1988

LAN performance gains achieved through the use of an enhanced performance architecture /

Kenneth James Wagner
Lehigh University

LAN PERFORMANCE GAINS ACHIEVED THROUGH THE USE OF AN ENHANCED PERFORMANCE ARCHITECTURE

by

Kenneth James Wagner

A Thesis

Presented to the Graduate Committee

of Lehigh University

in Candidacy for the Degree of

Master of Science

in

Industrial Engineering

Lehigh University

1987

This thesis is accepted and approved in partial fulfillment of the requirements for the degree of Master of Science.

12/14/87
(date)

Gregory L. Timbie
Professor in Charge

L. E. Kline
Chairman of Department

ACKNOWLEDGEMENTS

Several people deserve recognition and thanks for their help during the writing of this thesis. The first of these is Mark Stickler who helped in the formulation of the topic, the proofreading of later drafts of the paper, and by being a reliable source of information. Dr. Richard Denton also provided a wealth of information which was required to write the thesis in addition to giving me direction in my approach to the topic. Finally, I would like to thank my advisor Dr. Gregory Tonkay for his help with the logistics of writing the thesis as well as the timely proofreading of various drafts.

I would also like to take this opportunity to thank my parents for their financial support of my education.

TABLE OF CONTENTS

ABSTRACT 1

1 Introduction 3
1.1 Background 3
1.2 Objective of Thesis 5
1.3 Topics Covered 5

2 Local Area Networks 8
2.1 Introduction 8
2.2 LANs in the Factory 10
2.3 Local Area Network Fundamentals 14
2.3.1 Topology 15
2.3.1.1 Dedicated 15
2.3.1.2 Star 17
2.3.1.3 Ring 19
2.3.1.4 Bus/Tree 21
2.3.2 Transmission Media 23
2.3.2.1 Twisted Pair 23
2.3.2.2 Coaxial Cable 24
2.3.2.3 Fiber Optics 25
2.3.3 Signaling Techniques 26
2.3.4 Media Access Control (MAC) 30
2.3.4.1 Contention 30
2.3.4.2 Token Passing 33
2.3.4.3 Contention vs. Token Passing . . . 35
2.4 The Open Systems Interconnect Model 36
2.4.1 Physical Layer 37
2.4.2 Data-Link Layer 39
2.4.3 Network Layer 39
2.4.4 Transport Layer 40
2.4.5 Session Layer 40
2.4.6 Presentation Layer 41
2.4.7 Application Layer 41
2.5 Subnetworks and Internetworking 42
2.6 Chapter Summary 45

3 Evaluation of Local Area Networks 46
3.1 Introduction 46
3.2 Network Performance Measures 47
3.3 Network Characteristics and Variables 53
3.4 Network Simulation 57
3.4.1 Simulation Modeling 58
3.4.2 Reasons for Network Simulation 59
3.4.3 A Network Simulation Methodology 61
3.5 Chapter Summary 63

4 Network Architectures 65
4.1 Introduction 65
4.2 Manufacturing Automation Protocol 66
4.2.1 The Physical Layer 71
4.2.2 The Data Link Layer 72
4.2.3 The Network Layer 74
4.2.4 The Transport Layer 78
4.2.5 The Session Layer 79
4.2.6 The Presentation Layer 80
4.2.7 The Application Layer 80
4.3 The Need for Enhanced Performance 81
4.4 PROWAY-LAN 84
4.4.1 The Physical Sub-layer 86
4.4.2 The Medium Access Control Sub-layer . . . 88
4.4.3 The PROWAY Link Control Sub-layer 92
4.5 Chapter Summary 93

5 Simulation Models 95
5.1 Introduction 95
5.2 Simulation Language Used 96
5.3 Model Development 99
5.3.1 Assumptions 99
5.3.2 Logic 102
5.3.2.1 MAP Model 112
5.3.2.2 EPA Model 116
5.3.3 Inputs and Outputs 118
5.4 Verification and Validation 121
5.5 Experimental Design 123
5.6 Results 127
5.7 Conclusions 132
5.8 Chapter Summary 136

6 Summary and Conclusions 138

LIST OF REFERENCES 142

VITA 144

LIST OF FIGURES

Figure 2-1. A four-tier model for factory communications. 11
Figure 2-2. Dedicated topology. 16
Figure 2-3. Star topology. 18
Figure 2-4. Ring topology. 20
Figure 2-5. Bus topology. 22
Figure 2-6. Nodes contending for network media. 32
Figure 2-7. ISO Open System Interconnect model. 38
Figure 3-1. Ideal throughput vs. offered load. 50
Figure 4-1. MAP seven layer model. 70
Figure 4-2. PROWAY-LAN three layer model. 87
Figure 4-3. Media Access Control machines. 90
Figure 5-1. Message arrival subroutine. 104
Figure 5-2. Network control subroutine. 105
Figure 5-3. Token passing subroutine. 106
Figure 5-4. Frame sending subroutine. 107
Figure 5-5. Message removal subroutine. 108
Figure 5-6. Time spent in system by the message vs. the time between message arrivals. 129
Figure 5-7. Average number of messages in queue vs. load. 133
Figure 5-8. Token holding time vs. load. 134

ABSTRACT

This paper uses two simulation models to demonstrate the 550% gain in response time achieved over MAP version 2.1 by the PROWAY-LAN which is used to represent an Enhanced Performance Architecture. This result suggests that an Enhanced Performance Architecture is more capable of providing for situations which require real-time network communications such as cell and process control. It also suggests that substantial performance gains can be achieved within MAP by shifting some of the processing responsibilities from the mid-layers of the architecture to the application layer thereby allowing for the elimination of some of the mid-layers and the delay incurred by passing data through them. The results obtained from experimentation with the models also demonstrated that the performance of the network architectures degrades rapidly as the capacity of the network is approached and surpassed.

Before the development and results of the simulation models are presented, the reader is provided with some preliminary information to increase his understanding of Local Area Network technology and performance evaluation. Accordingly, the paper begins by introducing the reader to the fundamentals of Local Area Networks and the International Standard Organization seven layer Open Systems

Interconnect model on which the MAP and PROWAY architectures are based. The next chapter covers the evaluation of Local Area Networks through discussion of performance measures, network characteristics, and the simulation of networks. A brief description of each of the MAP and PROWAY architectures is then given along with a discussion of the needs for an Enhanced Performance Architecture such as the PROWAY-LAN. The final chapter then covers the logic of the simulations models and the results obtained from the experiments run using those models.

1 Introduction

1.1 Background

Computer networking has become one of the most important subjects facing manufacturers today. In recent history, the declining cost of computer based systems and the push for manufacturing automation have caused many organizations to hastily install computer controlled equipment in their factories. This has left these firms with what has become commonly known as "islands of automation." Many companies are now beginning to realize the potential power of the information which is presently stranded on these islands and therefore the importance of integrating this equipment. The problem which these organizations now face is getting the equipment, which has been obtained from a variety of vendors, to communicate. This problem is caused by the fact that many of the equipment vendors have their own proprietary methods of integrating equipment, making communications between devices provided by different vendors difficult if not impossible. The solution to this problem lies in computer networking standards. There are many organizations working on such standards today. The group of standards which are presently gaining the widest acceptance are the ones selected by General Motors for the Manufacturing Automation Protocol (MAP).

The need for a non-proprietary communications protocol which would support the integration of multi-vendor automation systems was the impetus behind the formation of the MAP Task Force at General Motors. Formed in 1980, its purpose was to prepare the specification of a standard which would allow diverse intelligent devices to exchange information in a cost-effective manner. The intent was not to develop a new set of protocols, but to choose from existing procedures which have already been implemented and well documented. GM hoped to gain a level of support for the MAP specification that would be sufficient to motivate vendor companies to produce products that adhere to it. MAP did indeed gain wide acceptance in the manufacturing community. In fact, control of the MAP specification has recently been turned over to the MAP/TOP (Technical Office Protocol) Users Group which is comprised of representatives from a large number of major manufacturing companies. In addition, many vendors are currently offering equipment which is fully compatible with the latest MAP specification.

Unfortunately, the MAP specification has been unable to suit the needs of all types of manufacturers. Many industries, particularly the process industry, require faster response times for their communications than are offered by the MAP specification. In an effort to offer the response these real-time control systems need, an Enhanced Performance Architecture (EPA) is being developed as part of

the MAP architecture. The EPA is basically a collapsed form of the MAP architecture which operates on subnetworks of the MAP backbone. The MAP committee has reviewed several architectures for possible inclusion in the MAP specification as the EPA architecture. At this point, it seems as though the PROWAY-LAN specification developed by the Instrument Society of America (ISA) will be used.

1.2 Objective of Thesis

The primary objective of this thesis is to simulate and compare the performance of the MAP and PROWAY-LAN architectures. The comparison will attempt to determine the performance gains, if any, obtained by using the PROWAY-LAN architecture. This will be done by comparing the relative amounts of delay that are incurred in passing through the layers of each of the specifications. It is hoped that the final result of the study will be in the form of a comparison between the response times achieved using each of the architectures.

1.3 Topics Covered

In order to better present the main topic of the thesis, the reader must first be given a basic understanding of the subject matter. The next three chapters of the thesis are, therefore, dedicated to presenting background information which will be important to the readers

understanding of the final chapter. The information is presented at an introductory level. It is intended that a reader with no prior knowledge of the subject matter be able to grasp the topics presented and to use the knowledge gained as a basis for understanding the material presented in later chapters. It is hoped that references to outside materials will not be required, however, such references are provided in case this does not hold true.

The second chapter will begin by covering the fundamentals of local area networks. Coverage here will include the basics topics of media, topology, and media access as well as the slightly more advanced topics of subnetworks and internetworking. This chapter will also introduce the seven layer Open System Interconnect (OSI) model developed by the International Standard Organization (ISO). This model was used as a basis for both the MAP and PROWAY specifications. In addition, the basic reasons for and applications of networks in the factory will be covered.

The third chapter will cover the evaluation of LANs. This chapter will include a discussion of the characteristics and variables of LANs which cause them to perform at different levels and how these levels of performance are measured and compared. The simulation of LANs will also be covered in this chapter to give the reader an understanding of how and why it is used.

Chapter 4 will present a brief summary of the MAP and PROWAY-LAN architecture specifications. This is provided to give the reader a good understanding of the composition of the specifications. It should also show the reader how the architectures differ from each other so that he will have a better understanding as to what causes them to perform differently.

In Chapter 5 the development and resulting data of two simulation models will be explained. One model will be used to simulate the performance of a network using the MAP architecture and the second will simulate the performance of a network using the PROWAY-LAN architecture. The chapter will include a discussion of the development of the models and the verification and validation of the models. The data resulting from the simulation runs will then be presented and compared. In addition, there will be a brief description of the language used to create the simulation models.

2 Local Area Networks

2.1 Introduction

Although Local Area Networks have been around for more then ten years, it is only recently that they are beginning to gain much recognition. The interest in this technology is spurred by the many benefits which it offers. Some of the advantages which networking provides include the sharing of information, the sharing of resources, the backup of vital systems, multi-vendor support, and the increased accessibility of equipment by users, regardless of the location of the user and the equipment (i.e. one terminal to access several computer systems). More and more computer users are beginning to realize the added power available from their present systems simply through the interconnection of them. In addition, with the decreased cost of computer power and the increased speed and capabilities of microprocessor based systems, many companies are beginning to move away from the large central mainframe computers and toward a distribution of smaller systems. In many cases this has created what has become commonly known as "islands of automation." In essence this means that both information and computer power have been trapped in small areas with no means of sharing between the different areas.

In the past, a company which installed microprocessor based equipment could expect to spend ten times the cost of

the component on integrating it with its environment. Computer communications were developed into networking to combat these problems. Stallings defines a local network as "a communications network that provides interconnection of a variety of data communicating devices within a small area."[1] This means that any device, including computers, terminals, and sensors, that communicates over some type of transmission media should be capable of connecting to the network. A local network is generally privately owned and usually travels only short distances, typically residing in one building or in a cluster of buildings located close together. Internetworking may connect several such local networks to form a Wide Area Network. Other characteristics include high data transfer rates (0.1 to 100 Megabytes per second) and low error rates (10^{-8} to 10^{-11}). It is important to note that since these are communications networks and not necessarily computer networks, all attached devices do not have to be intelligent.

This chapter will cover the uses of Local Area Networks in the factory environment as well as some of the fundamentals of networking. This information is intended to give the reader a cursory understanding of the subject matter so that he can better comprehend the material presented later in the writing. If the reader does not

---

[1] Stallings, Local Networks: An Introduction (New York, Macmillan, 1984), p. 2.

fully understand the fundamentals presented or wishes to gain a more detailed understanding, he should refer to the book Local Networks: An Introduction by William Stallings.

2.2 LANs in the Factory

In the past, the main concern of factory communications was the transfer of programs to robots and CNC machine tools and the collection of manufacturing data. Recently, the requirements placed on factory communications (and therefore factory networking) have grown considerably. It is now desirable for intelligent manufacturing devices to have the capability of exchanging information with other intelligent devices. This allows the shop floor equipment to monitor and control production. When design information is added, these devices can also compare design specifications to manufacturing output and make adjustments in the production process to eliminate the differences. All of this is accomplished without the need for human intervention resulting in a faster responding system which produces a better quality product.

In a paper presented at the 1986 International Conference on Industrial Electronics, Control and Instrumentation, Henrik A. Schultz [16] presented a four tier hierarchy for factory communications (see Fig. 2-1). The first tier of this model is corporate level communications. This involves all information transfers

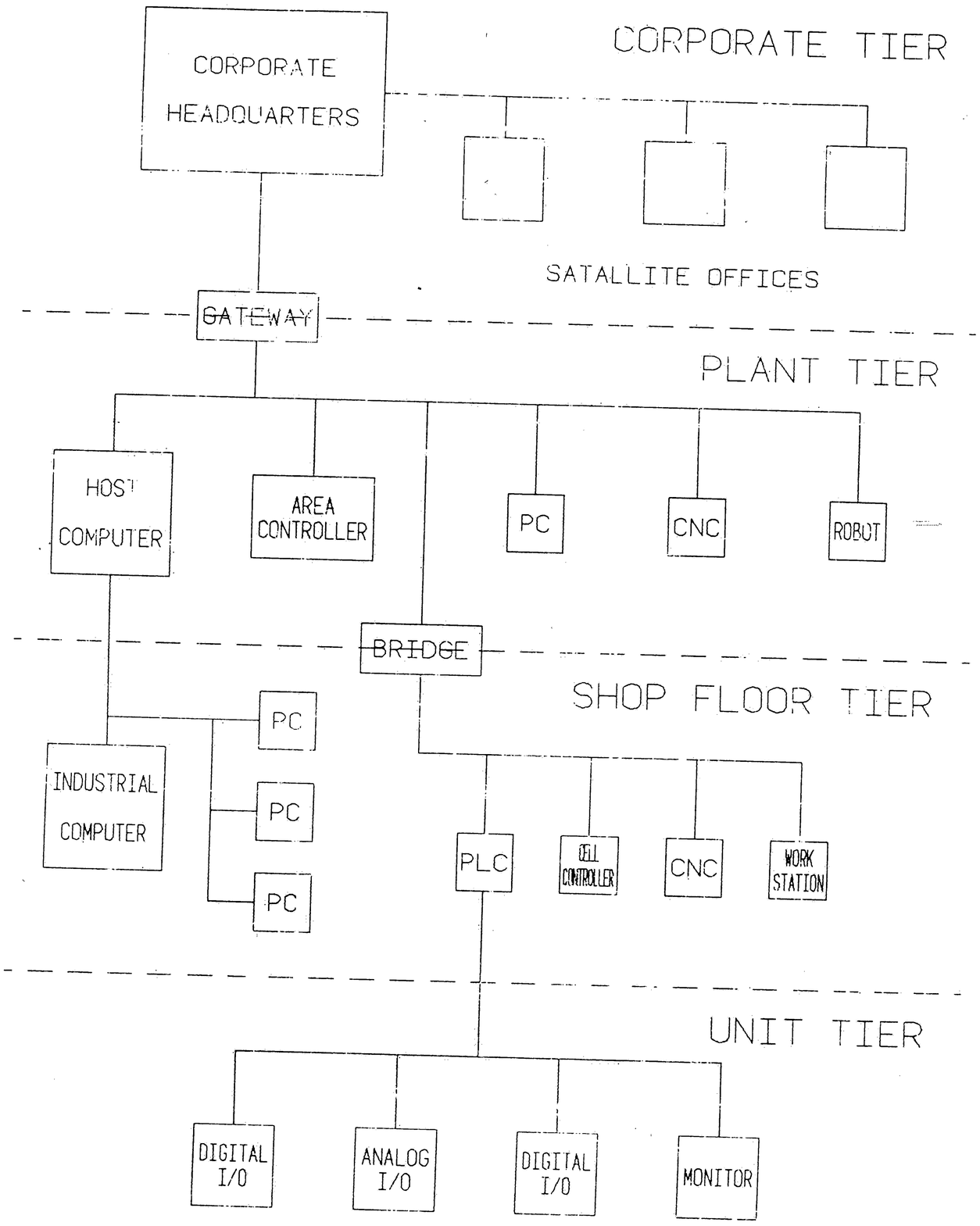

Figure 2-1. A four-tier model for factory communications.

required for the administrative division of a company to support the primary function of the company. This includes operations such as data processing, payroll, and management discussions pertaining to the overall operation of the company.

The next tier, plant level communications, supports all supervisory and monitoring activities to integrate diverse functional areas within a factory. Applications in this area would include scheduling, materials management, work order distribution, maintenance management and other overall plant management information exchanges. Communications in this tier are considered to be time sensitive; however, they do not involve real-time control and are therefore not time critical. Devices which would be interconnected to support these communication requirements would range from host computer systems to programmable logic controllers (PLCs).

The next tier is on the shop floor itself. This level includes the interconnection of programmable devices and controllers. An application of this level could be a small manufacturing cell in which a cell controller communicates with robots and machine tools within the cell. The type of communications carried out at this level often require real-time responses.

The lowest tier of this hierarchy is the unit level communications. Although this level does involve the exchange of signals, it does not involve communications in

the same sense as the communications within the other tiers. In this case the communications involve the signalling carried out between sensing, actuating and display elements and their controlling device, typically a computer based control unit. An example of the connections at this level would be the connection of a controller to the axis drives of the machine tool. This type of connection is not currently handled by LANs and it is not probable that LAN technology will ever become involved with factory communications at this level. Therefore, there has been no attempt at standardization at this level.

There are several factors present in factories which are not present in the office environment that make the requirements of a LAN for an office quite different from those of a LAN in a factory. One of the more important of these factors is the "real-time" requirements of the control and feedback which exists in the factory. These real-time requirements create a demand for some type of upper bound on the message delivery time. In essence, this seems to dictate that the message transfer routine must be deterministic (meaning the capability of setting limits on its operation must exist). These control situations also require that there be high data integrity. Low error rates are necessary for the same basic reason that the deterministic message delivery times are required. It should not be difficult for the reader to imagine the

usefulness of an error filled message even if it arrives well within the prescribed time limit. Maintaining this data integrity is especially difficult in a factory where there is a high level of electromagnetic interference present. This requires that the data channel be shielded.

LANs are now being used in both discrete parts manufacturing (for file service and control of robots and machine tools) and the process industries (for process control according to the input of sensors). To perform this task, factory LANs (as well as all other types of LANs) must provide a means of connecting a wide range of devices from a variety of vendors.

2.3 Local Area Network Fundamentals

There are four basic topics of discussion that should help the reader to better understand what local area networks are and what types of considerations are involved in their development. These areas are the type of topology, transmission media, transmission techniques, and media access control methods used to implement the network. Each of these areas will be discussed in the following section. This information is intended to give the reader a fundamental understanding of computer networking. All areas which are important to local area networks are not covered, only those which are most important to the understanding of the remainder of the material presented in this writing.

2.3.1 Topology

The network topology refers to the manner in which the devices will be connected together. There are four basic types of topologies:

(1) dedicated

(2) star

(3) ring

(4) bus/tree.

Although this paper is primarily concerned with network architectures which use a bus topology, all four types will be discussed briefly so that the reader may gain an understanding as to why the bus/tree systems are the most commonly used.

2.3.1.1 Dedicated

The dedicated or mesh topology (see Fig. 2-2) is actually not really a topology at all. Rather it is the alternative to some type of network which uses one of the other types of topologies listed above. In this approach each device on the network is connected to all other devices on the network. This may be a practical solution when two, three, or even four devices must communicate with one-another. However, as the number of devices increases beyond that level, this interconnection scheme becomes a jumble of wires.

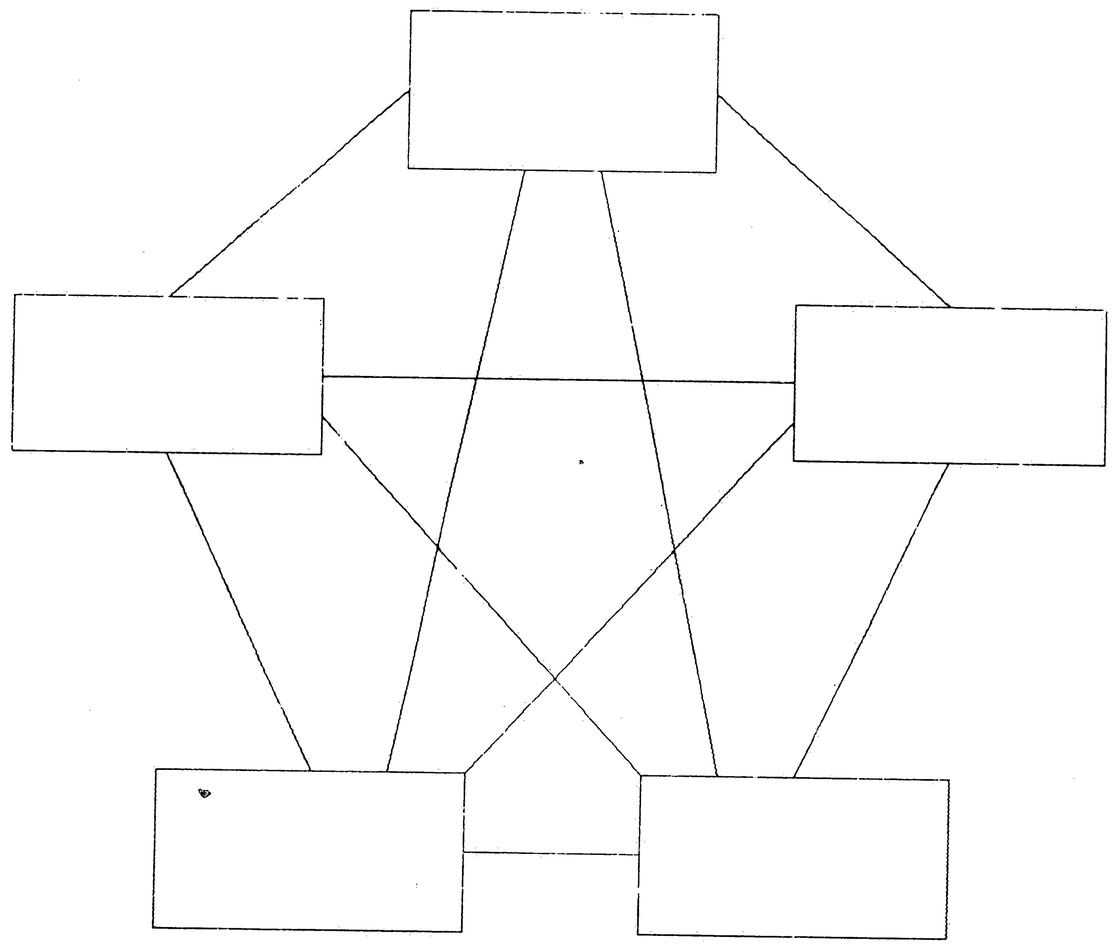

Figure 2-2. Dedicated topology.

The problem with this type of network should be easy for the reader to see. Each device on the network has a direct, dedicated link with all other devices on the network. Therefore, if there are N devices, then N(N-1) links are required, and each device must have (N-1) I/O ports. The cost of such a system, in terms of the hardware alone, will therefore grow with the square of the number of nodes on the network.[2]

2.3.1.2 Star

With the star topology (see Fig. 2-3), each node in the network is connected to one central node. All communications are then passed through and controlled by this central node. If a node wishes to send a message to another node on the network, it sends a request for communications to the central node. The central node then sets up a dedicated circuit between the two nodes that wish to communicate. The link between these nodes then becomes a virtual point-to-point link.

The advantage of this topology is that it requires very little intelligence at each of the communicating devices. All such logic is stored in the central node. The nodes only need to have the logic required for simple point-to-point communications. This topology, however, has an inherent disadvantage. It requires that the central

---

[2] Stallings, p. 54.

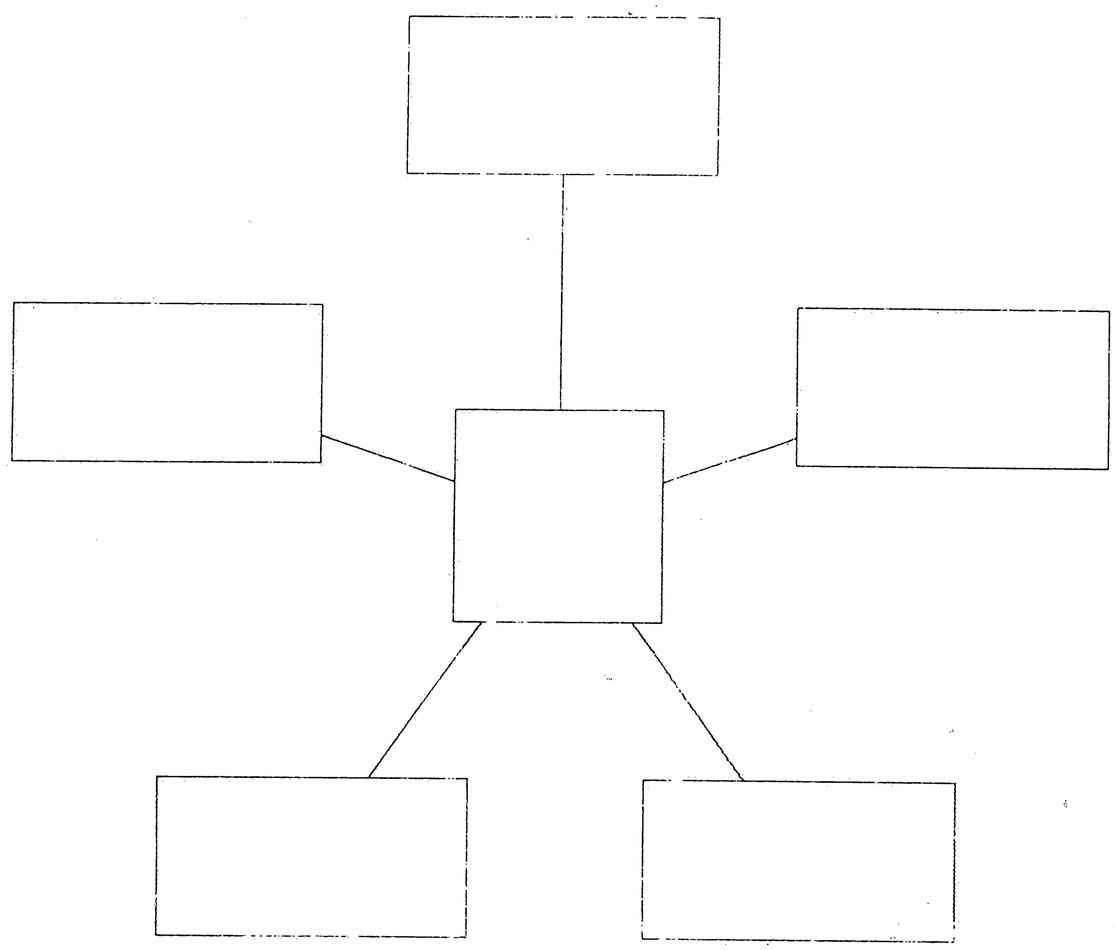

Figure 2-3. Star topology.

node have rather complex control logic. In addition, the network relies completely upon the central node. If this node should go down, the entire network will cease to function, unless a "hot" backup is provided.

2.3.1.3 Ring

In the ring topology (see Fig. 2-4), adjacent nodes on the network are connected to one-another in a point-to-point fashion. When all of the nodes are connected, a closed loop is formed. Nodes then communicate with one-another by sending packets of information around the ring. Communications are generally unidirectional. Therefore if the communications travel clockwise and a node wishes to send information to a node that is adjacent to it but in the counterclockwise direction, the message must travel the entire ring until it reaches its destination. In fact, in most architectures the message is removed from the ring by the same station that sends it. This is done as a form of verification. If a station receives a message exactly as it sent it, then it assumes that all other nodes on the ring (including the destination node) also received it in tact. In cases where this scheme is used the message does, in fact, travel the entire ring.

In contrast to the star topology, each node on the network contains all of the logic necessary to communicate. Control of the network is therefore decentralized. The

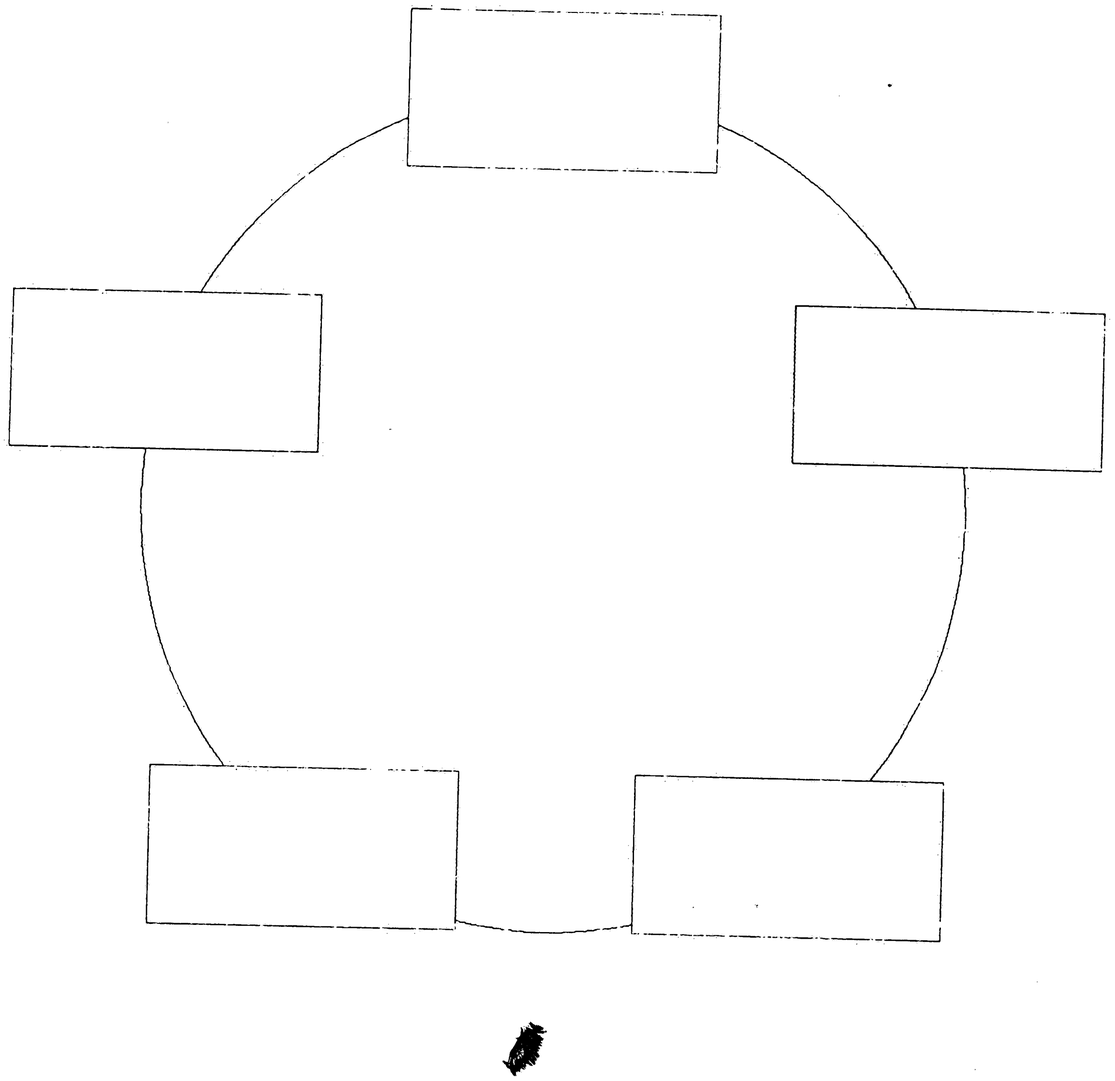

Figure 2-4. Ring topology.

nodes are usually connected to one-another with repeaters which simply pass the information they receive onto the next repeater station. The nodes themselves are then responsible for all other aspects of communication (packetizing and media access control). One of the major disadvantages with this type of topology is that if one of the repeaters on the network goes down or if the ring is broken for any other reason, communications are not possible.

2.3.1.4 Bus/Tree

The bus topology (see Fig. 2-5) is the topology that is used in both of the architectures covered in this paper. In this method all nodes on the network connect directly to one central communications media. This central media is known as the backbone. With this method only one device on the network can communicate at a time. In addition, all nodes on the network must have the logic necessary for all aspects of communication including the logical and physical portions. Any signal placed on the media by any station on the media can be received by all other stations on the media. Because of this, the bus topology is also known as broadcast.

The bus topology uses one backbone to which all devices are attached with drop cables. The tree topology is only slightly different in that it allows for branching from the backbone. These methods differ from the star and the ring

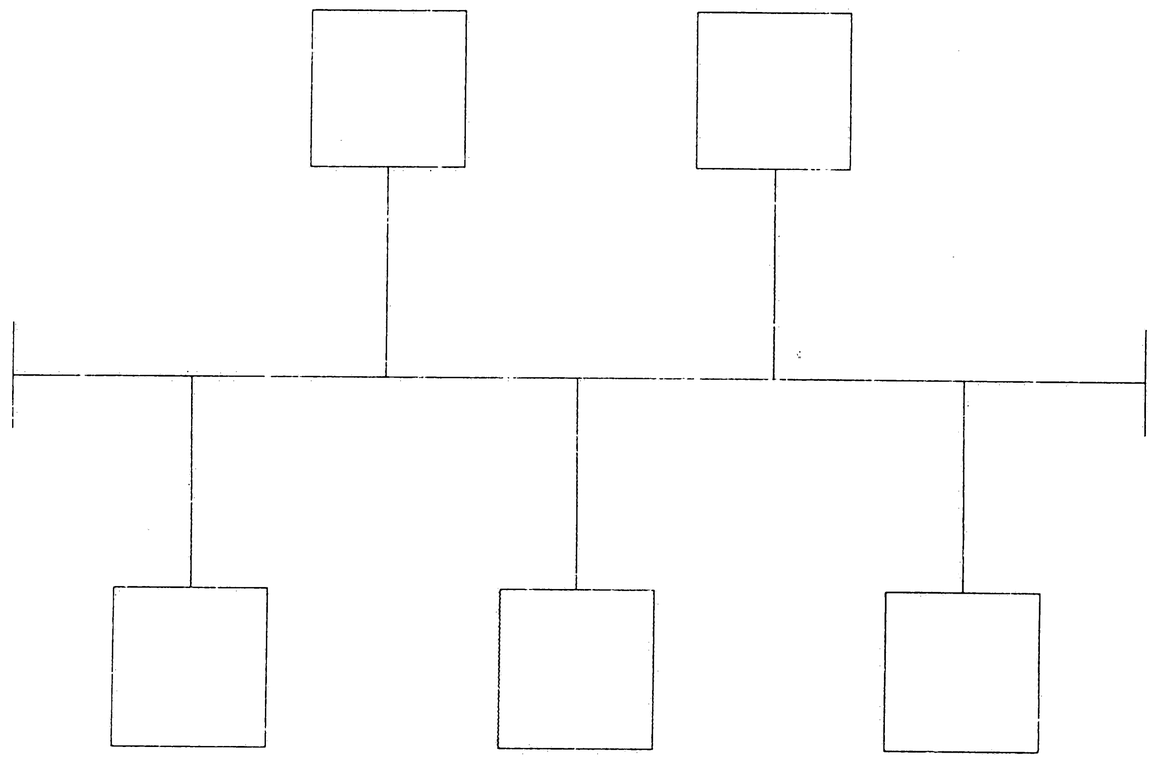

Figure 2-5. Bus topology.

approaches in that the transmission media does not actively participate in the communication of information by physically passing (repeating) the data transmitted. It is merely a channel over which the nodes exchange information. More detail will be given on this topology when the two architectures are discussed.

2.3.2 Transmission Media

There are several basic types of transmission media used in factory networking today. These media include:

(1) twisted pair wire

(2) coaxial cable

(3) fiber optic cable.

The choice of which of these is best suited for a particular situation is dependent upon the data rate required, the modulation technique and signal type used by the network architecture chosen, data integrity considerations (immunity to electromagnetic and radio frequency interference present in the environment), ease of tapping the media for nodes, the overall length of the network, and the cost. Each of the media will be discussed according to these considerations in the following paragraphs.

2.3.2.1 Twisted Pair

Twisted pair consists of two wires arranged in a spiral pattern. The twist length is designed such that the signals

on the two individual wires will not interfere with one-another. The pair typically has a thickness from 0.4 to 1.4 millimeters.[3] The wire itself is usually made of copper or copper coated steel covered in a protective sheath. This type of media is very popular. It is most widely used for the interconnection of telephones.

Twisted pair can be used with either digital or analog signalling but can only be used in baseband networks. It is the least expensive media in both purchase and installation cost. Tapping the twisted pair is very easy. Data rates are in the Megabytes per second (Mbps) range. Possible transmission lengths (without the aid of repeaters) are in 10s of meters.[4] The major disadvantage to using this type of media is its limited bandwidth. This limited bandwidth reduces the data capacity of the medium. Low resistance to electrical noise is another disadvantage of this media. Immunity can be helped by proper shielding of the wires. However, high noise resistance is not easily achieved.

2.3.2.2 Coaxial Cable

Coaxial cable is employed most commonly in cable television technology. It has been cited as the best media for network communications due mostly to its versatility.

---

[3] Stallings, p. 59.

[4] Mark Stickler, "Local Area Networks (LANs) for the Office Environment", CIM Lab Seminar Series, Lehigh University, June 30, 1987.

Like the twisted pair system, coax consists of two conductors. In this case, however, one conductor is encased within the other with a dielectric material between them. The inner conductor can be either a solid or a stranded wire and the outer conductor is either a solid or braided cylinder. The two wires are surrounded by a protective sheathing. The overall diameter of the wire can be anywhere from 1.0 to 2.5 centimeters.[5]

This media is capable of supporting data rates in the 10s of Mbps and distances in the 100s of meters.[6] It has relatively good immunity to the effects of EMI and RFI and is therefore capable of achieving good data integrity in the factory setting. Both baseband and broadband signaling can be used. Tapping into the media is easy although it is not quite as easy as with the twisted pair system. The cost of this media is high and stable. Most of the cost can be attributed to the high cost of installation (the cable itself is not very expensive).

2.3.2.3 Fiber Optics

Fiber optics is the most rapidly growing network communications media technology today. It began to emerge several years ago and is now very popular in the telecommunications industry. This media is comprised of an

---

[5] Stallings, p. 61.

[6] Stickler, "Office LANs."

extremely thin (50 to 100 micrometers) strand of glass which is used to conduct an optical ray.[7] The fiber is surrounded by a cladding material to isolate the fiber and keep the optical signal from escaping. Signals are injected into the fibers with either a LED or a laser and are read by a photo-detecting diode. The fibers are very flexible but are also brittle.

The optical fiber offers the greatest transmission distance and the highest data rate of all the media discussed. The data rates are in the 100s of Mbps and distances of several kilometers.[8] Fiber optics also have the highest level of data integrity because the optical signal is not effected by EMI or RFI. The greatest disadvantage of using fiber optics as a transmission media is the high cost. A good portion of this cost is due to the extreme difficulty involved in tapping the media. However, new developments in this and other areas are causing the cost of this media to fall rapidly. At present only baseband signalling can be used but this too may change with the development of new technologies.

2.3.3 Signaling Techniques

There are three signal transmission techniques commonly used for LAN applications: baseband, broadband, and

---

[7] Stallings, p. 62.

[8] Stickler, "Office LANs."

carrierband. The method chosen has a significant impact on the performance of the network and represents one of the largest differences between various architecture specifications. Consequently, there has been much discussion as to which method best provides for the needs of a factory network. Since the determination of which method is most effective is entirely dependent upon the environment in which the network is installed and the variety of possible environments is essentially limitless, this paper will not attempt to determine which method is best suited for factory networking in general. Instead, a description of the operation and advantages and disadvantages of each method will be presented. It will be up to the reader to determine which method is best suited for a particular situation.

The most simplistic of the three techniques is baseband signaling. With this method the data is placed on the medium in the form of a digital signal represented by voltage fluctuations.

One of the major disadvantages of using baseband is that the signal usually consumes the entire bandwidth of the medium making frequency division multiplexing impossible thereby reducing the data capacity of the medium. Another disadvantage is that transmission distances are generally limited to 1 kilometer. This is due to the fact that the signal disperses as it travels along the medium causing a

reduction in the difference between voltage levels which represent high and low values making the signal difficult to interpret at distances greater than 1 kilometer. Repeaters which reconstruct and retransmit the signal can, however, be used to extend the achievable transmission distance.

The major advantages of using this technique stem from the fact that no modulation of the signal is required. Because no modems are used the cost of the network installation will be reduced (compared to using those techniques which require modems). In addition, the design and maintenance of the network is easier.

Broadband systems use an analog signal to carry data on the medium. This allows for the use of frequency division multiplexing (FDM) to divide the medium to be into many carrier channels thereby increasing its data capacity. Broadband systems usually transmit signals on one range of frequencies and receive signals on a second range of frequencies. The channels provided by FDM must, therefore, be divided among transmit and receive. In addition, the signals must pass through a device known as a "head-end" which translates the incoming signal to the outgoing frequency and retransmits it.

The major advantage of using broadband signalling is the ability to divide the medium into many channels. This not only increases the data capacity of the medium but also allows it to be used for several purposes. For example,

some of the available channels can be used for a LAN while others are used for carrying voice and video signals. A second advantage of this method is that the analog signals can propagate a longer distance (compared to baseband signals) before becoming distorted and unreadable. Distances of 10 kilometers can generally be covered before re-amplification of the signal is required.

The major disadvantage of using broadband systems is the cost. This extra cost is for the purchase and maintenance of modems and other devices which are required to support analog signaling.

Carrierband signalling is a relatively new technique. It is essentially a combination of the techniques of baseband and broadband methods. It uses the same signaling technique as broadband systems, however, a single channel is used for both transmitting and receiving the signal.

An advantage of this method (when compared to broadband) is that the design of the modems is much more simple. In addition, since signals are transmitted and received on the same channel, the head-end remodulator is not required. These factors work to reduce the cost of the network installation.

The primary disadvantage of this method is that the medium cannot be used for other purposes (such as the transmission of voice and video signals). Therefore the data carrying capacity of the medium is reduced.

2.3.4 Media Access Control (MAC)

The MAC is basically the set of rules each node on the network follows to gain access to the transmission media. Many methods of MAC have been developed over the years. In most cases these methods can be classified as either a contention scheme or a token passing scheme. This section will discuss the most popular of each of these types of media access control methods. Carrier Sense Multiple Access with Collision Detection (CSMA/CD) will be covered as an example of the contention method and Token Bus as an example of the token passing method.

2.3.4.1 Contention

Probably the most widely used method of media access control today is CSMA/CD (also known as listen-while-talk). In this method, a node wishing to send a message to another node first listens to the media. If the node does not hear any other nodes communicating on the network, it sends its message. This would be a very simple and straight forward method of media access control except that the propagation delay of communications introduces some problems. For example, suppose two nodes at the opposite ends of the network wish to communicate with the nodes adjacent to them (see Fig. 2-6). Node A at one end of the network listens to the media, detects no traffic, and begins to transmit its

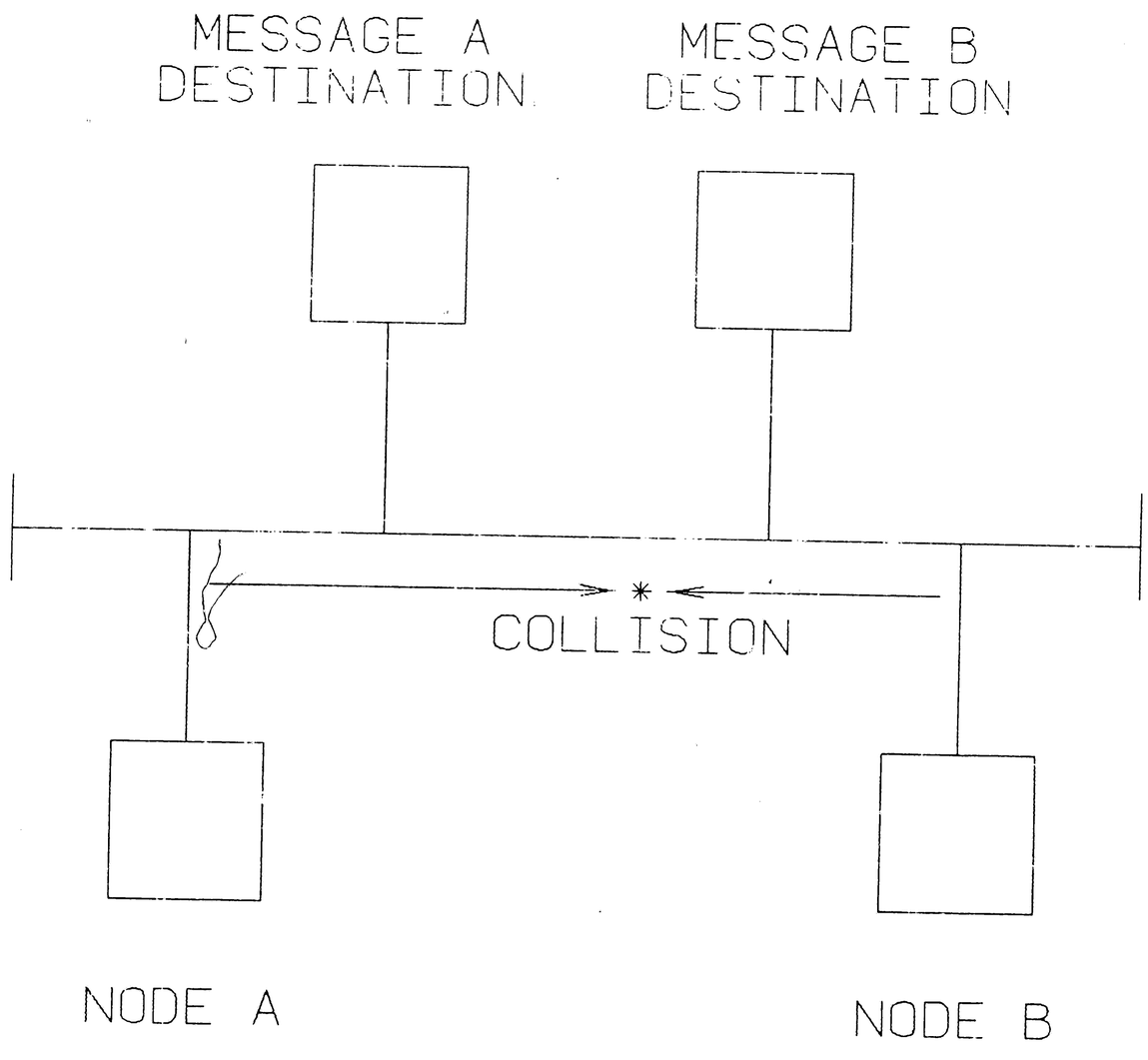

Figure 2-6. Nodes contending for network media.

message just before node B at the other end of the network performs the same task. Node B then listens to the media but does not hear the message node A is sending because that message has not yet had time to propagate to it. This would result in both of the nodes attempting to use the media at the same time (a collision) which destroys the signals from both of the transmitting nodes. Both of the nodes will detect this collision by determining that the signal on the media does not sound like the signal it has sent. When a collision is detected the nodes immediately stop sending their messages. A jamming signal is then sent over the media to let all of the other nodes know that a collision has occurred. Nodes A and B will then each wait for a random amount of time before attempting to gain control of the media again. Each time a node is involved in a collision before it has an opportunity to send its message, it doubles its delay time (known as a binary exponential back off algorithm). In this manner, one of the two nodes in contention for the media will eventually start sending long enough before the other node so that the message has time to propagate to the other node, thus preventing a collision.

One of the requirements of this type of media access is that there is a minimum message size. This minimum size is required so that a node cannot send its entire message before detecting a collision with the message of another

node. The minimum size is therefore two times the maximum propagation delay which is the delay for the two nodes which are farthest apart on the network.

2.3.4.2 Token Passing

In the token passing schemes the LAN is logically represented as a ring even if it has a bus topology. Each node in this logical ring knows the identity of its predecessor and successor. A token, which is actually a packet of control information, is then passed from station to station. The station which maintains control of the packet also controls the media. It can send messages to and request responses from other nodes on the network. A node need not have possession of the token to respond to the request of a node that does. In fact, a node does not even have to be a part of the logical token ring to respond. It simply must be at an addressable location. In this case, a node which is part of the logical ring can request data from a node that is not on the ring (but is addressable) and temporarily give that node the right to transmit on the media. A node must, however, have possession of the token to transmit a message or to send such a request to another node on the network. The token is transferred from the controlling node to its successor in the logical ring when the controlling node has finished with its communications or when a maximum token holding time has been reached.

Unlike the contention scheme, token passing requires some type of maintenance. There are several functions which each station on the logical ring must be capable of doing. The first of these is ring initialization. In this function a node will issue a claim-token packet if it does not detect bus activity for a certain period of time. This station will then have possession of the token and the network will continue to operate normally. If several stations send a claim-token packet simultaneously, the conflict is resolved by an address based contention scheme. The second function, addition of a node, is performed periodically by a token holding node. To give nodes not yet in the logical ring the opportunity to join, the token holding node issues a solicit-successor packet. A responding node is sent the token. When it receives the token it links itself to the appropriate predecessor and successor stations. The deletion of a node, the third function, is performed very easily. When a node which wishes to drop out of the logical ring receives the token, it simply (logically) splices its predecessor and successor together. The final function is fault management. The network must have a means of responding to failures such as double tokens, lost tokens, and broken rings. When a station holding the token detects another station which holds a token, it immediately drops its token. This will leave the network with either one or zero tokens. If one remains, normal operation continues.

If there are no tokens, the ring will go through a re-initialization procedure. When a token holding node detects that the (logical) ring is broken, it will issue who-follows packets until it finds a node which is properly connected to the ring. If no such node is found, all communications on the network are stopped as the token holding node drops the token and begins to listen to the media.

2.3.4.3 Contention vs. Token Passing

As with the signaling techniques, no one scheme is best in all situations. Each has advantages and disadvantages. The major advantage of CSMA/CD is its simplicity. The relatively complex logic in token passing requires more intelligence at each of the nodes. On the other hand, the token passing scheme is deterministic. This means that the amount of time between the opportunities for a particular node to use the media has an upper bound. In the contention based systems this time can only be represented statistically. This determinism also allows the network to perform well under heavy loading conditions where a contention based system may spend more time attempting to resolve collisions than it does in actual communications. The amount of control offered by token passing also allows for the prioritizing of nodes and the overall regulation of network traffic. In addition, since the control of the

media is predetermined, a minimum message length is not required as it is for the purpose of collision detection in the contention based scheme. This eliminates the need for bit stuffing for small messages and provides for better utilization of the media. The disadvantage created by the additional control of token passing is the high overhead. Nodes on the network will have to wait for the token to communicate even if no other node on the network is interested in controlling the media.

2.4 The Open Systems Interconnect Model

In 1977 the International Standards Organization (ISO) formed a committee to establish a standard framework for the development of communications architectures. The work of this committee resulted in the creation of the Open Systems Interconnect (OSI) model. The model was built with a layered approach. ISO used the following structural guidelines for the model:

1. A layer should be created where a different of abstraction is needed.

2. Each layer should perform a well defined function.

3. The function of each layer should be chosen with an eye toward defining internationally standardized protocols.

4. The layer boundaries should be chosen to minimize the information flow across the interfaces.

5. The number of layers should be large enough that distinct functions need not be thrown together in the same layer out of necessity, and small enough that the architecture does not become unwieldy.[9]

In following these rules, the ISO arrived at a seven layer model (see Fig. 2-7). Each of these layers is discussed in the following paragraphs starting with the lowest layer.

2.4.1 Physical Layer

This layer is concerned with the mechanical and electrical aspects of transmitting a bit stream from one station to another. It is the only layer which maintains a physical connection between the two devices which are communicating. All other layers connect virtually with their peers. The logic in this layer is not concerned with the message or even the characters being sent between the two communicating nodes. It is only concerned with sending bits. A responsibility of this layer associated with the transmission of bit streams is detecting errors in the bit patterns. Design issues of this layer include the type of cable plant used, connectors used, signal integrity, and protection from the physical environment (i.e. lightning protection).

---

[9] Tannenbaum, Computer Networks (Englewood Cliffs, N.J., Prentice-Hall, Inc., 1981), p. 15.

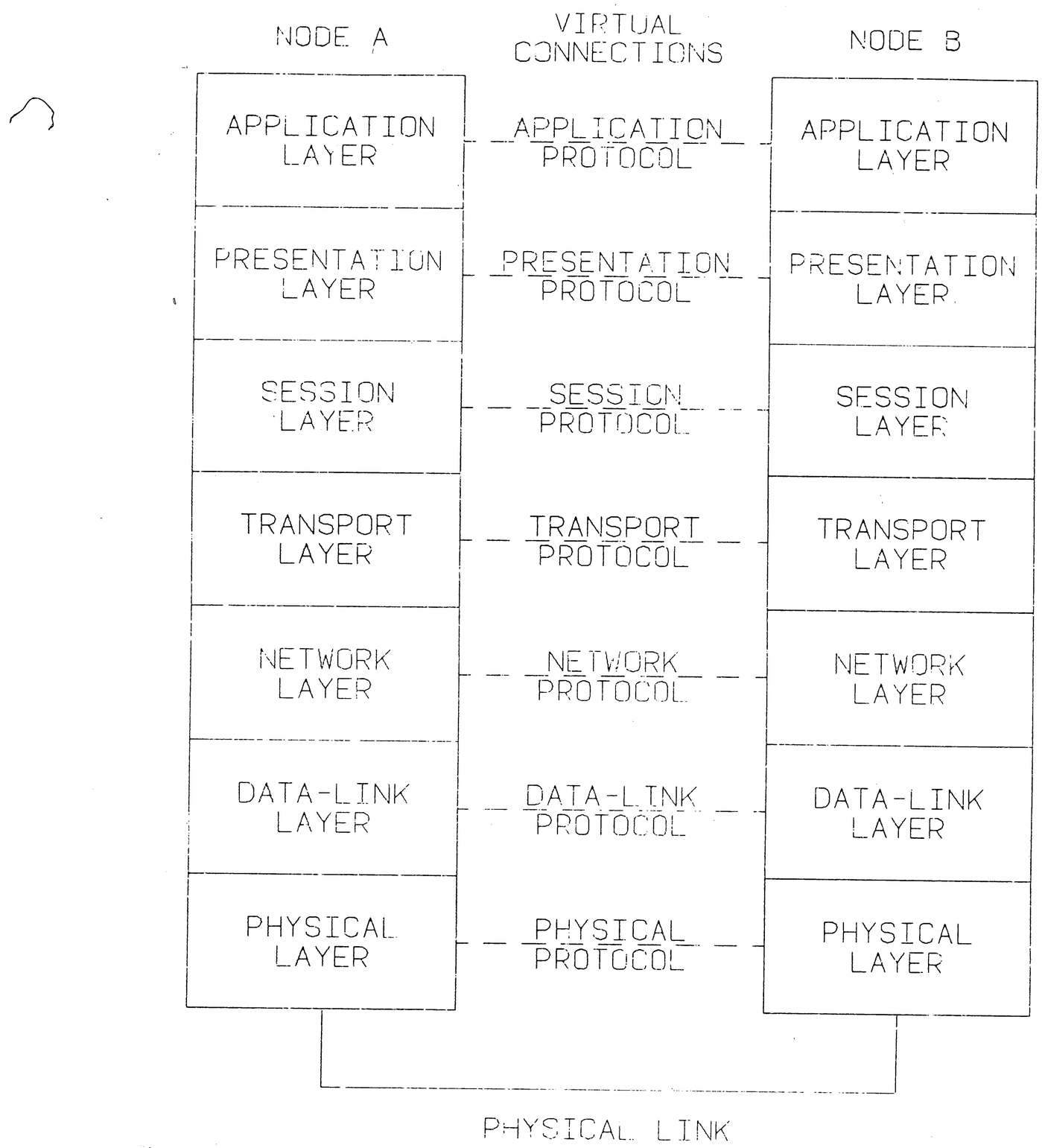

Figure 2-7. ISO Open System Interconnect model.

2.4.2 Data-Link Layer

The Data-Link Layer has two major responsibilities. The first of these is to break the message up into bit patterns for the physical layer. It also attempts to make the physical layer more reliable. In addition, it adds flags at each end of the frames of data. It also often adds sequence numbering to ensure that the segments of the message arrive or are reassembled in the proper order. Timing is another function of this layer. It is here that information is placed in buffers for situations in which the sending device is faster then the receiving device. In some cases, error checking routines are also included in this layer. These routines would be structured to detect burst errors as opposed to single bit errors for which the physical layer is responsible.

The second major responsibility of this layer is Media Access Control. It is here that the functions described in Section 2.3.4 are performed. These functions provide a means to activate, maintain, and deactivate the (logical) link between two devices on the network.

2.4.3 Network Layer

The network layer function is to establish a connection between the two communicating nodes. This mainly involves the addressing and routing of the information packets. It is the job of this layer to make the connection between the

communicating nodes transparent to all of the higher layers in the model. It is also the responsibility of this layer to monitor and control the traffic on the network as a whole. Therefore, this layer handles the gathering of statistics and accounting information on the operation of the network. If the network has a system of priorities, they are also maintained by this layer.

2.4.4 Transport Layer

The function of the transport layer is to ensure that the session layer information is transmitted in the proper sequence, with no errors, and no duplication. It is often thought of as the first end-to-end layer because it is the first one concerned with the delivery of the message and not the actual transmission of the message. This layer is responsible for recovering from errors rather then detecting them. In general it is responsible for the disassembly and assembly of session layer messages.

2.4.5 Session Layer

This layer is responsible for establishing the connection between two nodes. It uses a data base to determine the address of users according to a name. It is also responsible for mapping the connection to the transport layer. Functions such as the synchronization between nodes and the buffering of data are also handled by this layer.

It provides a means for the user to establish, maintain, and terminate a connection with another user.

2.4.6 Presentation Layer

The presentation layer is responsible for transforming the application layer information into a format that can be understood by the destination node. It is here that character codes, file formats, and data types are translated to maintain uniformity across the network. The translation is into a common network format rather than into the format the end node expects. The reverse translation is then performed at the destination node. It is in this layer that data encryption is performed for network security purposes.

2.4.7 Application Layer

The final layer of the OSI model is the application layer. This layer is basically the layer with which the user interfaces, making all other layers transparent. It provides services such as log in and password checking as well as text formatting. It is here that all of the applications of the network are performed. It is also here that vendor programs such as electronic mailing systems reside to give the user the ability to communicate with other nodes on the network and obtain information from them.

2.5 Subnetworks and Internetworking

In many organizations it is either impossible or not impractical to interconnect all devices in one network. This has brought about the development of subnetworks and internetworking. Several smaller networks linked together into one larger network, called a catenet, offer many advantages. Among these are improved performance, reliability, security, convenience, and geographical coverage.

Performance is improved for the simple reason that the number of devices attached to a single network is reduced. The fewer nodes there are on the network, the faster it will respond to user requests. In the case of contention based systems this is due to the increased probability that a particular node will be able to attain control of the network without experiencing a collision. In the case of token passing systems this is due to a reduction in the amount of control overhead required and the amount of time between possessions of the token. In both cases the amount of delay experienced by the end user will be reduced.

Reliability is increased because failures in one subnetwork will not effect the operation in another subnetwork. If the network goes down for some reason, then just those devices in the subnetwork will be effected. If all the devices were connected to a single network, then the operation of all devices on the network may be affected.

Convenience is improved by subnetworking for several reasons. First, installation is made easier as smaller areas of the organization can be networked and proven before they are added to the catenet rather then attempting to install one large network. Second, different areas of the organization may lend themselves to different types of network architectures. Subnetworking allows for the utilization of the best possible network solution for each individual area. The third reason is related to the advantage of improved geographical coverage. Consider a case in which a network must interconnect devices in two different buildings which are separated by some obstacle, such as a highway or a river, which makes stringing a cable difficult. A subnetwork could be placed in each building and the two could be joined by a bridge (which will be discussed later) which utilizes microwave technology to cross the gap between the buildings.

The geographical scope of the network can also be improved by internetworking. In this case, high speed data links can be used to join subnetworks that are located far apart. For instance, if a company wishes to network its factories in several different cities, than high speed bridges can be used to internetwork subnetworks at each of the locations.

There are two major types of internetworking; homogeneous and hybrid. In homogeneous internetworking, two

(or more) similar networks are linked together to form the catenet. The two networks are connected by a device known as a bridge. This bridge must perform several simple functions. First, it must read all frames transmitted on network A and transfer those frames that are addressed to network B to network B and perform the same function for B to A traffic. Second, it must use the appropriate media access control protocol to access each of the two networks (the MACs may or may not differ between the two networks). The bridge must not change the content or format of the frames it transfers between the two networks. It also should not encapsulate the information (e.g. add a header). Finally, the bridge must have the capability to buffer data.

When several different classes of networks are joined together to form the catenet, it is known as hybrid internetworking. Using this scheme a LAN, a Computerized Branch Exchange (CBX), and a High Speed Local Network (HSLN) may all be interconnected. As mentioned earlier, the different types of networks may be used in different situations so that the best possible solution to the network needs can be chosen in several different areas. In most cases, such networks are interconnected using gateways. Like the bridge, the gateway has several distinct functions which it must perform. First, it must provide a physical link between the two networks. Second, it must provide the routing and delivery of data between the processes which

reside on the two different networks. Third, the gateway must provide a means of tracking the services provided by the networks (for accounting purposes). The final obligation of the gateway, and probably most difficult to perform, is to provide for any differences between the two networks being connected. These differences can include the addressing schemes, packet sizes, time-outs, error recovery routines, status reporting, routing techniques, and MAC.

2.6 Chapter Summary

This chapter has attempted to provide the reader with a basic background in factory LANs. The reader should now have an idea of why an organization would want to put a network into its factory. He should also have a fundamental understanding of what a Local Area Network is and what is required to install one. Other topics, such as the ISO-OSI model and subnetworks, were discussed so that the reader will be able to understand material presented in later chapters. It is intended that the reader should have a sufficient understanding of this material that he will not have to use outside references to understand the material to follow.

3 Evaluation of Local Area Networks

3.1 Introduction

The most important aspect of developing a networking solution is to determine whether or not the proposed design will perform adequately when in operation. The cost of actually installing and testing a network solution is prohibitively expensive so the network design must be proven before it is actually implemented. Consequently, much work has been done on developing methods by which networks can be tested in the design phase. This chapter is intended to give the reader an understanding of how simulation can be used to evaluate how well a local area network will perform. So that this material can be presented properly, the reader must first gain an understanding of the criteria used to measure the performance of networks as well as the factors that affect that performance. The first section of this chapter will therefore discuss what measures are used to judge network performance and what is characterized as good network performance. The next section will review some of the more important performance-affecting factors covered in Chapter Two and will also introduce some new factors. This section will attempt to show the reader how changes in these factors affect the overall performance of the network. It will be difficult to cover the first two sections independently, so there will be some cross-referencing

between them. The final section will cover the simulation of local area networks. Simulation is the method which will be used to evaluate the performance of the two network architectures that will be studied in detail in later chapters.

3.2 Network Performance Measures

The main reason that network performance changes under varying loads is that networks accomplish communications between nodes by transferring packet switched messages on a shared media. This means that the characteristics of the media and the type of media access control used will combine to determine how capable the network will be of handling the communication requirements imposed on it.

The amount of demand placed on a network can be explained with two factors: the offered load and the input load. The offered load is the actual total amount of data that is presented to the network for transmission. This includes all data that is put onto the network. In addition to the actual information to be transferred, the headers and footers added to this information are counted. The offered load is also comprised of control packets, such as the token in token passing, and packets that must be retransmitted due to errors. This factor is often expressed in terms of the total data capacity of the network media. The second factor, the input load, is defined as the amount of data

generated by the stations attached to the network. This is a function of both the number of stations on the network and the type of stations (which determines the amount and rate of data they supply to the network). The input load is not separate from the offered load but is a subset of it.

There are three factors which are commonly used to judge the performance of a local area network. They are utilization, delay, and throughput. Utilization is defined as the fraction of the total capacity of the network being used. Because local area networks offer such a large bandwidth and high data rates, this factor is not very important. It must, however, be considered since the data capacity of local area networks does have a cost and one of the primary responsibilities of the network designer is to keep such costs to a minimum.

The second factor which is commonly used to measure the performance of a network is the delay. The delay is the amount of time which passes between the time that a packet of information is ready for transmission, and the successful completion of that transmission. This time is comprised of the time a station waits for access to the media and the amount of time it takes the message (and its acknowledgements) to propagate across the media. This is the factor which is probably the most noticeable to the end user.

The last of these factors, the throughput, is the probably the most important of the three. The throughput is defined as the actual amount of data being transmitted between nodes on the network. It is only concerned with successfully transferred data including the overhead to the packet transmission (headers and trailers) but not including data transmissions involving errors. It is usually represented by the number of bits transmitted. It is also often normalized and expressed as a fraction of the total capacity of the media. The result of such a calculation is a percentage value. This value is sometimes interpreted as the utilization of the network which is another reason why the utilization, as previously discussed, is considered to be of less importance.

Throughput and delay are generally plotted against the offered load. Both factors increase with the increase of load on the network. The throughput of the network should increase along with the offered load to the point where the data capacity of the network is reached. Ideally, this increase would be in equal proportions until a saturation point is reached where the throughput levels off (see Fig. 3-1). In reality, however, the throughput will increase at a slower rate then that of the offered load. This is mainly due to the overhead and errors which are incurred in the data transmissions. The delay increases more rapidly relative to the offered load and does so without bound.

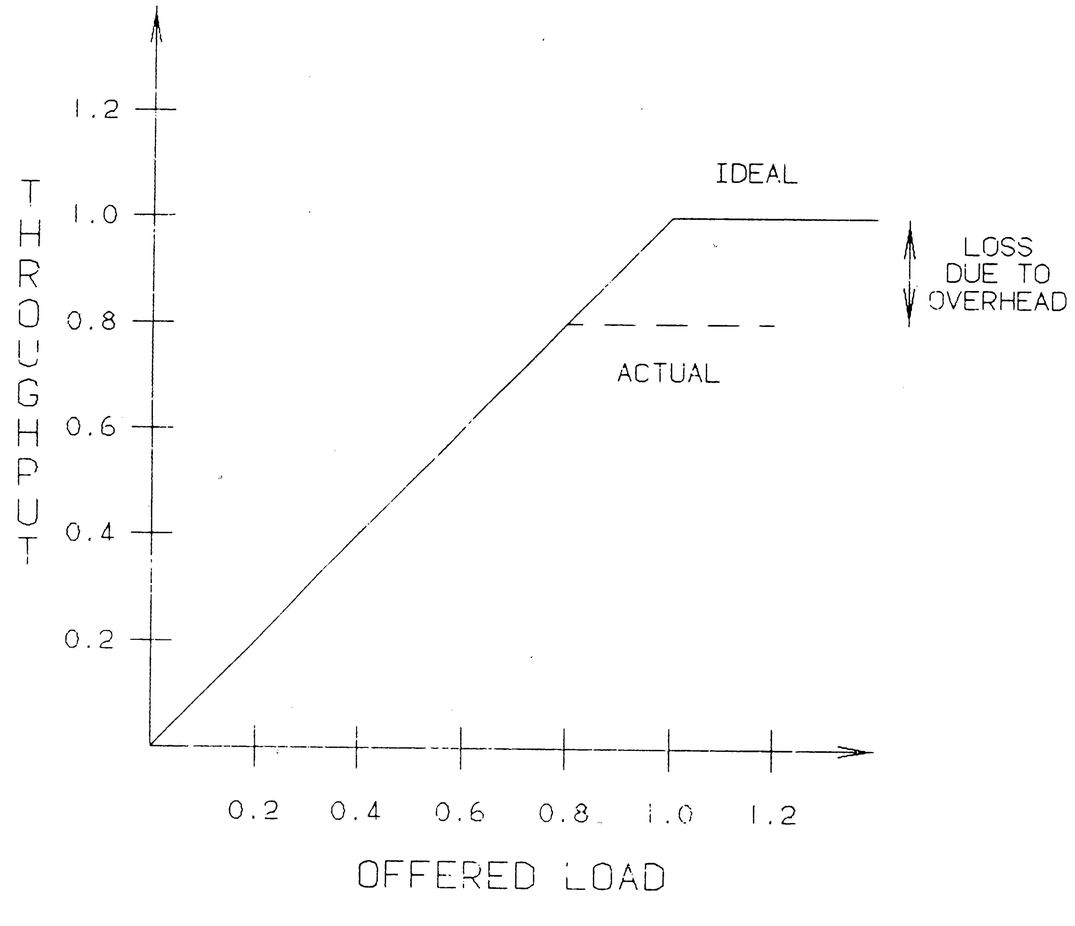

Figure 3-1. Ideal throughput vs. offered load.

This is due to the backlog of data transmission which develops that prevents a steady state from being reached.

It should be clear to the reader how each of these three factors should ideally appear. In an ideal situation the utilization and the throughput of the network should be as high as possible and the delay should be as low as possible. In order to put these requirements into a more easily defined perspective, Stallings [19] discusses three regions of network performance. His coverage of these regions will be summarized in the following paragraphs.

The first of the three regions which Stallings [19] defines is a region of low delay through the network. Here the data capacity of the network is more then adequate to process the load offered to it. Unfortunately, with this low delay there is also a low utilization of the network capacity. The second region is a region of high delay. In this case the network becomes a bottleneck to communications where too much time is spent in the control of communications relative to the actual transmission of data. Utilization of the media would be high in this case. The final region is the region of unbounded delay. In this region the offered load is greater than the data capacity of the network.

When designing the network according to these parameters, the designer should aim for the boundary between the first and second region. It should be clear that the

third region must be avoided. The statement that delay grows without bound in this region should be enough to indicate that there is a possibility that the delay may be infinite and that the message may never reach its destination. For a similar reason the second region should also be avoided. In this instance a surge of data on the network will increase the amount of delay incurred to unacceptable levels. In addition, operation of a network in this region indicates that the resources are not being used efficiently. The only remaining region in which the network can operate is the first region. However, this region also has its drawbacks. In this case there is a low utilization of the network which suggests that a more cost-effective solution is possible. For this reason the designer should aim for the boundary between the first and second regions. A network operating in this area will not contribute significantly to the overall delay of message transfer and at the same time will not represent an excessive solution to the requirements of the network.

Another parameter which is frequently used to determine how well a network performs is the product of the bandwidth (B) and length of the transmission media (d). The most useful way of looking at this is to compare the bit length of the transmission media (which is (B x d) divided by the propagation velocity (V)) to the length of a typical frame

or data packet (L). This gives the parameter "a" and the equation:

$$a = \frac{Bd}{VL} \qquad (3.1)$$

which is also equal to the propagation time on the medium divided by the time it takes a node to get a packet onto the medium. This parameter determines the maximum utilization of a local area network (1/(1+a)).[10] The utilization of the channel decreases as the value of "a" increases thereby decreasing the throughput of the network. Therefore, it is desirable to keep the value of "a" as low as possible. One of the ways in which this can be done is to increase the frame size for messages. Such an increase would, however, be wasteful if the frame size becomes larger than the average message. This would cause the bandwidth of the media to be wasted which would decrease the overall performance of the LAN.

3.3 Network Characteristics and Variables

There are three primary areas which have a strong impact on the performance of a local area network. They are: the channel characteristics, the network protocol used, and the network loading conditions. The first two of these are variables which can be controlled by the network designer. It is his job to choose the combination of these

---

[10] Stallings, p. 240.

characteristics which will allow the network to perform as prescribed above, in the most cost effective manner. Each of the characteristics and its effect on network performance will be discussed in the following paragraphs.

The channel or media characteristics determine, in part, what the data capacity of the network will be. These characteristics are comprised of those factors which determine the parameter "a" in equation 2.1. Bandwidth refers to the range of frequencies over which data can be transmitted. When this value is large, more data can be put onto the network at any given time. A good analogy for this is a highway. If cars are used to represent data and the number of lanes on the highway are used to represent the bandwidth, it is easy to see that when the number of lanes is large, more cars can pass a given point within a period of time than when the number of lanes is small. Using with the same analogy it can also be seen how the propagation velocity affects the amount of data that can travel on the media. The propagation velocity is the rate of speed at which the data can travel across the media. With the highway analogy this would be equated to the speed at which the cars are traveling. If traffic is moving at 30 MPH, fewer cars will pass a given point within a given period of time than if traffic is moving at 55 MPH.

The way in which the network designer chooses these factors is in his selection of the cable plant used. As

discussed in Chapter Two, different types of cable systems offer different bandwidths and propagation velocities. Bandwidth ranges from the most narrow of twisted pair to the widest of 75 ohm coax cable. For most types of cable used in local area networking the propagation velocity is approximately two thirds the speed of light or 2 X 10^8 meters per second.[11] The propagation velocity in fiber optic cable is equal to the speed of light.

A related characteristic is the topology or physical layout of the cable plant. Both the overall distance covered by the network and the distance between nodes on the network will have an effect on how well communications are carried out. One of the main reasons for this is the propagation delay increase incurred with an increase in distance that the message must travel. Both message transfers and acknowledgements will become more time consuming with a greater distance traveled. The overhead of control packet (the token) transfer will also be affected by the distance that information must travel on the network.

The overhead of network control is also strongly related to another of the characteristics which has a substantial impact on network performance; the loading conditions. Loading conditions are dependent upon the number of nodes or stations on the network and the offered load of each of those stations. As the number of stations

---

[11] Stallings, p. 237.

on the network increases, the amount of time required for the control of the network also increases. In the case of a token passing scheme, this is due to the fact that each of the stations must be given an opportunity to control the token (and therefore the network). When each of the stations holds the token, it can send a message. However, even if the station has no message to send, it still has control of the token for a short period of time. The load each station offers the network also has an impact for a similar reason. If a node has a lot of data to send, it will maintain control of the token until it is finished (or until a set time period expires) which adds to the delay other nodes on the network will experience.

The last of the three factors which affect network performance, the network protocol, is probably the factor over which the network designer has the most control. The network protocol can be further divided into three layers. The first of these is the physical layer. This layer is generally of little concern to the network designer as it does not usually detract from the network performance. Physical means of data transmission and reception have little trouble in keeping up with the timing of the remaining layers.[12] The second layer, the data-link layer, adds some delay to message transfers. This delay is mostly attributed to the addition of header and trailer bits to

---

[12] Stallings, p. 243.

each frame and some control overhead such as message acknowledgements. The simulation models which will be presented in Chapter 5 are very concerned with these delays.

The media access control, the third subdivision of the network protocol, also has a very strong effect on the network performance. The two methods most widely used were covered in Chapter Two. Recall from that discussion that the topic of which of these methods (contention or token passing) performs the best has been debated extensively. Also recall that it is a debate which has been left unresolved as it is too dependent on the situation to proclaim a best overall solution.

3.4 Network Simulation

Before the topic of network simulation can be introduced the reader should have an understanding of what a simulation model is and how it is used. The first sub-section is, therefore, targeted toward the first two of these questions. The second sub-section will attempt to give the reader an idea of what would motivate someone to simulate a network design. In addition to these, there will also be a section which covers a methodology used to develop simulation models.

3.4.1 Simulation Modeling

A simulation model is basically a mathematical description of some system that is designed to demonstrate how that system will behave under various conditions. In discrete event simulation, the type used in this research, entities and their attributes travel through different states of a system. The dependent variables of these entities change discretely at specified points in simulated time known as event times.[13]

When simulation first came into existence, models were usually developed with high level computer languages. With the growth of this practice came the advent of simulation languages. These languages provide a simplified means of defining the queues, resources, and activities which are commonly used in simulation models. The languages are most often comprised of a collection of subroutines written in a high level language (such as FORTRAN). The user is usually allowed to access these routines and to create his own code (in the high level language) to augment the capabilities of the modeling facility. One such language, Simulation Language for Alternative Modeling or SLAM II, was used to develop the models which will be used to compare the two network architectures in Chapter Six. A brief description of this language will appear in that chapter.

---

[13] Pritsker, Introduction to Simulation and SLAM II, 3rd ed. (West Lafayette, Indiana, System Publishing Corp., 1986), p. 52.

3.4.2 Reasons for Network Simulation

The most evident way to evaluate the performance of a network would be to measure it. Unfortunately, this is not always possible. In most cases this is due to the fact that the system is not yet installed. It can also be for the reason that measurement of an existing system is difficult or impossible. The next most obvious method of determining how well a network performs is to develop a model of the system. In many cases this method is also a more practical means of determining system performance. This is especially true during the design of a system where several proposed solutions must be tested. The drawback of the modeling approach is that the abstraction of the system into model form can result in the inaccurate representation of the system.

There are two ways in which a model can be developed; analytically and through simulation.[14] Analytical modeling involves the abstraction of the system into a form with which applied mathematics can be used to develop an equation to represent the system. These mathematical models will usually represent only a single system resource. Other components of the system and their interaction are represented in a very simplistic manner or are ignored

---

[14] Sauer and MacNair, Simulation of Computer Communication Systems (Englewood Cliffs, N.J., Prentice-Hall Inc., 1983), p. 4.

all-together. This fact, coupled with the high degree of abstraction required, often result in models with questionable accuracy.

The alternative to this approach is to develop a simulation model. The major advantage of using a simulation model is the capability of broad and inclusive coverage of system resources. This can however also become a disadvantage as a high level of detail may cause the model to become impractically large. Other disadvantages include the cost associated with model development and execution, the required statistical analysis of resulting data, and the difficulty of validating the accuracy of the model. These disadvantages are, however, outweighed by the amount of information obtained about a network solution through the development and execution of a simulation model.

Pritsker [14] sights four processes which a simulation model can be used for. The first is: "as an explanatory device to define a system or problem."[15] Dr. Plebani, who teaches simulation in the Industrial Engineering Department at Lehigh, is fond of the saying: "he who simulates knows." By this he means that one must have a very thorough understanding of a system to simulate its operation. Formulating a model is, therefore, a very good way to gain a detailed understanding of a system. The second use for a simulation model is: "as an analysis vehicle to determine

---

[15] Pritsker, p. 1.

critical elements, components and issues in a system."[16] Models can be used to find problem areas such as bottlenecks that are otherwise difficult to identify. The third use, "as a design assessor to synthesize and evaluate proposed solutions,"[17] is one of the largest uses in network simulation. In these cases the model is used to determine how a system that is not yet installed will perform. The final use, "as predictors to forecast and aid in planning future developments"[18], can also be related to network simulations. Network designers will often maintain the original model to aid in the fine tuning of the network once it is installed.

3.4.3 A Network Simulation Methodology

The methodology which will be used to develop the simulation models in Chapter Six has been adapted from the methodology introduced in a paper by Chlamtac and Jain [5]. Their methodology will be paraphrased in the following paragraphs.

The methodology described by Chlamtac and Jain [5] emphasizes the modularity and efficiency of the simulation program while obeying the rules of correctness and ease of modification which are important to all large programs. It

---

[16] ibid.

[17] ibid.

[18] ibid.

does this by decomposing the network architecture vertically along its protocol layers and horizontally within those layers. This will separate the software specifications such as protocols and operating systems from the hardware on which it runs. The components are then linked together via a generic framework for execution. This separation of the functional components of the system has two advantages. First, it eases the model development and validation by breaking the large system into several smaller, more manageable, independent systems. Second, it adds flexibility to the model because the functions are treated independently.

The basic principle behind the methodology is a top down design coupled with bottom up modeling. With this approach the designer first determines the network requirements at the highest end-user level and then determines other requirements while working down toward the lowest protocol layer. Once this process is complete, the designer works his way back up the layers finally translating the protocol requirements into input/output requirements for each protocol layer. In this way a detailed model of each protocol layer is developed separately.

The network model itself is comprised of procedures and structures that specify and interconnect the various network elements such as nodes, topology, protocols, and resources

and manage the dialogue between them. The model framework consists of a group of four managers which specify and control these elements. The protocol manager is responsible for specifying the protocols for execution and the protocol interfaces. It also handles all communications processes, generates resource requests, and presents a common node manager interface. The resource manager controls the resource utilization, interprets and executes resource requests, collects resource related statistics, and presents a common node manager interface. The node manager specifies the node resources and protocols and provides an interface between the resource manager and the protocol manager. The final manager is the network configurator. This specifies the number and type of nodes on the network and describes how they are related to one-another. This framework again gives the methodology modularity. Each of the managers can be created independently as long as the manager interfaces are developed correctly.

3.5 Chapter Summary

This chapter has touched on three major subject areas. The first two, network performance measures and characteristics, were covered to give the reader a general understanding of what affects the performance of a network solution and how that performance is measured. The remainder of this paper will be mainly concerned with the

relative performance of two network architectures which differ, primarily, in the number of OSI-type layers that the data passes through during communications. Factors such as media characteristics and network loading will be held equal so they will not effect the outcome of the studies. Simulation models will be used to compare the two architectures so the reader should have a good understanding of the methodology that will be used before proceeding to Chapter Six. If the reader requires more information on simulation in general he should refer to Pritsker [14]. If more information is required on simulation as it pertains to local network modeling he should refer to Sauer and MacNair [10] or the article by Chlamtac and Jain [5]. The reader is reminded that this is not the exact methodology that will be used to develop the model. The methodology has been adapted since this research is not concerned with many of the design aspects it is intended to cover. Instead it is interested only in the relative performance of architectures which use different protocol layering. It should also be noted that the reason for the use of simulation models is that means of measuring the performance of the architectures is required. Although the other reasons behind network model development which were cited in this chapter are valid for instances where a network is being designed, they are not applicable to the research presented in this paper.

4 Network Architectures

4.1 Introduction

This chapter will introduce the two network architectures that will be simulated and compared in Chapter Five. The architectures are the Manufacturing Automation Protocol (MAP) developed originally by General Motors Corporation and the PROWAY-LAN specification developed by the Instrument Society of America. The PROWAY-LAN specification is one of several which was considered for inclusion in the MAP Version 3.0 specification as an Enhanced Performance Architecture (EPA) for use in time-critical applications. Both architectures adhere closely to the ISO-OSI seven layer model discussed in Chapter Two although the PROWAY-LAN specification represents a collapsed form of this model using only the first two layers.

The information in this chapter is a brief summarization of the information contained in the MAP Version 2.1 specification [11] and a draft of the PROWAY-LAN specification [9]. These documents contain very detailed descriptions of the specifications and how they should be implemented. This chapter will, however, only summarize the information about the architecture of the protocols. It will describe how the specifications define each of the seven layers of the ISO-OSI model. All subsequent topics,

such as network management requirements, will be ignored. If the reader wishes to learn about these areas as well, he should refer to the references listed above.

It should be noted here that, although version 3.0 of the MAP specification has been released, this paper uses version 2.1 of the specification. There are two reasons for this. First, it is assumed that most MAP based networks in operation today are based on the 2.1 specification. Second, since the version 2.1 specification has been available to the public for quite some time, more research related to it has been done and consequently more information is available on it than for the version 3.0 specification.

4.2 Manufacturing Automation Protocol

The main barrier to factory-wide integration has been communications. The reason for this is that factory equipment is usually purchased from a variety of vendors rather then a single vendor. This would not be a problem except that these vendors often choose different standards of communications and equipment interfacing. In order to get this equipment to communicate, companies were forced to develop application specific interfaces for each piece of equipment. This is a very difficult and costly process.

General Motors Corporation formed the Manufacturing Automation Protocol Task Force in order to develop a standard of communications that would help to combat the

problem of interfacing intelligent factory equipment. They had begun to notice a trend in which up to fifty percent of equipment purchase costs were allocated to interfacing the new equipment with existing equipment.[19] The MAP Task Force was to stop this trend by developing a set of factory networking guidelines that would allow equipment such as robots, CNC machine tools, programmable logic controllers, and computers to communicate with each other. GM hoped to gain sufficient support of the MAP standard from equipment vendors that computer-based systems would eventually posses true plug-in compatibility. MAP did indeed gain acceptance from these vendors as well as a large number of other manufacturers. GM has since relinquished control of the MAP specification development to the MAP Users Group so that other companies could also play a part in deciding how the standards can be developed to best suit the needs of all manufacturers.

The MAP Task Force has published several documents on the MAP standard. The intention of the documents is to give their equipment suppliers enough information so that they can develop products that meet the specification. The first MAP specification document was published in October of 1982. This document provided descriptions of general networking considerations and information on implementation. One and a

---

[19] MAP Task Force, Manufacturing Automation Protocol Specification, Version 2.1, ch. 1 p. 1.

half years later, in April of 1984, a second document was published. This issue contained more complete and detailed information and came to be known as MAP Version 1.0. Later came Version 2.0 with new standards and finally Version 2.1 in March of 1985. Version 2.1 was the first document in which the specification was complete enough for compatible product design and actual MAP implementation. Many vendors now provide MAP compatible products although not all have plug-in compatibility. This is mainly due to the fact that there is presently room for various interpretations of the specifications. A Version 3.0 document has been published which should eliminate this problem as well as introduce some new standards such as an Enhanced Performance Architecture.

The primary intention of the MAP Task Force has been to establish a uniform set of standards that would allow for communications between diverse intelligent factory devices in a cost-effective and consistent manner. To do this the task force established a set of three objectives to be met:

1. Define a MAP message standard which supports application-to-application communication.

2. Identify application functions to be supported by the message format standard.

3. Recommend protocol(s) that meet our functional requirements.[20]

---

[20] MAP Task Force, p. iii.

The task force also decided that it should chose from existing standards to achieve this goal rather then developing a set of new standards.

To meet these objectives the MAP committee choose to adhere closely to the International Standards Organization Open System Interconnection model which was introduced in Chapter Two. This model was adopted because of its wide acceptance and also its applicability. In addition, the protocol also uses standards from other groups such as the Institute of Electrical and Electronic Engineers, the American National Standards, Institute, the Instrument Society of America, the Electronic Industries Association, and the National Bureau of Standards.[21]

The following sections will give a brief summary of the standards the MAP protocol uses for each of the seven layers of the ISO-OSI model (see Fig. 4-1). The descriptions often include the mention of a standard from one of the above groups which the MAP Task Force has specified for the layer. A knowledge of what these standards are or what they entail is not essential to the understanding of material presented later in the thesis. They are mentioned only to give the reader who has knowledge of these standards an indication of the direction of MAP.

---

[21] Allan, "Factory Communications: MAP Promises to Pull the Pieces Together," Electronic Design, 15 May 1986 p. 104.

NO.	NAME	MAP PROTOCOL SPECIFICATION
7	APPLICATION	ISO CASE, MAP SUBSET ISO FILE TRANSFER MAP MMFS MAP DIR. SERVICES MAP NETWORK MANAGEMENT
6	PRESENTATION	NULL
5	SESSION	ISO SESSION KERNAL
4	TRANSPORT	ISO CLASS 4 TRANSPORT
3	NETWORK	ISO CLNS INACTIVE AND SUBSET
2	DATA-LINK	IEEE 802.2 TYPE I IEEE 802.4 TOKEN BUS
1	PHYSICAL	IEEE 802.4 10 MB BROADBAND

Figure 4-1. MAP seven layer model.

4.2.1 The Physical Layer

The purpose of the physical layer is to provide a physical connection between the nodes on the network and to provide a means by which the connection can be activated and deactivated.

The MAP standard calls for the installation of broadband coaxial cable according to CATV industry standards. A mid-split configuration and a data rate of 10 Mbps are required. The mechanical and electrical interfaces should be performed according to the IEEE 802.4 token passing standard. The broadband backbone structure will tie into both MAP and non-MAP networks through bridges, routers, gateways.

There are two reasons for the selection of the token passing scheme. First, it is deterministic. This is important for discrete and continuous process control where uncertainties in message delivery times cannot be tolerated. The second reason is that token passing can support a message priority scheme. Select nodes on the network can, therefore, be given the opportunity to send messages more often than other nodes and/or at specified times.

Many of the reasons for the selection of broadband cable stem from the wide bandwidth of such systems. This bandwidth not only permits the transmission of voice and video signals in addition to data, but also permits multiple networks to exist simultaneously on the same media. This

allows for a smooth transition to MAP and also minimizes the amount of wiring modifications required. A final reason for the selection of broadband is that it is a part of the IEEE 802.4 communications standard.

4.2.2 The Data Link Layer

The data link layer provides for and manages the transmission of individual packets of data. In addition, it is also responsible for the detection and correction of errors created by the physical layer.

The IEEE 802 standard project, which MAP adheres to in this case, divides this layer into two independent sub-layers: the media access control (MAC) sub-layer and the logical link control (LLC) sub-layer. The MAC sub-layer is responsible for the management of access to the physical media by the data packets. The LLC sub-layer is responsible for addressing, error checking, and other functions which insure the accurate transmission of data between nodes.

MAP specifies the IEEE 802.4 token-passing bus configuration which uses a bus topology which is logically represented as a ring. This standards handles all of the functions encompassed in the MAC sub-layer. The LLC sub-layer is provided for by the IEEE 802.2 Logical Link Control specification. This protocol uses two services to define a multi-point peer-to-peer relationship. The first of these services, connectionless-oriented, allows for the

exchange of data between two logical link control entities without the establishment of a data link connection. The second of the services, connection-oriented, establishes a data link connection and provides for message sequencing acknowledgements, flow control, and error recovery. MAP specifies that only the first of these two services is required. This is due to the fact that higher layers in the model will be used to handle the functions performed by the second service.

The final specification for this layer is the number of address bits used. The MAP protocol calls for a 48 bit address field as opposed to a 16 bit field. This is used to allow for unique addressing both within and between local area networks. The 48 bit field can be segmented into network, station, and entity IDs to facilitate this.

Token passing was chosen for the MAP specification for four reasons. The first of these is that it is the only data link protocol which is supported on broadband by the IEEE 802 standard. Second, many vendors are already using token-bus based schemes. This makes the transition to full MAP compatibility easier for these vendors. The support of the message priority scheme, mentioned earlier, is the third reason for the selection of the token-bus scheme. The fourth and final reason for this selection, which was also illuded to earlier, is the fact that messages can be delivered in a specified time limit. This is a result of

both the deterministic nature of token passing and the ability to use a priority scheme.

Four reasons can also be cited for the selection of the IEEE 802.2 LLC standard. First, it will support data transfer at very high rates. Second, it can be used on a variety of media. Third, it provides the connectionless service required. Finally, it is hoped that once the standard becomes widely accepted, VLSI chips will emerge and substantially reduce the cost of devices utilizing it.

4.2.3 The Network Layer

The purpose of the network layer is to provide end-to-end message routing between nodes. This routing is performed for both nodes on the same subnetwork and nodes on different subnetworks.

The MAP specification breaks the network layer down into four sub-layers: the Inter-Network sub-layer (3.4), the Harmonization sub-layer (3.3), the Intra-Network sub-layer (3.2), and the Access sub-layer (3.1). Sub-layers 3.3 and 3.1 are interface layers. They provide the necessary logical interface between their adjacent sub-layers. They do not provide routing functions. Those functions are performed by sub-layers 3.2 and 3.4. Sub-layer 3.3 may translate Inter-Network addresses into Intra-Networks addresses.

The Inter-Network sub-layer contains the information necessary for the end node to end node internetwork routing of data packets. It allows messages to traverse multiple local area networks without requiring routing information for each of the particular networks. The Subnetwork Independent Convergence Protocol (SNICP) acts as a vehicle for the provision of this service. The MAP committee has chosen the Draft International Standard 8473 which is known as the Data Communications Protocol For Providing the Connectionless-Mode Network Service (P_CLNS) as the actual standard to be used. P_CLNS provides the data and control information exchange through a connectionless transmission, Internet Protocol Data Unit (IPDU) encoding of the information, interpretation procedures for data and control information, and a formal description which must be met to be in compliance with the standard.

The Harmonization sub-layer, formerly referred to as the Network Interface Sub-layer, supplements the lower layers and sub-layers so that uniform services can be provided to the Internet sub-layer. In most cases this is done by mapping inter-network addresses into intra-network addresses. This allows the use of a local area network as a link in a larger network. The Harmonization sub-layer adapts the global routing requirements to the available local routing sub-layer services. When the internetwork protocol is connectionless and the intra-network is

connection oriented, the harmonization layer will be a full protocol between entities which establish intra-network connections. If the next sub-layer, the intra-network sub-layer, is null, then the harmonization sub-layer is also null.

The Intra-Network sub-layer contains the local routing protocols. It is concerned with all routing and switching of messages within the immediate local network where the immediate network is defined as the set of nodes which communicate with a common routing protocol. This common protocol may involve X.25 packet switching or a vendor proprietary method. Ideally this sub-layer, and therefore the Harmonization sub-layer, will be null and the Inter-Network sub-layer will perform these functions.

The Access sub-layer provides the necessary interface to the Data Link layer. This sub-layer will only be a full protocol when the services sub-layer protocols differ from the data link layer protocols. If, for example, the Inter-Network sub-layer is connectionless and the data link layer protocol is connection oriented, then the Access sub-layer must provide a means of data link connections between entities.

These sub-layers work together to perform several functions. First, they must convert global address information into local routing information. Second, they must maintain tables and/or algorithms for message routing.

Application layer address and routing data bases are often used to support these first two functions. The third function they perform is to provide a means for establishing and terminating network connections when appropriate. Finally, they must switch all incoming messages to their correct outgoing paths. Not all sub-layers are required in all situations. There are some cases, such as communications within one-vendor networks, in which the functions of some of the sub-layers are already provided.

The last topic covered in the Network Layer specification is the address structure and routing. The ISO-8348 DAD2 specification provides the address structure that the MAP Committee has specified should be utilized by the Data Communications Protocol for Providing Connectionless Mode Network Service (CLNS). This standard will be used to further define the destination and source address parameters that are defined by the CLNS protocol. These are Network Service Address Protocol (NSAP) addresses as defined in the Internal Organization of the Network Layer (IONL). The syntax and semantics which describe NSAP are also, therefore, described in ISO-8348 DAD2. Although it is not required that these address encoding schemes include routing information, GM has decided that the underlying address structure to be chosen will imply and/or describe hierarchical routing information which will be utilized by

intermediate systems. This will be the specification until a widely accepted standard emerges.

4.2.4 The Transport Layer

The purpose of the transport layer is to provide a transfer of data between Session layers in a way that is transparent to the end user. This service is to be network-independent. The National Bureau of Standards states that "the transport protocol exists to provide one fundamental service, the reliable, transparent transfer of data between transport users."[22]

For this layer, the MAP standard specifies the National Bureau of Standards, Class 4 Transport Protocol (NBS-Class 4). This class of transport provides three basic services. It provides flow control, the ability to multiplex user transmissions to the network, and error detection and recovery. Class 4 is the only NBS Transport Protocol class which supports a datagram oriented network and a level of control that seems to support a wide variety of network sub-layers. This protocol also boasts the largest amount of support among U.S. manufacturers.

There are three basic differences between the NBS and ISO standards for Class 4. The first of these is that the NBS standard supports the concept of datagrams while the ISO standard does not. ISO also does not support the graceful

---

[22] MAP Task Force, ch. 3 p. 48.

closure of a connection while the NBS standard does. Finally 'The Status of Connections' function is not supported by ISO.

The transport protocol provides two services to the transport user: transport connection management and data transfer. Connection management services allow the user to create and maintain a data path to another user. It is comprised of four functions: establishment, closure, disconnection (abortion), and status checking of connections. The data transfer service provides a means by which data can be exchanged between two users. The exchange can occur in three ways. The first is a normal data transfer, the second is an expedited or urgent data transfer and the last is a unit data transfer which does not require an actual transport connection.

4.2.5 The Session Layer

The purpose of the session layer is to enhance the transport service by providing a means of managing and structuring the data transfer provided on a transport connection. Interactions between users can be structured as either simplex, half-duplex, or full-duplex by the session connection.

As a standard the MAP committee has chosen to follow the internationally accepted ISO Session Standard. MAP specifies that only full-duplex communications should be

performed. The minimum subset of the ISO Session for MAP connectivity is specified by the Kernal and Duplex functional units. The use of available options or tokens is not required.

4.2.6 The Presentation Layer

The purpose of the presentation layer is to provide a standard data format for use by the application layer. At this time the MAP committee specifies a null implementation for this layer. No Presentation Protocol Control Information is provided by the MAP Version 2.1 specification.

4.2.7 The Application Layer

The highest layer of the ISO-OSI model is the application model. The only purpose for this layer is to provide a means through which application programs can access the local area network.

In MAP, the Application Layer consists of three routine specifications which provide basic services to the end user. The first of these is the Common Application Service Elements (CASE) specification developed by the ISO. As defined by the ISO, CASE provides several services. Of these, the MAP architecture specifies only the association control. This routine allows applications in different nodes to establish logical connection known as associations.

The commands provided allow user programs to establish, terminate, and abort associations without having knowledge of how the lower layer protocols actually perform these functions.

The second service specified by MAP is File Transfer Access and Management (FTAM) which was also developed by the ISO. FTAM is used to create and delete files, read the characteristics of a file, and to transfer files. It establishes its own associations to perform these functions.

The final Application Layer Interface specified by MAP is the Manufacturing Message Format Standard (MMFS) developed by GM. MMFS provides a common command language format which various devices on the network use to communicate. This specification may be replaced with the Electrical Institute of America Manufacturing Message Service (MMS) which is similar to MMFS but provides more services.

4.3 The Need for Enhanced Performance

Although the MAP specification provides for the integration needs of intelligent devices in the discrete parts manufacturing and assembly industries, several shortcomings have been identified. One of the largest of these shortcomings is the inability to provide the response times needed for real-time control. At present, there is no effective means for real-time message sending between

multi-vendor cell controllers, robots, and automated equipment. This has led to need for the development of an Enhanced Performance Architecture (EPA) within the MAP standard. This architecture must provide a specification for nodes that are fully compatible with MAP and can transfer messages at a rate which is consistent with real-time requirements.

Like many discrete parts manufacturers, the process and process control industries are beginning to feel competitive pressure from off-shore companies. The process industries are also beginning to be plagued with greater uncertainties in raw material availability and costs, manufacturing costs, and final product market value. This has led to a push for better integration and control of processes in much the same way as there has been a push for integration in manufacturing. It has also greatly reduced the feasibility of obtaining a cost-effective single vendor automation solution as is the case in discrete parts manufacturing. These factors have led to the growth of the process industries interest in MAP standard and to the subsequent formation of the MAP Process Industries Initiative.[23]

The MAP Process Industries Initiative (MPII) is a committee of the MAP/TOP users group which was formed in 1986. The committee was formed to perform three basic

---

[23] Crowder, Robert S. "Enhanced Performance and MAP: Part I." MAP/TOP Interface August 1986, p.3.

tasks. The first of these is to give potential MAP users in the process industries an awareness of what MAP is and how it works and to keep these potential users up-to-date on the evolution of the specification. The second goal of the group is to determine how applicable the MAP specification is to the needs of the process industries. Finally, and possibly most importantly, the group wishes to foster process industry involvement in the actual development of the MAP specification so that the standard will meet the needs of process applications.[24]

One of the first actions taken by the MPII Committee was to publish a paper [12] which proclaimed the needs and technical issues that are important to the process industries. One of the primary interests is in real-time network performance with message transactions occurring within a few milliseconds in some cases. A second concern is in the ability of process I/O device networks to be integrated with MAP networks. Third, MPII is concerned with the availability (redundant communications media and processors), reliability (a mean time of years between failures), and maintainability of the network. In addition to these, the report also mentions other process industry network needs such as the ability to withstand harsh environments, intrinsic safety and low power consumption,

---

[24] MAP Process Industry Initiative (MPII) Working Group MAP in the Process Industry pp. 3-4.

security, and conformance to the ISO communications standard.

The MPII Committee report also contains limited coverage of the type of topology that would be expected of the MAP EPA. In this topology, process I/O devices and the process controller would be connected to a sub-network segment. These devices would communicate with each other using an EPA which would represent two layers of the ISO-OSI model. These nodes, known as MINI-MAP nodes, would not be capable of communicating directly with the full seven-layer MAP nodes on the backbone network. A MAP/EPA node which is capable of communications using either the MINI-MAP protocols or the full-MAP protocols would be required to act as a protocol translator to facilitate such communications. Such a configuration will both allow for the tight timing considerations of the process control devices and reduce the amount of traffic on the backbone network.

4.4 PROWAY-LAN

One of the standard specifications which seems to fulfill the requirements of MAP/EPA and MPII is ISA-d72.1, PROWAY-LAN. In fact, this is one of the standards being considered for inclusion in the MAP Version 3.0 specification. The development of the PROWAY-LAN specification is sponsored by the Instrument Society of America (ISA). The ISA committee (ISA SP72) which

formulated the standard worked closely with the IEEE 802 committee to encourage compatibility between the two standards.

The PROWAY standard specifies protocols for interconnection of stations by way of a Local Area Network using the Token-Bus media access method. Unlike the MAP specification, it does not only draw on existing standards. Instead, it uses many of its own standard specifications which were developed with the IEEE 802.4 specification in mind and therefore closely resembles it although it is slightly more restrictive in some areas. Some of the networking elements which the specification covers include: the electrical and physical characteristics of the transmission medium, the signaling method used, transmission frame formats, and token bus media access. The standard applies to serial transmission over a shared electrical transmission line (coaxial cable). ISA committees are, however, presently working on possible fiber optic solutions.

The standard provides definitions of protocols, interfaces, and media for the first two layers of the ISO-OSI reference model. It is intended that compliance with this and the standards of higher OSI layers (which will probably be provided for by vendor proprietary software) will allow communications to take place between devices which comprise a distributed industrial or process control

system over a shared Process Data Highway. The standard is applicable to both continuous and discrete process control systems in a wide range of factory automation situations. It is not intended for the transmission of data which is not directly related to outputs which cause the transfer of materials or energy.

The PROWAY specification uses a three layer model to describe its primary functional areas (see Fig. 4-2). The layers are comprised of the PROWAY Link Control (PLC) layer, the Media Access Control (MAC) layer, and the Physical Signaling (PHY) layer. The PLC and MAC layers make up the ISO-OSI Data Link layer and the PHY layer represents the Physical layer. The following paragraphs contain a brief description of each of the three layers.

4.4.1 The Physical Sub-layer

There are several functions which are to be provided for by the physical layer specification. The first of these is a means to allow for the physical transmission of signals between stations on the network that conform to the standard and that are connected by a single channel bus using a coaxial cable media. The communication channel must also be capable of high bandwidth and low error-rate performance. In addition, the specification of this layer must provide high network availability and the ease of installation and maintenance in a wide range of environments.

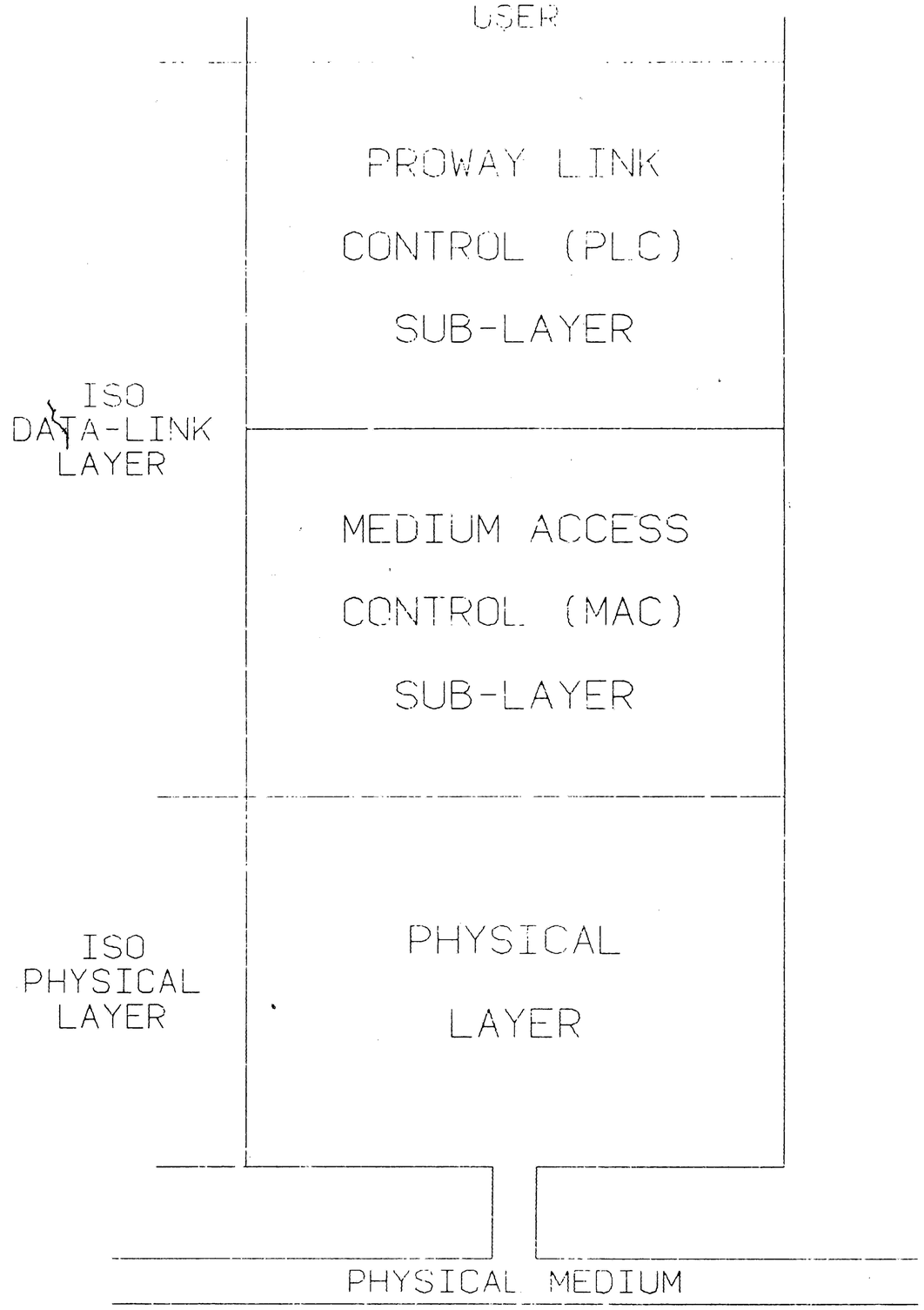

Figure 4-2. PROWAY-LAN three layer model.

The physical layer specifies that stations should be connected to a trunk of a single channel coaxial cable bus systems by drop cables and impedance matching taps. The trunk cable is specified as RG-6 type, 75 ohm, semi rigid coaxial cable. A more flexible coaxial cable such as RG-59 is specified for the drop cables. The topology suggested is that of a highly branched tree. All cabling and taps must be of a non-directional nature. These selections were made because they represent the most viable solution at the present level of technology. As advances are made, however, this specification may be replaced by another type of physical medium such as fiber optics.

The specification also calls for phase continuous frequency shift keying modulation. The standard data signaling rate that is to be achieved using this technique is 1 Mbps. In addition, the manchester encoding scheme is to be used to represent data, non-data, and pad-idle symbols. In this scheme separate data and clock signals are combined into a single self-synchronizing data stream which can be transmitted on a serial channel.

4.4.2 The Medium Access Control Sub-layer

The MAC sub-layer provides sequential access to the shared bus medium by passing control of the medium from station to station. A station on the network determines when it should have control of the medium by recognizing and

accepting control from its predecessor. Similarly, it recognizes when it should relinquish control of the network to its successor station.

The means by which this control is performed is very similar to the IEEE 802.4 Token Bus scheme which was discussed in Chapter Two. The PROWAY token bus is slightly more restrictive than the IEEE standard. However, they are enough alike that the token bus description in Chapter Two will suffice for the level of detail with which this writing is concerned. One of the ways the PROWAY specification differs from the 802.4 specification is that it divides the functions of medium access control into four groups which are provided for by separate "machines" within the layer (see Fig. 4-3). The "machines" consist of an Interface machine (IFM), an Access Control machine (ACM), a Receive machine (RxM), and a Transmit machine (TxM).

The IFM acts as an interface and buffer between the MAC layer and other layers in the protocol. It accepts and interprets incoming service primitives and outputs the appropriate responses. It also queues service requests when necessary. Finally, the IFM performs the address recognition function accepting only the appropriately addressed PLC frames.

The ACM is the most important and complex of the four machines. It is responsible for coordinating the control of the transmission media with the ACMs of all other stations

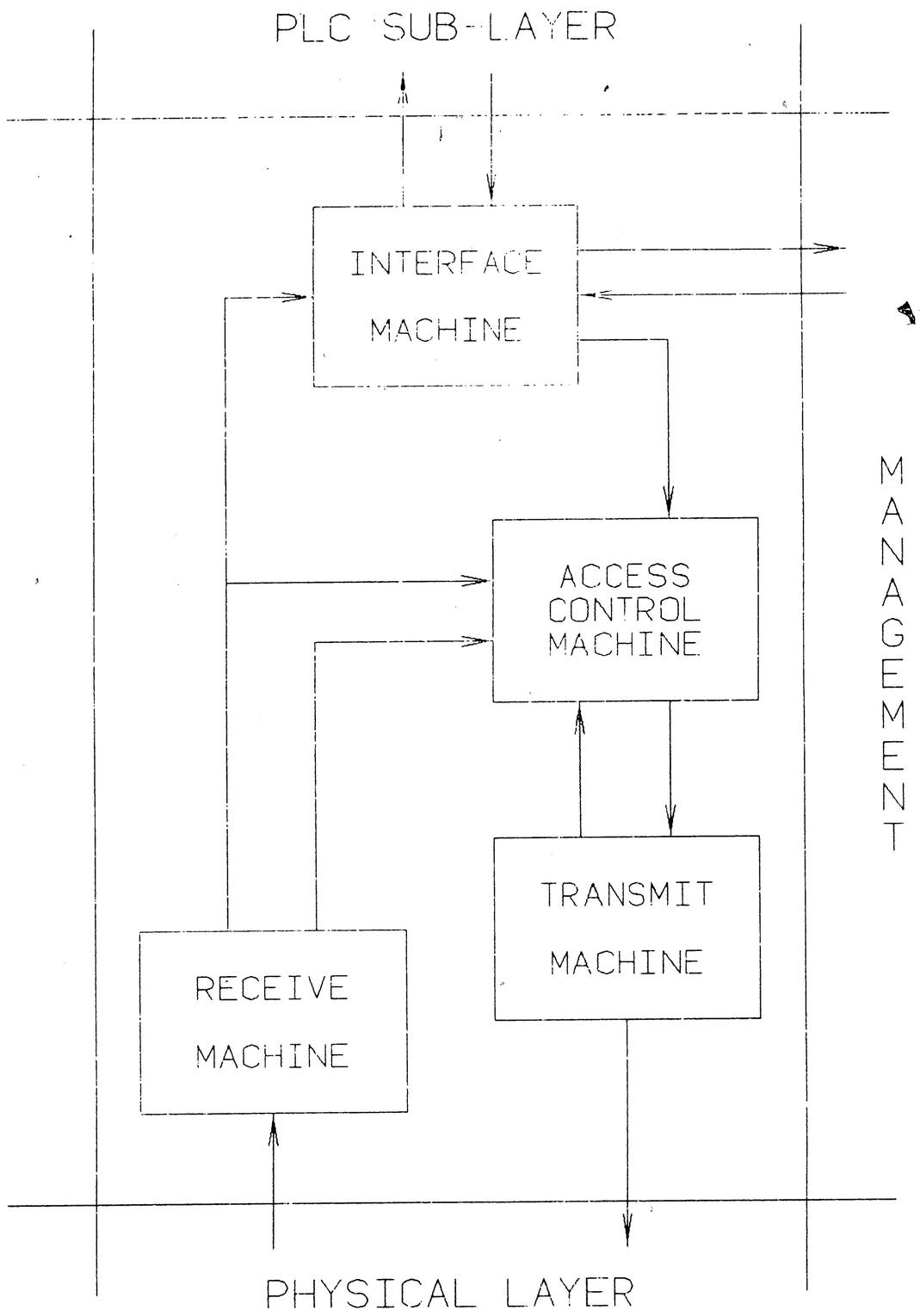

Figure 4-3. Media Access Control machines.

on the network segment. It is also responsible for managing and providing different levels of quality of service to the PLC layer. The ACM waits for acknowledgements of transmitted frames when required and retransmits those frames for which no acknowledgement is received. This machine is also responsible for the token ring maintenance functions (see Chapter Two). In addition, it is responsible for the detection of and recovery from failures in token passing.

The RxM is responsible for accepting symbol inputs from the physical layer, assembling them into frames, and passing those frames onto the IFM and ACM machines. This is done through the recognition of the frame start and frame end delimiters, the checking of the frame check sequence, and the validation of the frame structure. The final function of this machine is to identify and indicate the reception of noise bursts and the bus quiet condition.

The fourth and final machine, TxM, performs functions which are basically the inverse of those performed by the RxM. It accepts a data frame from the ACM and breaks it down into symbols which it transmits to the physical layer in a specified format. It also adds the start frame delimiter, the end frame delimiter, and the frame check sequence to the symbols.

4.4.3 The PROWAY Link Control Sub-layer

The PLC sub-layer provides for the data transfer and control services between peer PLC entities. This is done by providing a means by which they can exchange link service data units (l_sdu) over the shared medium. The data transfer can take place in an open-loop, closed-loop, point-to-point or multi-point manner. These services are known as: send data with acknowledge (SDA), also known as acknowledged connectionless service; send data with no acknowledge (SDN), also known as unacknowledged connectionless service; request data with reply (RDR), also known as connectionless reply service; and remote station recovery (RSR).

The SDA service allows the local user of a station to send data to the user of a single remote station. The local user then receives confirmation as to whether the remote user received the data. The SDN service allows the local user of a station to send data to a single remote station, a group of remote stations, or to all remote stations. In this case the sending station does not receive confirmation that the data was received by the remote stations. It does, however, receive confirmation that the transmission was complete. RDR allows a local user to request data which was previously submitted by the user of a single remote station. The requesting station will either receive the requested data or an indication that the data was not available. The

RSR service allows a local user to request activation of a hardware output in a single remote station. The requesting user will then receive confirmation of whether or not the recovery request was received.

The functions provided in these services are partitioned into two groups. The functions in each group are then performed by an independent state machine. The first of the two machines is the local state machine. This machine handles all requests from and confirmations to a local user. All local requests result in the transmission of a request frame. The functions provided for by the local PLC machine include: accepting local user requests, generating request frames, receiving response frames, and passing confirmations to the local user. The second machine, the remote state machine, passes indications to the remote user, manages the shared data areas, and returns requested data to local users. The function provided for by this machine include: receiving request frames, passing indications to the remote user, generating response frames, accepting remote user requests to update a shared data area, and passing update confirmations to the remote user.

4.5 Chapter Summary

This chapter has presented a brief summary of the MAP and PROWAY-LAN architecture specifications as they appear in the MAP Version 2.1 specification [11] and the PROWAY-LAN

specification draft [12]. The discussion is intended to give the reader a general idea of how each of these architectures approaches the ISO-OSI seven layer model. If the reader has experienced difficulty in understanding the material presented in this chapter or he wishes to gain a more detailed understanding of the specifications, he should refer to the appropriate specification document.

The reader should now have an understanding of what is involved in communications between nodes which use these architectures. The simulation models which will be presented in the next chapter will be used to determine the amount of time required to pass information between nodes through these layers and the effect this has on the overall response time of the system.

5 Simulation Models

5.1 Introduction

One of the major objectives of this paper is to compare the performance of the MAP and PROWAY architectures discussed in the previous chapter. To do this, the performance of networks with these architectures must be observed while operating under identical loading conditions.

As discussed in Chapter 3, simulation is one of the best available methods to study the performance of a Local Area Network. There is no other suitable means of monitoring the behavior of two networks which are implemented with all factors held the same except those which differ as a direct result of the architecture definitions. Even the development and operation of such networks would not provide the flexibility in experimental design and the ease of obtaining information about the performance characteristics as the simulation of the networks does.

This chapter will present the development of the two simulation models which will be used to demonstrate the relative performance of the network architectures introduced in Chapter 4 and the results obtained from conducting experiments with those models. Before this information is disclosed, the reader will be given an introduction to the simulation language used to create the models.

5.2 Simulation Language Used

The language used to develop the simulation models is SLAM II (Simulation Language for Alternative Modeling II). Like many simulation languages available, SLAM is basically a collection of FORTRAN subroutines which provide the user with various functions commonly required in simulation model development. A partial list of these functions includes:

(1) the maintenance of simulated time,

(2) the scheduling of events which occur in this time frame,

(3) the alteration of attributes associated with entities in the system,

(4) the generation of random numbers,

(5) the generation of numbers according to various statistical distributions, and

(6) the collection of statistical information pertaining to occurrences during the execution of the simulation.

The primary means through which these functions are accessed is in the development of a system network. Here the word "network" is used in a different sense then it has been previously in this paper. In this case a network is used as a logical representation of the flow of entities through a system. Nodes are used to represent queues,

servers, and decision points in the model.[25] Branches from these nodes are used to define the path which the entity takes while traveling through the system and the times required for the entity to travel along the path.

SLAM provides symbols for the branches and the various nodes which represent the basic functions offered. The simulation modeler uses these symbols to create a graphical representation of the model. A simple translation step in which the modeler converts the symbolic representation of the network into network commands, which the SLAM parser interprets, is then performed. Other SLAM commands are added to this code to provide for functions such as variable initialization, the identification of statistics to be collected, and the control of the execution (i.e. run length) of the model.

The other means through which the basic functions of SLAM can be accessed is by direct calls to the subroutines which perform the functions through FORTRAN subroutines. SLAM provides the ability for the user to create his own FORTRAN subroutines to augment the network model he has created. This greatly enhances the capabilities of SLAM by allowing users to model operations which are difficult or impossible to replicate using the network statements provided.

---

[25] Pritsker, p. 63

SLAM also provides the capability to create models which represent discrete systems, continuous systems, or a system which has both discrete and continuous aspects. Discrete modeling refers to situations in which events occur at discrete points in time. This is the type of modeling which has been used to represent the network architectures. Continuous modeling refers to situations in which events occur in a manner which is continuous over time. Such systems are modeled using a combination of differential and/or difference equations. The models developed for this paper have no continuous aspects, therefore this type of system modeling will not be discussed in any further detail.

Discrete event modeling in SLAM operates in the same manner as was discussed in the general explanation of discrete event simulation presented in Section 3.4.1. Actions on entities in the system occur at specified points in time. SLAM performs the duties necessary to maintain the simulated time clock. The modeler must define the points in time at which events occur during model execution and the logic associated with each event. The times at which events occur are defined by scheduling the event. An analogy for this would be when someone writes an appointment for a specific time on his calender. In this case the person would use a wall clock or his watch to determine the time and would go to his appointment when the time written on the calender is reached. Similarly, SLAM has an event calender

and maintains a simulated time clock. When a scheduled event is reached in simulated time, SLAM will call a routine which executes the logic associated with that event.

5.3 Model Development

To demonstrate the performance differences between MAP and an Enhanced Performance Architecture (EPA), models have been developed to simulate the operation of each of these architectures. This section will attempt to give the reader an understanding of the logic involved in the development of the two simulation models. Before this information is presented, the assumptions made for the development of the models will be discussed. The next section will first cover aspects of logic which is common to both models. Aspects of model development which are unique to the individual models are then covered in separate sub-sections.

5.3.1 Assumptions

To develop the network models, it was necessary to make a number of assumptions. One of the major assumptions is that the network operates in a steady state and with no errors. This means all functions related to the maintenance of the token bus are unnecessary. No new nodes will request to join the logical ring and no nodes on the ring will be dropped. Also the token will not be lost and all nodes addressed will acknowledge when data is sent to them.

Instead of modeling the occurrence of such events, a token overhead delay time is associate with each passing of the token. This allows for the consideration of the delay caused by the maintenance functions while simplifying the model.

Another assumption of the model is that all messages have equal priority. Whereas there are usually four transport classes (high priority, class 4, class 2, and class 0) in the transport protocols of both architectures representing different levels of message priority, for these models it is assumed that all messages on the network are of the same type so one message will not have priority over another message in any given message queue. It is also assumed that a node can only transmit one message at a time. This means that all frames associated with the sending of one message must be sent before the node can start processing a second message. This simplifies the model significantly.

An exception to this equal priority rule lies in message acknowledgements. In the original model, acknowledgements were not considered to have a higher priority over MMFS messages. This caused the network to lock up with all nodes waiting for an acknowledgment from one of the other nodes in the network and those acknowledgements queued behind MMFS messages which were also awaiting acknowledgements. Consequently, it was determined

that as each node received the token (gained control of the bus) it should send all acknowledgement frames in its queue before sending MMFS messages. Thus, it can be said that two levels of priority are represented (high priority and class 4). This assumption is required only for the MAP model. This is because the EPA architecture piggybacks its acknowledgements (they are transmitted in full duplex with the message).

Delay times associated with the physical transmission of data which are measured in microseconds were assumed to have no effect on the final results of the experiments run using the models and were therefore excluded. This is because most of the delay times used are measured in milliseconds. For this reason, delays associated with such things as the propagation of signals from one node to another are not included in the models.

Because the times presented by Crowder [6,7] were utilized in the models, the assumptions he made in developing these times must also be used. Among these is the assumption that a permanent association is not maintained between nodes on the network. Associations must therefore be established and dropped to facilitate the sending of MMFS and MMS frames. The establishment of associations is assumed to require a total of four control frames. To drop the association, two control frames are

required. More detail will be presented on this in later sections.

It must also be assumed that the models developed are representative of sub-networks which support 20 nodes on a 500 meter length of cable and that no communications are required with devices on the main backbone network. This is the configuration which Crowder [6,7] used as a framework for the development of delay times which are used in this model.

The last notable assumption made by Crowder [6,7] that was carried over for model development concerns the length of messages (in bits). For both models, all messages are assumed to be of the same size and this size is the maximum allowed by the architecture protocol.

In certain situations some of these assumptions may have a detrimental effect on the accuracy of the models. However, since these models are used only to make a relative comparison of the network architectures and not to judge the performance of either architecture alone, they should have no effect on the final determinations made from their use. This is only true because all assumption are applied to both models.

5.3.2 Logic

In order to take full advantage of the flexibility provided by SLAM and to make the process of model

development easier, all logic for the models was implemented using FORTRAN subroutines which were subsequently linked to the SLAM program code. As mentioned above, SLAM allows the use of its routines by directly calling the subroutines from user written FORTRAN code. This method was used in the development of both models to provide access to the facilities of SLAM required for the model (such as the scheduling of events). The network facilities provided by SLAM are used only for the initialization of network variables and the identification of statistics to be collected.

The models consist of five logical areas: message arrivals (at each node), network control, token passing, frame sending, and message removal. In the model, these areas are represented by individual subroutines (see Figs. 5-1 to 5-5). The general logic in each of these subroutines will be discussed in the following paragraphs. The actual program code can be found with the copy of this thesis that is maintained by the Lehigh University Industrial Engineering Department.

The message arrival routine runs in parallel to the main control routine of the model. It is used to determine the load on the Local Area Network (model). In this routine, when a message arrives at a node, it is immediately placed in the message queue of the node. This is done by placing an entity and its attributes in a file which

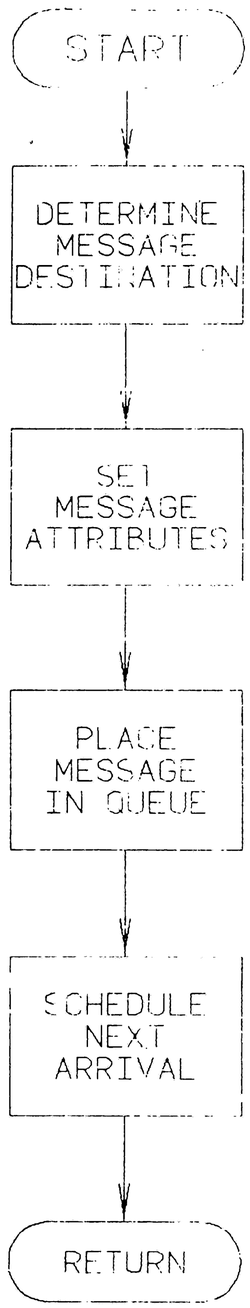

Figure 5-1. Message arrival subroutine.

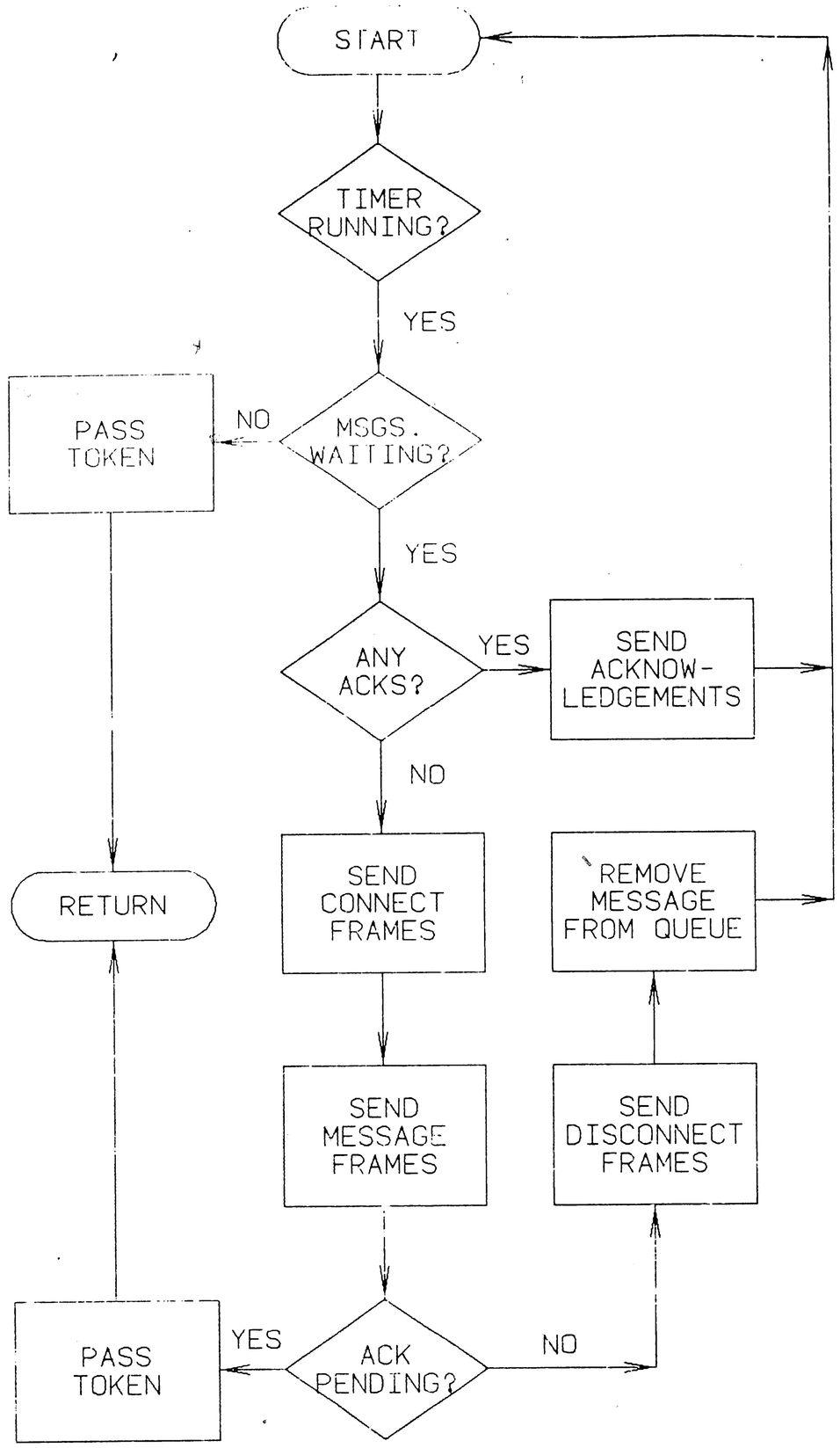

Figure 5-2. Network control subroutine.

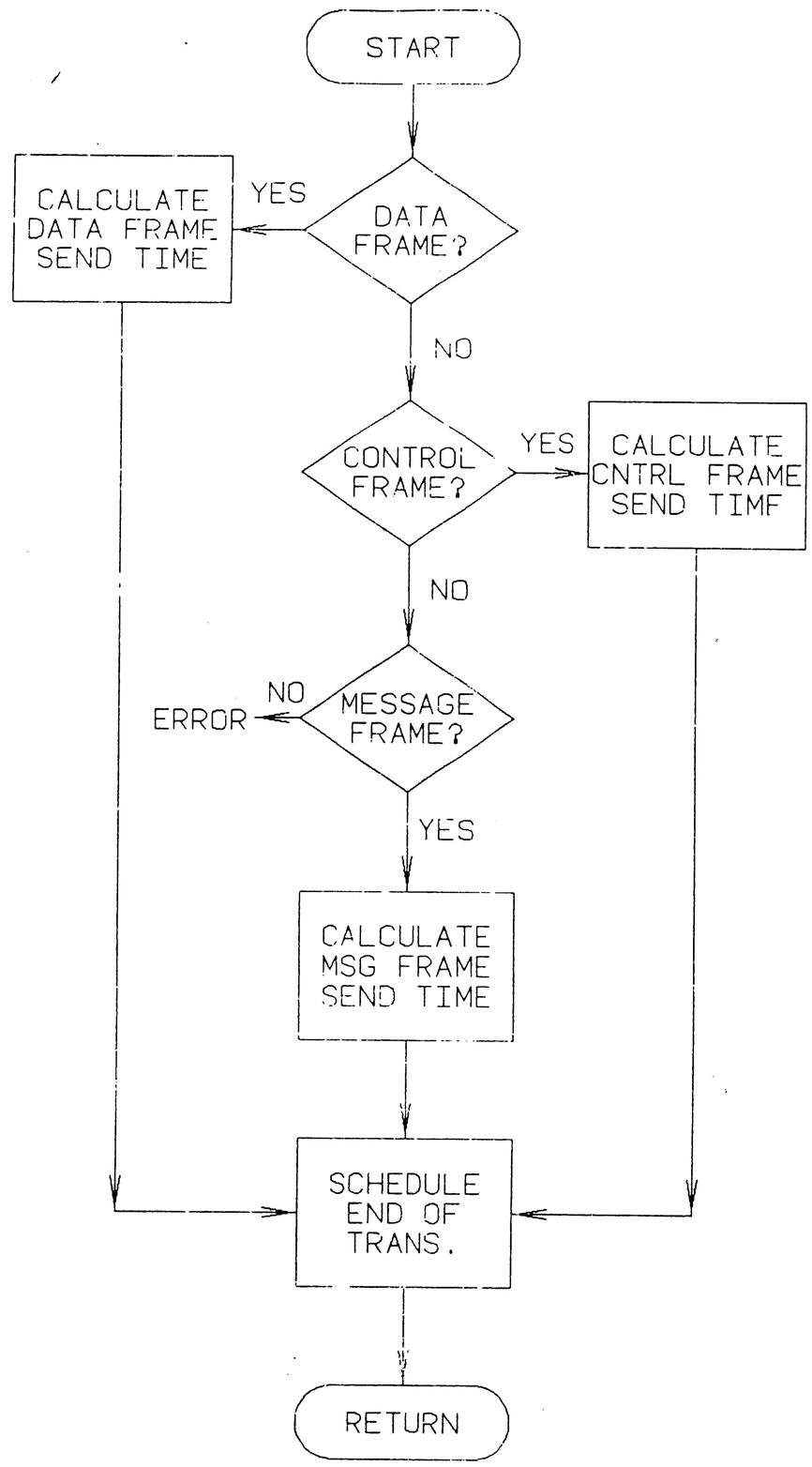

Figure 5-3. Token passing subroutine.

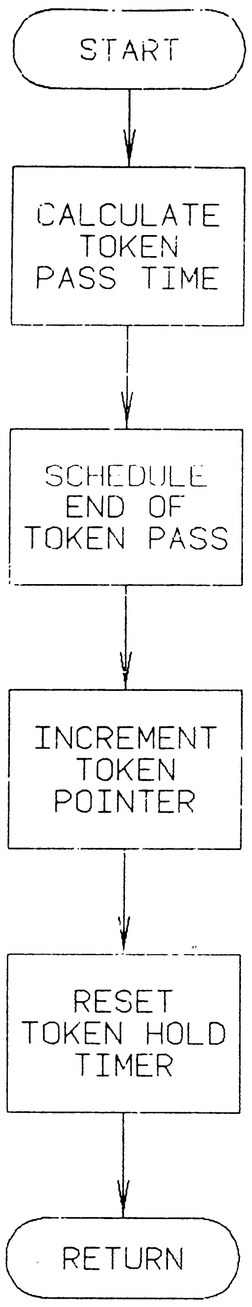

Figure 5-4. Frame sending subroutine.

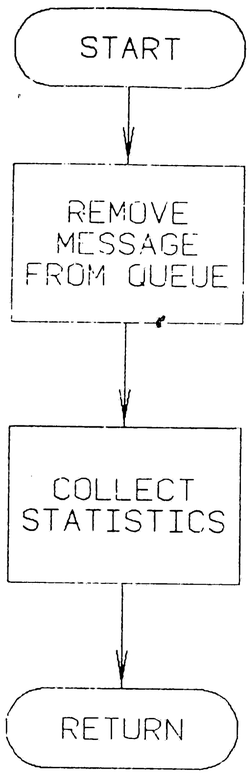

Figure 5-5. Message removal subroutine.

represents the message queue. A separate file exists for each node on the network. The next message arrival is also scheduled at this time. This is done by using the SLAM scheduling routine to schedule the event of a message arrival at the node. The time between the present time (in the simulation) and the time of the next message arrival determines the load on the network. This value is initialized in the SLAM network.

It is important to clarify what is meant by a message arrival at a node. In these network models, the load on the network is represented as the number of messages sent by each node per unit time. Each message is comprised of two MMFS or MMS frames plus the control frames required to establish and drop the association required between the sending and receiving nodes for message transfer. It is assumed that all communications on the network consist of this type of message. Therefore, the arrival of a message at a node is used to signify that the application layer of that node has determined that it has information to send to another node on the network. The rate of these arrivals is constant and is equal for all nodes on the network. The node to which the message is sent would normally also be determined by the application layer. In this case, however, the receiving node is randomly selected from all other nodes on the network.

The network control routine contains the main body of logic required to simulate network performance. It represents the logical operations performed by the token holding node. It is here that the decisions of when to pass the token and when to transmit the data frames are made. In this routine the message queue of the token holding node is checked to determine if there are any frames pending. If there are, the frames are transmitted by calling the sending routine. If there are no frames pending in the message queue of the token holding node or if the maximum token holding timer has expired, the token is passed to the next station. This routine also initiates the removal of the message from the queue and the collection of statistics associated with the transmission of the message by calling the statistics collecting routine when the last frame associated with the message has been sent.

Because this routine embodies the main logic of the architecture, this is where the major differences between the two models exist. A more detailed explanation of the logic in this routine will be presented for each of the models in later sections.

The token passing routine does not contain the logic which determines when the token is to be passed. As mentioned earlier, this logic is contained in the network control routine. Instead, this routine is used to determine the amount of time required to pass the token to the next

node in the logical ring. In addition, this routine also resets the maximum token holding timer and increments the token pointer to indicate which node has control of the network.

Like the token passing routine, the frame sending routine contains only the logic required to determine the amount of time which is required to transmit the individual frames of data. It is called by the network control routine which determines when frames are to be transmitted. This is another routine which differs significantly between the two models. A more detailed explanation of this routine will, therefore, also be presented in later sub-sections.

The last routine, the message removal routine, is also called by the network control routine. In this routine SLAM facilities are used to remove the message from the queue. Because this routine is called only when the message transaction is complete (all frames associated with the message have been sent) it is also used to collect statistics related to the transfer of the message.

The model contains two additional subroutines. An initialization subroutine is used to initialize the variables of the model according to user inputs from the SLAM network. It also schedules the arrival of the first message at each node and starts token rotation.

An event subroutine is used both to schedule message arrivals and to advance the system clock according to the

delay incurred from token passing and the sending of data frames. This routine is called by SLAM whenever an event time is reached. In the case of a message arrival event occurrence, another message arrival is scheduled for the node to which a message just arrived. The message arrival routine is, therefore, actually a part of this subroutine.

Whenever an event occurs which is associated with the end of a delay time, the token controlling node is free to perform another action. Therefore, the network control routine is called whenever such an event time is reached. In this manner, no secondary actions are allowed to take place during the time which a token is being passed or a frame is being sent.

5.3.2.1 MAP Model

Crowder [6] describes the MMFS message sending transaction as a six step process. The first step involves establishing a (logical) transport connection between the sending and receiving nodes. This requires the sending of three control frames. The second step is to establish a CASE association between the nodes. This requires the sending of two control frames. The next step is the transfer and action of two MMFS messages which requires two data frames and some end user program processing. The receiving station must then send an acknowledgement frame to the sending station. This requires one data frame. The

acknowledgement cannot be overlapped. The CASE association and the Transport connection are then released. This requires the sending of four control frames (two for the release of each association).

Each of the frames transmitted over the network requires some processing time for the logic contained in the various network layers. This adds delay to the transmission of the frames. For the MAP model this time consists of a combination of logical link control protocol and transport layer delays. The difference between the control and data frames discussed above lies in the fact that the transport layer delay for control frames is twice the transport layer delay for data frames. In the case of frames sent for the MMFS messages, additional delays are incurred from the application layer (user program and CASE logic latency).

The MAP model represents the delays associated with sending data frames by identifying three types of frames transmissions. The first type is for pure data frames. The delay from sending this type of frame consists of the logical link control layer delay plus two times the transport layer delay (one for each node). This type of transmission is used only for message acknowledgements. The second type of transmission is for control frames. The delay from sending this type of frame is the same as that experienced for the data frames with the exception of doubling the transport layer delay (for each node). This

type of frame is used for the establishment and release of the transport and CASE associations. The final type of transmission is for the sending of MMFS frames. These frames contribute the same amount of delay as the pure data frames plus additional delays from the CASE protocol latency at each node and end user program latency for the logic required for MMFS. The transmission type to be used in sending the frame is identified when the subroutine providing this logic is called.

The messages in the queue of each node are represented as entries in a file with the file number being equal to the node number. Each entry has seven attributes associated with it. The first attribute is the time at which the message was placed in the queue. The second attribute is the number of the node to which the message is to be sent.

The next three attributes are used to represent the frames that must be sent to complete the message transaction. The third attribute of the entity, therefore, represents the remaining number of connect frames that must be sent. The fourth attribute represents the remaining number of MMFS frames. The fifth attribute represents the remaining number of release frames. The sixth attribute is a flag which indicates whether or not the acknowledgement to the message has been received.

The seventh attribute is used only for entities which are acknowledgements which must be sent from a receiving

node to a sending node. The value of this attribute is equal to the number of the node to which the acknowledgement must be sent. When this value is not equal to zero, all other attributes of the entity are equal to zero.

The network control routine uses these attributes to determine what types of frames must be transmitted (if any). Before attempting to transmit these frames, the token holding node first determines whether there are actually messages in its queue and if the token hold timer has not been exceeded. If both of these conditions are met, the queue is then searched for pending acknowledgements. If acknowledgement messages are pending, then these frames are sent. If no acknowledgements are pending, then frames of the first message in the queue are transmitted. The messages are thereby transmitted according to a First-In-First-Out priority scheme.

When sending a message, the token holding node first checks for and transmits all connect frames. When all connect frames have been sent, the MMFS frames are then transmitted. When this task is complete, an acknowledgement message is placed in the message queue of the receiving node. Transmission of the release frames cannot take place until this acknowledgement is sent by the receiving station. Once the acknowledgement is received and all release frames have been sent, the message is removed from the queue and the node begins to process the next message.

5.3.2.2 EPA Model

Message transactions are handled differently by the EPA. As Crowder [7] explains, sending EPA messages is a three step process consisting of: initiating a MMS dialogue, transferring the MMS messages, and then terminating the MMS dialogue. The initiation and termination of the MMS dialogue requires two frames each. The two MMS messages transmitted also require two frames. With this architecture, the message acknowledgement is sent in full duplex. Therefore, for the purposes of this model, it is as though no acknowledgement is sent at all (since no delay is incurred in sending it).

The delays associated with sending these frames also differ from those in the MAP model. In this case, delay in sending the frames is incurred when the data passes through the SDA link control protocol of each end system and from the MMS protocol at each end system. The overall delay caused by sending frames for the initiation and termination of the MMS dialogue is, therefore, the sum of these two delays. When the frames being sent are the actual MMS frames, a delay caused by user program latency must also be added. The frame sending routine represents each of these types of frame transmissions (control and MMS) separately. When the control program determines a frame must be sent, it

identifies which type of frame is to be sent when calling the frame sending routine.

Messages are represented in the same manner in the EPA model as they are in the MAP model. In this case, however, only five attributes are associated with each message entry (as apposed to the seven used for the MAP mode). The first attribute represents the time at which the message entered the queue and the second attribute represents the number of the node to which the message must be sent. This portion is identical to the MAP model. The next three attributes represent the number of MMS dialogue initiation frames remaining, the number of MMS frames remaining, and the number of MMS dialogue termination frames remaining, respectively. The two extra attributes used in the MAP model are for the message acknowledgement process. Since this is not represented in the EPA model, these frames are not required.

The logic in the control routine is essentially the same as that in the control routine of the MAP model. The difference between the two is that no acknowledgements are used in the EPA model. The process of sending frames is, therefore, much more simple for this model. The routine will check for pending frames for the first message in its queue. Each of the frames will be sent in the proper order (dialogue establishment, MMS, dialogue termination) according to its type until all frames for the message have

been sent or the maximum token holding time for the node expires. When all frames associated with the message have been sent, the message is removed from the queue.

5.3.3 Inputs and Outputs

Both models were developed so that the user would have the option of altering the characteristics of the networks. This was done to allow the user to run a variety of experiments using the models.

The first variable which the user can set is the number of nodes on the logical ring of the network. The second variable which can be altered is the frequency of message arrivals. The model was developed so that messages arrive at the nodes in a constant distribution. The value the user enters, therefore, represents the time between message arrivals at each node. The combination of these first two variables determines the load on the network.

The next two variables which the user can set are used for token passing operations. The first of these is the token overhead delay time. This is the time associated with the delay of passing the token to the next node in the logical ring. It should include the delay time incurred for all token bus maintenance operations such as adding tokens to the ring. It should also contain an allowance for the average delay time caused by errors in token passing and the amount of time required to physically pass the token.

The remaining variables are used to represent the delay times associated with passing data through the various layers of the network architecture. For the MAP model these times include: the logical link control delay, the transport protocol delay per MAP end station, the delay associate with CASE logic per MAP end station, and the delay caused by user program (application layer) latency. For the EPA model these times include: the SDA link control delay per MAP end station, the delay associated with MMS and the user interface per MAP end station, and the delay caused by user program (application layer) latency.

The intended use of the models is to judge the performance of the network architectures under the conditions prescribed by the above inputs so that they can be relatively compared. To do this the models must produce results which represent the performance of the network. For this purpose, the results of model execution are displayed in the form of various statistics collected during the run of the simulation.

The primary statistic which is collected is the average amount of time each message spends in the system. This is the amount of time that elapses between the time that the message arrives at a node and the time the message is removed from the queue. It is used as an indication of the response time provided by the network architecture (under the specified loading conditions). In addition to the mean

value for all messages, the standard deviation, coefficient of variation, maximum value, and minimum value are also presented.

A second statistic which is collected during model execution is the average token holding time. This value represents the average amount of time nodes maintain control of the network (hold the token) before passing it on to the next node. Just as with the message statistics, the mean, standard deviation, coefficient of variation, maximum value, and minimum value are calculated and output.

In addition to these values, statistics are collected on the activity in each message queue at each node. The values recorded here include: the average number of messages in the queue, the standard deviation for this average, the maximum number of messages in the queue, the number of messages in the queue at the time the simulation stops, and the average amount of time each message spends waiting in the queue. In the case of this model, the average amount of time the message spends waiting in the queue actually consists of the waiting time plus the message transmission time. Therefore, this result is interpreted as the average amount of time a message spends in the system on a per node basis.

5.4 Verification and Validation

Verification and validation are means of determining whether or not the model is an accurate representation of the system being simulated. Verification is usually performed first. This is "the process of establishing that the model executes as intended."[26] After the model has been verified, it is validated. This is "the process of establishing that a desired accuracy or correspondence exists between the simulation model and the real system."[27]

Verification of the MAP and EPA models was performed by two separate methods. First, the code for the models was thoroughly checked for any logic errors which might exist. Since it is often difficult for the programmer to spot his own mistakes, the models and an explanation of their intended function were also presented to other individuals. These individuals also checked the logic of the models to determine if any errors existed. The second means of verifying the models was to observe their dynamic performance. This was done by printing out virtually all information available on the operation of the models during their execution. In this manner, each logical operation performed by the models was observed. The activities which occurred were then compared to the expected operation of the

---

[26] Pritsker, p. 11.

[27] ibid.

simulated networks. This process was very time consuming and may not have been possible for a more complex model.

The validation of the models is significantly more difficult to perform then the verification. In an ideal situation, the validation would be performed by comparing the results of the simulation to those obtained directly from the real system being modeled. This was, of course, impossible for these models. The only means of doing this which remained was to compare the results obtained from the execution of the model to results obtained through manual calculations (which were performed by Crowder [6,7]). When these results were compared, there was a significant difference between them. The values obtained from model executions were more than ten times greater than those calculated by Crowder [6,7]. However, it was expected that such differences would be present due to the dynamic operation of the model which could not be duplicated through manual calculations. In fact, further examination revealed that the difference between the values was equal to the amount of time the messages spent waiting in their queues for transmission.

The confidence gained from the verification of the models was very high. This means that the models perform as intended. The confidence gained from the validation of the models was also high. However, in this case this means that the models appear to be a good representation of the

operation of the network architectures as they were described by Crowder [6,7]. It does not necessarily provide an extremely high level of confidence that the models are a good representation of actual networks operating under these conditions. To have confidence that this holds true as well, an assumption must be made that the analysis of the network architectures presented by Crowder [6,7] is valid.

5.5 Experimental Design

The design of the experiments primarily involves the setting of the variables of the models (such as the delay times and loading conditions). However, there is one other major factor which must be considered. This is the run length of the simulation. The run length is the amount of simulated time for which the model executes. Care must be taken to select a value which allows the model to reach a steady state of performance. If steady state performance is not achieved, the results obtained from the simulations will not accurately represent the operation of the system which is being simulated.

One means of assuring that the model reaches this state is to track a value as the model executes and stop the simulation only when that value reaches a steady state. This is the method most commonly employed when creating simulations. However, since two separate models are being used, this method was not employed for these experiments.

Instead, experiments were run using the models to determine a time at which it was certain that both models would achieve steady state operation (provided the conditions under which the model is executed allow this to occur). This is done because identical conditions are desired for the execution of the models. By stopping each model independently, it would be possible for the models to run for different lengths of (simulated) time. This could increase the variability between the models, thereby making a relative comparison of their results more difficult. The final run length chosen was 15 minutes (of simulated time). This value includes a considerable buffer for those situations which may require more time to reach steady state than usual.

All of the delay times used were taken directly from the articles by Crowder [6,7]. For the MAP model, he explains that the link control protocol produces 2ms of delay, the transport protocol produces 4ms of delay at each node, the CASE logic produces 4ms of delay at each node, and the latency of end user programs produces 5ms of delay. For the EPA model, delay times are given as: 1ms for the SDA link control protocol; 5ms for the MMS and user interface logic at both nodes; and 5ms for user program latency.

The time which Crowder calls one-way access plus transmission time was taken as the delay incurred from token maintenance overhead. Because the value was not presented

as the token maintenance delay, simulation runs with values ranging from 0.1ms to the 4ms specified by Crowder [6] were performed to determine the effect on the overall performance of the network. When the values were maintained across both models, they did not have a significant effect on their relative performance. The values given by Crowder were, therefore, maintained as the delay due to token maintenance activities and physical transmission for all subsequent experiments.

Another value directly related to token control which must be specified is the maximum token holding time allowed. This value determines the maximum amount of time for which a particular node can send the frames associated with messages in its queue before it must relinquish control of the network to the next node in the logical ring. In most cases the value is set much higher than the predicted average token holding time. Therefore, the maximum token holding time should rarely be reached. If the token is continuously being passed as a result of the expiration of this timer, the network is most likely loaded beyond capacity. The selection of this time should, therefore, be irrelevant. However, it can be used to place an upper limit on the maximum amount of time that elapses between opportunities for individual nodes to obtain control of the network (the token rotation time) and must, therefore, be set. The value

chosen for these experiments is 400ms. This value was based on the findings of preliminary executions of the models.

The last two variables which had to be set were the number of nodes on the network and the number of messages sent by each of these nodes per unit time. To maintain consistency with the development of the model, the number of nodes on the networks was taken as twenty. This is the number of nodes which Crowder [6,7] uses for both the MAP network and the EPA network.

The determination of the number of messages sent by each node per unit of time was not quite as simple. As mentioned earlier, this value actually represents the frequency with which each node must send information to another node on the network. Each of these information transactions is represented in the models by the transfer of two messages (MMFS messages in the MAP model and MMS messages in the EPA model). The loading of the networks is, therefore, determined by the amount of time which elapses between the transactions at each node.

As mentioned in Chapter 3, the load on networks is usually represented as a percentage of capacity or in Mbps. It is rarely defined in the terms described above. In addition, it would be very difficult (if not impossible) to translate from a percentage or Mbps value into a number of transactions per unit time value. These factors made it

difficult to obtain loading condition information in the form in which it was required.

Because a value which was accompanied with a high level of confidence could not be obtained, experiments were run for a variety of times between messages. The starting value was one second between messages. This value was then increased by one second for nine subsequent model executions. This provides a range of loading conditions under which the relative performance of the two network architectures can be compared.

The final experiment performed with the simulation models consisted of running both models ten times under the conditions described above. The results obtained from these executions were then compared relatively to determine the percentage difference in the performance of the architectures.

5.6 Results

There was one very interesting development during the running of the experiments. It was discovered that once the load on the network was such that the architecture could keep pace, the average amount of time required for a message transaction remained very constant regardless of a reduction of the load on the network. However, if the load on the network was greater than the capacity of the architecture, the amount of time required for a message transaction grew

without bound as did the number of messages awaiting transmission in the queue of each node.

As expected, the load at which this transition occurs is significantly different for the two architectures. The MAP architecture was not capable of supporting the number of transactions required for messages which would arrive with a frequency greater then one message every five seconds at each node. The EPA was capable of supporting the processes required for messages which would arrive at each of the nodes as frequently as one every two seconds.

Because the overloading of the network had very detrimental effects on the performance of the network architectures, only those values obtained from model executions in which the loading conditions did not exceed the capacity of the network will be used to compare the performance of the two architectures. Earlier, the point was made that the architectures should be compared under the exact same loading conditions. However, since the performance of both architectures remains almost constant regardless of the load, provided they are not loaded beyond capacity, it appears that the only distinction required for the relative comparison is that the networks be operating in a steady state.

The results of the simulation runs for the various loading conditions for each of the models can be found in Figure 5-6. The graph in this figure plots the average

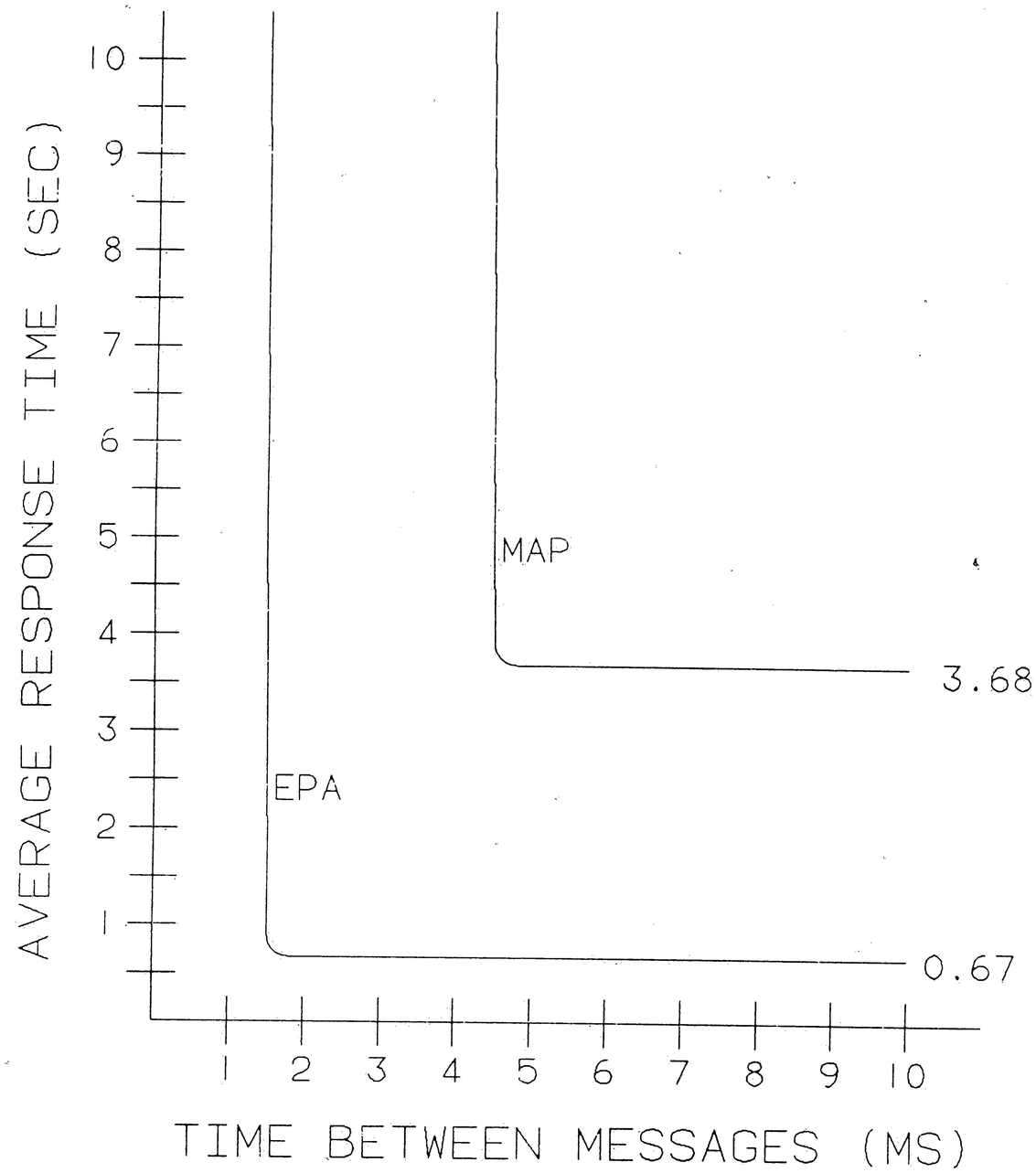

Figure 5-6. Time spent in system by the message vs. the time between message arrivals.

amount of time each message spent in the system against the time between the message arrivals at each node. When studying this graph, the reader should realize that the overall load on the network decreases as the time between message arrivals increases. The reader is also reminded that the results presented here should be used only in a relative comparison between the two models. They are not necessarily representative of the actual performance of the architectures.

In the MAP model, messages transactions took an average of 3.68 seconds. The average value for the standard deviation was 0.653 with a standard deviation between values obtained from different loading conditions of 0.005. The coefficient of variation for each of the simulation runs had a mean value of 0.178 and a standard deviation between values of 0.002. The minimum amount of time required was never less then 0.20 seconds and the maximum was never greater than 4.50 seconds.

The EPA model required an average of 0.67 seconds to process a message transaction. The average value for the standard deviation for this time within each execution was 0.369 with a standard deviation between values obtained from the different loading conditions of 0.0. The coefficient of variation for each of the simulation runs had an average value of 0.551. The standard deviation of this value for the different loading conditions was also 0.0. The minimum

amount of time required in this architecture was never less than 0.062 seconds and the maximum was never greater than 1.28 seconds.

If it is assumed that the response time of the architectures can be represented by the amount of time required to process a message transaction, then results of the simulation executions show that the Enhanced Performance Architecture provides a response time that is nearly 550% faster then that provided by the MAP Architecture.

The results also show that this increase in transaction processing speed also acts to increase the maximum load which can be handled by the EPA. This increase was found to be in the neighborhood of 250% over MAP.

Another comparison between the architectures can be made by looking at the coefficient of variation values within the individual model executions. These values were more than three times as great in the EPA model as they were in the MAP model. This shows that the MAP architecture spent a greater majority of its execution time processing message transactions than the EPA model did. This seems to indicate that the EPA architecture dispatched the message transactions more rapidly than the MAP architecture did which supports the results above.

The average number of messages awaiting transmission in the message queue of each node is also of some importance. Unlike the average message transmission times, these values

did decrease with the decreasing load on the network (see Fig. 5-7). In general the average values for the MAP model where higher then those of the EPA model. In addition, the maximum number of messages in a queue never rose above one for the EPA model where it was as high as six for the MAP model. These facts again support the notion that the EPA architecture outperforms the MAP architecture although no quantitative measure can be derived from this observation.

An additional aspect of the operation of the networks which was observed was the average amount of time each node held the token. Unlike the average message transaction time, this value did vary with the load on the networks. The value decreased as the load on the network was reduced and seemed to approach a minimum which is assumed to be the token maintenance overhead time (see Fig. 5-8). As expected, the value did not approach the maximum token holding time allowed except in those cases where the load on the network was greater then the capacity of the network.

5.7 Conclusions

It is clear from the results presented above that the EPA outperforms the MAP architecture significantly in terms of response times. This performance improvement was expected based upon the fact that a reduced number of OSI layers are utilized by the Enhanced Performance

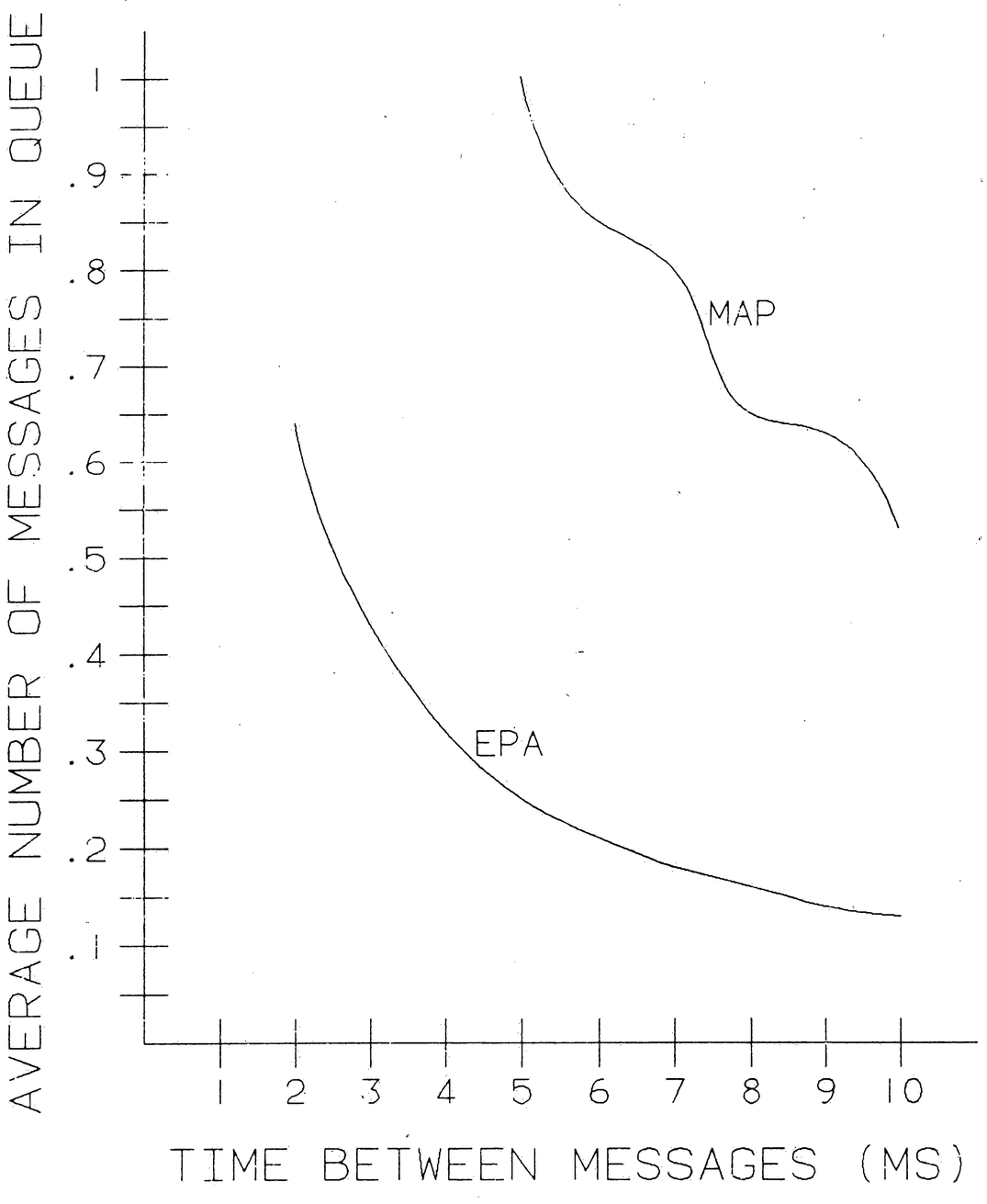

Figure 5-7. Average number of messages in queue vs. load.

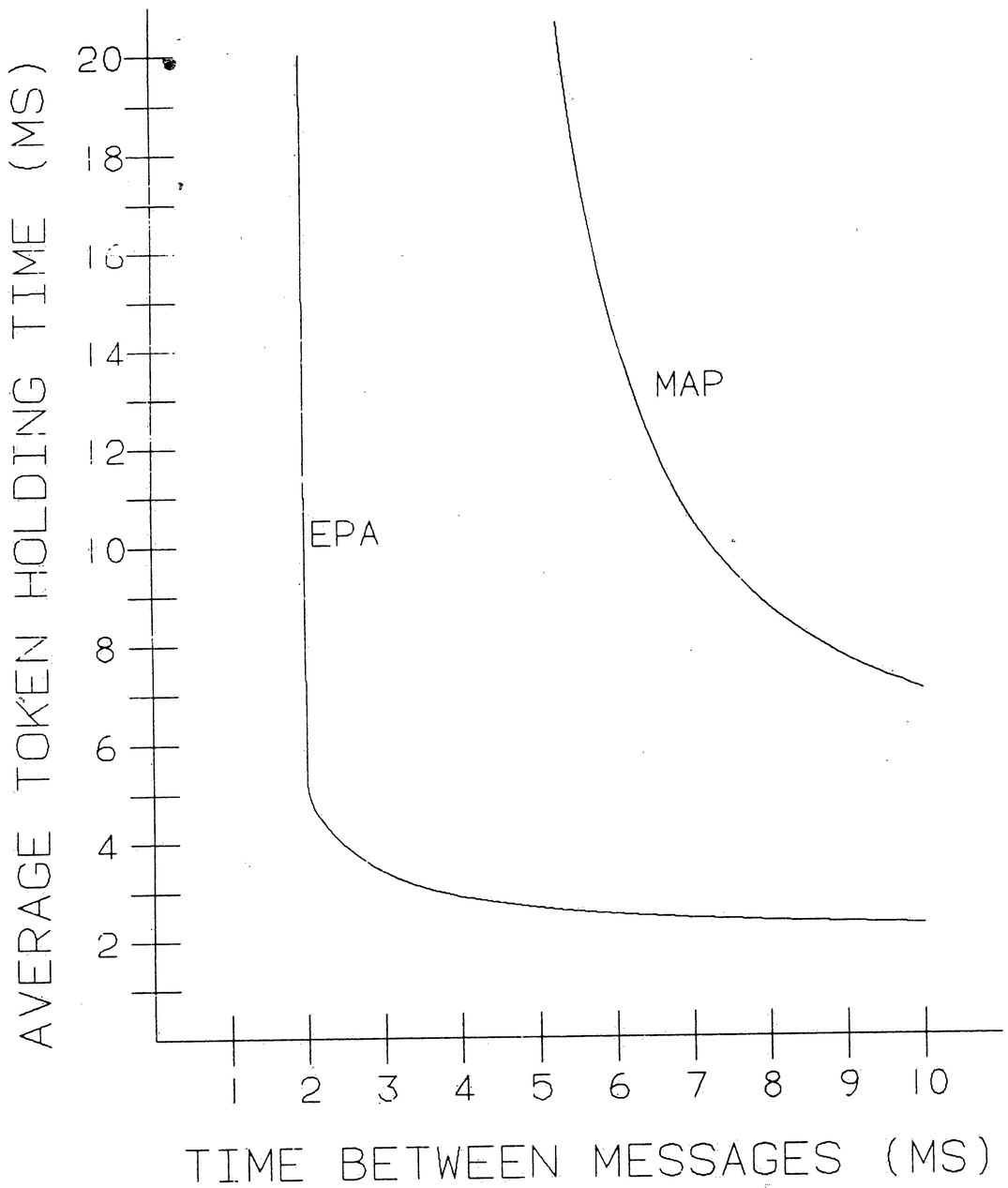

Figure 5-8. Token holding time vs. load.

Architecture. The magnitude of this improvement was not expected to be as great as the results show.

There are other factors, in addition to the response time, that must be considered when comparing the two architectures. Primary among these are the considerations of availability, reliability, maintainability, and costs which were listed as concerns of the MPII committee in the MPII White Paper [12]. Because the EPA architecture used in this comparison was specifically developed with these needs in mind, it is better suited to provide for them.

A conclusion can, therefore be drawn that, in those situations where real time control is required, such as process or cell control, the EPA is better suited to provide for the specific needs of the application.

It should be noted that the MAP architecture as described by Crowder [7] and subsequently represented by the model presented in the previous chapter is not representative of the optimal performance which can be achieved with the MAP architecture. It is also true, however, that the models do not make assumptions about cell controller capabilities or cell size that would demonstrate the maximum performance gains over MAP which are achievable with an EPA.[28]

---

[28] Crowder, Robert S. "Enhanced Performance and MAP: Part II." MAP/TOP Interface November 1986, p. 4.

Some may argue that the version 3.0 MAP architecture will offer substantial improvements in performance over version 2.1. The opposite will in fact be true. Following the analogy presented by Crowder [7], version 3.0 will simply add more delay due to the definition of the presentation layer. However, an EPA is presented as part of the version 3.0 specification. This architecture within MAP, for which the PROWAY specification was a candidate, may perform equally as well, if not better than, the EPA studied in this paper.

The results of the experiments performed also show that the performance of both network architectures degrades very rapidly as the capacity of the network is approached and surpassed. This demonstrates that the network loading conditions must be carefully considered during the design phase of network development.

5.8 Chapter Summary

After reading this chapter, the reader should have an understanding of the magnitude of the performance gains provided by an EPA when compared to the basic MAP architecture. Relatively detailed information on the simulation models developed and the experiments run using those models has also been presented. This information is provided so that the reader can understand the logic behind

the model which is intended to convince him that the model and the experiments are indeed valid.

6 Summary and Conclusions

This paper began by presenting a brief discussion of the place of Local Area Networks in the manufacturing environment. It then went on to introduce the reader to the fundamentals of Local Area Networks. Topics covered for this purpose included network topologies, types of transmission media, signaling techniques, and methods of media access control. This introduction was intended to give the reader the background information necessary for the understanding of material presented later in the paper. The ISO-OSI model was also introduced at this time to give the reader a framework for understanding the descriptions of the two network architectures compared later in the paper. It is hoped that the information provided in this chapter eliminated the need for the reader to consult outside references to comprehend the topics covered in the following chapters.

The objective of the next chapter was to provided the reader with information about how the performance of various LAN architectures is judged. Accordingly, topics such as network performance measures and the network characteristics and variables which effect performance were covered. This chapter also introduced the topic of network simulation. This material was presented to provide the reader with an understanding of why simulation is used to model Local Area

Networks as well as what is involved in the creation of such models.

Next the reader was introduced to the two network architectures which were compared in the final chapter of the paper. The descriptions presented were intended to serve two purposes. The first of these was to give the reader a basic understanding of what must be defined in the specification of Local Area Network architectures and how each of the specification covered fulfills these specification requirements. The second purpose was to demonstrate the differences between the two architectures so that the reader could form an understanding of how and why they perform differently.

Finally the simulation models used to compare the performance of these network architectures were presented along with results obtained through experimentation with the models. The logic used to develop the models was covered in a relatively high level of detail as was the design of the experiments performed using the models. This was done to give the reader an understanding of the models and the experiments that is sufficient to give him confidence in the results obtained.

These results disclosed that the EPA provides significant improvements over the MAP architecture in response times achieved for the transfer of information on the network. From this the conclusion was drawn that an EPA

should be used on subnetworks which interconnect devices which required real time responses. It is important that the reader notes the distinction of subnetwork in this conclusion. The results do not suggest that the EPA outperforms the MAP architecture for general networking needs. MAP is much better suited for the needs of the larger (backbone) network for many reasons. It is beyond the scope of this paper to list these now. However, an example would be the fact that the EPA uses carrierband and therefore does not have the multi-channel capabilities of the MAP architecture.

A more general conclusion that can also be drawn from the results of the final chapter is that performance gains can be achieved over the MAP architecture by shifting some of the processing responsibilities from the mid-layers to the application layer thereby eliminating the need for at least three of the OSI layers defined by MAP. This will ultimately result in a time savings from removing the delay associated with the passing of data through these layers.

There are several areas which were pointed out during research on this topic which would prove interesting for further research. The first of these involves the type of load which is placed on the network architectures. In the models developed for this research, messages would arrive at each of the nodes on the network according to a constant distribution. This representation conforms to situations in

continuous processing and discrete parts manufacturing in which the devices communicating on the network behave in a consistent manner. However, in some industrial situations, this consistent communication behavior is periodically interrupted by bursts of activity on the network. Such burst may represent the downloading of new control logic to devices on the network. It would be interesting to incorporate such a situation into the simulation models presented and observe the effects on the average response times of each of the networks. It would also be interesting to expand the models to include communications between devices on the subnet and devices on the backbone. Unfortunately, the information required to model this type of activity could not be obtained at the time of this writing.

LIST OF REFERENCES

[1] Allen, Roger, "Factory Communication: MAP Promises to Pull the Pieces Together," Electronic Design, 15 May 1986, pp. 102-112.

[2] Archambault, Jean-Luc, "An IEEE 802.4 Token Bus Network Simulation," NBSIR 84-2966 October 1984.

[3] Baker, Donald G., Local-Area Networks with Fiber-Optic Applications, Englewood Cliffs, N.J., Prentice-Hall, 1986.

[4] Chandy, K. Mani and Sauer Charles H., Computer Systems Performance Modeling, Englewood Cliffs, N.J., Prentice-Hall, Inc., 1981.

[5] Chlamtac, I. and Jain, R., "A Methodology for Building a Simulation Model for Efficient Design and Performance Analysis of Local Area Networks," Simulation, Feb 1984, pp 57-66.

[6] Crowder, Robert S., "Enhanced Performance and MAP: Part I," MAP/TOP Interface, August 1986, pp 1-4.

[7] Crowder, Robert S., "Enhanced Performance and MAP: Part II," MAP/TOP Interface, November 1986, pp 1-4.

[8] Iglehart, Donald L. and Shedler, Gerald S., "Simulation Output Analysis for Local Area Computer Networks," Acta Informatica, 21 (1984), 321-338.

[9] ISA SP72 Committees, ISA-ds72.01, Proway-LAN An Industrial Data Highway, Draft Standard, Research Triangle Park, North Carolina: Instrument Society of America, 1984.

[10] MacNair, Edward A. and Sauer, Charles H., Simulation of Computer Communication Systems, Englewood Cliffs, N.J.: Prentice-Hall, 1983.

[11] Manufacturing Automation Task Force, General Motors' Manufacturing Automation Protocol Specification Version 2.1, Detroit Michigan: General Motors Publications, 1985.

[12] MAP Process Industries Initiative (MPII) Working Group, MAP in the Process Industry, 1986.

[13] Pimentel, Jean R., "Performance Simulation of the IEEE Token Bus Protocol," NBS Special Publication 500-127 June 1985, pp. 5-34.

[14] Pritsker, A. Alan B., Introduction to Simulation and SLAM II, 3rd ed., New York: John Wiley & Sons, 1986.

[15] Sastry, A. R. K. and Atkinson M. W., "Simulation of the IEEE 802.4 Token Passing Bus Protocol Using SIMSCRIPT," NBS Special Publication 500-127, June 1985, pp. 52-60.

[16] Schultz, Henrik A., "The Role of MAP In Factory Automation," IEEE 1986 IECON Proceedings, vol. 2, pp. 607-613.

[17] Stacy, Alan H., The MAP Book: An Introduction to Industrial Networking, Santa Clara, CA: Industrial Networking Incorporated (Part number: 16652-01), 1987.

[18] Stallings, William, Data and Computer Communications, New York: Macmillan, 1985.

[19] Stallings, William, Local Networks: An Introduction, New York: Macmillan, 1984.

[20] Stickler, Mark G., "Local Area Networks (LANs) For The Office Environment," CIM Lab Seminar Series, Lehigh University, 30 June 1987.

[21] Stickler, Mark G., "Local Area Networks: Methods of Determining Their Needs and Modeling Them for Design and Performance Analysis," Masters Thesis, Department of Electrical Engineering, Lehigh University, 1986.

[22] Tanenbaum, Andrew S., Computer Networks, Englewood Cliffs, N.J.: Prentice-Hall, 1981.

[23] Watson, W. B., "Modeling and Monitoring a LAN, One Experience," NBS Publication, pp. 32-55.

[24] Webb, Michael K., "Local Area Networks Aid Factory Automation," Design News, 6 February 1984, pp. 179-183.

VITA

Kenneth James Wagner was born on March 27, 1964 in Buffalo, New York to Paul Walter and Marilyn Jane Wagner. He received his Bachelor's degree in Industrial Engineering from Lehigh University in June of 1986. He immediately began working toward his Mater's degree in Industrial Engineering at Lehigh. He will receive his Master's degree in January of 1988. While attending Lehigh he has worked for the Industrial Engineering Department's Computer Integrated Manufacturing Laboratory. His duties at the CIM Lab included: work with NC milling machines, the development of postprocessors to allow the programming of NC machine tools with CAD systems, and the instruction of undergraduate students and industry personnel. His interests lie mainly in automation and the application of computer-based technology to manufacturing and manufacturing support activities.