodes
PROLINK CORP. PROloop	10 Mbps	Coaxial Cable	62	350 meters
PROTEON, INC. ProNET	10 Mbps 80 Mbps	Twisted Pair, Coax, & O.F.	255	node to node .1-10 Km.
RACAL-MILGO planet	10 Mbps	Coaxial Cable	500	950 ft. be-tween taps
SCIENTIFIC DATA SDSNET	1 Mbps	Coaxial Cable	255	1000 meter
SIECOR CORP. Fiberlan-Net 10	10 Mbps	Optical Fiber	4000	2.5 Km.
STANDARD DATA Disc-less Network	3 Mbps	Coaxial Cable & Optical Fiber	255	75 Km.
STRATUS COMPUTER StrataLink	12.5 Mbps	Coaxial Cable	255	25 miles
SYTEK, INC. LocalNet	1.5 Mbps	Coaxial Cable	24000	50 Km.
TECMAR, INC. ComNet	10 Mbps	Coaxial Cable		
UNGERMANN-BASS Net / One	5 Mbps 10 Mbps	Coaxial Cable & Optical Fiber	36000	2800 meters
WANG LAB. WangNet	12 Mbps	Coaxial Cable	62535	4 miles
WESTERN DIGITAL NetSource / PC-LAN	1 Mbps		254	10000 ft.
XEROX CORP. Ethernet	1.5 Mbps	Coaxial Cable	1024	1.5 miles

TABLE A.2 CONTINUE				
COMPANY & LAN NAME	BIT RATE	MEDIUM	CONNEC-TIONS	MAX. DISTANCE
XYPLEX, INC. The XYPLEX System	1 Mbps	Coaxial Cable	255	6 miles
MARTIN MARIETTA	100 Mbps	FIBER OPTICS		
HONEYWELL	100 Mbps	FIBER OPTICS	500 Stations	100 Km. Ring Circumfere-nce

TABLE A.3 LANs COMPARISON		
COMPANY & LAN NAME	TOPOLOGY	GATEWAYS USED
AMECOM Government / Military UBITS (Universal Bus Information Transfer System)	BUS	OSI Level 4 gateways
APOLLO COMPUTER DOMAIN Distributed Opreating Multi-Access Network	BUS	IBM gateways
APPLE COMPUTER AppleNet	BUS	Ethernet gateways
APPLITEK UniLINK	BUS	BSC; SDLC; HDLC; X.25 gateways
CODEX 4000 Series LAN	BUS	
COMPLEX SYSTEM XLAN	BUS	
COMPUTER AUTOMATION COMMERCIAL SYSTEMS DIVISION SyFAnet	BUS	SNA; X.25 gateways
CONCORD DATA SYSTEMS Token / Net	BUS	
CONTEL INFORMATION SYSTEMS ConTelNet	BUS	X.25 gateways
CORVUS SYSTEM Corvus Omninet	BUS	SNA gateways

TABLE A.3 CONTINUE		
COMPANY & LAN	TOPOLOGY	GATEWAYS USED
CR COMPUTER SYSTEMS X-Net	BUS	BSC; HDLC; X.25; X.21; SNA / SDLC gateways
DATA GENERAL ZODIAC Network Bus	BUS	X.25 gateways
DATAPOINT ARCnet	BUS	SNA; HDLC; X.25; TLX; TWX gateways
DAVONG SYSTEMS MultiLink		
THE DESTEK GROUP DESNET	BUS	Ethernet gateways
DEVELCON ELECTRONIC Develnet	BUS	X.25; Ethernet gateways
DIGITAL EQUIPMENT DEC Ethernet/dataway		
FOX RESEARCH 10-NET	BUS	SNA; Ethernet; DECNet gateways
GATEWAY COMM. G/NET	BUS	BSC; SDLC; HDLC; SNA; Ethernet gateways
GENERAL TELENET ETHERCOM	BUS	Ethernet gateways
GOULD MODWAY	BUS	MODBUS; ADCE gateways
HARRIS HNET Campus / Work Group	BUS	SNA; 2780 / 3780 gateways

TABLE A.3 CONTINUE		
COMPANY & LAN	TOPOLOGY	GATEWAYS USED
IDEAS IDEAS LAN	BUS	BSC; SDLC; X.25 gateways
INTECOM LANmark	STAR	Ethernet; T-1; 3270 gateways
INTERACTIVE SYSTEM / 3M ALAN VIDEODATA LAN /I	BUS	TTY gateways
INTERPHASE CORP. LCN 5180	BUS; STAR	SDLC; HDLC gateways
INTERSIL SYSTEMS GEnet	BUS	DECNet gateways
MICOM SYSTEMS INSTANET	UNCONSTRAINED	X.25 gateways
NCR CORP. Mirlan	BUS	
NESTAR SYSTEMS PLAN 20 / 30 / 40	BUS	IBM; Telex server gateways
NETWORK SYSTEMS HYPERbus HYPERchannel [H.C]	BUS	Link adapter to carrier facilities; Network adapters for CPU / CPU transfer gateways
NOVELL, INC. NETWARE / S	STAR	SNA; Ethernet; Omninet gateways
ORCHID TECH. PCNET	BUS	
PERCOM DATA CORP. Precomnet		Ethernet; IBM 3270 gateways
PRAGMATORNICS TIENET	BUS	BSC; SDLC gateways

TABLE A.3 CONTINUE		
COMPANY & LAN	TOPOLOGY	GATEWAYS USED
PRIME COMPUTER RINGNET	RING	Access to most standard protocols ; X.25 gateways
PROLINK CORP. PROloop	RING	BSC; SDLC gateways
PROTEON, INC.	RING	HDLC; X.25; IBM; TCP/IP; DECNet gateways
RACAL-MILGO planet	RING	
SCIENTIFIC DATA SDSNET	BRANCHING NON ROOTED TREE	
SIECOR CORP. Fiberlan-Net 10	BUS; STAR	Ethernet gateways
STANDARD DATA Disc-less Network	BUS	X.25; Ethernet gateways
STRATUS COMPUTER StrataLink	RING	HDLC; SDLC; BSC gateways
SYTEK, INC. LocalNet	BUS	X.25; BSC; Ethernet gateways
TECMAR, INC. ComNet	BUS	
UNGERMANN-BASS Net / One	BUS	Ethernet; V.35 gateways
WANG LAB. WangNet	BUS	Wang Data Switch; Remote microwave; Satellite gateways
WESTERN DIGITAL NetSource / PC-LAN	RING	
XEROX CORP. Ethernet	BUS	

TABLE A.3 CONTINUE		
COMPANY & LAN	TOPOLOGY	GATEWAYS USED
XYPLEX, INC.	BUS	
NBI, INC.	BUS	
MARTIN MARIETTA	RING	
HONEYWELL	RING	

A.2 Access Time

Access time is the time required for a node in a local area network to gain control of the transmission medium so that the node may complete transmission of its packet. It is possible, in a ring topology implementing a token passing protocol, to set the maximum bound on the access time. On the other hand setting the maximum bound on the access time for a contention protocol is not possible because any packet may suffer a collision.

A.3 Network Length Constraints

The physical separation between two nodes connected via a link in a given local area network has upper and lower bounds. Such bounds are functions of the characteristics of the transmission medium being used, the power level in the transmitted signal, the receiver dynamic characteristics, and coupling losses.

A.3.1 Timing due to Propagation Delay

In both coaxial and twisted pair medium the propagation delay is a function of the medium's propagation constant. The propagation constant is a complex function and it is composed of two parts, a real part called the attenuation constant and an imaginary part called the phase constant. The attenuation constant contributes to the manor in which the transmission medium affects the magnitude of a signal propagating through it, and the phase constant contributes to the manner in which the transmission medium affects the phase of the signal propagating through it.

In general, propagation through an optical fiber medium is due to different modes, and every mode has its own propagation constant. The propagation constant has different magnitudes for different modes, hence different propagation delays. This phenomena in optical fiber results in the receiver in a system with many modes (multimode optical fiber) receiving many different replicas of the transmitted signal spread over a time period corresponding to the fastest and the slowest of the group velocities of the various modes hence another form of dispersion.

The followings are some realistic values for propagation delay in commercially available mediums:

1- The propagation delay for a 500 meter long coaxial cable is 1.66x10 E-6 second [FRANTA]

2- The propagation delay for a ring of 1000 meter in circumference circle of optical fiber is 5085 nanosecond. [JOHN87]

The propagation delay values mentioned above suggests that propagation delay is a function of distance. These values imply that the distance between any two consecutive nodes in the network system should be kept as short as possible. The separation between two consecutive nodes in a local area network usually has minimum bound, the value of such bound is different for different local area networks. For example, it is recommended in an Ethernet system the separation between two consecutive nodes must be more than or equal to 2.5 meter. This is to avoid having standing wave phenomena in the system.

A.3.2 Signal Strength Limitations

As it stated in section A.3.1 the real part of the propagation constant is the attenuation constant. For a given transmission medium the attenuation constant defines the allowable transmitted frequencies through the transmission medium. Signals within the allowable frequencies propagate through the transmission medium and experiences amplitude and phase distortions. In addition to attenuation, dispersion also needs to be considered when designing a local area network using optical fiber medium.

The power that the transmitting circuitry can provide is another parameter that should be considered when accounting for signal strength in the process of designing a local area network.

A.3.3 Dynamic Range

In a local area network, the maximum distance between a sending node and a designated node is a function of the amount of power the sending node can output. For example, if the designated node bandwidth is such that α dBm < RSP< β dBm, where (RSP) is the power recognized by the designated node, then the distance between the two nodes should be such that the power received at the designated node is between α and β. In a ProNET 10 system, for example, the amount of power that a sending node can provide is adjustable, so it is possible to have a different length constraints between nodes. On the other hand in a ProNET 80 system the amount of output power from a sending node is fixed, therefore the distance between nodes is fixed by the dynamic range of the receivers.

A.4. Fault Tolerance and Reliability Features

A.4.1 Netware

In most applications software is needed to interface a node to a given LAN. In general, LAN operating system software (in some cases hardware also) uses a peer-to-peer approach to workstations and file service. Any node, in these LANs, can be used simultaneously as a workstation and a shared network resource. With some LANs, all workstations run the same operating system software regardless of whether their resources are shared with the network or not. In other cases, workstations that will also offer resources to the network require additional operating system components.

A.4.1 Hardware

Different LANs use different techniques for improving reliability. The following are examples from commercially available systems.

A local area network using the FDDI standard for a 100 Mbps token ring with a fiber optic transmission medium uses the following to improve reliability:[COMM86]1- Node bypass switch: This feature is used to solve the problem of a known broken node or a powered down node.

2- Counter rotating ring connections: The counter rotating ring is required for all nodes directly attached in the ring. The second ring is used as a standby ring or for concurrent transmission. If a link fails in the active ring then the backup ring is used for transmissions. If a node fails then the two rings are folded into one ring, which is approximately twice as long, maintaining full connectivity.

3- Concentrators: The concentrators may be used to attach nodes to the ring. Each node has a direct link to the concentrator. This allows any combination of nodes to be switched out of the ring while retaining full connectivity for the remaining nodes.

A local area network using the HYPERchannel system, on the other hand, may have four trunks dedicated for one connection between two nodes. This can be used for improving the network reliability in that if onc trunk fails the other three trunks can still be used for continuing the communication between the two nodes. The fault in the trunk can be detected in two ways: [HYPERchannel literature]

1- Each of the connected nodes to the four trunks may have a self testing circuitry which tests the trunks at random for faults. This will require space and time.

2- Software diagnostics and management may be used by the nodes connected to the four trunks. This software will require memory allocation and part of the system bandwidth.

A.5 Guarantee of Data Delivery and Data Latency

Local area networks using token ring protocol use a "readbit" method through which a receiving node can acknowledge the of receiving a message. The sending node recognizes this bit when it strips the frame from the ring. A designated node in a local area network using HYPERchannel system will use an acknowledgement of message delivery packet to inform the sending node that its message has been received by the designated node. In a local area network utilizing Ethernet the sending node assumes that its message is received the designated node. No guarantee of data latency is available for a network using Ethernet. If a collision is detected a sending node will try to send its message to the designated node 16 times and after that the sending host must reinitialize transmission.

The latency per physical connection in a network using FDDI system is 600 nanoseconds, the total latency is obtained through multiplying the total number of nodes in the network by 600. The token transmission time is 0.00088 milliseconds, to obtain maximum overhead due to token transmission time the total number of nodes is multiplied by 0.00088. [JOHN87]

A.6 Ease of Expansion

An Ethernet local area network is easy to expand. Adding a node to a system already in use is done through the use of a vampire tap. During the expansion the network retains its full connectivity. On the other hand in a token ring local area network adding a node may require a break in the ring, this is true if the local area network was not an FDDI system. Removing a node from a non FDDI token ring local area network requires the addition of a connector which results in an extra loss in the system.

A.7 Special Features

Table (A.4) includes LAN-to-LAN bridges that can be purchased separately for any particular local area network interface boards or system.[Connectivity, June 28,1988].

Table (A.5) includes telephone number of some LAN product companies.

Table (A.6) shows different companies or organizations and their corresponding software at every ISO protocol layer.

TABLE A.4 BRIDGES & GATEWAYS					
Product	Company	LAN supported	Bridges to what ?	Speed	Price $
ACS 4030	Advanced computer comm.	Ethernet	Ethernet	128 Kbps	4,975
N110 / E	Applitek	Ethernet	UnitLAN	10 Mbps	8,000 - 14,000
N110 / TI	Applitek	UnitLAN	T-I, RS-449	56 Kbps to 2 Mbps	
IB / 1	Bridge Comm.	Ethernet	Broadband	5 Mbps	6,000 - 11,000
IB / 2			Ethernet		
IB / 3			T-I;V.35 RS-422 RS-232	19.2 Kbps to 2.048 Mbps	
Gator Box	Cayman sys.	Ethernet LocalTalk	Ethernet LocalTalk	10 Mbps	3,495
Ethermodem III Bridge	Chipcom	Ethernet	Ethernet Broadband	10 Mbps	9,950
Marathon Bridge		Ethernet IEEE 802.4	Broadband Token bus		
HyBridge	Cisco Sys.	Ethernet	Ethernet		6,200
Series 4100	Concord Co.	IEEE 802.4 (MAP)	IEEE 802.4 (MAP)		9,950 - 11,900
Series 4200		Ethernet	IEEE 802.4 Broadband Token bus	10 Mbps	

TABLE A.4 BRIDGES & GATEWAYS

Product	Company	LAN supported	Bridges to what ?	Speed	Price $
ILAN-1	CrossComm	Ethernet or Token ring or StarLan	Fiber backbone	10 Mbps	4,900 - 15,000
ConnectLAN	Hally Sys.	Ethernet	Ethernet Broadband T-I, DDS Fiber optic	2.048 Mbps	7,300 - 10,500
InterBridge	Hays Micro-computer Products	AppleTalk	AppleTalk	19.2 Kbps	799
28647A	Hewlett Packard	StarLAN	Ethernet		4,475 - 6,975
28648A		Ethernet			
LAN Span	Infotron Sys.	Ethernet	T-I, V.35, RS-449	56 Kbps to 2.048 Mbps	11,495
8023	LANEX	Ethernet	Ethernet StarLAN Broadband Fiber optic		3,995
IB 3000 IB 30 IB 10	Micom Interlan	Ethernet, Thin-wire Ethernet	Ethernet, Broadband, Thin-wire Ethernet, Starlan (IB 10)	10 Mbps	2,295 - 4,495 (1,000 for network manage-ment option)

TABLE A.4 BRIDGES & GATEWAYS					
Product	Company	LAN supported	Bridges to what ?	Speed	Price $
MLB / 1000	Microcom	Token ring or Ethernet	2 wire modem	19.2 Kbps	5,499 - 12,499
MLB / 1500			S-Interface	64 Kbps	
MLB / 2000			4 wire leased line, V.35, RS-449 / 422	56 Kbps	
MLB / 2500				112 Kbps	
Bridge Plus	Netways	Ethernet, StarLAN, IEEE 802.3	Ethernet, StarLAN, T-I, Fiber optics & Remote links	10 Mbps (Ethernet & StarLAN), 1.544 Mbps (others)	5,695 - 8,000
Remote Ethernet Bridge	RAD Network Devices	Ethernet	T-I, V.35, RS-422 / 232	9.6 Kbps to 2.048 Mbps	6,950 - 7,950
RetixGate 2244 & 2255 Series	Retix	Ethernet, StarLAN	Ethernet, StarLAN		1,950 - 2,850
NetBridge	Shiva	AppleTalk	AppleTalk	230 Kbps	399
8050	Sytek	Ethernet	LocalNet 6000 broadband	2 Mbps	7,000 - 9,500
8080			Ethernet		
8200			IEEE 802.4 Broadband Token Bus	10 Mbps	

TABLE A.4 BRIDGES & GATEWAYS					
Product	Company	LAN supported	Bridges to what ?	Speed	Price $
TransLAN 350	Vitalink Comm.	Ethernet	T-I, V.35, RS-449, RS-232	9.6 Kbps to 2.048 Mbps	12,000 - 18,500
TransLAN III					
TransRING 550		Token Ring			
LN, CN	Wellfleet Comm.	Ethernet, IEEE 802.3	Ethernet, T-I, Broadband, Leasd line	10 Mbps (Ethernet), 1.544 Mbps (T-I)	11,800

TABLE A.4 BRIDGES & GATEWAYS		
Product	**Company**	**Features**
ACS 4030	Advanced computer comm.	Packet filtering, X.25 support
N110 / E	Applitek	Packet filtering
N110 / TI	Applitek	Packet filtering
IB / 1	Bridge Comm.	Packet filtering
IB / 2		
IB / 3		Packet filtering, supports up to 8 synchronous lines
Gator Box	Cayman sys.	Packet filtering, transparent protocol translation, Kinetics FastPath emulation. Network management software
Ethermodem III Bridge	Chipcom	Packet filtering, spanning tree-loop detection, 24,200 packets per second filter rate, remote management, programmable filtering, LAN monitoring.
Marathon Bridge		Packet filtering, 15,000 packets per second rate
HyBridge	Cisco Sys.	brouter capabilities
Series 4100	Concord Co.	Packet filtering
Series 4200		

TABLE A.4 BRIDGES & GATEWAYS		
Product	Company	Features
ILAN-1	CrossComm	
ConnectLAN	Hally Sys.	Packet filtering, up to 4 synchronous lines, brouter capabilities, alternate routing, distributed load sharing, security access and control
InterBridge	Hays Micro-computer Products	2 synchronous lines, data compression
28647A 28648A	Hewlett Packard	
LAN Span	Infotron Sys.	
8023	LANEX	
IB 3000 IB 30 IB 10	Micom Interlan	Packet filtering, brouter capabilities, IEEE 802.1 network management and spanning tree, remote bridge management

TABLE A.4 BRIDGES & GATEWAYS		
Product	Company	Features
MLB / 1000	Microcom	Data compression
MLB / 1500		
MLB / 2000		
MLB / 2500		Data compression
Bridge Plus	Netways	Packet filtering, brouter capabilities, security filtering, link monitoring statistics, extensive diagnostics, automatic address learning and purging, non-volatile memory
Remote Ethernet Bridge	RAD Network Devices	Supports up to 4 synchronous lines
RetixGate 2244 & 2255 Series	Retix	Packet filtering, automatic configuration, multimedia access, network management option
NetBridge	Shiva	Packet filtering, brouter capabilities, network manager software for creating and managing zone access privileges
8050	Sytek	Packet filtering
8080		
8200		

TABLE A.4 BRIDGES & GATEWAYS		
Product	**Company**	**Features**
TransLAN 350	Vitalink Comm.	Supports up to 8 synchronous lines, brouter capabilities
TransLAN III		
TransRING 550		
LN, CN	Wellfleet Comm.	Packet filtering, 16 synchronous lines (LN), 52 synchronous lines (CN), brouter capabilities, concurrent routing option, D4 / GSF compatibly, integrated T-1, integrated CSU, voice support
NETWORK SYSTEM	EN601	Filters Ethernet messages by source and destination address using a listen and learn procedure.

TABLE A.5 LAN PRODUCT COMPANIES TELEPHONE NUMBER	
COMPANY NAME	TELEPHONE NUMBER
ADVANCED COMPUTER COMMUNICATIONS	805-963-9431
AMP INCORPORATED	717-564-0100
APPLITEK	301-330-8700
AT&T	1-800-37202447
BEST	608-565-7200
CAYMAN	617-494-1999
CHIPCOM	617-890-6844
CISCO SYSTEMS, Inc.	415-326-1941
CONDENOLL	914-965-6300
CODEX	617-364-2000
COMPUTROL	203-544-9371
CONCORD	617-460-4646
CORVUS	408-281-4100
CROSS COMM CORP.	617-481-4060
HAYES	404-449-8791

TABLE A.5 LAN PRODUCT COMPANIES TELEPHONE NUMBER	
COMPANY NAME	TELEPHONE NUMBER
HEWLETT PACKARD	301-258-2000
HONEYWELL	612-541-6500
LANEX	301-595-4700
MARTIN MARIETTA	301-682-0900
MICOM-Interlan	617-263-9929
MICROWAVE FILTER COMPANY; Inc.	1-800-448-1666
NETWORK SYSTEMS	404-255-6790
RAD data communication	201-587-8822
SHIVA	617-864-8500
SIGNETICS	408-991-2000
SYTEK	415-966-7300
VERSITRON	301-497-8600
WELLFLEET	617-275-2400

TABLE A.6 LAN COMMUNICATION PROTOCOLS	
COMPANY OR ORGANIZATION	THE APPLICATION LAYER
3Com	3+ MS-Net
DEC / Net	Network Management Local Area Terminal Protocol
NOVELL	Netware
HP	HP Network Services Office Share MS-Net
SUN	Network File System
UC Berkeley	Berkeley Services
DARPA	ARPA Services
Xerox / XNS	XNS Application Services
IBM / SNA	PC Network MS-Net IBM SNA Transaction Services
ISO / CCITT	Virtual Terminal Service Job Transfer & Manipulation Authorization Service Office Document Architecture. Directory File Transfer Access Management Common Mgt. Information Protocol Remote Operation Service Commitment Concurrency Recovery Message Handling Service X.400 Manufacturing Messaging Service (RS-511) Association Control Service Element (ACSE)

TABLE A.6 LAN COMMUNICATION PROTOCOLS	
COMPANY OR ORGANIZATION	THE PRESENTATION LAYER
3Com	Server message Block Protocol (SMB)
NOVELL	Netware File Service Protocol (NFSP)
HP	Server message Block Protocol (SMB)
SUN	Exchange Data Representative Protocol (XDR)
Xerox / XNS	XNS Courier
IBM / SNA	Server message Block Protocol (SMB) Presentation Services
ISO / CCITT	Connection Oriented Presentation Protocol

TABLE A.6 LAN COMMUNICATION PROTOCOLS	
COMPANY OR ORGANIZATION	THE SESSION LAYER
3Com	NETBIOS
DEC / Net	Session Control Protocol
NOVELL	NETBIOS Emulator
HP	Interprocess Communication NETBIOS
SUN	Remote Procedure Call (RPC)
UC Berkeley	BSD Socket
Xerox / XNS	XNS Courier
IBM / SNA	Data Flow Control NETBIOS
ISO / CCITT	Connection Oriented Presentation Protocol

TABLE A.6 LAN COMMUNICATION PROTOCOLS

COMPANY OR ORGANIZATION	THE TRANSPORT LAYER
3Com	MS-DOS Interval Network Driver Protocol (MINDP)
DEC / Net	Network Services Protocol (NSP)
NOVELL	Netware Core Protocol (NCP)
DARPA	Internal Name Server Protocol (INSP) User Datagram Protocol (UDP) Packet Exchange Protocol (PXP) Transmission Control Protocol (TCP)
Xerox / XNS	XNS Tansport
IBM / SNA	Transmission Control
ISO / CCITT	Transport Protocol

TABLE A.6 LAN COMMUNICATION PROTOCOLS

COMPANY OR ORGANIZATION	THE NETWORK LAYER
3Com	MS-DOS Interval Network Driver Protocol (MINDP)
DEC / Net	Routing Protocol Maintenance Operation Protocol (MOP)
DARPA	Internal Control Message Protocol (ICMP) Internal Protocol (IP) Address Resolution Protocol (ARP)
Xerox / XNS	Network Internetwork Datagram Protocol (IDP)
IBM / SNA	Path Control
ISO / CCITT	X.25 Packet Level Protocol Internetwork Protocol End System Intermediate System ES-IS

TABLE A.6 LAN COMMUNICATION PROTOCOLS	
COMPANY OR ORGANIZATION	THE DATA LINK LAYER
DEC / Net	Ethernet Data Link Control
IEEE	802.2 Logical Link Control 802.3 CSMA/CD Media Access Control 802.4 Token Passing Bus Media Access Control 802.5 Token Passing Ring Media Access Control
ANSI	FDDI Token Ring Media Access Control

TABLE A.6 LAN COMMUNICATION PROTOCOLS	
COMPANY OR ORGANIZATION	THE PHYSICAL LAYER
DEC / Net	Ethernet 10 Mbps 50 ohm coax Thin LAN 10 Mbps 50 ohm coax Broadband 10 Mbps 75 ohm coax
IEEE	10 BASE5 10 Mbps 50 ohm coax 10 BASE2 10 Mbps 50 ohm coax 10 BROAD 36 10 Mbps 75 ohm coax 1 BASE5 1 Mbps twisted pair 10 BASET 10 Mbps twisted pair Carrierband 1 Mbps Phase Continuous FSK 75 ohm coax Carrierband 5,10 Mbps Phase Coherent FSK 75 ohm coax Broadband 1, 5, 10 Mbps Multilevel Duobinary AM/PSK 75 ohm coax 1, 4 Mbps shielded twisted pair
IBM	16 Mbps shielded twisted pair 16 Mbps fiber optics
ANSI	FDDI Physical Layer Protocol 100 Mbps fiber optic FDDI Physical Media Dependent Interface

Bibliography

[AKER87] J.D. Akers, "Lecture Notes", Mississippi State University, 1987.

[ALBE86] Harold J. Alber and Wayne A. Thomas, "A Dual Channel High Speed Fiber Optics Multiplex Data Bus System", (white paper), Systems Engineering Avionics Facility, Wright-Patterson AFB, 1986.

[AMP82] "Designer's Guide to Fiber Optics", AMP Incorporated, 1982.

[AWST88] (No author listed), "Honeywell Design for Space Station Systems Network Passes Critical Review", Aviation Week and Space Technology, May 30, 1988.

[BOEI88] Timothy Mosher, Glenn McNeil, and Michael Nehl, "DISTRIBUTED DATA BUS STUDY FOR SPACE TRANSPORTATION", Procurment Instrument Identification Number F33601-85-D0070, Report number AFWAL-TR-87-3111.

[BORG85] Flaminio Borgonovo, Luigi Fratta, Fabio Tarini, and Paolo Zini, "L-Express-net: The Communication Subnetwork for the C-NET Project", IEEE Transactions on Communications, v. COM-33, July 1985.

[COHN88] Marc Cohn, "U.S Navy Safenet Lightweight Protocol Selection", Transfer, v.1 #2, March/April 1988.

[DALY84] James C. Daly, Fiber Optics, CRC Press, 1984.

[FDDI88] "FDDI Physical Layer Medium Dependent (PMD), Draft Proposed American National Standard," American National Standards Institute Committee X3T9.5, Gene Milligan, Chairman, Floyd Ross, V. Chairman, 1988.

[FRAN84] William R. Franta and John R. Heath, "Measurement and Analysis of HYPERchannel Networks", IEEE Transactions on Communication, v. C-33, March 1984.

[GEOR82] Chris J. Georgopoulos, Fiber Optics and Optical Isolators, Don White Consultants Inc., 1982.

[GOPA84] P.M. Gopal and J.W. Wong, "Analysis of a Hybrid Token-CSMA/CD protocol for Bus Networks", Performance of Computer Communications Systems, Elsevier Science Publishers B.V. (North-Holland), 1984.

[HEAT86] Sharon K. Heatleya, "A Simplified Discrete Event Simulation Model for an IEEE 802.3 Local Area Network", GLOBECOM '86, IEEE Global Telecommunications Conferenece: Communications Broadening Technology Horizons Conference Record, IEEE, 1986.

[JOHN 85] Marjory Johnson and Larry Webster, "Local Area Network Extensible Simulator (LANES)", NASA Ames Research Center, 1985.

[JOHN87] Marjory Johnson, "Proof that Timing Requirements of FDDI Token Ring Protocol are Satisfied", IEEE Transactions on Communications, v. COM-35, June 1987.

[JOHN88] Marjory Johnson, "Performance Analysis of FDDI", Research Institute for Advanced Computer Science, NASA Ames Research Center, April 1988.

[JONE88] William B. Jones Jr., Introduction to Optical Fiber Communications Systems, Holt, Rinehart and Winston Inc., 1988.

[KEIS83] Gerd Keiser, Optical Fiber Communications, McGraw-Hill Book Co., 1983.

[KRAU84] Johon D. Kraus, Electromagnetics, McGraw-Hill Book Co., 1984.

[LIAO85] Samuel Y. Liao, Microwave Devices and Circuits, Second Edition, Prentice-Hall Inc., 1985.

[MARA80] Madhav Marathe, "Design Analysis of a Local Area Network", Proceedings of the 1980 Computer Networking Symposium, IEEE, 1980.

[MEYE86] John W. Meyer, "SAE AE-9B Draft Standard High Speed Token Passing Data Bus for Avionics Applications", Proceedings of the 7th AIAA/IEEE Digital Avionics Systems Conference, IEEE, 1986.

[PROT84] "Operations and Maintenance Manual for the ProNet Model p1200 Multibus Local Area Network System," Proteon Inc., 1984.

[PROT86] "ProNet-10 Series p320x Counter-Rotating Ring Operation and Maintenance Manual," Proteon Inc., 1986.

[RELI83] M.J. Relis, "New MIL Standard Opens Way for Optical Links in Aircraft", Electronic Design, August 4, 1983.

[ROSS86] Floyd E. Ross, "FDDI: a Tutorial", IEEE Communications Magazine, May 1986.

[SEVI87] Marjory Johnson and K. C. Sevick, "Cycle Time Properties of the FDDI Token Ring Protocol", IEEE Transactions on Software Engineering, v. SE-13 #3, 1987.

[SHER85] Ken Sherman, Data Communications, Second Edition, Reston Publishers, 1984.

[SPIE86] James E. Spieth and Walter D. Seward, "Simulation Model of a High-Speed Token-Passing Bus for Avionics Applications", Proceedings of the 7th AIAA/IEEE Digital Avionics Systems Conference, IEEE, 1986.

[SPIE88] IEEE Spectrum Magazine, June 1988.

[STALL84] William Stallings, Local Networks: an Introduction, MacMillan Publishing Co., 1984.

[STALL85] William Stallings, Data and Computer Communications, MacMillan Publishing Co., 1985.

[YARI85] Amnon Yari, Optical Electronics, Third Edition, Holt, Rinehart and Winston Inc., 1985.

Report Documentation Page

NASA — National Aeronautics and Space Administration

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.

4. Title and Subtitle

SPACELAB SYSTEM ANALYSIS-
A STUDY OF COMMUNICATIONS SYSTEMS FOR ADVANCED
LAUNCH SYSTEMS

5. Report Date

October 1988

6. Performing Organization Code

7. Author(s)

F. M. Ingels
J. K. Owens
S. P. Daniel
F. Ahmad
V. Couvillion

8. Performing Organization Report No.

MSU-EE-FIN-10B-88

10. Work Unit No.

9. Performing Organization Name and Address

Department of Electrical Engineering
Mississippi State University
Mississippi State, MS 39762

11. Contract or Grant No.

NAS8-36717

13. Type of Report and Period Covered

INTERIM FINAL REPORT
June 87 to Oct 88

12. Sponsoring Agency Name and Address

National Aeronautics and Space Administration
Washington, D.C. 20546-0001

14. Sponsoring Agency Code

15. Supplementary Notes

Prepared by Mississippi State University for Sherman Jobe

16. Abstract

An analysis of the required performance of internal avionics data buses for future launch vehicles is presented. Suitable local area networks that can service these requirements are determined.

17. Key Words (Suggested by Author(s))

LAN-Local Area Network
FDDI-Fiber Distributed Data Interface

18. Distribution Statement

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of pages	22. Price
Unclassified	Unclassified	209	