EY-0149E-SG-0001

Network Concepts

A Self-Paced Course

digital

EY-0149E-SG-0001

Network Concepts

A Self-Paced Course

Student Workbook

Prepared by Educational Services
of
Digital Equipment Corporation

Printed in U.S.A.

CONTENTS

SG STUDENT GUIDE

COURSE DESCRIPTION 3
PREREQUISITES. 3
COURSE ORGANIZATION. 5
COURSE GOALS 6
COURSE MAP DESCRIPTION 6
COURSE MAP 7
OPTIONAL COURSE RESOURCES. 8
    Books. 8
    Articles 10
PERSONAL PROGRESS PLOTTER. 11

1 DDP EVOLUTION AND NETWORK CONCEPTS

INTRODUCTION 15
OBJECTIVES 16
RESOURCES. 16
EVOLUTION OF DDP 17
WHY DDP? 17
DATA PROCESSING. 20
    Batch Processing 20
    Interactive Processing 22
        Transaction Processing 23
        Query Processing 24
        Statement-Oriented Query Languages 24
        Form-Oriented Query Languages. 24
        Uses of Transaction and Query Processing 24
DATA PROCESSING SYSTEMS. 25
    Centralized Systems. 25
    Decentralized Systems. 27
    Distributed Systems. 27
DATA COMMUNICATIONS. 30
NETWORKS 32
    Terminal Networks. .32
    Distributed Processing Networks. 33
    Network Functions. .34
        Program-to-Program Communications. 34
        File Transfers Between Systems 34
        File Accesses Between Systems. 34
        Down-Line Loading. 35
    Data Storage 35
        Sequential Organization. 36
        Indexed Organization 36
    Access Modes 38
        Sequential Access. 38
        Random Access. .38

Interrelated Files 39
File Transfer. 39
Remote File Access .40
DATA SECURITY. 40
Access Controls. 41
Encryption 41
EXERCISES. 42
SOLUTIONS. 44

2 NETWORK FUNDAMENTALS

INTRODUCTION 49
OBJECTIVES 49
RESOURCES. 49
NODES. 51
Host Processing. 51
Remote Processing. .53
Front End Processing 55
Remote Concentration 56
EXERCISES. 57
SOLUTIONS. 58
LINES. 59
Switched Lines 60
Circuit Switching. .62
Message Switching. .63
Packet Switching 64
Private Leased Lines 65
TOPOLOGIES 66
Point-to-Point 66
Multipoint 67
Line Contention. 67
Roll-Call Polling. .68
Hub Polling. 68
Centralized Star 68
Hierarchical Tree. .69
Ring 70
Distributed Mesh 71
EXERCISES. 72
SOLUTIONS. 74

3 COMMUNICATIONS FUNDAMENTALS

INTRODUCTION 79
OBJECTIVES 80
RESOURCES. 80
COMPUTER CODES 81
ASCII. 81

EBCDIC 81
SBT. 81
Baudot 81
LOCAL LINKS. 82
Twenty Milliampere Loops 82
EIA Lines. 83
Problems and Restrictions. 83
ASYNCHRONOUS AND SYNCHRONOUS TRANSMISSION. 84
EXERCISES. 87
SOLUTIONS. 88
LONG-DISTANCE TRANSMISSION .89
Public Switched Circuits 89
Private Leased Circuits. 90
MODULATION AND MODEMS. 90
Amplitude Modulation (AM). 91
Frequency Modulation (FM). 92
Phase Modulation (PM). .93
Two-State and Multistate Signals 94
Interface Standards. 95
EIA Standards Overview (RS-232-C Vs. RS-449) 95
ISO 2593/CCITT V.35 Standards Overview 96
Modem Features 97
Testing. 97
Transmit Only/Receive Only 97
Originate Only/Answer Only 98
Autodialing and Autoanswering. 98
Acoustic Coupling. 98
Modem Control. 98
LONG DISTANCE MEDIA. 98
Open Wire Pairs. 99
Wire Cables. 99
Coaxial Cables 99
Microwave Towers 99
Satellites 99
SIGNALLING SPEED AND LINE CAPACITY 99
Subvoice Band. 100
Voice Band 100
Wideband 101
LINE CONDITIONING. 101
EXERCISES. 103
SOLUTIONS. 104
TRANSMISSION MODES 105
TWO-WIRE AND FOUR-WIRE CIRCUITS. 105
MULTIPLEXING 106
THE COMMUNICATIONS INTERFACE 108
Modem Control. 108
Serial/Parallel Conversion 108
Buffering. 109
Multiplexing 110
Interface Features 110

EXERCISES. 111
SOLUTIONS. 112

4 ARCHITECTURE OVERVIEW

INTRODUCTION 115
OBJECTIVES 115
RESOURCE 115
NETWORK GOALS. 117
FUNCTIONS AND LAYERS 117
    The Network User 119
    The User Functions Layer 120
    The Network Functions Layer. 120
    The Data Link Functions Layer. 121
        Framing. 121
        Error Checking 121
        Sequencing 123
        Transparency 123
        Line Control 123
        Synchronization. 123
        Timeout and Start-Up Control 124
    The Physical Link Layer. 124
    Benefits of Layering 124
INTERFACES 124
PROTOCOLS. 126
EXERCISES. 127
SOLUTIONS. 129
DATA LINK PROTOCOLS. 131
    Character-Oriented Protocols (BISYNC). 132
        Synchronization Sequence 133
        Message Fields 134
        Transparency 135
    Byte-Count Oriented Protocols (DDCMP). 135
        Classification Field 136
        Count Field. 137
        Flag Field 137
        Sequence and Response Fields 137
        Address Field. 137
        CRC 1 and 2. 137
        DDCMP Synchronization. 137
    Bit-Oriented Protocols (HDLC and SDLC) 139
        Beginning and Ending Flag. 139
        Address Field. 139
        Control Field. 140
        Information Field. 140
        Frame Check. 140
MESSAGE FLOW 142
EXERCISES. 143
SOLUTIONS. 144

5 DIGITAL NETWORK ARCHITECTURE AND DECnet

INTRODUCTION 147
OBJECTIVES 148
RESOURCES. 149
DECnet FUNCTIONS 151
Task-to-Task Communications. 151
Remote File Access 153
Terminal Facilities. 153
Remote Terminal Facilities 155
Network Management 156
Down-Line Loading. 157
Up-Line Dumping. 158
Loopback Testing 158
Routing. 159
Multipoint 159
THE DNA LAYERS 160
User Layer 160
Network Management 161
The Network Applications Layer 161
The Session Control Layer. 162
The Network Services Layer 163
The Transport Layer. 164
The Data Link Layer. 168
DECnet MODULE INTERFACES 168
DNA PROTOCOLS. 170
USER SERVICES. 173
User-to-User 175
Remote Application 175
Network Management 175
DATA FLOW. 175
Data Flow at the Source Node 177
Data Flow Across the Network
to the Destination Node. 177
Data Flow at the Destination Node. 178
EXERCISES. 179
SOLUTIONS. 183

6 SYSTEM NETWORK ARCHITECTURE

INTRODUCTION 189
OBJECTIVE. 190
RESOURCES. 190
OVERVIEW 191
END USERS. 192
NETWORK ADDRESSABLE UNITS. 192
Logical Units. 192
The System Service Control Point 192
The Physical Unit. 193

Distribution of NAUs 193
Boundary Functions 195
SESSIONS 195
SNA LAYERS 197
End-to-End Services. 198
Transmission Services. 198
User Access to the Network 199
Interface Between Layers 199
PEER PROTOCOLS 199
Request/Response Units 201
Headers. 202
Creating a Data RU 203
The Request/Response Header. 203
Transmission Header. 203
Data Link Header and Trailer 203
Data Flow. 203
FUNCTION SUBSETS 204
LU Types and Function Profiles 204
PU Types 204
Type 1 Terminal. 205
Type 2 Cluster Controller. 205
Type 4 Communications Controller 205
Type 5 Host. 205
Domains. 206
IBM SOFTWARE PRODUCTS. 207
VTAM, TCAM, and EXTM 207
CICS, IMS, and TSO 208
The Network Control Program (NCP). 208
SNA CAPABILITIES 210
EXERCISES. 211
SOLUTIONS. 213

7 X.25

INTRODUCTION 217
OBJECTIVE. 217
RESOURCES. 217
X.25 STANDARD. 219
DTE and DCE. 219
X.25 LEVELS. 222
Level 1 - Physical Interface 223
Level 2 - Frame Level. 223
Level 3 - Packet Level 224
X.25 USER RESPONSIBILITIES 226
EXERCISES. 227
SOLUTIONS. 228

8 LOCAL AREA NETWORKS

INTRODUCTION 231
OBJECTIVES 232
RESOURCES. 232
OVERVIEW 233
LOCAL AREA NETWORK TOPOLOGIES. 234
TRANSMISSION TECHNIQUES. 238
LAN SERVICES 238
    Private Branch Exchange (PBX). 239
    Broadband. 239
    Baseband 239
PROTOCOLS. 240
MEDIA. 242
SOFTWARE AND HARDWARE CONSIDERATIONS 242
RELATIONSHIP TO THE AUTOMATED OFFICE 244
EXERCISES. 245
SOLUTIONS. 246

9 COMMON CARRIERS

INTRODUCTION 249
OBJECTIVES 250
RESOURCES. 250
U.S. DOMESTIC CARRIERS 251
    The Bell System. 251
        Dataphone Service. 251
        Wide Area Telephone Service (WATS) 251
        Leased Lines 251
        Dataphone Digital Service (DDS). 252
    Western Union. 252
    Satellite Carriers 252
VALUE-ADDED AND PUBLIC PACKET-SWITCHING NETWORKS . . . 253
    Public Packet-Switching Networks (PPSNs) 255
        ARPANET. 256
        TYMNET 256
        TELENET. 257
        DATAPAC. 257
        TRANSPAC 257
        DATEX-P. 257
        PSS. 258
INTERNATIONAL CARRIERS 258
EXERCISES. 259
SOLUTIONS. 261

GL GLOSSARY

NETWORK CONCEPTS COURSE CRITIQUE

FIGURES

1-1 Computer Cost Vs. Performance 19
1-2 Typical Batch Processor Components. 20
1-3 Typical Batch System Operation. 21
1-4 Typical Interactive Terminal Usage. 22
1-5 Sample Transaction Processing System Message. 23
1-6 Typical Form-Oriented Screen Display. 25
1-7 Centralized System. 26
1-8 Decentralized System. 28
1-9 Distributed System. 29
1-10 Data Communication Mediums. 31
1-11 Typical Terminal Network. 32
1-12 Typical Distributed Processing Network. 33
1-13 Typical Company File Of Employees 35
1-14 Typical Sequential File 36
1-15 Typical Index File. 37

2-1 Host-Oriented Network 52
2-2 Remote Processing Node. 53
2-3 Front End Processor 55
2-4 Remote Concentrator 56
2-5 Typical In-House Network. 59
2-6 Typical Network Between Cities and States 60
2-7 Circuit Switching .62
2-8 Message Switching .63
2-9 Packet Switching. .64
2-10 Virtual Call Switching. 65
2-11 Point-To-Point Topology 66
2-12 Multipoint Topology 67
2-13 Centralized Star Topology 68
2-14 Hierarchical Tree-Shaped Topology 69
2-15 Ring Topology 70
2-16 Distributed Mesh Topology 71

3-1 The Four Basic Elements of a 20 mA Loop 82
3-2 Typical Asynchronous Character Frame. 84
3-3 Typical Synchronous Transmission. 85
3-4 Public Switched Circuits. 89
3-5 Analog Signal 90
3-6 Digital Signal. 90
3-7 Amplitude Modulation. 91
3-8 Frequency Modulation. 92
3-9 Phase Modulation. .93
3-10 Four-State Signal .94
3-11 Four-Wire Trunk Line. 106
3-12 Frequency Division Multiplexing 107
3-13 Time Division Multiplexing. 107
3-14 Communication Interface 108

9-1 Packet-Switching. 254
9-2 Public Packet-Switching Network 255

TABLES

5-1 DNA Protocols 172

7-1 Summary of X.25 Packets 225

8-1 Basic Local Area Network Architectures. 237

SG

STUDENT GUIDE

COURSE DESCRIPTION

This self-paced course deals with the concepts of distributed data processing networks. It is designed so you can progress at your own pace. At the completion of the course you will have a firm foundation in distributed data processing concepts and terminology. This will enable you to:

- Effectively communicate with technical administrative personnel concerning distributed data processing networks.
- Pursue further distributed data processing network training.

PREREQUISITES

This course is an entry level course and does not require previous subject matter knowledge or distributed data processing network experience to be successfully completed. It is assumed that the student has had a minimum of six months experience in a computer related environment.

COURSE ORGANIZATION

This course is divided into nine modules as follows:

- DDP Evolution and Network Concepts
- Network Fundamentals
- Communications Fundamentals
- Architecture Overview
- Digital Network Architecture (DNA) and DECnet
- System Network Architecure (SNA)
- X.25
- Local Area Networks
- Common Carriers

Within each module, you will find:

- An introduction identifying the material covered in the module.
- Objectives providing a concise statement of what you should be able to do after completing the module.
- Resources identifying optional resources that may be read if you want more information.
- One or more major sections of information.
- Module exercises representing the types of problems you should be able to solve, based on the material covered in the module.

While reading the material in this course, remember that your retention of the concepts presented will greatly increase if you take the module test after reading each module. You can evaluate your own performance by checking your answers against those provided.

If you believe you have the skills called for in the module exercises without completing the instruction, take the test for that module, and check your answers with the answer sheets provided. Even if you can successfully complete the module test without reading the module, you may benefit from material provided in the module. Therefore, it is suggested that you review material even in those modules for which you feel your skills are adequate.

After successfully completing a module test, make sure the course administrator records your completion of that module in the Master Progress Plotter (in the Administrator Guide). Your

Personal Progress Plotter is located at the end of this module. Ask the course administrator to also initial this in the spaces corresponding to each module completed.

For your convenience a glossary has been provided at the end of this workbook. If a term is not defined or is insufficiently defined, refer to the glossary for a better definition, or ask the course administrator for assistance.

COURSE GOALS

After successfully completing this course you should be able to:

- Describe the purpose of a distributed data processing network.
- Define and illustrate network concepts.
- Define and illustrate the concepts of data communication.
- Identify and describe the functions of data communication equipment.
- Describe the functions of each International Standards Organization (ISO) layer.
- Describe the features of DNA and DECnet.
- Describe the features of SNA.
- Describe the features of X.25.
- Describe the features of Local Area Networks.
- Identify and describe the functions of common carriers.

COURSE MAP DESCRIPTION

The course map shows how each module relates to the other modules and to the course as a whole. Before beginning a particular module, it is recommended that you first complete all modules with arrows leading to that module. These prerequisite modules present material necessary to understanding the module about to be studied. If you have no preference, it is best to study the modules in the order presented.

COURSE MAP


DNA AND DECNET
SNA
X.25
LOCAL AREA NETWORKS
COMMON CARRIERS
ARCHITECTURE OVERVIEW
COMMUNICATION FUNDAMENTALS
NETWORK FUNDAMENTALS
DDP NETWORKS AND EVOLUTION
STUDENT GUIDE

OPTIONAL COURSE RESOURCES

Books

1. Communications Architecture for Distributed Systems, R.J. Cypser, Reading, Massachusetts, 1978.

2. Computer Networks, Andrew S. Tanenbaum, Prentice-Hall, 1981.

3. Computer Networks and Distributed Processing, James Martin, Prentice-Hall, 1981.

4. DATAPRO Reports on Data Communication, A McGraw-Hill Company, 1978.

5. DECnet DIGITAL Network Architecture General Description, Digital Equipment Corporation, Order No. AA-K179A-TK.

6. Design and Strategy for Distributed Processing, James Martin, Prentice-Hall, 1981.

7. Distributed Processing: Current Practice and Future Developments, Vol. 2: Technical Report, Q.E.D. Information Sciences, Inc., Wellesley, Massachusetts, 1978.

8. Distributed Systems Handbook, Digital Equipment Corporation, (EP-P1195).

9. DNA Data Access Protocol (DAP) Functional Specification, Version 5.6.0, Digital Equipment Corporation, Order No. AA-K177A-TK.

10. DNA Digital Data Communications Message Protocol (DDCMP) Functional Specification, Version 4.1.0, Digital Equipment Corporation, Order No. AA-K175A-TK.

11. DNA Maintenance Operations Protocol (MOP) Functional Specification, Version 4.1.0, Order No. AA-K178A-TK.

12. DNA Network Management Functional Specification, Version 2.0.0, Order No. AA-K181A-TK.

13. DNA Network Services (NSP) Functional Specification, Version 3.2.0, Digital Equipment Corporation, Order No. AA-K176A-TK.

14. DNA Session Control Functional Specification, Version 1.0.0, Digital Equipment Corporation, Order No. AA-K182A-TK.

15. DNA Transport Functional Specification, Version 1.3.0, Digital Equipment Corporation, Order No. AA-K180A-TK.

16. IBM Systems Network Architecture (SNA) Format and Protocol Reference; Architecture Logic, IBM.

17. IBM Systems Network Architecture (SNA) General Information, IBM.

18. Introduction to DECnet, Digital Equipment Corporation, Order No. AA-J055A-TK.

19. Introduction to Local Area Networks, Digital Equipment Corporation, (EB-22714-18).

20. Introduction to Minicomputer Networks, Digital Equipment Corporation, (EK-05300).

21. Provisional Recommendations X.3, X.25, X.28, and X.29 on Packet-Switched Data Transmissions Services, CCITT, Geneva, 1978.

22. Technical Aspects of Data Communication, Second Edition, John E. McNamara, Digital Equipment Corporation, (EY-AX018-DP-001).

23. Telecommunications and the Computer, Second Edition, James Martin, Prentice-Hall, 1976.

Articles

1. Cravis, Howard. "Local Networks Prove Practical for Datacom Systms in Close Proximity", Datamation, 27:2 March 1981, p. 98.

2. Kinnucan, Paul. "Local Networks Battle for Billion-Dollar Market", High Technology, 1981.

3. Rowan, James T. Southern Bell, Atlanta, Georgia, "A Look at a Popular Switching Technique", Data Communications, March 1981, p. 117.

4. Sideris, George; Contributing Editor. "Network Hardware is Key Element in Connection Cost", Electronic Design, Sept. 1981, pp. SS-53.

5. Sideris, George; Contributing Editor. "Software Helps Networks Grow With Compatibility", Electronic Design, Sept. 1981, p. SS-39.

PERSONAL PROGRESS PLOTTER

Module	Comments	Date Completed	Sign-off Initials
DDP Evolution and Network Concepts			
Network Fundamentals			
Communications Fundamentals			
Architecture Overview			
DIGITAL Network Architecture and DECnet			
SNA			
X.25			
Local Area Networks			
Common Carriers			

DDP EVOLUTION AND NETWORK CONCEPTS

1

INTRODUCTION

Distributed Data Processing (DDP) is the term used to describe systems with multiple processors. Although the term may have a different meaning to different people, in its most common usage, the term distributed implies that the processors are in geographically separate locations. It is quite possible that the term may be used to imply an operation using multiple computers which are not connected at all.

It seems that everybody is talking about DDP, but nobody seems to agree on exactly what it is.

- Some people think of distributed processing as any computer system that uses remote intelligent terminals (typically keyboard type) to reduce the load on the main computer.

- Other people think that you have to have minicomputers in order to have distributed processing.

- And still others say that if you don't have some kind of a communication network, you don't really have distributed processing at all.

All of these people are wrong.

Distributed processing is a lot more than just the equipment that is used to do the work. Basically, distributed processing is a way of managing your information-handling requirements: the centrally planned distribution of a company's computer processing among two or more computer systems.

This module will concentrate on answering the following question -- What is data communications? Why do we need DDP? How does DDP fit into my present data processing needs?

OBJECTIVES

1. Describe the purpose of a Distributed Data Processing network.

2. Describe the features of the following data processing operations:

   - Batch Processing
   - Interactive Processing

3. Define the following data processing systems:

   - Centralized
   - Decentralized
   - Distributed

4. Describe the purpose of Data Communications.

5. Describe the features of the following types of networks:

   - Terminal Networks
   - Distributed Processing Networks

6. Describe features of the following data processing techniques:

   - Data Storage
   - Data Security

RESOURCES

1. Computer Networks, Andrew S. Tanenbaum, Prentice-Hall, 1981.

2. Computer Networks and Distributed Processing, James Martin, Prentice-Hall, 1981.

3. Design and Strategy for Distributed Processing, James Martin, Prentice-Hall, 1981.

4. Distributed Processing: Current Practice and Future Developments, Vol. 2: Technical Report, Q.E.D. Information Sciences, Inc., Wellesley, Massachusetts, 1978.

5. Distributed Systems Handbook, Digital Equipment Corporation, (EB-P1195).

EVOLUTION OF DDP

In the late 1940's and early 1950's, computers were very expensive. Programmers and users were highly trained. In those days there were problems of how to reduce hardware cost, how to improve its performance, and how to make computers easier to use. Though we have some of these same problems today, we have managed to reduce them considerably.

In the 1960's, batch processing was well accepted. Computer users put their programs and data on cards, and passed the cards on to a computer operating staff for input and processing. The computer operating staff would run these programs on a priority basis and send the line printer reports back to the users.

In the 1970's, computer hardware cost began to drop. Inexpensive but powerful mini and micro computers became available. Many companies who at one time could not afford computers began using them at the departmental level.

The evolution of DDP is not unique in itself. The paths followed in its development are pure reflections of the historical trends in computer technology.

WHY DDP?

With distributed processing, complex problems, such as a company's information processing needs, can be broken down into manageable parts. With distributed processing, the problems associated with each manageable part can be solved by using the computer system best suited to it.

To understand the concept of distributed data processing, one doesn't have to understand a lot about computers, just a few basic facts. The most basic fact is this -- different computers are good at different kinds of jobs.

Large-scale computers are typically used as the heart of a central computing system. This type of computer is best suited for batch-oriented jobs. This means that it is best at handling large amounts of data, doing high-speed computation, and taking care of one job at a time without much interference or interruption from outside.

Minicomputers on the other hand, are better at handling smaller amounts of data with a high degree of interaction from the person who's using the computer. Minicomputers will respond quickly to requests for data, and will allow users to hold conversations with the computer and to change their minds and modify things as they go along. They're more efficient in data entry applications, in handling a rapidly changing data base (the entire collection of information available to a computer system), in editing and word-processing applications, and in situations where uncorrected errors pose a critical problem. Also, minicomputers can be connected to other equipment to collect data and monitor instruments while the process is going on.

Microcomputers, like minicomputers, are good at handling small amounts of data. A microcomputer can be thought of as a "small minicomputer" and is usually designed to fit on a single chip. Though a microcomputer may not have all the processing power of a minicomputer, it is proving to be a very cost effective data processing tool. For example, microcomputers are very effective in network environments where they may be used to control data communication lines.

The personal computer market is another area which is becoming very popular. Most large computer companies (together with a multitude of newer companies) are now offering computer systems geared for personal use in the home and small business. Networking software packages are also being offered to be used with them. Some of the more recent and most popular software packages include Source, Compuserve, and Videotex.

Distributed processing is the technique of utilizing not only large-scale computers, but also minicomputers and microcomputers, each one doing the job it does best. The way a business uses and configures these different kinds of computers is determined by a company's management data processing goals.

To get a better understanding of the importance of DDP, refer to Figure 1-1 for a cost vs. performance comparison of today's computers. After reviewing this figure, it should be quite clear that the cost of computers today is declining with respect to increased performance.

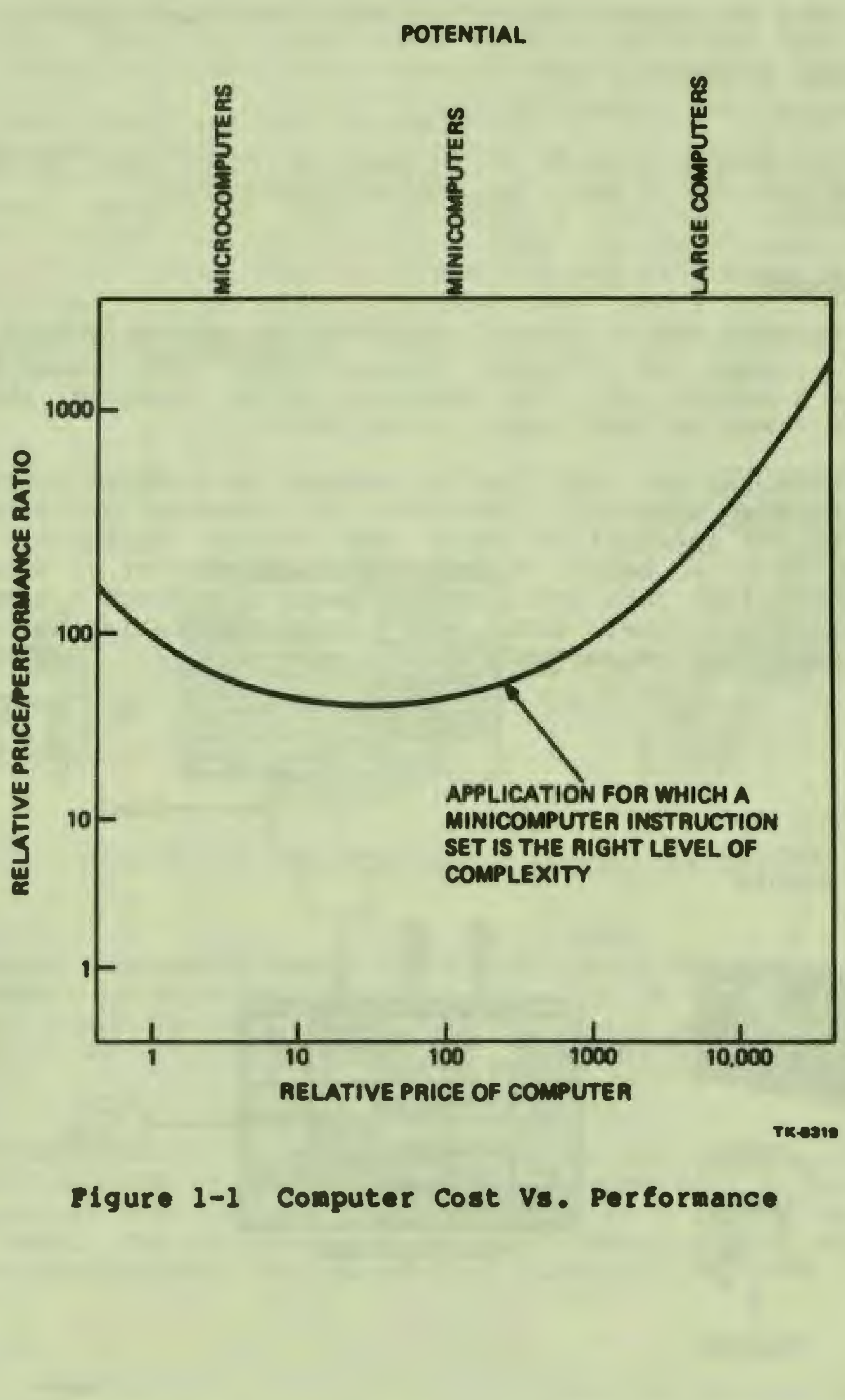


Figure 1-1 Computer Cost Vs. Performance

DATA PROCESSING

Most data processing operations fall into two large groups:

- Batch processing and
- Interactive processing.

Some processing systems use a combination of both batch and interactive methods in their processing operations.

Batch Processing

In commercial applications, batch processing is a method for processing groups of related transactions (for example, file updates) as a single job. The computer cycles through the same sequence of steps for each unit in the batch.

In Figure 1-2 you will find an example of a batch system. A batch processor typically consists of a central processor with data storage and appropriate input and output equipment. Also associated with a batch system will be some sort of **data entry** devices. A typical data entry device used in a batch system would be a terminal. Other data entry devices, called **key-to-tape** devices, store data on magnetic tape.

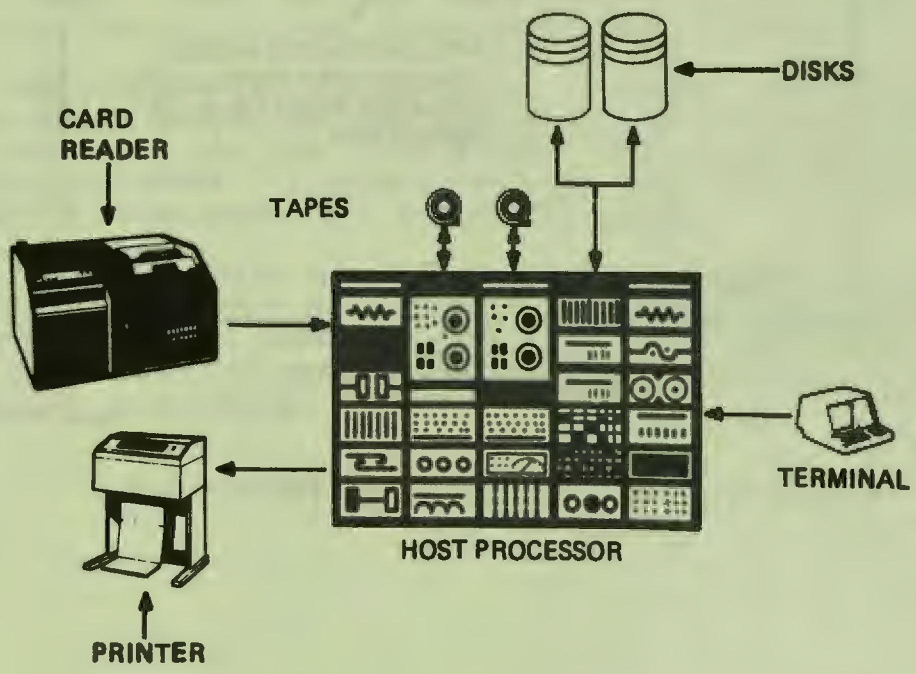


Figure 1-2 Typical Batch Processor Components

Each department in an organization may have its own data entry personnel. At set times, departments forward batches of data on cards or tapes to the batch processor. In a batch system, the user does not communicate directly with the data processor. Also, there is no on-line access to the processor's stored data.

Lets look at the example of a batch system operation illustrated in Figure 1-3. Suppose it is required that an update be made to a master employee file. A data-entry operator in the personnel department must prepare a batch of new and modified records. These records are then sent to the batch processor, which reads them one after another and uses them to update its master file. If the personnel department needs information about certain employees, it makes a request to the computer personnel and waits for the results.

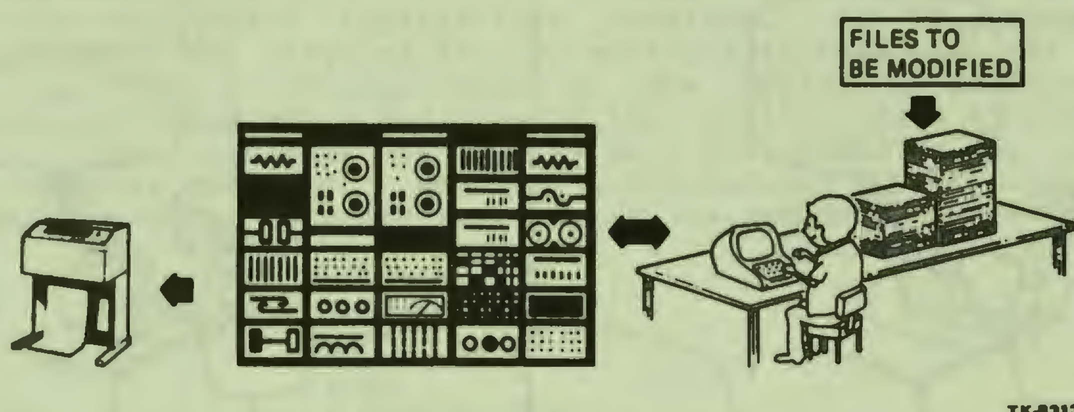


Figure 1-3 Typical Batch System Operation

Batch processing makes efficient use of the computer, which is capable of repeating the same sequence of steps over and over. However, batch processing can be inefficient and unacceptable for the user who needs information and the results of processing immediately.

Technological improvements now permit the building of computers that have more power, but cost the same as bigger, older machines, or even to have the same power as previous models, and cost less. The minicomputer and microcomputer both emerged along this constant-power, decreased-cost pathway.

Interactive Processing

In an interactive system, the user communicates directly with the processor by means of an interactive terminal with a keyboard and a video screen or printer. The user sends any data and processing requests to the processor by typing them on the keyboard. The processor returns data and messages by displaying them on the screen or by printing a hard copy on the printer. The terminal user can run system programs, develop and execute applications, and get on-line access to stored data.

The interactive user can correct mistakes immediately (see Figure 1-4), rather than waiting for the return of a batch job some hours later, and can base present work on immediately preceding results.

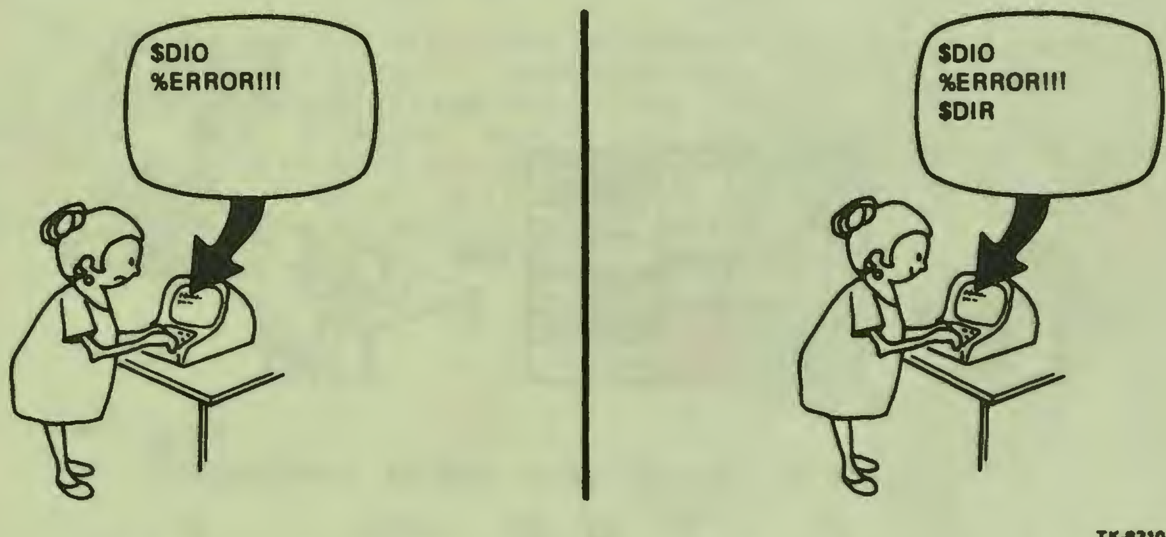


Figure 1-4 Typical Interactive Terminal Usage

Interactive systems range from microprocessor-based terminals to central processing computers that operate clusters of timeshared terminals. In timesharing, the processor divides its time among many terminals in such a way that each terminal appears to be the only terminal which has the use of the computer. Small interactive computers are also used for laboratory experiment control, machinery control, and other real-time applications that can not fit into a batched environment.

Two types of interactive processing used in commercial environments are:

- Transaction processing and
- Query processing.

Transaction Processing -- An interactive computer system that is used for data processing applications by people with no computer-specific training is called a transaction processing system, simply because the data is processed in single, interactive transactions, rather than in traditional batched fashion.

Transaction processing systems differ from timesharing systems, because the users are different and the nature of the processing is different. In a typical interactive system, the user has access to all the facilities of the computer:

- Its operating system,
- Language compilers,
- Utility programs, and so on.

In a transaction processing system, the user communicates only with selected applications programs. As illustrated in Figure 1-5, these programs display messages to help the user enter data and make processing requests. The application performs the transaction immediately and returns the results. In a transaction system, the user does not have to be a programmer to communicate with the data processor. However, a transaction system does need a computer expert to write the interactive applications programs.

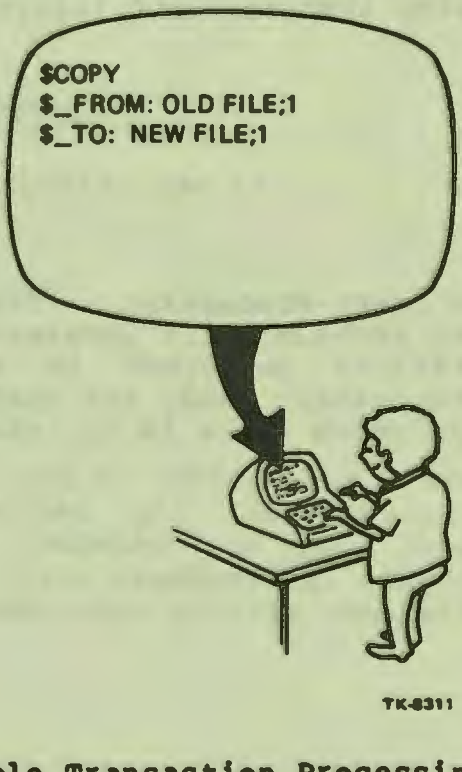


Figure 1-5 Sample Transaction Processing System Message

Query Processing -- Transaction processing systems can replace much of the record keeping previously done with paper forms. Once data is on-line, a query system can be used to obtain accurate, detailed information stored in the computer system.

In a query processing system, a non-programmer can define processing operations (that is, create applications for special jobs) by issuing instructions in a query language. There are two types of query languages:

- Statement-oriented and
- Form-oriented.

Statement-Oriented Query Languages -- In a statement-oriented query language, the user makes a request by typing phrases according to the language's rules of punctuation and syntax. For example, a sales executive who wants to open a Paris office can sit at a terminal and type:

GET ME THE NAMES OF ALL SALES PERSONNEL WITH A MINIMUM OF FIVE YEARS EXPERIENCE WHO SPEAK FRENCH.

Form-Oriented Query Languages -- In a form-oriented query language, the user designs the forms that the processor displays on the screen. Figure 1-6 illustrates a typical form-oriented screen display requesting that specific information be entered in the spaces provides:

- NAME
- ADDRESS,
- CITY, and
- STATE.

Uses of Transaction and Query Processing -- Transaction processing is for routine jobs (for example, file updates). Each transaction involves the same operations performed in the same sequence. Query systems, on the other hand, are useful for special data processing requests, for which there is no standard applications software.

TK-8312

Figure 1-6 Typical Form-Oriented Screen Display

DATA PROCESSING SYSTEMS

Data processing systems can be:

- Centralized,
- Decentralized, or
- Distributed.

Centralized Systems

In the simplest type of centralized system (Figure 1-7), a single computer handles all the processing for a complete organization. This computer can be programmed for batch processing or interactive timesharing. The first data processing systems used in business were greatly centralized.

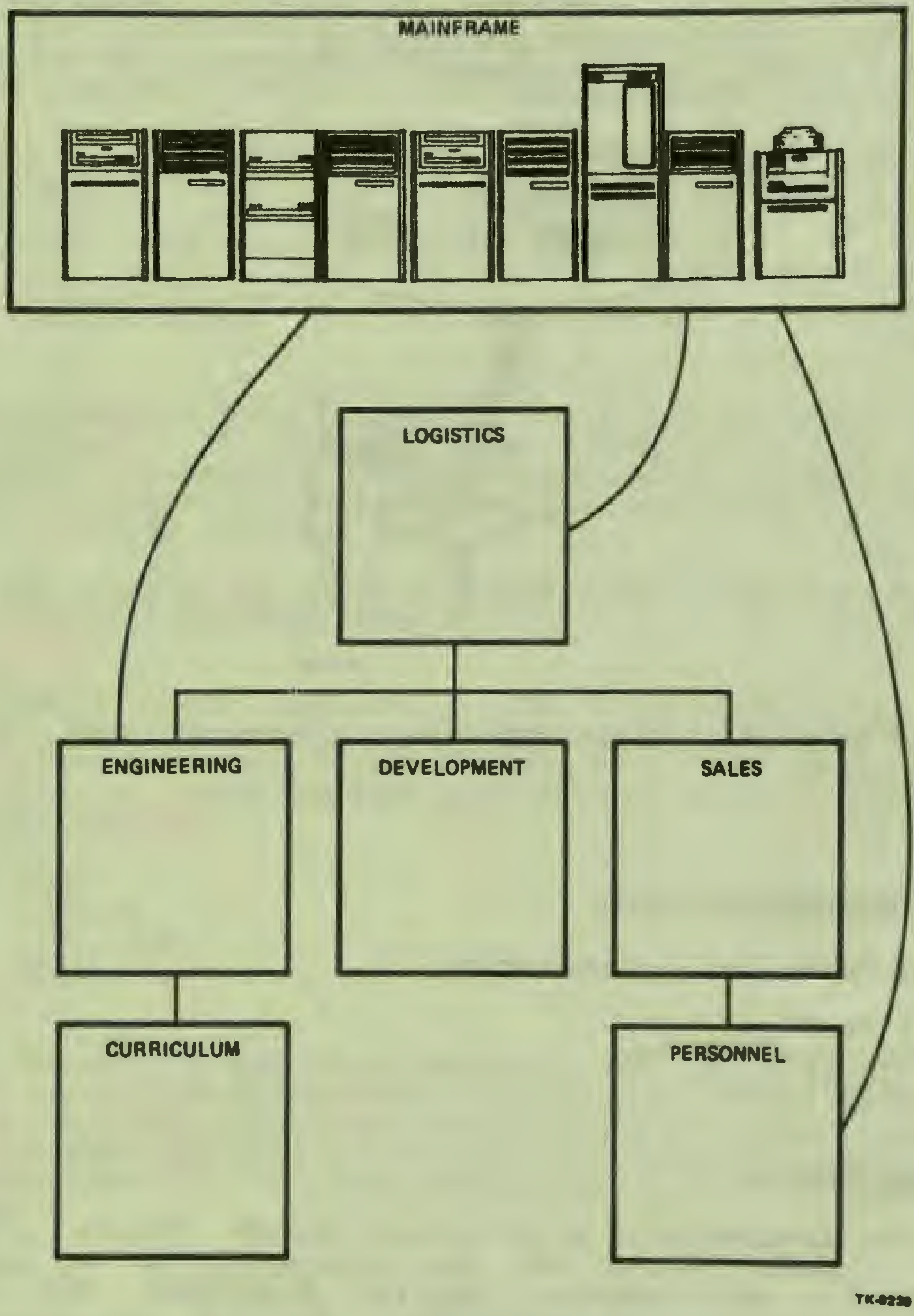


Figure 1-7 Centralized System

Decentralized Systems

The first decentralized systems (Figure 1-8) were created to loosen the restrictions of centralization. Departments were provided with computers and their own applications software. This made computing power much more accessible to the personnel who needed it in their jobs. However, different machines running different applications and creating different types of files often restricted each system to the department for which it was originally designed.

Distributed Systems

In a distributed processing system, as in a decentralized system, each department in an organization can be provided with its own computer and applications programs. In distributed processing (Figure 1-9), however, care is also taken to maintain compatibility among machines, software, and files. This makes it possible for computers in different departments to share their resources.

Some of the factors which are causing the trend to distributed processing include:

- Low cost processors
- Increasing sophistication of end users
- Need for higher availability
- High cost of phone lines
- Central control of remote development
- System interlinking
- Remote data base access
- Security
- Reduced CPU load
- Centralized and decentralized capabilities

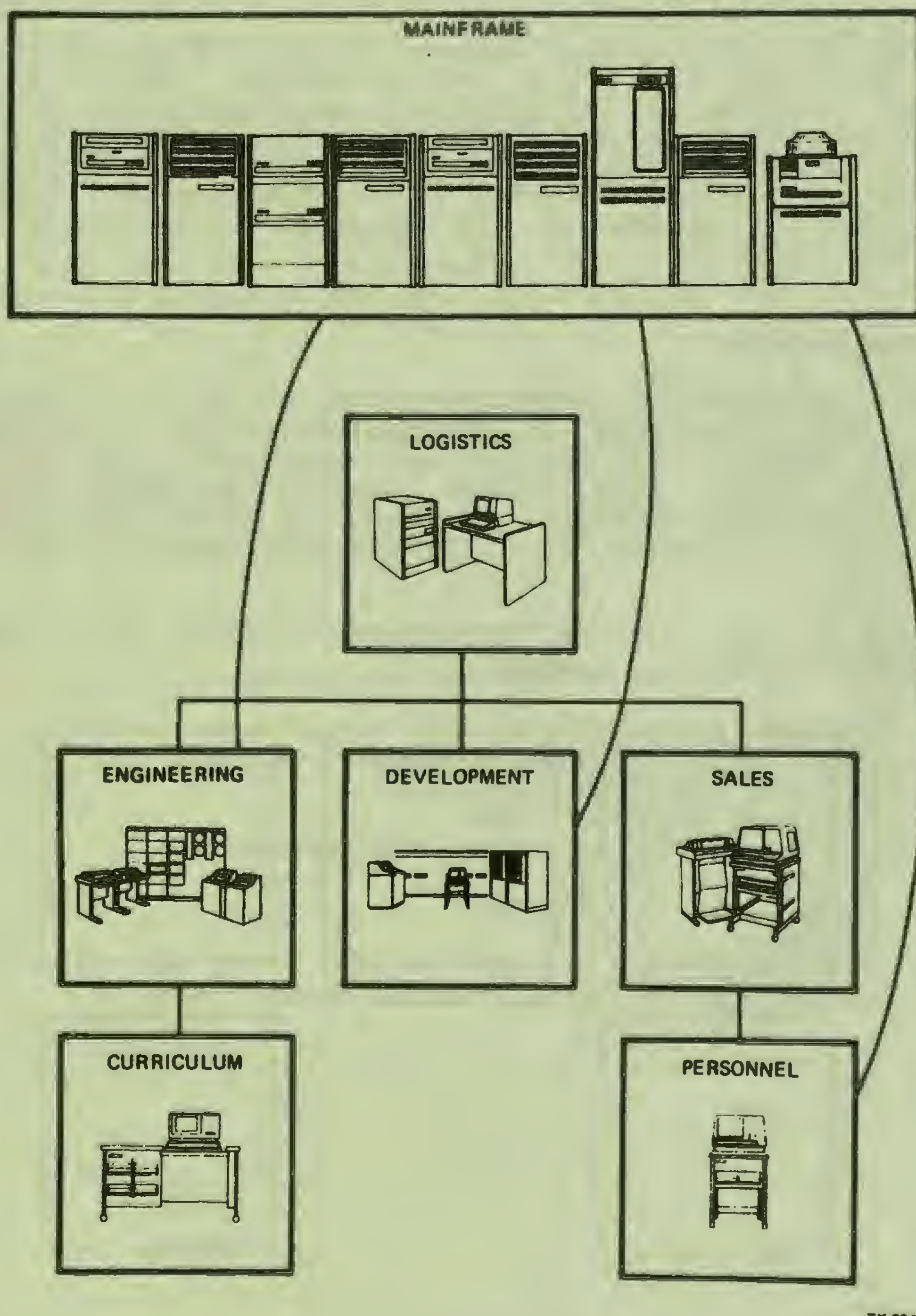


Figure 1-8 Decentralized System

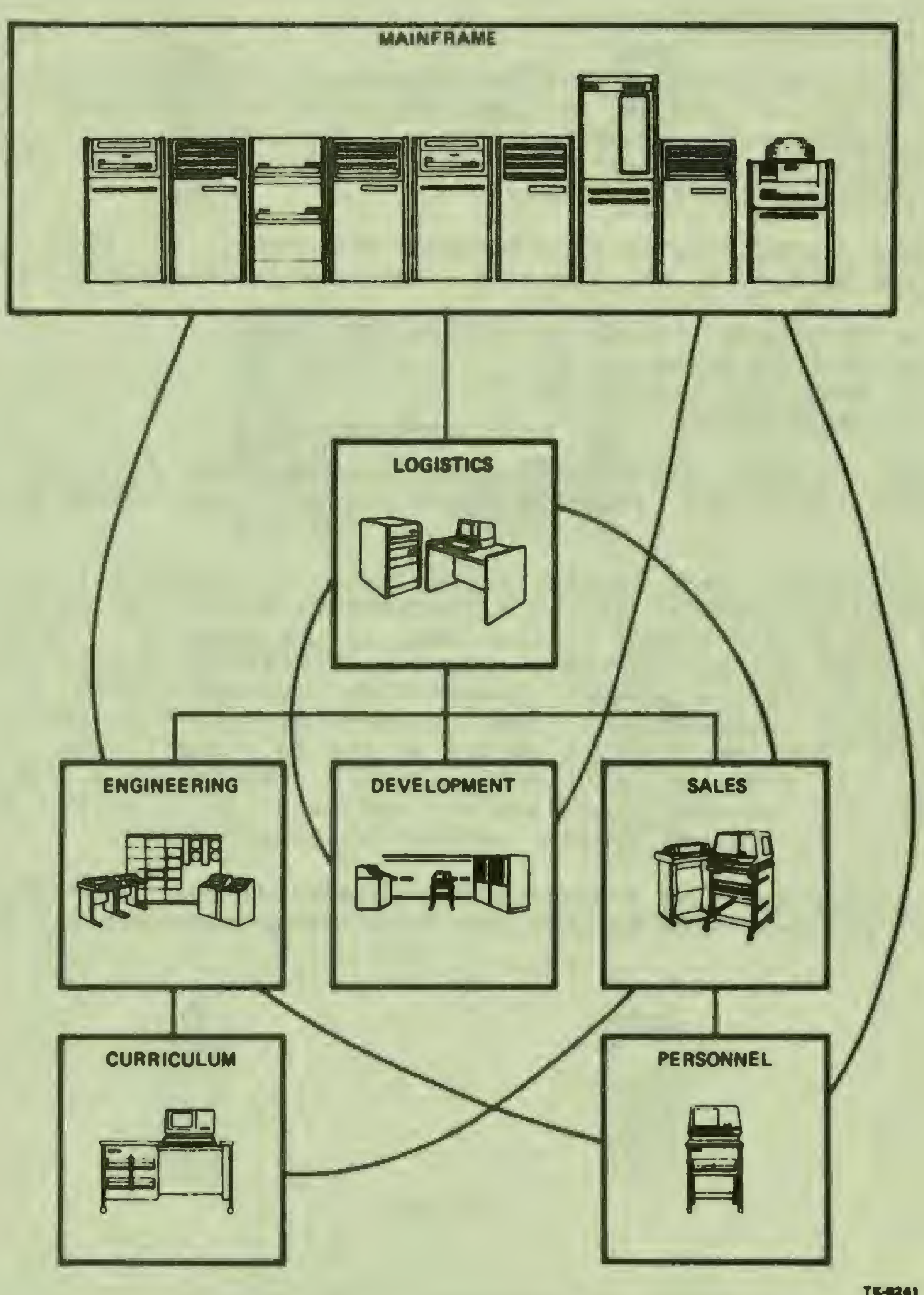


Figure 1-9 Distributed System

DATA COMMUNICATIONS

Data communication provides a computer the ability to access data from one system to another, using the different types of transmission media. Using data communication to transfer information is very similar to using a telephone, except that data is transferred instead of voice.

The fastest way to move data is to change it into signals that can be sent in one of the following ways (see Figure 1-10):

- Telephone lines,
- Coaxial cables,
- Microwave towers, or
- Satellites.

In most networks, electronic communications links (physical paths) are provided by the telephone companies and other common carriers.

Electronic data communications makes it possible to access an interactive computer system a thousand miles away as conveniently as if the terminal were located next to the computer. Being able to overcome distance makes a computer information system radically different from a manual information system. With a manual information system, you must either go to the data file, or transmit your request to a person at the file and count on that person to find the right data. With a computer in a distributed system, you can stay where you are and make use of files, in your system or in another system, without an intermediary.

A more detailed discussion of communications fundamentals and common carriers will be provided in later modules of this course.

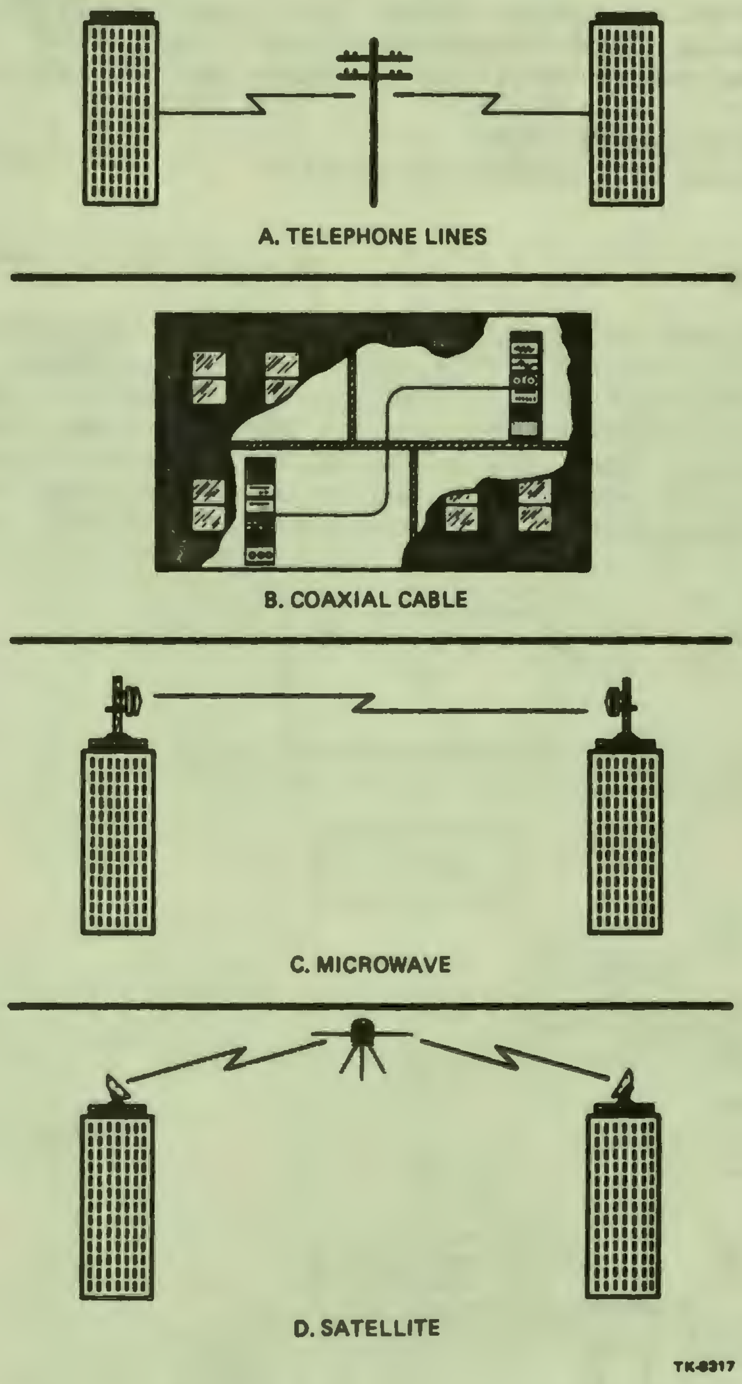


Figure 1-10 Data Communication Mediums

NETWORKS

A data processing system that utilizes available computer systems, terminals, and communications facilities is called a **network**. The two most commonly used networks of today are:

- Terminal Networks and
- Distributed Processing Networks.

Terminal Networks

In the simplest type of network (called a terminal network, see Figure 1-11), electronic links are used to connect remote job entry devices and interactive terminals to a central computer. A remote job entry device accepts batch data that has been prepared on cards or magnetic tape and sends it to a central processor. An interactive terminal enables a user at a remote location to engage in interactive processing with a central timesharing computer.

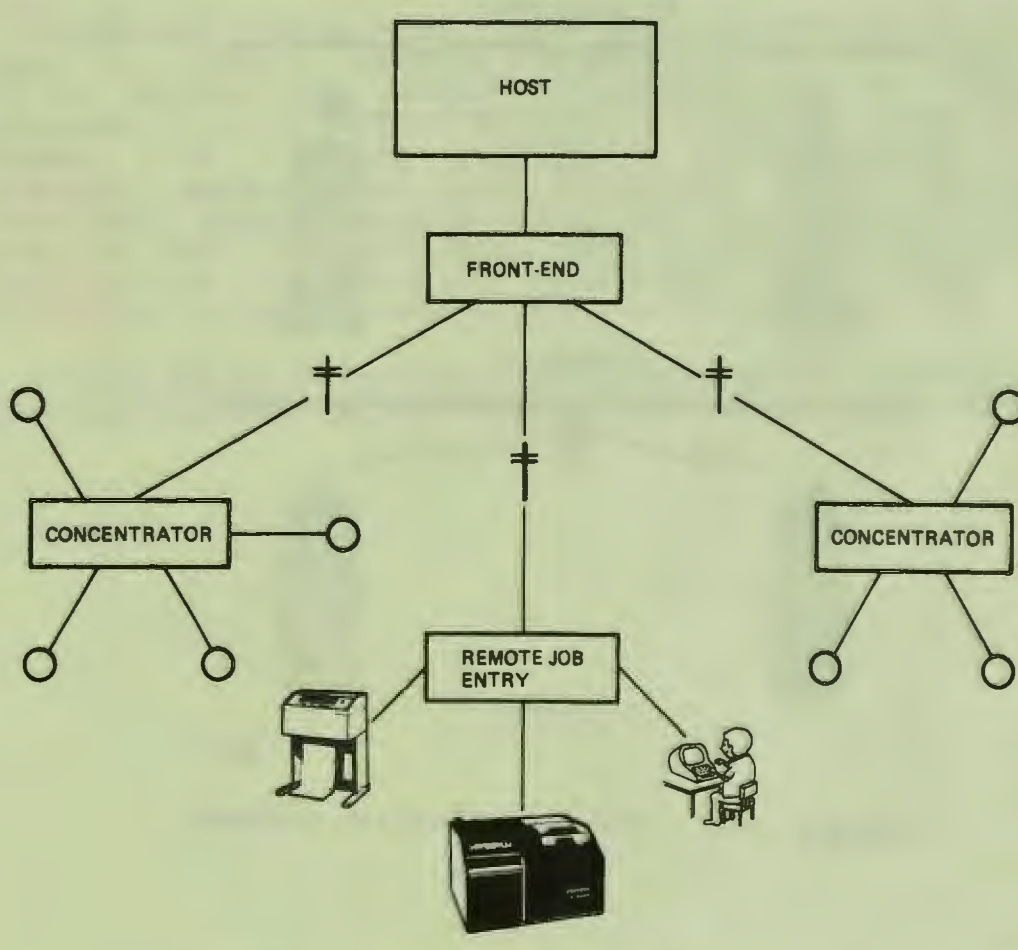


Figure 1-11 Typical Terminal Network

In a network, the central processor is called a network host. This processor, along with its data base and peripheral devices (external devices such as a tape drive or disk), make up a host **node**. A node can be defined as an end point of any branch of a network, or a junction common to two or more branches of a network. A computer or an intelligent terminal would be two examples of a node. In a later module, Network Fundamentals, terminal networks will be further defined.

Distributed Processing Networks

The computers in distributed processing systems (Figure 1-12) can be connected by electronic links to form distributed **processing networks**. Each processor (plus terminals and peripherals) is a node in the network. Users in different departments can use their computer for local processing and can also communicate with other nodes as needed.

Distributed processing networks may or may not have a central host node that controls communications. In a host-oriented network, the subsidiary nodes are usually referred to as **satellites**. In a network without a host, all nodes have equal status.

In the Network Fundamentals module of this course, ways that computers can be linked to create distributed processing networks will be described.

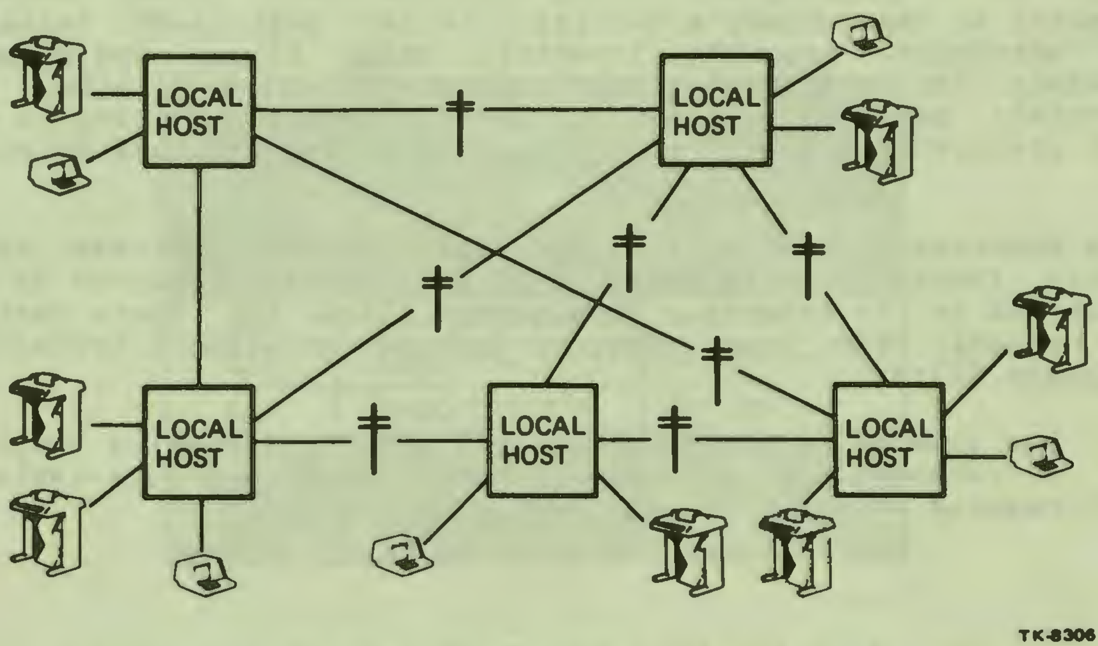


Figure 1-12 Typical Distributed Processing Network

Network Functions

One of the most important features of a network is its flexibility. A properly designed network will allow its users to link together their computers and peripherals in almost any manner they wish. Once a network is in place, new nodes may easily be added. Furthermore, the network and existing nodes may be upgraded or modified to meet current user needs without extensive reprogramming.

The capabilities of distributed processing networks include:

- Program-to-program communications,
- File transfers between systems,
- File accesses between systems, and
- Down-line loading of programs.

Program-to-Program Communications -- In program-to-program communications, a program running on one computer sends messages over a network link to a computer running another program.

File Transfers Between Systems -- In file transfers between systems, data stored at one site is copied and transferred to a storage device at another site, without communication between system programs.

For example, dealerships and warehouses can record inventory usage on a minicomputer during business hours. After hours, the computer at the company's central site can poll each dealership and warehouse, get the inventory usage files, and prepare a schedule for maintaining the inventory at each site. File transfers between systems are useful to organizations in which data prepared and stored at one site is needed at other sites.

File Accesses Between Systems -- File accesses between systems enable remote sites to make use of data bases maintained at other locations in the network. This method allows the remote users to get exactly the item they are looking for without transferring complete files.

For example, a ticket agent at a suburban shopping mall can use a terminal to determine which seats are available for performances at any downtown theatre.

Down-Line Loading -- Down-line loading is a method that enables a central processor to load executable programs (or complete systems) into remote computers or terminals that do not have the ability to store all the programs which may be needed at any one given time. With down-line loading, application programs can be created and modified at a central site and sent (down-line loaded) to remote sites for execution.

Data Storage

A particularly important function of a network is to provide access to remote storage units. This requires the ability for reading, writing or updating complete files and for reading, writing, deleting, and inserting single records.

All company information finds its way into some type of file. In data processing, a file is an organization of related data. In a personnel file, for example, the data is related by the fact that every entry in the file describes an employee of the same company.

As an example, in Figure 1-13 you will find a company file of employees organized according to employee number. Each record in the file contains all relevent data about the employee (as an example, marketing analyst Lester Young). Each record in our simple employee file has three fields (segments): a badge number field, a name field, and a department field.

BADGE NUMBER	NAME	DEPARTMENT
3209	YOUNG, LESTER	MARKETING
3236	PARKER, CHARLES	ACCOUNTING
3251	GORDON, DEXTER	PURCHASING

TK-6842

Figure 1-13 Typical Company File Of Employees

Many organization types are used in computer files. Two of the most common organization types will be briefly discussed:

- Sequential Organization and
- Indexed Organization.

Sequential Organization -- In a sequential file, records are listed in the order in which they are written into the file by the applications program. Each record, except the first, has another record preceding it. Each record, except the last, has another record following it. The physical order of the records is always the same as the order in which they were written (see Figure 1-14). A sequential file may be stored a number of different ways, such as on magnetic tape or on a disk.

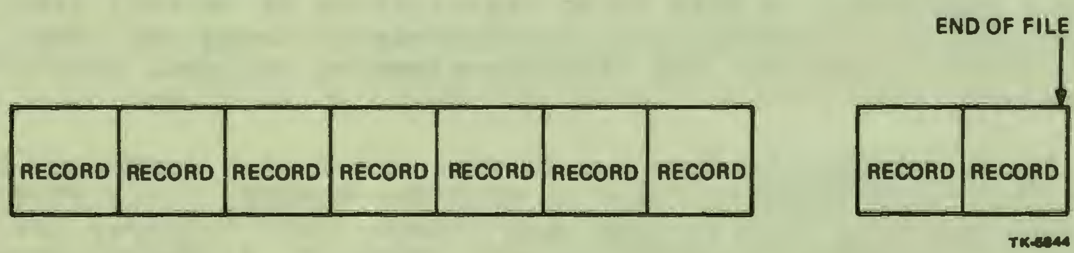


Figure 1-14 Typical Sequential File

Indexed Organization -- The physical ordering of records in an **indexed** file is transparent to the applications program. In indexed files created by DIGITAL's Record Management Services (RMS), for example, the RMS software has complete control over the placement of records.

As the application writes records into the file, RMS creates an index according to a key value in each record. This key value is a string of characters specified by the user.

In our personnel file, the key value is the employee number. As the application writes employee records into the file, the file software creates an ordered index of numbers. Each entry in the index includes a pointer to the actual record.

The user must define one key - the primary key. In addition, alternate keys may be defined. RMS creates an index for each key defined. Figure 1-15 shows a personnel file with three indexes. An indexed file can be stored on disk media only.

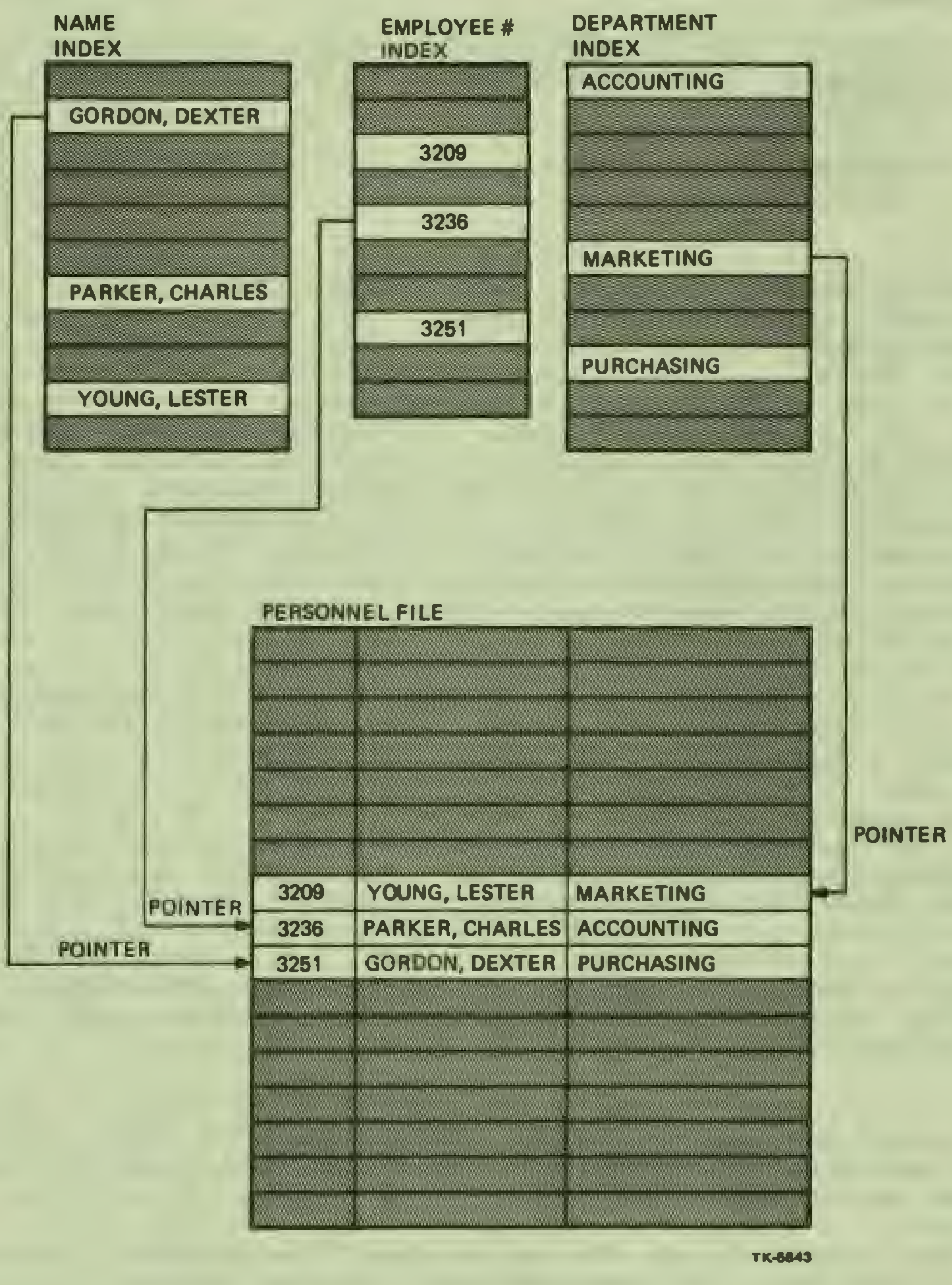


Figure 1-15 Typical Index File

Access Modes

The method used to get and store records in a file is called an access mode. Two access modes used in commercial applications are:

- Sequential access and
- Random access.

Sequential Access -- The user can get records in sequential mode from both sequential and indexed files. The user can also store records in order in both sequential and indexed files.

To read a certain record in a sequential file in sequential mode, for example record number 15, the application program must open the file and read through the first 14 records before it finds the record it wants. Once the program gets record 15, it can go on to read the remaining records. It cannot, however, read any preceding record, for example record 10, without first closing the file, opening it again, and starting at the beginning. To write a record into a sequential file in sequential mode, the user must go to the end of the file.

To read records in order from an indexed file created by RMS, the user specifies an index. RMS retrieves records in the sequence represented by the index. For example, if you specify the name index for the personnel file, RMS will read the records in the alphabetical order of employee names. To write records into an indexed file in sequential mode, the user must be sure that each record has a primary key value that is equal to or greater than that of the preceding record.

Random Access -- The user can get random access to records in an indexed file. Sequential files, however, cannot be accessed in a random manner.

To get a record from an RMS indexed file, the user specifies an index and a key value. RMS searches the index for the key value and then uses the pointer linked with the value to find and return the record.

To write a record into an indexed file, the user passes the record to RMS. RMS uses internally stored information to locate the key fields and update all indexes.

Interrelated Files

Each department in an organization creates its own set of files, setting them up in a way that best fits its applications. In many cases, a file can be used only for the application for which it is designed. As a result, if the same information is needed by other departments, the information must be repeated in as many files.

For example, Harry Edison, a customer of the First and Only Bank, has a checking account, a savings account, and an outstanding commercial loan. Harry's name, address, occupation, and other personal information appear in three separate application files: a checking account file, a savings account file, and a loan file.

To prevent the assembly of redundant information and the creation of files and applications which are not compatible, the data base can be designed so that there is a relationship among all files, records, and data fields. Recently, in fact, the First and Only Bank designed its data base, creating a master customer file. This file includes all the information common to the three earlier files, plus pointers to additional files which have information about customer checking accounts, savings accounts, and loans.

File Transfer

A facility which is built to conveniently copy data files (parts of the system data base) from one system to another is called a utility. This utility can be used for resource sharing by having one system invested with expensive, but cost-effective, large disk storage units. Intersystem file transfers can increase data reliability by periodically sending copies of all updated files to an archive system where special precautions (such as using fire-resistant storage vaults) are taken to protect the data.

The simplest file transfer utility may just send binary (ones and zeros) data images from one system to another (transfer a file as a sequence of bits). More sophisticated utilities may do translations from one format to another. For example, a character file (textual form data) could be translated from the character set of one computer system to the character set of another. Or an indexed record file could be transferred as records, and rebuilt into the indexed format of a different system.

Remote File Access

The next increase in sophistication beyond file transfer is the ability to access files located on a remote system. Normally a file is accesed by "opening" it and then performing a set of positioning, reading, and writing operations on it.

In the case of remote file access, opening a file consists of:

1. Determining which system had the needed file,

2. Establishing a communications session (message exchange) with a cooperating program (a file transfer utility) on that system, and

3. Performing an open function for the file on the remote system.

Having opened the file, the program writing or reading the file issues standard file-access commands (e.g., read, write, update). Instead of executing these commands, the system on which the program is executing transmits these commands to the system at which the data is located, via the communications network. System-provided utility programs access the remote file as requested, using the communications network to transfer data between file and program.

DATA SECURITY

A data processing network that can transmit data from one computer to another and store it in data bases must also be capable of preventing unauthorized access to the information it handles.

Two types of security controls are used to protect data against unauthorized access both from within the organization and from without. These are:

- Access controls and
- Cryptographic controls.

Access Controls

Access controls determine which files a network user can read, edit, or delete. This can be done by assigning clearance levels to users and by including a protection code number as part of each file specification. The code specifies the set or subset of users who can gain access to the file. It can also specify the type of access allowed:

- Read-only access or
- Read/write access.

Encryption

If a network is unable to protect the transmission medium against wiretappers, data can be encrypted; that is, put into code so that it cannot be understood except by those who have a key.

This method requires an enciphering and deciphering mechanism for both the sender and the receiver (for example, an LSI chip which has the encryption algorithm). In addition, the sender and receiver are issued one or two keys, depending on the type of encryption used. A key is a sequence of bits used by the encryption algorithm to translate a message into a code, or to turn the code back into the original message. Keys can be changed often to stop code breakers.

EXERCISES

Label the following statements True (T) or False (F).

___ 1. Distributed processing is a lot more than just the equipment that is used to do the work.

___ 2. Distributed processing is the technique of using large-scale computers, minicomputers, and microcomputers in business, each one doing the job it does best.

___ 3. A distributed processing system may include both interactive and batch data processing.

___ 4. An indexed file can be stored on either a disk or a magnetic tape, but not on punched cards.

___ 5. Transaction processing needs interactive communications between the user and the computer.

___ 6. An indexed file can have only one index.

___ 7. A Program-to-program communication is possible only in centralized networks.

___ 8. The fastest way to move data is through the U.S. mail.

___ 9. A data base can have many related files.

___ 10. Down-line loading is a method used to send large data files over the network.

___ 11. As increased data processing needs arsie in an organization, there is an increasing need to interlink separate systems.

___ 12. A field is a segment of a data record.

___ 13. A remote job entry device enables a user to transmit data to a batch processor.

___ 14. A network with distributed processing may still maintain a centralized data base.

___ 15. Batch processing is used for operations that require fast results.

___ 16. File transfers between systems enable a network user to prepare data at one location and store it at another.

___ 17. A query language can be used only by programmers.

___ 18. Remote file access goes beyond the abilities of simple file transfers, in that it provides the ability to access files located on a remote system.

SOLUTIONS

Label the following statements True (T) or False (F).

T 1. Distributed processing is a lot more than just the equipment that is used to do the work.

T 2. Distributed processing is the technique of using large-scale computers, minicomputers, and microcomputers in business, each one doing the job it does best.

T 3. A distributed processing system may include both interactive and batch data processing.

F 4. An indexed file can be stored on either a disk or a magnetic tape, but not on punched cards.

T 5. Transaction processing needs interactive communications between the user and the computer.

F 6. An indexed file can have only one index.

F 7. Program-to-program communication is possible only in centralized networks.

F 8. The fastest way to move data is through the U.S. mail

T 9. A data base can have many related files.

F 10. Down-line loading is a method used to send large data files over the network.

T 11. As increased data processing needs arise in an organization, there is an increasing need to interlink separate systems.

T 12. A field is a segment of a data record.

T 13. A remote job entry device enables a user to transmit data to a batch processor.

T 14. A network with distributed processing may still maintain a centralized data base.

F 15. Batch processing is used for operations that require fast results.

T 16. File transfers between systems enable a network user to prepare data at one location and store it at another.

F 17. A query language can be used only by a programmer.

T 18. Remote file access goes beyond the abilities of simple file transfers, in that it provides the ability to access files located on a remote system.

2

NETWORK FUNDAMENTALS

INTRODUCTION

A network consists of a configuration of network manageable components connected by communication paths. In a data processing network, the network manageable components are called nodes. Nodes may be made up of different things, such as processors and their software, and intelligent terminals.

Network nodes may be connected by various types of electrical data tranmission media - telephone lines, privately strung wires, coaxial cables, microwave towers, or satellites.

For as many different business applications there are as many different networks configurations. The reason for different types of networks is due to the different needs of users.

In this module, we will examine nodes and lines and see how they can be arranged to create different network configurations called topologies.

OBJECTIVES

Define and illustrate the following network concepts:

- Host and remote nodes
- Front end processing and remote concentration
- Point-to-point and multipoint fixed lines
- Contention and polling procedures for multipoint lines
- Common switching techniques
- Common network topologies

RESOURCES

1. Computer Networks, Andrew S. Tanenbaum, Prentice-Hall, 1981.

2. Computer Network and Distributed Processing, James Martin, Prentice-Hall, 1981.

3. Design and Strategy for Distributed Processing, James Martin, Prentice-Hall, 1981.

4. Distributed Systems Handbook, Digital Equipment Corporation, (EB-P1195).

NODES

A node is a processing system, a combination of computer hardware and software, that has the ability to communicate over a network line. Nodes differ greatly in size and computing capability. All nodes, however, must be capable of accepting messages from other nodes, transmitting messages to other nodes, or doing both. Nodes handle both data processing and communications processing.

Data processing includes all the applications that may be developed for organizations such as:

- Payroll,
- Inventory control,
- Information collecting, and
- File keeping of all types.

Communications processing includes all the operations needed to transmit data over network lines (for example, message formatting, error checking, polling, routing and addressing, and automatic dialing and answering). We will examine communications processing operations later in this course.

Nodes can provide different processing needs in networks such as:

- Host processing,
- Remote processing,
- Front end processing, and
- Remote concentration.

Host Processing

If a network positions most or all of its data processing and communications processing at one central node, the main computer at that node is referred to as the host. The host computer controls data bases and generally monitors the network.

As defined in the previous module, the simplest type of host-oriented network is called a terminal network. A terminal network normally consists of a cluster of remote terminals connected to a single host node. Terminals in this environment may include remote job entry devices, used for batch processing, and interactive terminals connected to a timesharing host (see Figure 2-1).

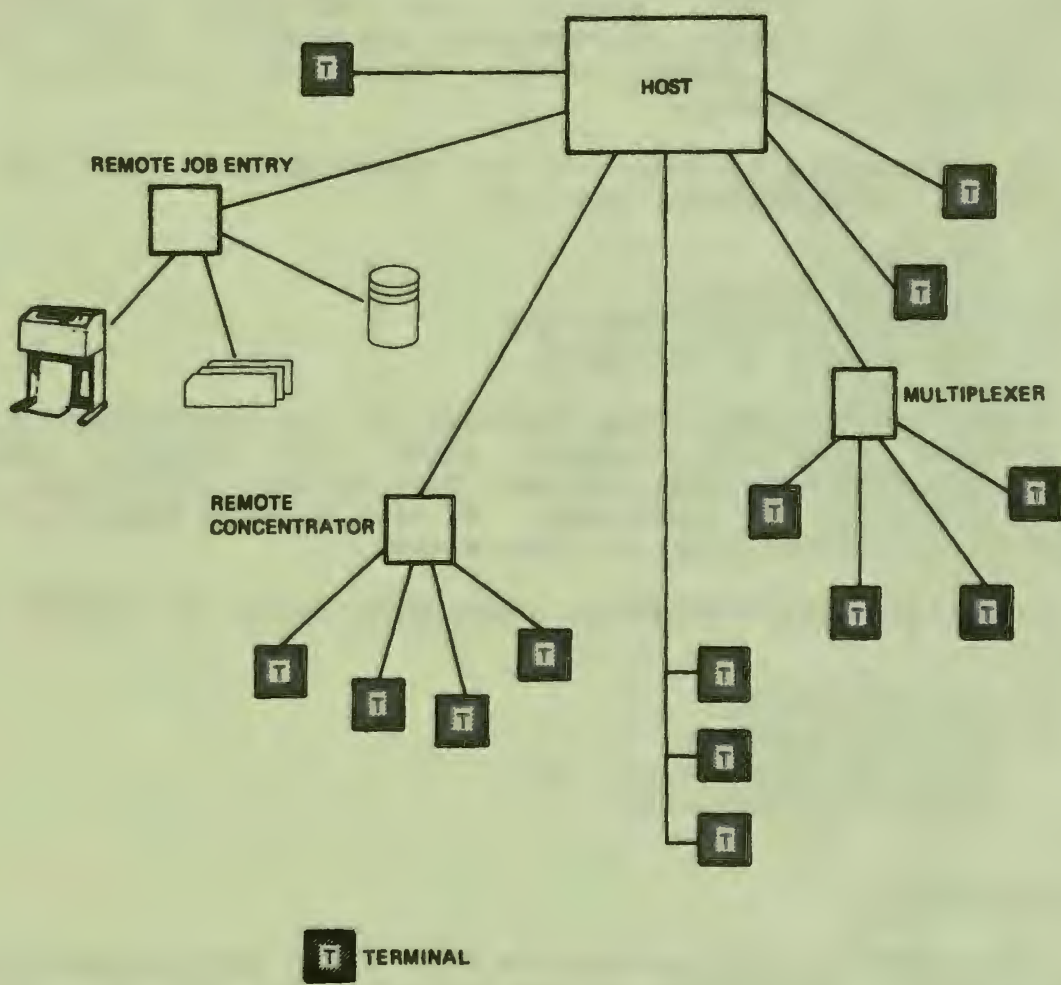


Figure 2-1 Host-Oriented Network

Note that by our definition, a terminal alone is not a network node, since a terminal has no processing power of its own. However, terminals which distribute the access to processing power are important components in all types of networks. Many special purpose terminals have been designed to handle specific types of business transactions such as:

- Banking,
- Supermarket checkout,
- Retail sales,
- Hotel reservations and registration,
- Airline reservations,
- Hospital admissions, and many more.

Distributed processing networks may also include host nodes. These host nodes may control network communications, programs, peripheral devices, data and other resources for a number of remote computing nodes.

Remote Processing

A **remote processing** node is a stand-alone computing system. For example, a minicomputer-based business system with a video terminal and peripheral devices is a stand-alone computering system that is connected to a network (see Figure 2-2).

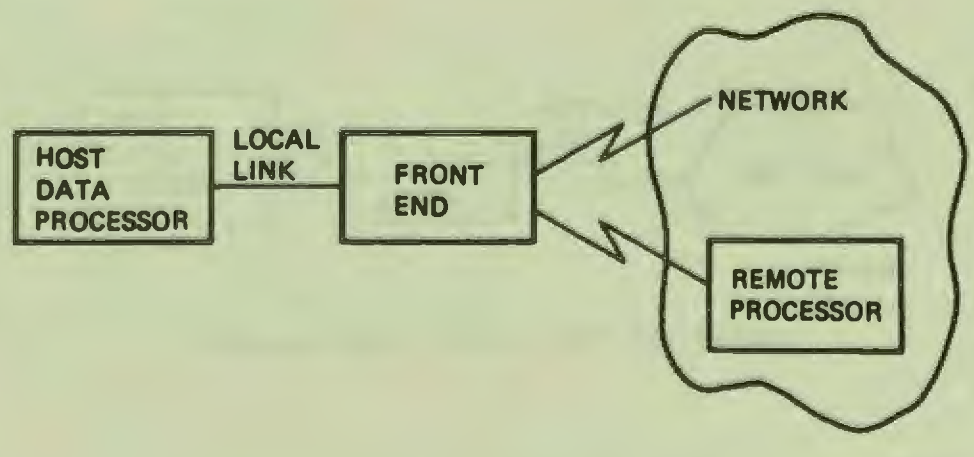


Figure 2-2 Remote Processing Node

In a host-oriented network, remote processing nodes handle data processing operations that would otherwise have to be performed by the host. The user can execute applications programs, maintain a local data base, and use the network to communicate with the host processor or other remote nodes as needed.

In many networks, remote computer systems can directly access the data base of the host computer. This puts an important resource in the hands of the remote user and greatly expands the range of applications available.

Remote processing nodes can also be used as remote job entry stations, thus enabling the remote user to send batch jobs to the host computer for processing and to accept output from the host.

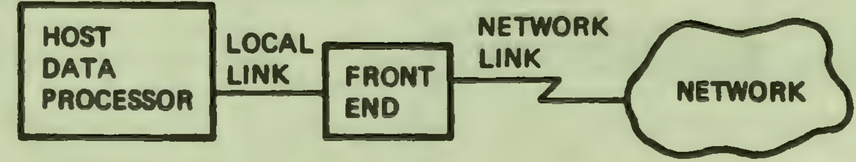
Remote job entry stations have usually been non-programmable hardwired devices. More and more, however, minicomputers are performing functions usually done by a hardwired remote job entry station. The operator can enter data on a video terminal, store it on a disk or tape, and transmit it in batches to the host. Data coming into the system can be similarly stored. When it is not performing as a remote job entry station, the computer can be used for other types of processing.

Front End Processing

Front end processing is used to relieve the host processor of communications functions. A front end processor can be a specially designed hardwired programmable processor or a general-purpose minicomputer. This type of processor can take over part or all of the communications tasks, leaving the host free to use its time and memory for data processing. By definition, a front end is connected to the host computer by a local line (see Figure 2-3).

Since front end processors can be configured with their own peripherals, they are capable of storing a portion of the system load. This can be a distinct advantage during peak load situations and can help smooth system throughput. The programmable front end can also be configured as a standard data processor. When not functioning in a communications mode, it can be used to perform data processing.

Because of its flexibility, the front end processor can easily be adapted to handle future network modifications. New system inputs, generated from changes in the number of terminals supported or from higher traffic loads, can easily be handled by programmable front ends. The ease with which such changes can be made relates to the overall system being broken down into more manageable, functional segments.



TK-5668

Figure 2-3 Front End Processor

Remote Concentration

Remote concentration is a procedure for decreasing the number of communication lines needed to connect a cluster of remote terminals to a host. A remote concentrator is a front end processor that has been removed from the host node and set up in the network as an independent control node. The concentrator accepts messages from many terminals over low-speed lines and transmits them to the host in a steady flow over a single high-speed, long-distance line (see Figure 2-4).

Because of its inherent stored program flexibility and modular expandability, the general-purpose minicomputer offers significant advantages when used as a remote concentrator. It has sufficient storage capacity and processing power to perform applications-oriented processing in addition to message receipt and transmission. In inquiry/response systems, for example, the minicomputer concentrator can be used to receive inquiry messages from remote terminals, process them to determine the specific information required, retrieve the information from on-line random-access disk storage units, and send it back to the inquiring terminal.

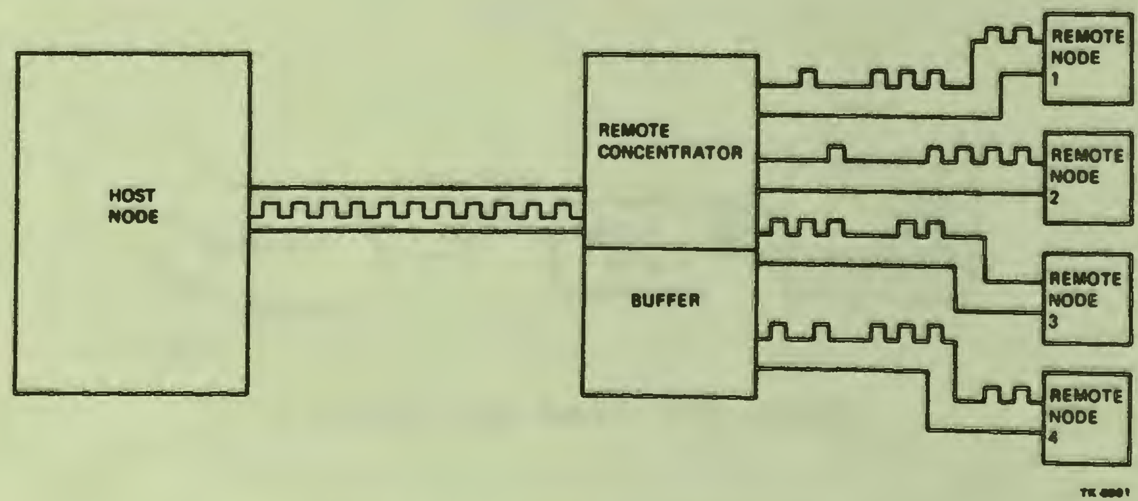


Figure 2-4 Remote Concentrator

EXERCISES

In the spaces provided, match the word or phrase to the correct definition. Each word or phrase may be used once, more than once, or not at all.

A. Front end processor

B. Controller

C. Remote terminal

D. Host central processor

E. Terminal node

F. Minicomputer-based business system

G. Modem

H. Remote concentrator

I. Remote job entry device

___ Connected to the host processor by a local line.

___ Relieves host computer of communications processing.

___ Handles processing for remote nodes.

___ Distributes access to host processor.

___ Transmits data for batch processing at the host node.

___ Takes in data from remote nodes and transmits it to the host over a fewer number of physical lines.

___ Provides the remote node with local processing power.

SOLUTIONS

In the spaces provided, match the word or phrase to the correct definition. Each word or phrase may be used once, more than once, or not at all.

A. Front end processor

B. Controller

C. Remote terminal

D. Host central processor

E. Terminal node

F. Minicomputer-based business system

G. Modem

H. Remote concentrator

I. Remote job entry device

__A__ Connected to the host processor by a local line.

__A__ Relieves host computer of communications processing.

__D__ Handles processing for remote nodes.

__C__ Distributes access to host processor.

__I__ Transmits data for batch processing at the host node.

__H__ Takes in data from remote nodes and transmits it to the host over a fewer number of physical lines.

__F__ Provides the remote node with local processing power.

LINES

Data processing networks use a wide selection of transmission media. In small in-house networks (refer to Figure 2-5), wires can be used to connect a host processor with interactive terminals in each department. For networks between cities and between states (refer to Figure 2-6), the telephone companies and other common carriers supply both public-switched and private leased lines.

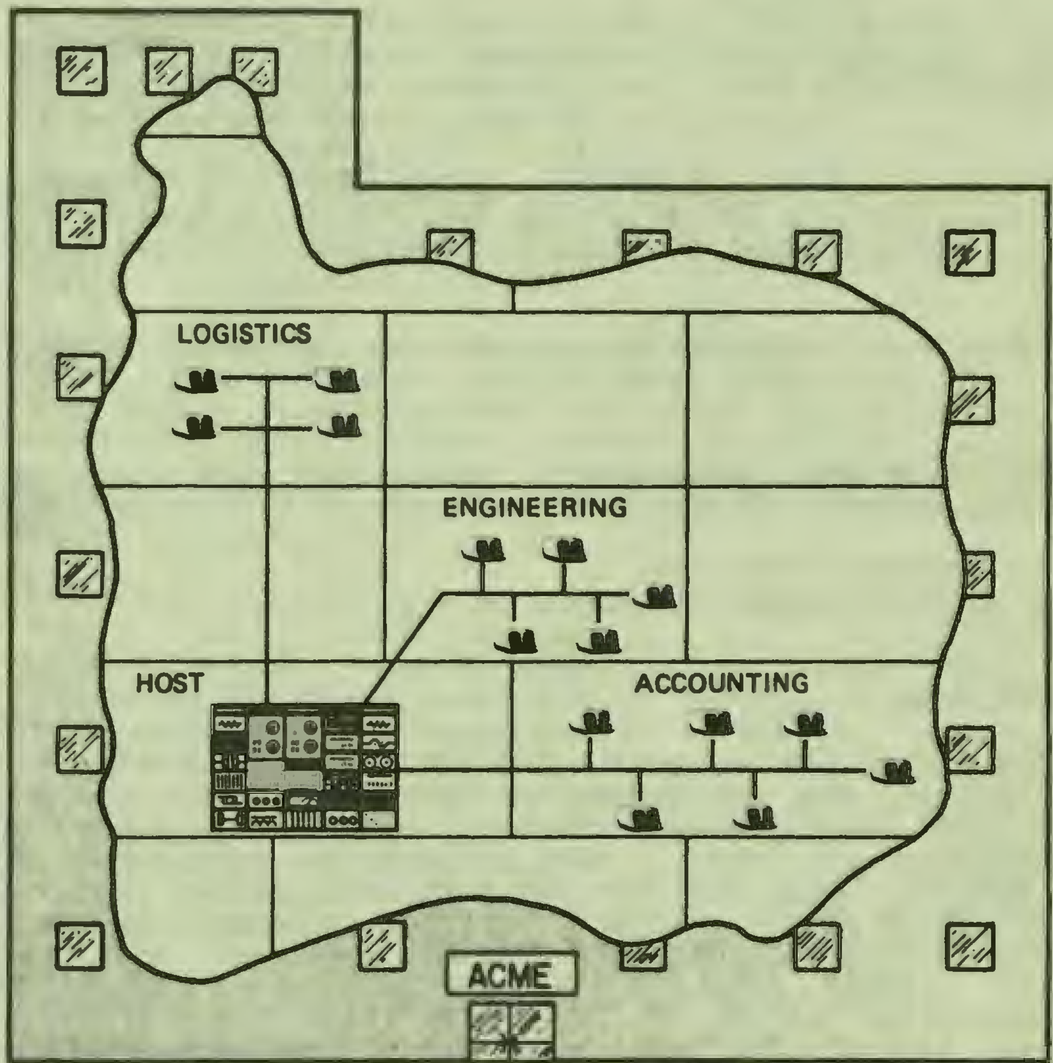


Figure 2-5 Typical In-House Network

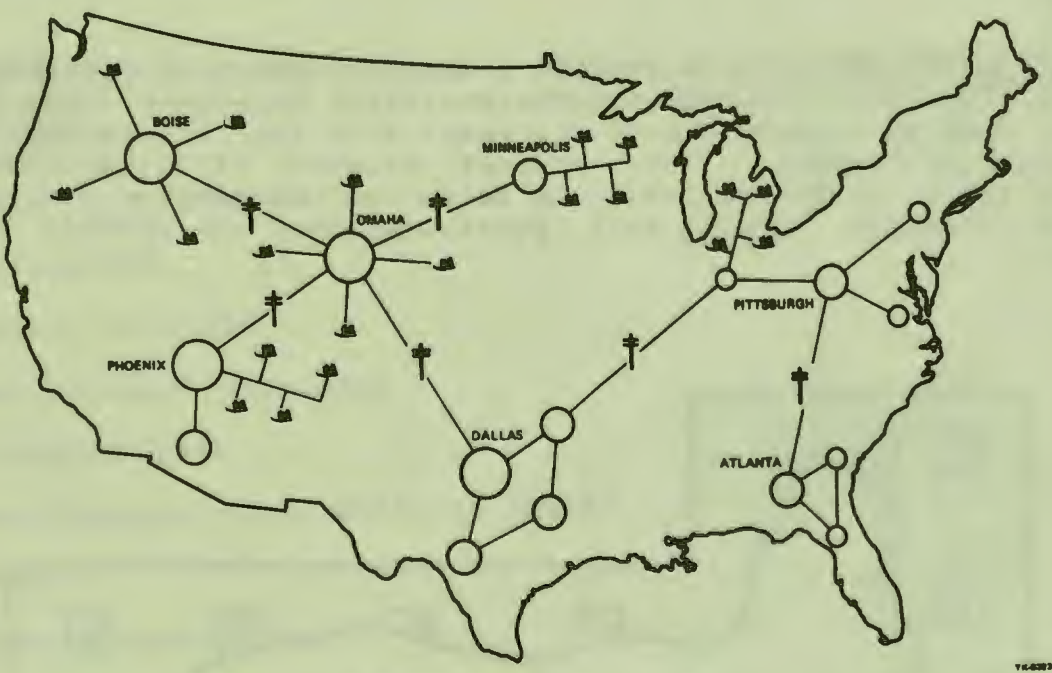


Figure 2-6 Typical Network Between Cities and States

No matter what transmission medium is used, the physical lines that connect the nodes in a network fall into two groups:

- Switched lines and
- Private leased lines.

Switched Lines

A **switched line** creates a temporary path between two nodes. The simplest way to accomplish this is by using the telephone system (or other common carrier facilities which provide communication services; usually used in conjunction with the phone system). When one node dials another node's phone number, the telephone system switching equipment selects a set of linking circuits (switches connect the circuit together) and creates a complete circuit from the dialing node to the node being dialed. This connection exists only for the duration of the transaction or for the transmission of batch data. The next time the nodes communicate, the message may be switched over a different route. Switched circuits can be used with telephones for voice communication, or with modems (special communication devices used for transmitting and receiving of data) for digital communication.

A switched connection has important advantages:

1. **Flexibility.** As long as both ends have compatible modems, no specific planning is needed.

2. **Cost-effectiveness.** The common carrier facility charge is usually based on the duration of the call. By dialing the call when data communications is needed, good usage can be made of the service paid for.

Significant disadvantages also exists when utilizing switched connections:

1. **Predictability.** The phone system is designed to complete a call by whatever means are possible when the call is made. There is no guarantee that a call will be completed with the same circuit on each call.

2. **Quality.** The quality of a specific call is particularly hard to predict. Not only will the circuits used vary, but the number of switches needed and the quality of the switching equipment will vary.

3. **Design Mismatch.** The telephone system was optimized for voice communications, not for data communications. Since most voice communications are only a few minutes in length, lengthy data communications (a timesharing session, for example) can impact overall phone service by tying up facilities for much longer than is likely with a voice communication.

   The converse is also true: optimizations made on the basis of voice properties can impact digital communications. For example, when people talk, most conversations have one person talking at a given time, with slight pauses (fractions of a second) between one person talking and the next. The phone system makes use of this on some connections, and turns the two-way link into a one-way link, providing communications in only one direction at one time. This is a minor inconvenience for voice communications, but a total disaster for many data communications.

Networks use three types of switching techniques:

- Circuit switching,
- Message switching, and
- Packet switching.

Circuit Switching -- Telephone systems use circuit switching (see Figure 2-7). In circuit switching, a person or terminal places a call by entering into the switching system the directory listed number of the person or terminal to be called; the switching system then sets up a connection. Every piece of information entered at the calling point is immediately conveyed to the called point. Furthermore, the connection between the two points is used solely by the two communicating parties. At the end of the communication, the connection is broken.

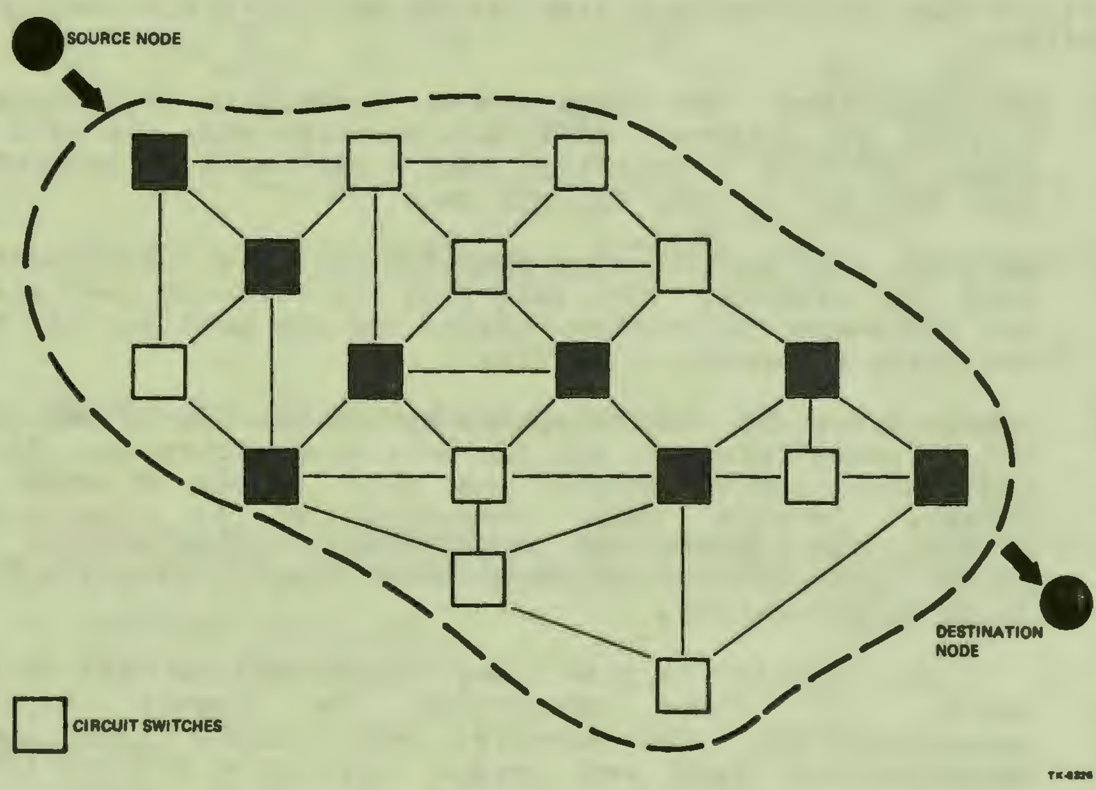


Figure 2-7 Circuit Switching

Traditional circuit switches connect different wire-pair paths by moving electromechanical contacts. Because this "old-fashioned" switch is made by electromechanical parts, it is very slow and may take seconds to set up. New solid state switches are currently being incorporated by the phone company. These newer switches can create connections in milliseconds or less. Also, due to the fact that they are solid state rather than electromechanical, they create less noise on the line which can disrupt communications.

Message Switching -- In message switching (as in packet switching) there is no preassigned end-to-end path for data to follow. Instead, each message is given an address at the source node. Switching nodes in the network, usually including minicomputers, use the address to create a path dynamically while the message is on the way.

In a message switching network, each message moves as a complete unit. Each switching node stores the message temporarily before sending it on the next step of its trip. This method is commonly referred to as the store-and-forward routing (see Figure 2-8).

Note that the route a message follows is determined by the current state of the network lines. Two separate messages moving between the same pair of nodes may use completely different paths.

Note, too, that because each message is copied to a storage device one or more times, transmission can be slow. As a result, message switching is good for noninteractive data communications only (for example, to transfer a large block of data from a remote node to a central host for batch processing).

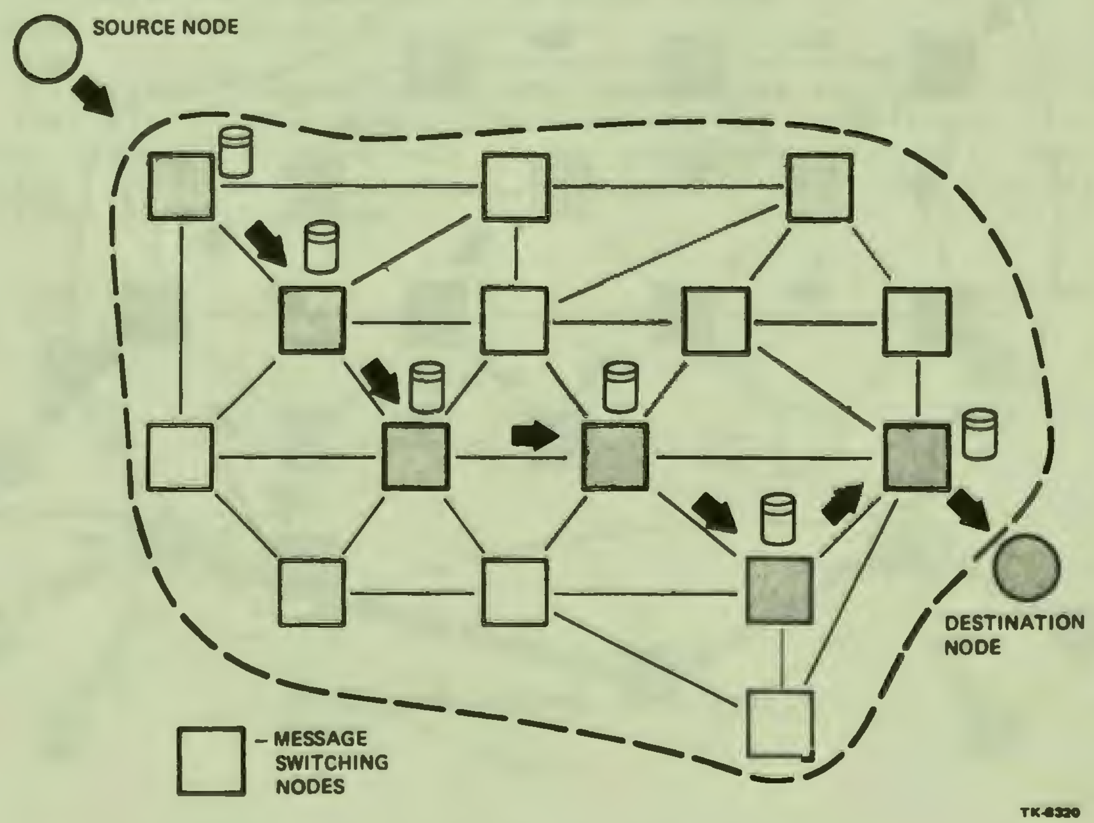


Figure 2-8 Message Switching

Packet Switching -- Packet switching is a high-speed form of message switching good for interactive transaction processing. With this method, a maximum message length is specified; for example, 128 characters. Longer messages are broken up into 128 character segments called packets. Each packet has a header which includes its position in the original message.

The sending node transmits packets to the first switching node in the network. This node may hold the packets for a short time, less than a second. Its purpose is to move each packet it receives as fast as possible by the best route available.

The process is repeated through as many switching nodes as necessary until all packets arrive at their destinations. Packets may move by different routes and arrive in a random order. Using the sequence number in the header of each packet, the destination node reassembles the message (see Figure 2-9).

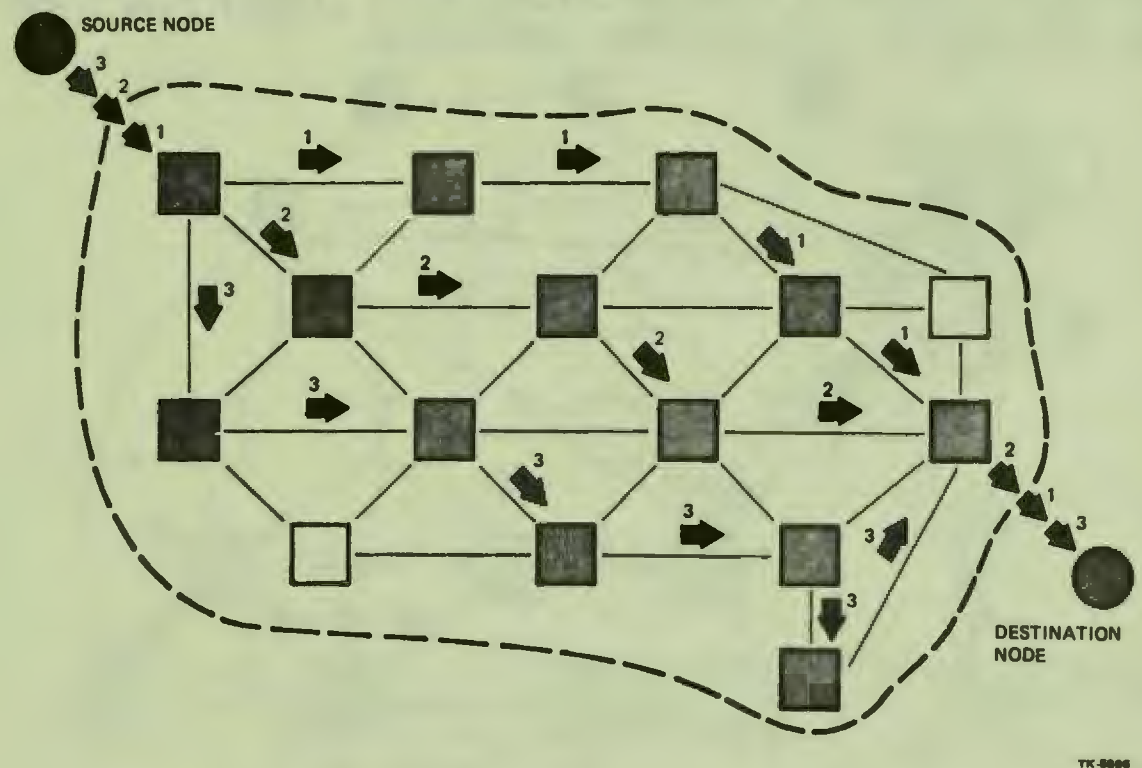


Figure 2-9 Packet Switching

This type of packet switching, in which the packets follow different routes through the network, is called datagram switching. In another form, called virtual call switching (Figure 2-10), the first packet determines the route that all packets will follow. This method, which combines features of packet switching, message switching, and time-division multiplexing, makes efficient use of the line, since packets from different sources can be interleaved in transmission.

Private Leased Lines

A private leased line (also referred to as a dedicated line), is a permanent fixed line providing a physical communications path between two or more nodes. Each time one node transmits a message to another node, the data moves by exactly the same route. Rather than using a different arbitrary switching path on each connection, the line is wired down. The advantage of a private leased line is the predictable quality. Most of the problems introduced by the switching equipment are eliminated.

With a private leased line, the telephone system provides additional signal quality guarantees at an additional cost (line conditioning). The quality of a conditioned line is not only predictable; it is predictably good.

The disadvantage of a private leased line is that you lose the flexibility and cost-effectiveness of a switched line. A private leased line goes only where you have planned, and the planning must be done some time in advance. And you pay for full-time use of the line.

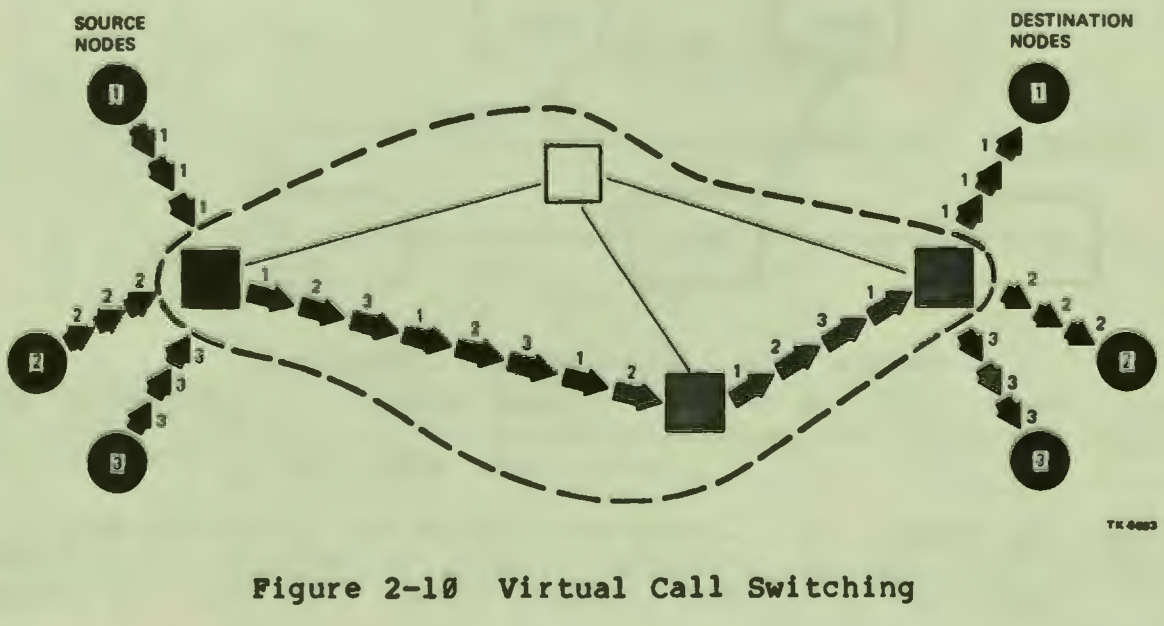


Figure 2-10 Virtual Call Switching

TOPOLOGIES

The form a network takes as nodes are connected together is called a **topology**. Most topologies fall into one of the following categories:

- Point-to-Point,
- Multipoint,
- Centralized star,
- Hierarchical tree,
- Ring, and
- Distributed mesh topology.

Point-to-Point

The simplest possible network topology structure is that of a **point-to-point** connection shown in Figure 2-11, in which two nodes are connected by a single communications link. The nodes may be terminals or processors. Whenever either node has data to transmit, the link is available. The link between nodes may be fixed as a private leased line or switched as on the dial-up (telephone system) network. If the dial-up method is used, the receiving node has to be available to receive in order for transmission to take place. If a busy signal exists, the sending node has to wait before sending its message across the link.

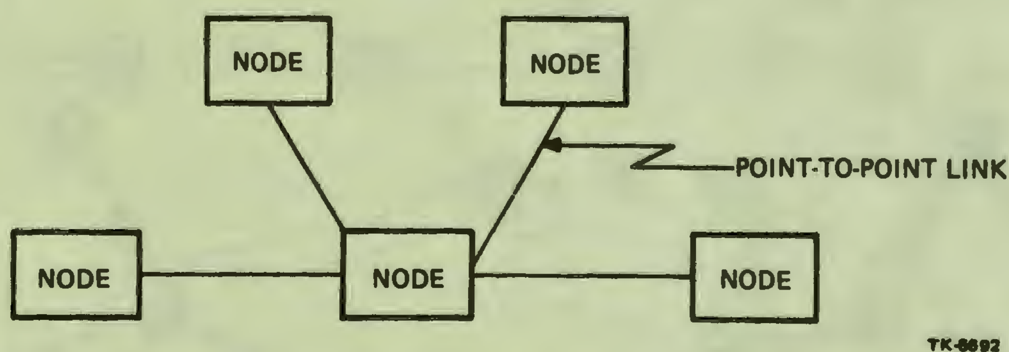


Figure 2-11 Point-To-Point Topology

Multipoint

In a multipoint topology, several remote nodes, often called tributaries or satellites, share the same physical line with one node designated as the control (see Figure 2-12).

Multipoint lines raise two problems not found in simple point-to-point configurations. How does the master node specify a particular tributary as the destination of a message? And, how do tributary nodes share the same line?

The first problem is solved by giving each remote node on a multipoint line a unique station address. The master node addresses each message it sends out. Each remote node constantly monitors the line for messages addressed to it. A node will take in messages that arrive with its address, and ignore all others. In some networks, remote nodes also provide a general broadcast address in addition to their unique address, enabling the master to send general messages to all nodes with a single transmission.

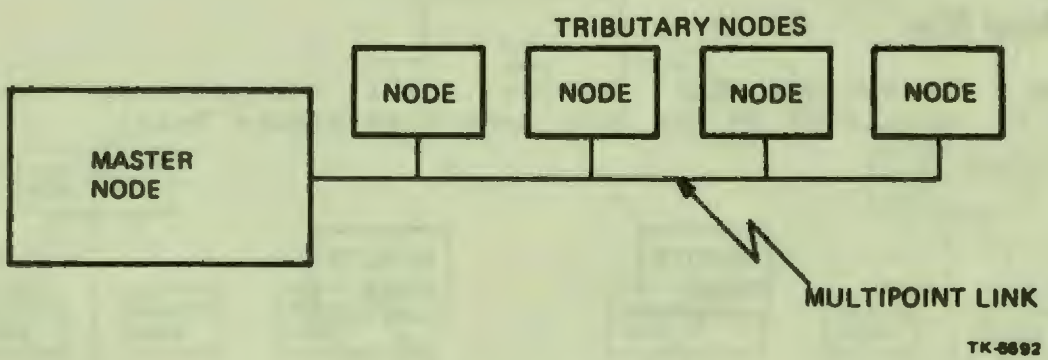


Figure 2-12 Multipoint Topology

There are three procedures for dealing with the problems of line sharing. These are:

- Line contention,
- Roll-call polling, and
- Hub polling (or token passing).

Line Contention -- Line contention is based on "first come, first served." If a node tries to send a message but finds the line in use, it gives up and tries again later.

This method is the simple solution. It needs no special software or processing time. However, it provides no way to prevent one node from taking complete control of the line.

Roll-Call Polling -- Roll-call polling solves this problem by giving control of the multipoint line to the master node. The master node **polls** each tributary node in turn, using a table stored in memory. If a node has no message to transmit, it gives a negative response. If one node usually has a large volume of data to send, its address can appear in the table more than once so that the host polls it more often than the other nodes.

Hub Polling -- Roll-call polling has one main problem: its query-and-response method keeps the host in constant dialog with the remote nodes. A modification of roll-call polling called hub polling (also referred to as token passing) decreases the host's involvement. With this method, the host polls the first node in the sequence. This node transmits a message if it has one, and then passes the query to the next node in the sequence. Each node in turn transmits a message and passes on the query or passes the query only. The last node on the line informs the host that the cycle is complete. The host then starts again at the beginning.

Centralized Star

In a **centralized star** topology, all communication in the system is controlled by the host node (see Figure 2-13).

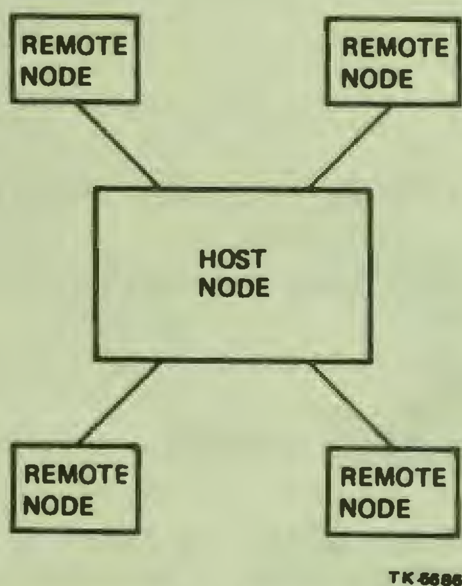


Figure 2-13 Centralized Star Topology

Remote nodes, which have terminal devices or small processors, communicate with each other only through the control of the host. Data can move toward the host or away from it. To relay data from one remote node to another, the host operates a switcher. Central control of communications and access to data bases and programs is maintained by the powerful host. However, if the host fails, the complete network goes down.

Hierarchical Tree

Hierarchical tree-shaped topologies assign levels of control (see Figure 2-14). Each of the supervisory nodes at the center level controls its own **cluster** of remote nodes. The supervisory nodes, in turn, report to the central host. (Hierarchical networks are widely used in industrial environments where computers control and supervise real-time processes.)

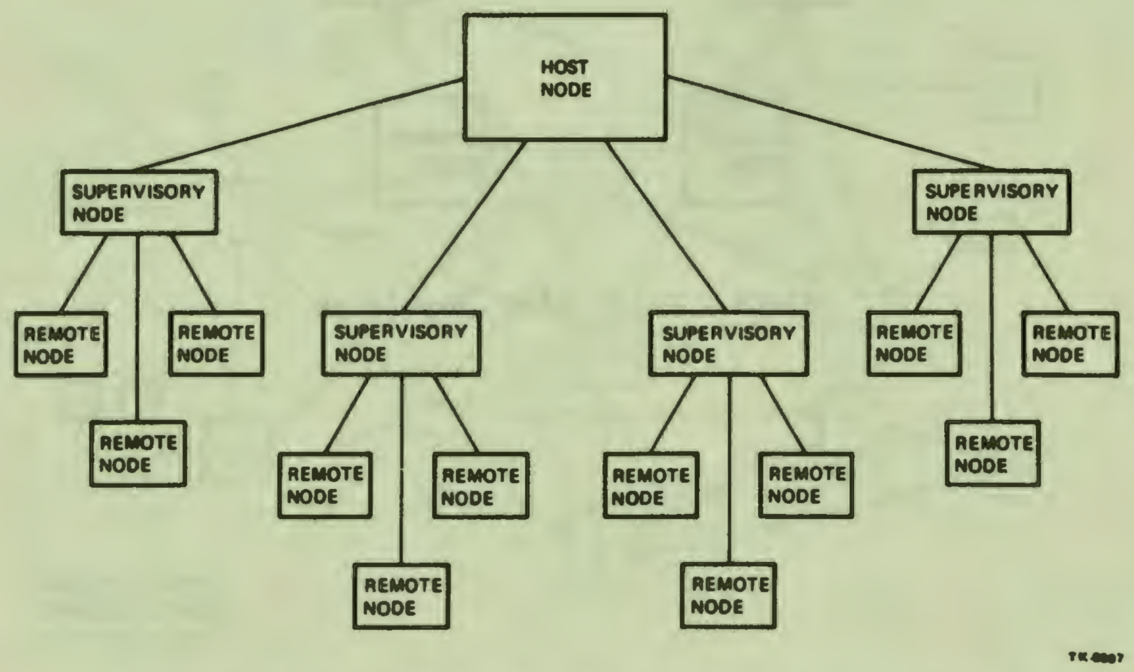


Figure 2-14 Hierarchical Tree-Shaped Topology

Ring

In a ring topology, communications between nonadjacent nodes are relayed through intermediary nodes (see Figure 2-15). Ring structures have the benefit of redundant lines. If one processor or line goes down, nodes can continue to communicate by sending messages the long way around the ring.

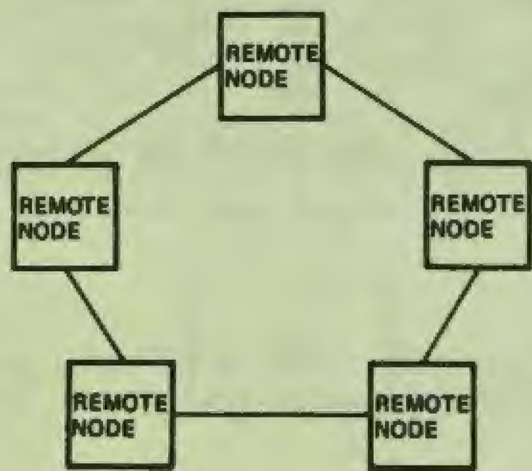


Figure 2-15 Ring Topology

Distributed Mesh

In a distributed mesh topology, many nodes are connected to the network by more than one physical line (see Figure 2-16). Multiple lines supply other communications paths if one line or processor fails. In many topologies of this type, no central host appears to control the entire network. Looking very closely at this type of structure, one could find a combination of the different types of structures previously discussed (star, multipoint, point-to-point, ring, etc.).

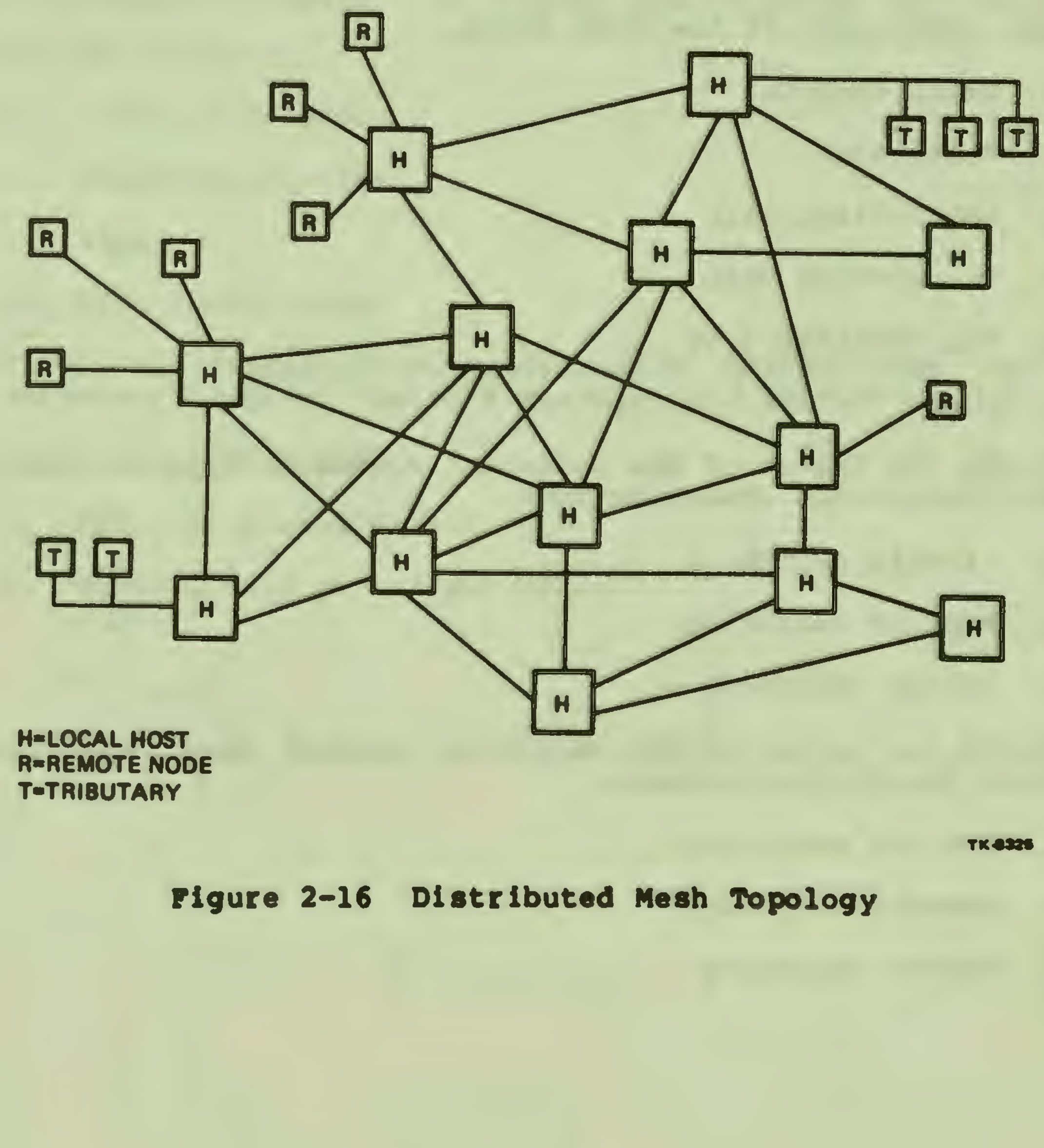


Figure 2-16 Distributed Mesh Topology

EXERCISES

1. Print "L" beside the line control method that needs the least involvement from the host. Print "M" besides the method that needs the most involvement from the host.

   ___ a. Contention

   ___ b. Hub polling

   ___ c. Roll-call polling

2. Circle the letter of the network or networks which will go down completely if the host fails.

   a. Point-to-point

   b. Multipoint

   c. Centralized Star

   d. Distributed mesh

   e. Hierarchical tree

   f. Ring

3. Circle the letter of the switching method(s) that are not good for interactive communications.

   a. Circuit switching

   b. Message switching

   c. Packet switching

4. Circle the letter of the switching method that specifies a fixed length for messages.

   a. Circuit switching

   b. Message switching

   c. Packet switching

5. Circle the letter of the switching method that needs a temporary point-to-point connection to be created before message transfer can occur.

   a. Circuit switching

   b. Message switching

   c. Packet switching

6. Circle the letter of the topology or topologies that do not need host nodes to control network communications.

   a. Point-to-point

   b. Multipoint

   c. Centralized star

   d. Hierarchical tree

   e. Ring

   f. Distributed mesh

7. Circle the letter of the transmission method that may send different parts of the same message by different routes.

   a. Dynamic switching

   b. Datagram packet switching

   c. Virtual call packet switching

SOLUTIONS

1. Print "L" beside the line control method that needs the least involvement from the host. Print "M" besides the method that needs the most involvement from the host.

   _L_ a. Contention

   ___ b. Hub polling

   _M_ c. Roll-call polling

2. Circle the letter of the network or networks which will go down completely if the host fails.

   (a.) Point-to-point

   (b.) Multipoint

   (c.) Centralized star

   d. Distibuted mesh

   e. Hierarchical tree

   f. Ring

3. Circle the letter of the switching method(s) that are not good for interactive communications.

   a. Circuit switching

   (b.) Message switching

   c. Packet switching

4. Circle the letter of the switching method that specifies a fixed length for messages.

   a. Circuit switching

   b. Message switching

   (c.) Packet switching

5. Circle the letter of the switching method that needs a temporary point-to-point connection to be created before message transfer can occur.

   (a.) Circuit switching

   b. Message switching

   c. Packet switching

6. Circle the letter of the topology or topologies that do not need host nodes to control network communications.

   (a.) Point-to-point

   b. Multipoint

   c. Centralized star

   d. Hierarchical tree

   (e.) Ring

   (f.) Distributed mesh

7. Circle the letter of the transmission method that may send different parts of the same message by different routes.

   a. Dynamic switching

   (b.) Datagram packet switching

   c. Virtual call packet switching

COMMUNICATIONS FUNDAMENTALS

3

INTRODUCTION

To understand the role of data communications, it is necessary to have an understanding of the basic components and techniques used to generate and process data on a communications link. Typically, data signals are transmitted over voice telephone channels, primarily because telephone facilities are available universally and, therefore, cost can be maintained at a reasonable level.

The most common situation in data communications is a computer interfacing with a remote terminal over voice telephone lines. This allows users at remote locations to use a central computer for data processing. The equipment used to perform this interaction has two general classifications:

1. **Data communications equipment (DCE)** - the equipment used to convey information between locations (for example, a modem).

2. **Data terminal equipment (DTE)** - the remote terminal, where information enters and exits from the data link for a user, and the computer, where information is processed and stored.

In this module, we will examine the main components of the DCE and DTE, and follow a message as it moves over a long-distance network link from its source to its destination. At each step we will examine the main communications equipment used and see the functions that each performs. We will pay special attention to the communications facilities provided by the telephone company because of their wide use.

OBJECTIVES

1. Define and illustrate the concept of "data communications."

2. Identify and describe the major functions of the following data communications concepts:

   - Computer codes,
   - Local links,
   - Transmission techniques,
   - Modulation techniques, and
   - Transmission characteristics.

3. Identify and describe the function of data communications equipment and facilites used in computer networks.

RESOURCES

1. Introduction to Minicomputer Networks, Digital Equipment Corporation, (EK-Ø53ØØ).

2. Technical Aspects of Data Communication, Second Edition, John E. McNamara, Digital Equipment Corporation, (EY-AXØ18-DP-ØØ1).

COMPUTER CODES

The first job in any data processing operation is to convert the data into a code that the computer can understand. In all computer codes, each alphabetic and alphanumeric character is represented as a sequence of binary digits, 1s and Øs. The most common codes have 5-, 6-, 7-, or 8-bit characters. These include:

- The American Standard Code for Information Interchange (ASCII)
- The Extended Binary Coded Decimal Interchange Code (EBCDIC)
- The Six-Bit Transcode (SBT)
- Baudot

ASCII

Most terminal equipment and computer manufacturers use the ASCII code, allowing compatibility between different makes of devices. ASCII characters are coded in seven bits, with an eighth bit available for use as a parity bit. (Parity is described in the Architecture Overview module.)

EBCDIC

IBM uses the Extended Binary Coded Decimal Interchange Code (EBCDIC). This code has 8-bit characters. Since the eighth bit is part of the code, there is no condition for character-by-character parity checking.

SBT

IBM has also developed an abbreviated 6-bit code, Six-Bit Transcode (SBT), which it uses to increase the efficiency of high-speed data lines.

Baudot

A 5-bit papertape code, Baudot, was the first serial asynchronous code developed for telegraph equipment. It is the most common code today, used by most of the world's printing telegraph communications networks.

LOCAL LINKS

Local or short-distance communication is generally accomplished by causing either voltage or current changes in a local wire circuit. This local wire ciruit is commonly referred to as a local link. Typical applications rarely exceed one mile, although the actual distance limit depends on individual circuit characteristics. Two of the most commonly used local links are:

- 20 milliampere loops (current) and
- EIA Lines (voltage).

Twenty Milliampere Loops

The scheme to modify the current flowing through the wire dates back to the early days of telegraphy. With this method, binary code (data) is created by turning off and on a constant current (20 milliampere) in both the sender and receiver.

Circuits with 20 milliampere (mA) currents are used because they are suitable for a wide range of applications. Such a circuit is often referred to as a 20 mA loop.

20 mA loops require two wires to complete the loop for transmitting data, and two wires to complete the loop for receiving data. A current loop (Figure 3-1) requires four basic elements:

- A current source
- Transmission wires
- A switch to interrupt the current flow
- A detector to sense current in the circuit

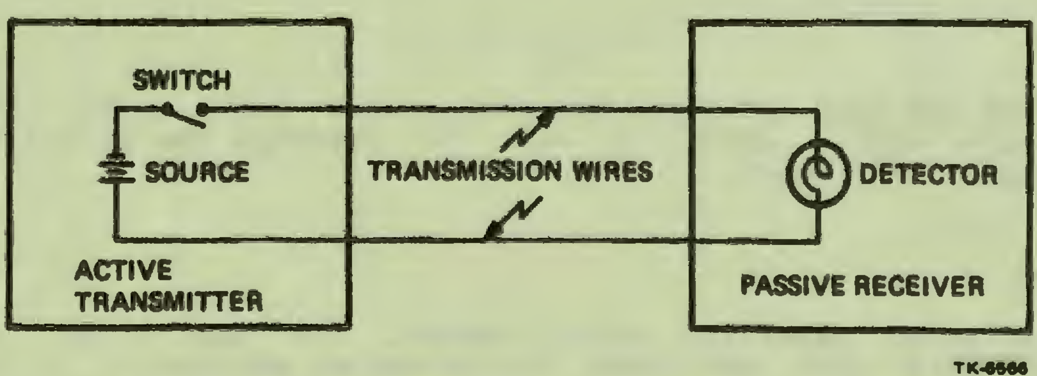


Figure 3-1 The Four Basic Elements of a 20 mA Loop

The current source can be located anywhere in the circuit. A device which contains the current source is referred to as the active device.

The other device which loops the current back to the source is called the passive device.

It is necessary that one active and one passive element always be used together.

EIA Lines

In the early days of data communications in the United States, American Telephone and Telegraph (AT&T) was virtually the only provider of data communications services. Therefore, the data communications equipment developed by Bell Laboratories and manufactured by Western Electric were the standards of the industry.

Manufacturers of computers and data terminal equipment who wanted to use the services provided by the Bell System needed to know the electrical characteristics of Bell's equipment, so that they would be able to interface to them. To solve this problem, the Electrical Industry Association (EIA) in cooperation with the Bell System, the independent modem manufacturers, and the computer manufacturers, developed a standard for the interface between DTE and DCE. The current standard is called RS-232, or more specifically, RS-232-C, reflecting the current (C) revision.

RS-232-C provides standards for both the physical and electrical specifications of a data line. To transmit data over an RS-232-C line, binary ones and zeros are represented by voltage levels.

In the RS-232-C, one wire provides for the transmit path and a second wire carries the signals of the receiver path. A third wire is used to provide a common ground between the transmitters and the receivers.

A more in-depth look at interface standards will be provided later in this module.

Problems and Restrictions

Both 20 mA loops and EIA lines have problems which limit their use. For example, 20 mA loops usually do not provide added wires for controlling the DCE. Therefore, they usually cannot be used for data communications that require some type of DCE. As we will see, most long-distance communications use some type of EIA DCE.

EIA lines are more open to electronic noise than 20 mA loops and, therefore, do not adjust well to electrically noisy environments, such as factories. Newer EIA standards (for example, RS-449) are currently under consideration to help alleviate such restrictions.

Because of such problems, 20 mA loops and EIA lines can be used for short local transmissions only. The maximum effective data rate drops rapidly as the length of the local link increases. The 20 mA loop is limited to about 5000 feet (1500 meters), although long-line drivers may be used to increase this distance. The current EIA standard limits cable length between DTE and DCE to 50 feet (15 meters), although it is possible to get acceptable performance from a longer EIA link. The actual maximum length of either type of link depends on various conditions:

- The type of cable used,
- The amount of noise in the environment,
- The data rate selected, etc.

ASYNCHRONOUS AND SYNCHRONOUS TRANSMISSION

Data can be transmitted over electric media asynchronously or synchronously. In an asynchronous transmission, each character generated by the transmitting end moves separately over the link. In asynchronous communication there is no common timing between the transmitting and receiving ends.

Each asynchronous character sent over the link has two bits added to it. The preceding bit is called the start bit and the trailing bit is called the stop bit (see Figure 3-2).

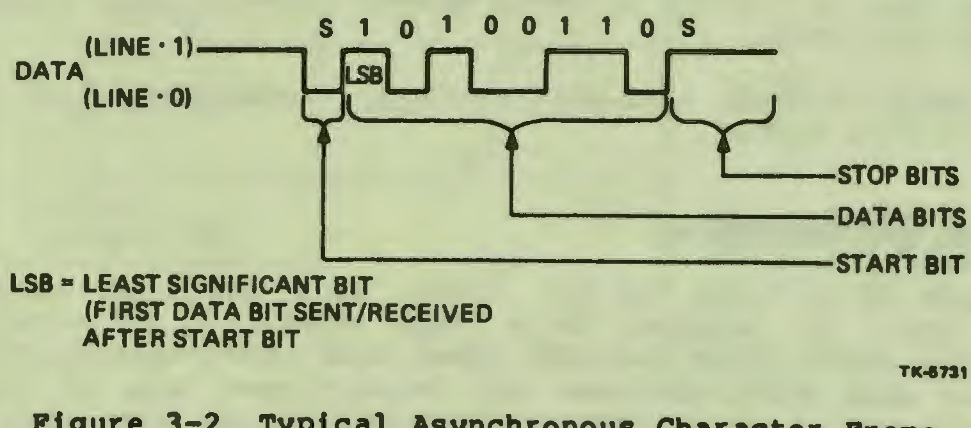


Figure 3-2 Typical Asynchronous Character Frame

The start bit is used to inform the receiving end of the line that a character is coming. The stop bit, which may be the length of one, one and one-half, or two bits, is provided to give the receiving end enough time to reset itself before the next character arrives.

Asynchronous mode is commonly used for low-speed, irregular transmissions (for example, those produced by an operator at an interactive terminal communicating with a computer). The main characteristic of asynchronous communication is that the receiver cannot predict when the next character will arrive, and must constantly monitor the line for start and stop bits.

Synchronous communication is characterized by a common timing source between transmitter and receiver. This timing is usually provided by the DCE.

In synchronous transmission, the transmitter stores up a string of characters and transmits them together at a high, steady rate of speed. One or more special synchronizing characters, called **sync**, at the beginning of the block of characters (usually 256 characters to a block) synchronizes the receiving end with the transmitting end (see Figure 3-3). Since all the bits representing data get to their destination in a steady stream, the receiving end must use a timing device to tell where one character ends and the next starts. If the receiving end is out of sync with the transmitting end by as much as a single bit, the message has no meaning.

Synchronous transmission is the usual method of computer-to-computer communication. It is also used with terminals designed to transmit characters in blocks.

Since start and stop bits are not required in synchronous transmissions, a standard ASCII 8-bit character can be transmitted without the 20% overhead (start and one-stop bit) required in asynchronus transmissions.

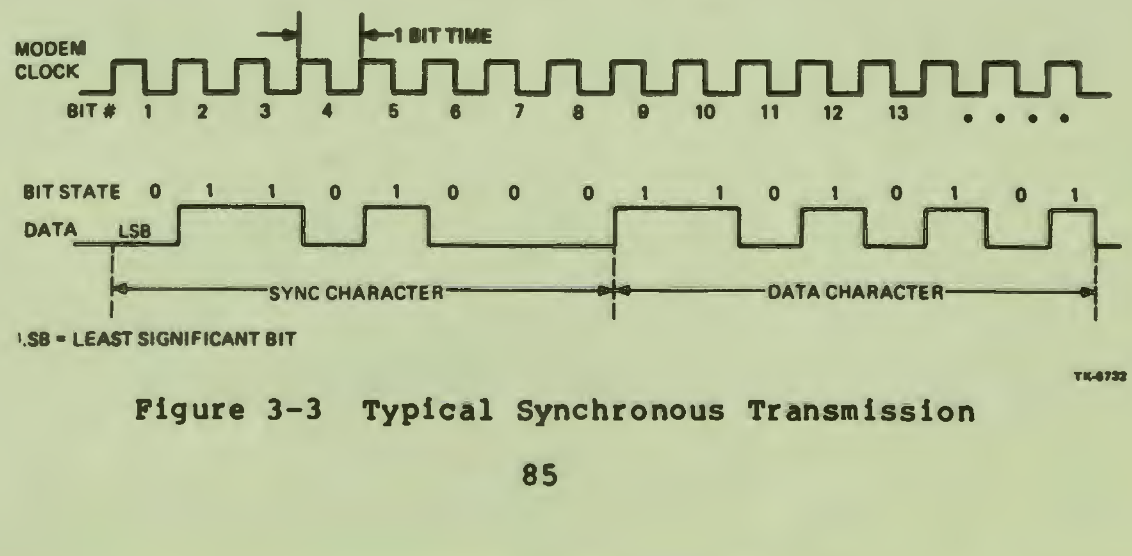


Figure 3-3 Typical Synchronous Transmission

EXERCISES

Match the following items with the phrase that best descibes them. Each item may be used one or more times, or not at all.

A. EBCDIC

B. Synchronous transmission

C. Baudot

D. 20 milliampere loop

E. ASCII

F. EIA

G. Asychronous transmission

H. SBT

___ Seven-bit character code with an added bit for parity.

___ Current turned off and on to represent binary data.

___ Specifies added wires for DCE control.

___ IBM code to increase efficiency on high-speed lines.

___ Each character generated moves separately over the link.

___ Needs a common timing device between transmitter and receiver.

___ Usually used for low-speed, irregular transmissions.

SOLUTIONS

Match the following items with the phrase that best descibes them. Each item may be used one or more times, or not at all.

A. EBCDIC

B. Synchronous transmission

C. Baudot

D. 20 milliampere loop

E. ASCII

F. EIA

G. Asychronous transmission

H. SBT

_E_ Seven-bit character code with an added bit for parity.

_D_ Current turned off and on to represent binary data.

_F_ Specifies added wires for DCE control.

_H_ IBM code to increase efficiency on high-speed lines.

_G_ Each character generated moves separately over the link.

_B_ Needs a common timing device between transmitter and receiver.

_G_ Usually used for low-speed, irregular transmissions.

LONG-DISTANCE TRANSMISSION

In most computer networks, long-distance transmission links are provided by the telephone companies and other common carriers. Circuits provided by the telephone company include:

- Public switched circuits and
- Private leased circuits.

Public Switched Circuits

The voice telephone network used by the public is a switched network. The user specifies a destination by dialing a number. The call is automatically switched through after dialing is complete.

A different circuit path is selected each time a call is made. When a number is dialed, the call goes to the local telephone exchange, where it is connected to the specified location or transferred to another exchange, as shown in Figure 3-4.

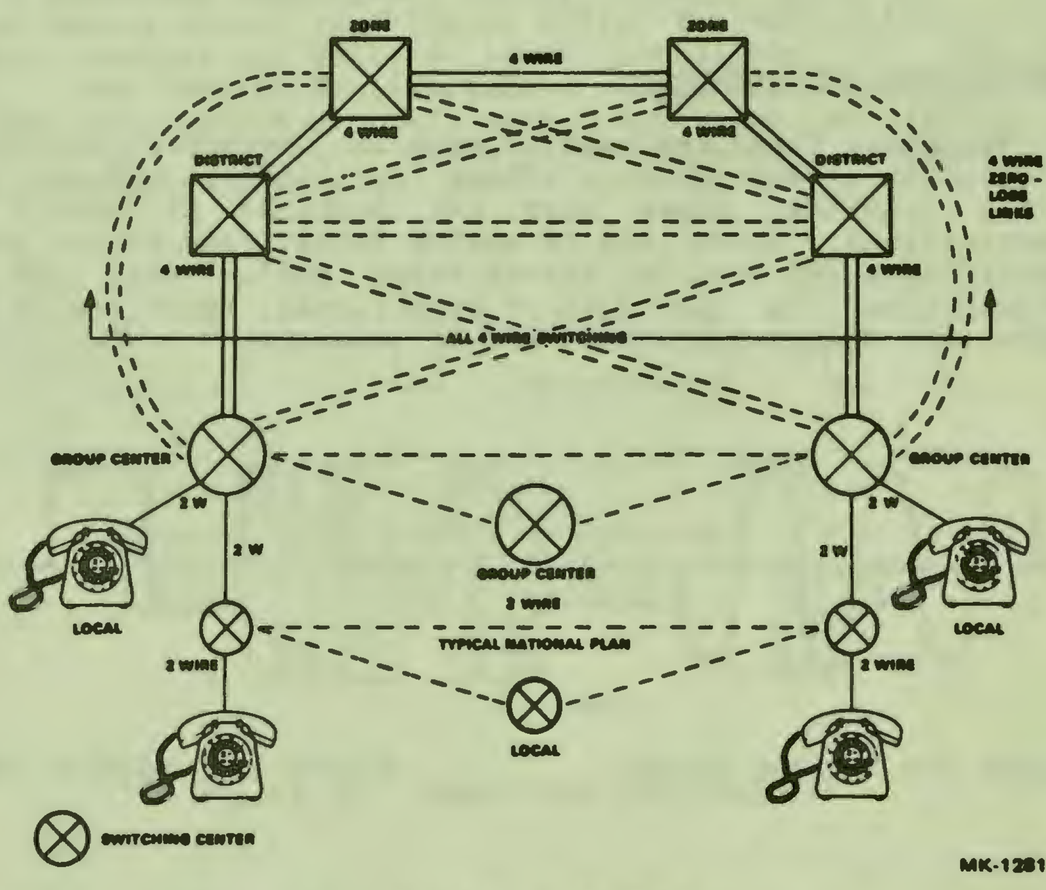


Figure 3-4 Public Switched Circuits

Public switched lines have the benefit of being always available. They are also lower in cost than leased lines when usage is low.

Private Leased Circuits

A leased line is a permanent circuit for private use within a data communications network. The line is either connected directly between two locations or routed through a telephone exchange. At the exchange, the leased line is connected in a way that makes it separate from the public equipment.

Four-wire leased lines are usually used when it is desirable to communicate regularly at speeds of 4800 bits per second or higher. To overcome some of the problems created by telephone lines at speeds of 1200 bits per second or higher, a line conditioning is available.

Line conditioning is available from the telephone company and is used to improve the signal quality of lines used for transmitting data at speeds of 1200 bps or greater.

MODULATION AND MODEMS

Telephone lines are used in computer networks because they are available everywhere. These facilities, however, present special problems, since they are designed to handle voice communications, which are in analog format (see Figure 3-5), and computer networks need a transmission medium that can handle communications in the form of binary code, which are in digital format (see Figure 3-6).

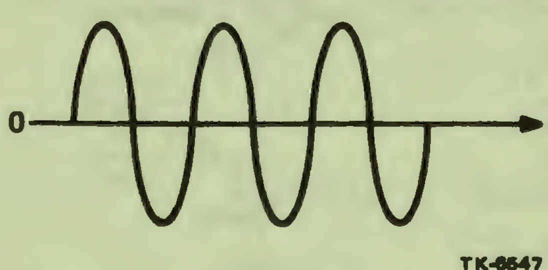


Figure 3-5 Analog Signal

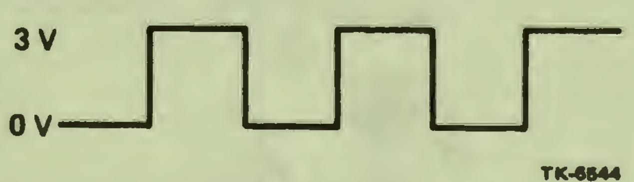


Figure 3-6 Digital Signal

Devices called modulators-demodulators, or commonly referred to as modems, were created to solve this problem. A modem is one type of DCE that is commonly used. Each data circuit needs a pair of modems. When transmitting data, the modem is used to convert the digital data from the terminal or computer into an analog signal for transmission on the telephone lines (a process called modulation). When data is received, the modem is also used to convert analog data received from the telephone line into a digital form used by the terminal or computer (a process called demodulation).

Three types of signal modulation commonly used in data communications are:

- Amplitude modulation (AM),
- Frequency modulation (FM), and
- Phase modulation (PM).

Amplitude Modulation (AM)

In amplitude modulation, binary digital data is modulated into an analog signal (see Figure 3-7). The modulation process is one which changes the voltage level (amplitude) of the analog signal. One amplitude represents a binary one, and another amplitude respresents a binary zero. Amplitude modulation makes efficient use of a voice grade telephone signal. However, it is very susceptible to problems caused by noise on the line.

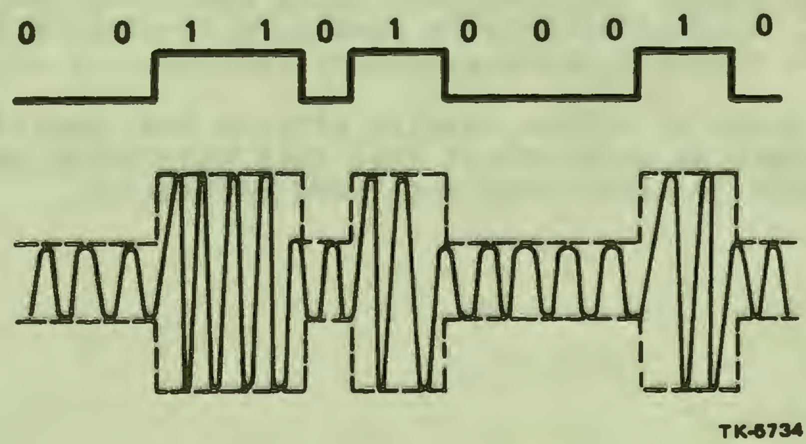


Figure 3-7 Amplitude Modulation

Frequency Modulation (FM)

In frequency modulation, binary digital data is also modulated into an analog signal (see Figure 3-8). This modulation process differs from AM in the fact that the frequency (rate of oscillation) of the signal is altered, rather than the amplitude. One frequency is used to represent a binary one, and another frequency is used to represent a binary zero.

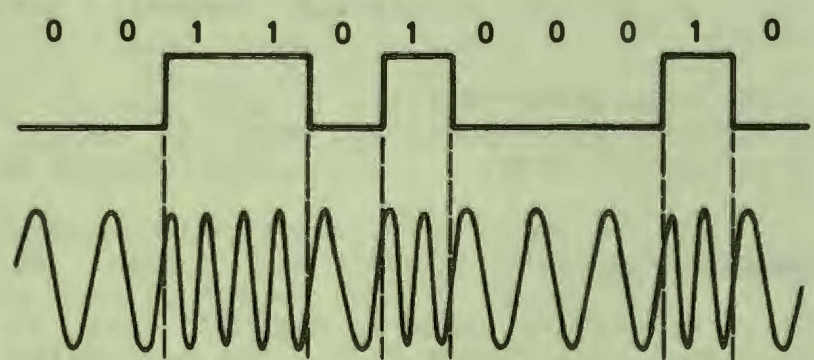


Figure 3-8 Frequency Modulation

Notice in Figure 3-8 that when it is desirable to transmit a binary zero, a low frequency is used. To transmit a binary one, the modem will transmit using a higher frequency.

Because noise on a line usually affects the amplitude of a signal, it should be quite clear that this modulation technique is less susceptible to noise than amplitude modulation.

Phase Modulation (PM)

In phase modulation, binary digital data is again modulated into an analog signal (see Figure 3-9). This modulation technique will use a single frequency and amplitude to represent binary ones and zeros.

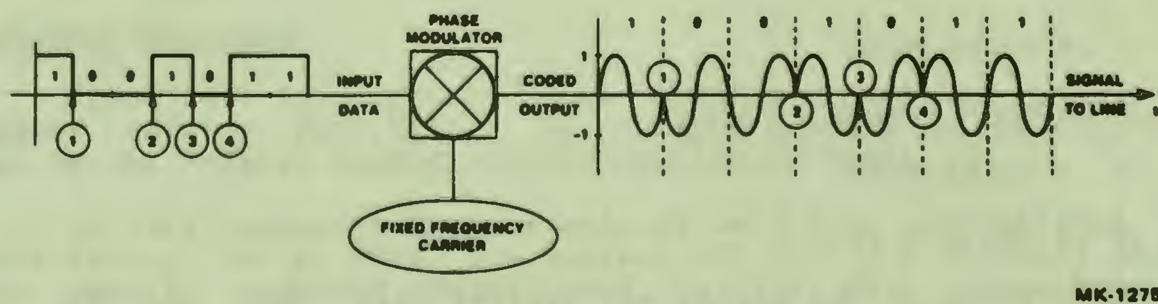


Figure 3-9 Phase Modulation

In order to avoid the problems on the line which affect the strength (amplitude) and speed (frequency) of a signal, it is necessary to come up with a modulation process which is slightly different than that of AM and FM. Phase modulation is what is used.

To distinguish between binary ones and zeros, a modem using PM will simply change the phase of the signal each time we go from a binary one to a zero or when we go from a binary zero to one (notice points 1, 2, 3, and 4 in Figure 3-9).

Two-State and Multistate Signals

Modulation in which one signalling state represents a one and a second state represents a zero is called binary modulation. Not all signals, however, are limited to two signalling states. In a multistate (or multilevel) signal, each state represents a combination of ones and zeros. In a four-state signal, for example (see Figure 3-10), two bits can be used to represent the four states as follows:

1 0 volts = 00

2 1 volt = 01

3 2 volts = 10

4 3 volts = 11

Each pulse of the signal now carries more than one bit of information. In Figure 3-10, each level corresponds to two bits (called dibits). That is, twice the information is being carried for the same number of pulses.

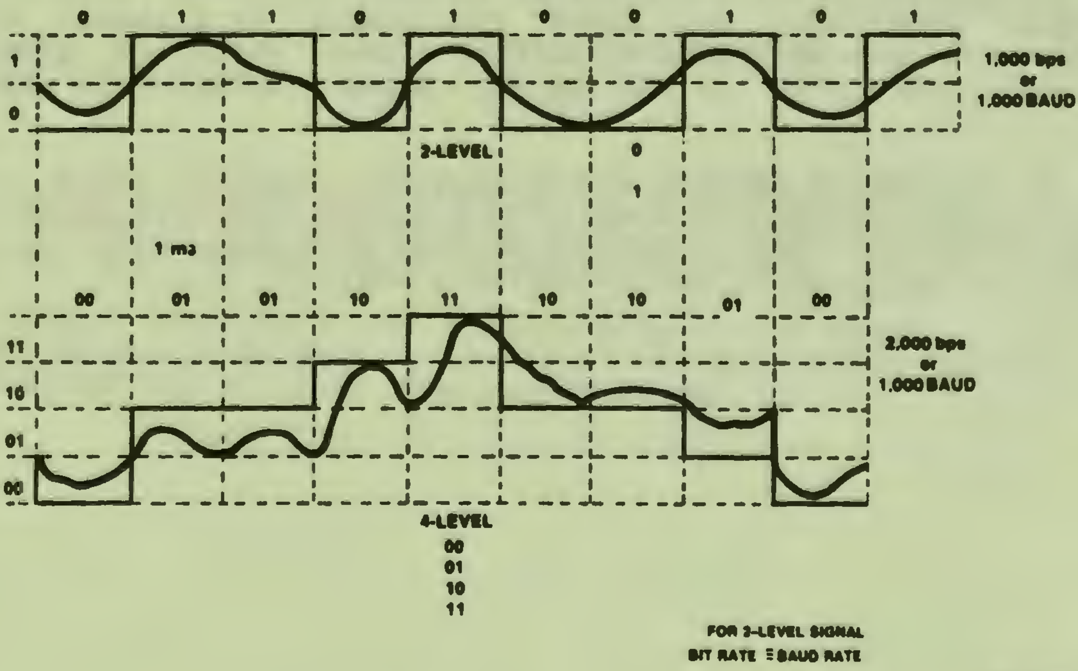


Figure 3-10 Four-State Signal

There are two disadvantages to multistate signalling. The first is an increase in modem cost and complexity. The second is an increased susceptibility of the signal to all forms of interference and line distortion. This means that high-speed modems are much more expensive, and require higher quality transmission lines than low-speed modems.

Interface Standards

Since most long-distance data transmissions take place over common carrier facilities, modems must conform to the standards set by the Federal Communications Commission (FCC).

In data communications, standards have been established for interfacing DTE to DCE. The current EIA RS-232-C standard is the most popular standard being used by the data communications industry. Because of its limitations, a new EIA standard, RS-449, is being introduced as a replacement for RS-232-C.

Another popular standard is the International Standards Organization (ISO) 2593 and the equivalent European Comite Consultatif Internationale de Telegraphic el Telephone (CCITT) V.35 standard.

EIA Standards Overview (RS-232-C Vs. RS-449)

The most common interface standard used in recent years has been EIA's RS-232-C. RS-232-C is presently the most widely used standard in the U.S.; it does, however, have some serious limitations for use in modern data communications systems, the most critical being that of speed and distance.

EIA RS-232-C restricts speed to 20,000 bits per second (20 kbps), and distance between DCE and DTE to 50 feet (15 meters). Because of this speed and distance restriction, the industry has developed a new standard, RS-449. RS-449 maintains a degree of compatibility with RS-232-C to accommodate an upward transition to RS-449.

The most significant difference between RS-449 and RS-232-C is in the electrical characteristics of signals between DTE and DCE. RS-232-C used only unbalanced circuits (thus, the restriction of speed and distance), while RS-449 uses both balanced and unbalanced circuits (balanced circuits are used to improve both speed and distance). Electrical specifications for these circuits are contained in EIA Standard RS-422A for balanced circuits, and EIA Standard RS-423A for unbalanced circuits.

RS-422A and RS-423A provide specifications for circuits which permit greater transmission speeds and allow for greater distances between DTE and DCE. The following lists the speed and distance characteristics of both RS-422A and RS-423A:

- RS-422A
  - maximum speed -- 2 Mbps (million bps)
  - maximum distance between DTE and DCE -- 200 feet (60 meters)
- RS-423A
  - maximum speed -- 20 Kbps
  - maximum distance between DTE and DCE -- 200 feet

The transition from RS-232-C to RS-449 will take some time. Therefore, any application interconnected between RS-232-C and RS-449 must adhere to the limitations set forth under RS-232-C.

ISO 2593/CCITT V.35 Standards Overview

These standards were developed for international use for data communications networks using wideband (high-speed) modems. These standards specifically deal with data transmissions greater than 48 Kbps using special leased line band circuits.

The preferred mode of operation identified by these standards is for synchronous lines using a scrambler/descrambler (similar to the encipher/decipher principle described in a previous module) mode of transmission.

Currently, cable length limitations corresponding to the actual operation are being studied.

Modem Features

While all modems perform the prime function of modulation/demodulation, there are other functions that can add to their capability, or in some situations restrict performance to better accomplish a specialized task. Some standard features are:

- Testing
- Transmit only/receive only
- Originate only/answer only
- Autodialing and autoanswering
- Acoustic coupling

Testing -- Three types of testing, when available, can be performed with modems:

- Analog loopback test
- Self-test
- Digital loopback test

The analog loopback test allows the operator to check out the local modem using any data pattern that can be generated. This test provides for local looping of transmit and receive data.

The self-test, by itself or in conjunction with other loopback tests, allows the operator to check out the local modem using a built-in message generator and message comparator. This does not require a data input and can be configured to check out both a local and remote modem.

The digital loopback test allows the operator to verify the data link through the local DTE, local modem, telephone line, and remote modem. This test requires that someone be present at both the local and remote ends to configure the modems for the test.

A special feature of some modems is the remote digital loopback test. This test gives you the testing features of digital loopback, but does not require that someone be present at the remote end to prepare the modem for the test.

Transmit Only/Receive Only -- With these features, it is possible for users to configure their modems to either transmit only or receive only. In some applications it may not be necessary for a modem be able to do both. Also a modem which only has one of these features will cost less than a modem with both.

Originate Only/Answer Only -- With these features, it is possible for users to configure their modems to either originate a call only or answer a call only. Again, in some applications it may be desirable to only have one of these features. As before, the cost will be less.

Autodialing and Autoanswering -- With switched modems, the link is created by placing a call. This can be done by an operator at the modem or by an autocalling unit (ACU). An ACU allows a computer to dial its own calls. This capability is called autodialing. In the same way, some modems automate the answering function. This capability is called autoanswering.

Acoustic Coupling -- Acoustic couplers are special modems that have a small speaker that changes digital signals directly into audible tones. These tones are picked up by a standard telephone handset, like the one you have at home, that fits into the coupler. Data is picked up from the handset speaker by a microphone in the coupler. The characteristics of a telephone handset limit acoustic couplers to a line speed between 300 bps to 1200 bps. Acoustic couplers are usually originate-only devices.

Modem Control

As discussed earlier, the RS-232-C standard allocates added wires for modem control. Modem control, which is performed by the computer's communications interface and its related modem, includes certain "handshaking" routines that must take place before and after a data transmission. These routines are made up of a sequence of questions and answers: "Are you there?", "Yes, I'm here."; "Can you receive a message?", "Yes, I can. Go ahead."; "Here it comes.", etc.

LONG DISTANCE MEDIA

Long-distance communications need a transmission medium that can carry some type of modulated analog signal. Long-distance media include:

- Open wire pairs,
- Wire cables,
- Coaxial cables,
- Microwave towers, and
- Satellites

Open Wire Pairs

Open wire pairs are uninsulated copper wires hanging from insulated cross arms on telephone poles. Most of these in the United States have been replaced with cables.

Wire Cables

Wire cables have hundreds of wire pairs. Each wire pair (a two-wire circuit) is capable of carrying one voice grade telephone channel.

Coaxial Cables

Coaxial cables can transmit at much higher frequencies than a wire pair. The cable consists of a hollow copper cylinder around a single wire conductor. The space between the cylinder wall and the wire conductor is filled with insulators. Coaxial cables, which can be combined into groups that handle over 18,000 telephone calls at the same time, have very little static, cross talk, or signal loss.

Microwave Towers

Microwave towers can transmit thousands of voice channels at one time. Microwave transmission is a line-of sight method in which signals are transmitted through the atmosphere by microwave towers set up 25 to 30 miles apart.

Satellites

A satellite in orbit functions as a high microwave tower. Satellites can handle thousands of voice-grade channels at the same time. By putting them in high orbit (22,000 miles), satellites can be made to travel at the speed of the earth's rotation. This makes them appear fixed in the sky above a certain point.

SIGNALLING SPEED AND LINE CAPACITY

The rate at which data can be transmitted is measured in bits per second (bps or b/s). The bit rate refers to the number of bits that can be transmitted in a second. Signalling speeds vary from about 50 bps to over a million bps (1 Mbps).

The capacity of a line to carry information is determined by its bandwidth. Bandwidth refers to the frequency range of a channel. It is the difference in Hertz (frequency speed in cycles per second) between the maximum frequency and the minimum frequency that can be transmitted over the line. Therefore, a frequency range of 5 to 10 Hz (Hertz) and another of 15 to 20 Hz have the same bandwidth, 5 Hz.

Data communication lines are classified according to bandwidth:

- Subvoice (narrow) band - 0 to 300 Hz
- Voice band - 300 to 3000 Hz
- Wideband - over 3300 Hz (3000 to 3300 Hz is not used)

These frequency ranges are general classifications. Many times, the actual bandwidths may overlap. The speed of transmission increases as bandwidth and frequency increase. Therefore, a narrow band is considered low speed, a voice band is considered medium speed, and wideband is considered high speed.

Subvoice Band

A subvoice channel has the narrowest bandwidth and the lowest speed of the three grades. Typical transmission speeds range from 45 to 150 bps. This is also called telegraph grade, since it is often used with telegraph and similar equipment. This band is normally used in asynchronous operations. The bits of each character are synchronized separately, but the characters themselves are generated at random intervals.

Voice Band

Voice band channels allow the complete 3K Hz bandwidth used for voice transmission to be used for data. Most voice grade services operate from 60 to 4800 bps. Speeds of up to 10,800 bps are possible with some modems, but at an increased cost. Voice grade provides a good, economical means of data transmission up to 4800 bps.

Wideband

Because of the bandwidth available with wideband grade (up to 240K Hz with leased lines), data can be passed at speeds from 19,200 to 500,000 bps. Even higher speeds are possible for special applications. Wideband grade is used when needed to transmit large volumes of data at high speed. The main problem with this grade is its high cost.

LINE CONDITIONING

Voice grade lines are low-fidelity data paths. To improve the quality of the line, the telephone company must condition it to provide the quality desired. Private leased lines are the only types of lines that can be conditioned. Switched lines can not be conditioned because the same data paths are not used every time. With private leased lines, the same physical path is used each time.

Many modems now have a special feature called adaptive equalization. Such modems are capable of adjusting themselves to make up, to some degree, for line characteristics. Adaptive equalization takes the place of a certain amount of conditioning.

EXERCISES

Match the following items with the phrase that best describes them. Each item may be used one or more times, or not at all.

A. Acoustic coupler

B. Phase modulation

C. Voice grade channel

D. Laser beam

E. Switched circuit

F. Frequency modulation

G. Wideband channel

H. Dibit

I. Leased line

J. Amplitude modulation

K. Modem

L. Coaxial cable

M. Satellite

___ Functions as a very high microwave tower.

___ Hollow copper cylinder around a single wire conductor.

___ Changes digital data into audible tones.

___ Performs digital/analog signal conversion.

___ Doubles the amount of data transmitted with each signal state.

___ Frequency range extends from 300 to 3000 Hz.

___ Most common modulation method at speeds up to 1800 bps.

___ Widely used, low-fidelity data paths.

___ Voltage is changed to represent ones and zeros.

SOLUTIONS

Match the following items with the phrase that best describes them. Each item may be used one or more times, or not at all.

A. Acoustic coupler

B. Phase modulation

C. Voice grade channel

D. Laser beam

E. Switched circuit

F. Frequency modulation

G. Wideband channel

H. Dibit

I. Leased line

J. Amplitude modulation

K. Modem

L. Coaxial cable

M. Satellite

<u>M</u> Functions as a very high microwave tower.

<u>L</u> Hollow copper cylinder around a single wire conductor.

<u>A</u> Changes digital data into audible tones.

<u>K</u> Performs digital/analog signal conversion.

<u>H</u> Doubles the amount of data transmitted with each signal state.

<u>C</u> Frequency range extends from 300 to 3000 Hz.

<u>F</u> Most common modulation method at speeds up to 1800 bps.

<u>C</u> Widely used, low-fidelity data paths.

<u>J</u> Voltage is changed to represent ones and zeros.

TRANSMISSION MODES

Almost all communications links, both local and long-distance, must be capable of handling messages. There are basically three types of transmission modes available to solve the problem of moving data:

- Simplex,
- Half-duplex, and
- Full-duplex.

A simplex line permits messages to move in one direction only. With this type of line, data can only be transmitted or received, but not both. One example of a simplex line is a line printer which is set up to output printed copies.

With half-duplex lines, data is allowed to move in either direction, but not in both directions at the same time. When one transmission is complete, the line must be "turned around" so that data can be sent in the opposite direction.

A full-duplex line gives you the same capabilities of a half-duplex line, but allows data movement in both directions at the same time.

TWO-WIRE AND FOUR-WIRE CIRCUITS

A standard telephone is connected to the local office by a two-wire loop. Signals going in both directions pass through this local loop.

Telephone offices in different cities, however, are connected by four-wire trunk lines. Two wires are used to send signals one way, and two to send signals the other. Long-distance calls must be transferred from a local two-wire loop to a four-wire trunk. A device called a hybrid coil makes this conversion. Figure 3-11 shows this arrangement.

For voice transmission, this system is acceptable. For data transmissions, however, the local two-wire loop is capable of handling half-duplex communications only. To get full-duplex service on a two-wire loop, the bandwidth must be divided between sending and receiving data.

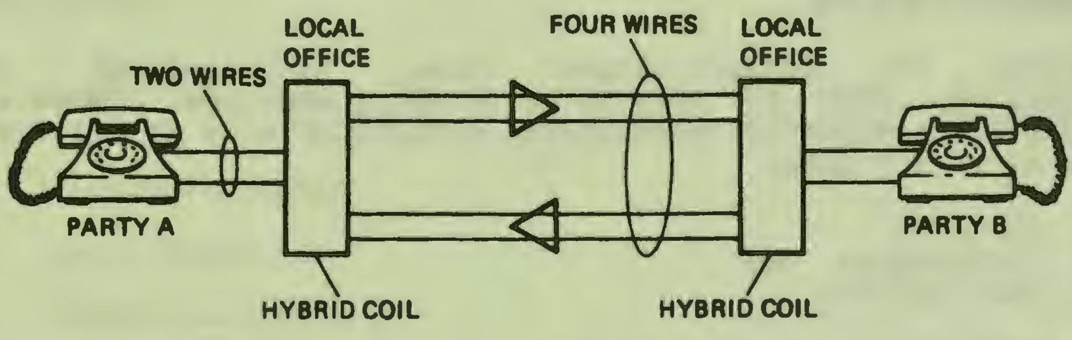


Figure 3-11 Four-Wire Trunk Line

Many users of leased lines get away from this problem by arranging for four-wire local service. This usually costs about 10 per cent more than standard two-wire service, and it provides full-bandwidth for full-duplex service.

MULTIPLEXING

In multiplexing, the physical link is divided in such a way that it can carry multiple messages at the same time. The multiplexer takes in data from several remote sources and sends it on to a host over the same number of subchannels. Since fewer physical links are needed to carry multiple circuits, this method significantly reduces the cost of each link.

Two multiplexing techniques are widely used:

- Frequency division multiplexing
- Time division multiplexing

Frequency division multiplexing (FDM) divides a voice grade or wideband transmission frequency into narrowband frequencies, one for each remote node that wants to communicate with a host. Figure 3-12 illustrates a frequency division multiplexer combining four voice grade channels from remote nodes into a single wideband channel for communication to the host.

Time division multiplexing (TDM) allocates a specific amount of time to each remote node that wants to communicate with a host. The multiplexer transmits a segment of data for each remote node in turn (see Figure 3-13). Usually, more subchannels can be created on a single physical link with time division multiplexing than with frequency division multiplexing.

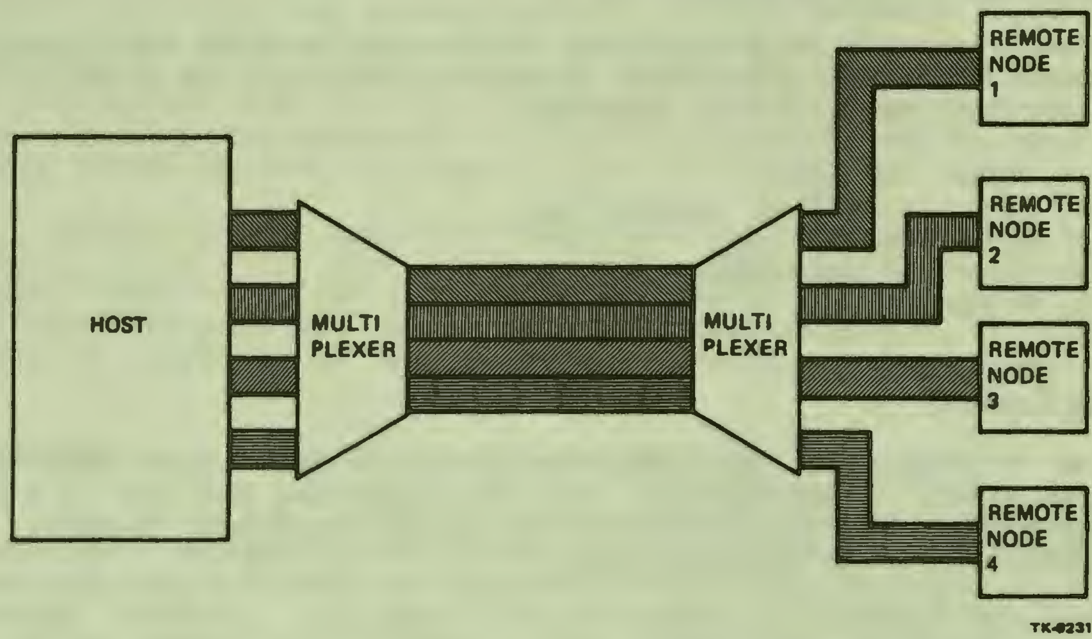


Figure 3-12 Frequency Division Multiplexing

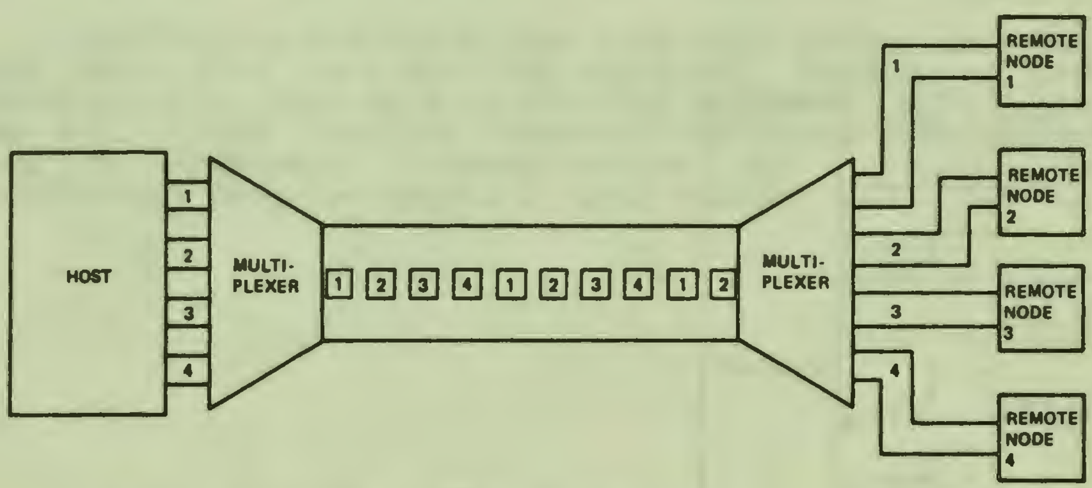


Figure 3-13 Time Division Multiplexing

THE COMMUNICATIONS INTERFACE

The communications interface is located between the modem and the computer. The interface is usually referred to as DTE. The functions of the interface include:

- Modem control,
- Serial/parallel conversion,
- Buffering,
- Character assembly, and
- Multiplexing.

Modem Control

As we have seen, the interface controls the modem and does the necessary "handshaking" at the beginning and end of a data transmission. The interface and the modem exchange control and data information in digital form.

Serial/Parallel Conversion

The interface changes entering data from the serial form in which it moves over the line, to the parallel form that the computer uses. For existing messages, it does the reverse (see Figure 3-14).

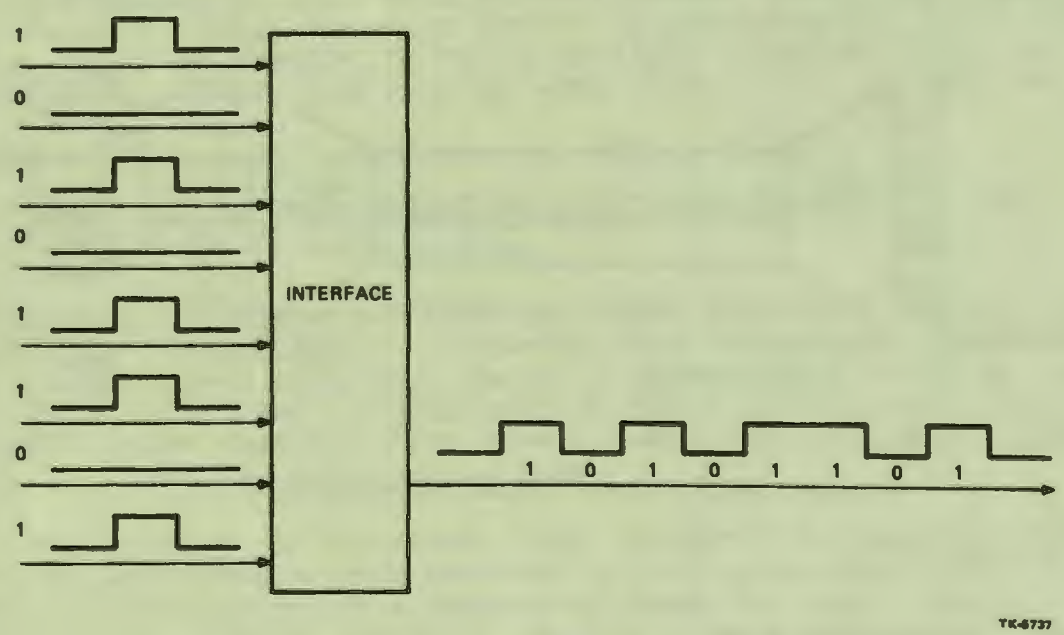


Figure 3-14 Communication Interface

In serial transmission, the bits of a character follow each other in order down a single wire. If the character code is made up of eight bits, it will take eight times as long to send one character as it will to send a single bit. In a parallel transmission, each bit in a character is sent down its own wire. In parallel transmission, a complete character can be sent in the same amount of time it takes to send a single bit.

Computers use parallel transmission to move data internally. Also, it is often used to communicate with peripheral devices and other computers located close at hand. However, all data transmissions of any significant distance, even down the hall, are done on serial links because of the cost of parallel lines.

Buffering

As entering data arrives serially from the modem, the interface assembles characters in a parallel output buffer. When the computer is ready to process a character, it takes it from the output buffer. If the computer does not recover the character from the output buffer before the interface receives the next character, the first character is lost. This is called an overrun. To minimize the occurrence of overruns, some interfaces use a method called double buffering. In high-speed interfaces or multiplexers, the buffer may be expanded by using a memory device called a silo or FIFO (first-in, first-out) buffer.

Asynchronous communications interfaces remove the start and stop bits from each entering character. Synchronous interfaces remove control elements from entering blocks of data and divide the bit stream into its component characters. Control elements used in synchronous communications are examined in the Architecture Overview module of this course.

Multiplexing

If there is more than one line, the interface attaches a label to the character to indicate which line it arrived on. In this way, the interface operates as a time-division multiplexer. It takes characters at the same time from many sources and passes them on to the computer over a single link. Since the rate at which a keyboard operator can send data is much less than the rate at which a computer can process it, it is typical for four, eight, sixteen or more input lines to be multiplexed in order to keep the computer busy.

Interface Features

Communications interfaces come in a number of forms:

- Synchronous and asynchronous,
- Parallel and serial,
- Single-line and multiline,
- With and without modem control, and
- Direct memory access.

Certain interfaces may be set to send and receive data at different speeds. This is known as split speed.

Some interfaces operate at a fixed speed and have no variable features. These are referred to as having fixed parameters. Of those interfaces that have adjustable parameters (such as transmit and receive speeds, character length and length of stop bit), some must be set by moving jumper wires on circuit boards. Others are set by small switches. In the more extended interfaces, parameters are set under program control by the host computer.

EXERCISES

1. Circle the letter of the phrase which most correctly completes the statement.

   Links which must be turned around at the end of each transmission are:

   A. Half-duplex

   B. Full-duplex

   C. Half and full-duplex

   D. Simplex

2. Circle the letter the phrase or phrases which correctly describe four-wire links.

   A. Used for intercity trunks.

   B. Can be installed for local leased-lines service.

   C. Uses two wires to send messages in one direction and two wires to send messages in the other direction.

3. Circle the letter of the statements which are true.

   A. A multiplexer takes in messages from n remote sources and transmits them to the host over m subchannels, where m is always less than n.

   B. A time division multiplexer allocates a specific amount of time to each remote node that wants to transmit data.

   C. Frequency division multiplexing cannot be used with voice grade lines.

   D. Parallel data transmissions are usually limited to two miles.

   E. In serial transmission, a complete character can be sent in the same time it takes to transmit a single bit.

SOLUTIONS

1. Circle the letter of the phrase which most correctly completes the statement.

   Links which must be turned around at the end of each transmission are:

   (A.) Half-duplex

   B. Full-duplex

   C. Half and full-duplex

   D. Simplex

2. Circle the letter of the phrase or phrases which correctly describe four-wire links.

   (A.) Used for intercity trunks.

   (B.) Can be installed for local leased-lines service.

   (C.) Uses two wires to send messages in one direction and two wires to send messages in the other direction.

3. Circle the letters of the statements which are true.

   (A.) A multiplexer takes in messages from n remote sources and transmits them to the host over m subchannels, where m is always less than n.

   (B.) A time division multiplexer allocates a specific amount of time to each remote node that wants to transmit data.

   (C.) Frequency division multiplexing cannot be used with voice grade lines.

   (D.) Parallel data transmissions are usually limited to two miles.

   (E.) In serial transmission, a complete character can be sent in the same time it takes to transmit a single bit.

ARCHITECTURE OVERVIEW

4

INTRODUCTION

Most networks start with a set of specifications and definitions called an architecture. The communications components, software and hardware, designed and assembled according to the specifications of the architecture are referred to as an implementation.

For example, the DIGITAL Network Architecture (DNA) is a set of specifications for a distributed processing network. The main implementation of DNA is an assembly of software products called DECnet. We will look at DNA and DECnet in the "DIGITAL Network Architecture and DECnet" module of this course.

Another very popular architecture is IBM's Systems Network Architecture (SNA). Both DNA and SNA architectures will be presented in this course.

An architecture typically includes:

- A definition of network goals,
- Specifications of network functions,
- Definitions of functional layers, and
- Definitions of interfaces and protocols.

OBJECTIVES

- For a typical network architecture, describe the following:
  - Network goals
  - Network functions and layers
  - Network protocols
  - Data link protocols
  - Message flow

RESOURCE

- Technical Aspects of Data Communications, Second Edition, John E. McNamara, Digital Equipment Corporation, (EY-AX018-DP-001).

NETWORK GOALS

A network architecture usually begins with a statement of network goals. The principal goal of most networks is transparency of communications functions. Once transparency is produced, users can exchange data without concern for the way communications functions are performed.

Other network goals include the support or non-support of the following features:

- Synchronous and asynchronous transmissions,
- Half-duplex and full-duplex lines,
- Point-to-point and multipoint configurations,
- centralized and decentralized topologies,
- packet-switching, and so on.

FUNCTIONS AND LAYERS

An architecture defines all the functions needed to produce its stated goals. In architectures, communications functions are usually arranged in logical layers of related functions. Each architecture that uses layering defines its own layers and creates sublayers as needed.

The International Standards Organization (ISO) is currently proposing that a standard be set identifying network architecture layers. The ISO is further proposing that their Open System Interchange (OSI) model (Figure 4-1) be used to identify the standard layers of a network architecture. The seven layers of the ISO OSI model are as follows:

- Application -- Provides the services to users of OSI.

- Presentation -- Resolves syntax differences between local system data formats and OSI format.

- Session -- Provides the means for communicating applications to organize and synchronize the dialogue between end systems.

- Transport -- Responsible for transfer of messages, in sequence, between end systems.

- Network -- Provides the means for routing individual messages, through multiple systems, to the required destination end system.

- Data Link -- Responsible for error free transmission of messages between adjacent systems.

- Physical -- Provides the means for transmission of bits, in sequence, between adjacent systems.

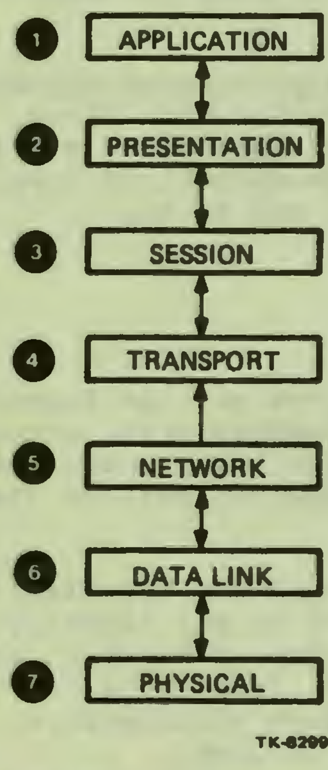


Figure 4-1 ISO Model for Network Architectures

Architectures that are designed for end-to-end communications, that is, communications between users at different nodes, are usually broken down into four broad functional layers. These four layers, and if applicable, their sublayers, perform the same functions as the layers of the ISO OSI model. Their relationships to the ISO model are as follows:

- User Functions Layer

  - Application layer
  - Presentation layer

- Network Functions Layer
  - Session layer
  - Transport layer
- Data Link Functions Layer
- Physical Link Layer

Figure 4-2 shows two network nodes, each equipped with these four layers of communications software and hardware.

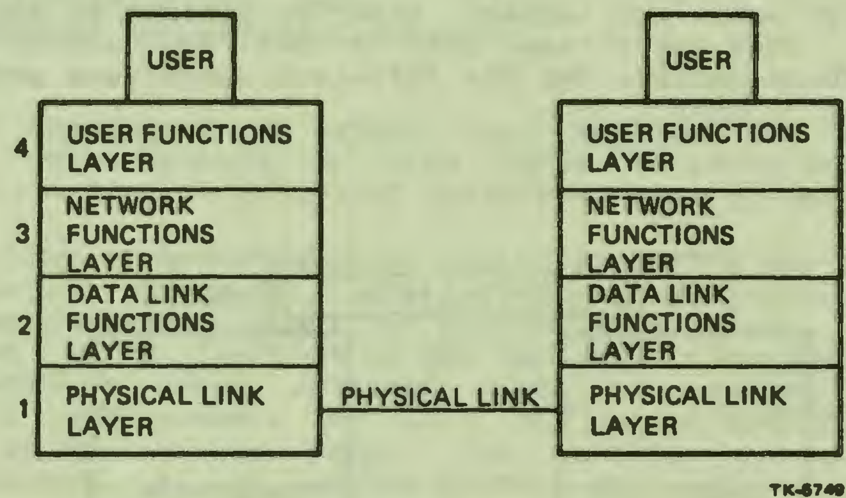


Figure 4-2 Typical Network Architecture Layers

The Network User

Positioned "above" the highest layer is the network user. A user can be either an applications program or an operator at a terminal. Network communications can include two programs that want to swap messages, two operators at interactive terminals, or an operator who wants to run a remote program or open a remote file.

The User Functions Layer

The highest layer, which we call the user functions layer, provides the user interface to the network. Software modules in this layer take requests and exiting data from the user and pass entering data to the user. Modules in this layer also provide user-oriented services, such as remote file access.

The Network Functions Layer

Modules in the network functions layer are normally responsible for establishing an end-to-end connection, called a virtual path, between a user application task or process at one node, and a user application task or process at another (see Figure 4-3). Once the virtual path is set up, users can swap messages without concern for the different operations performed at lower layers.

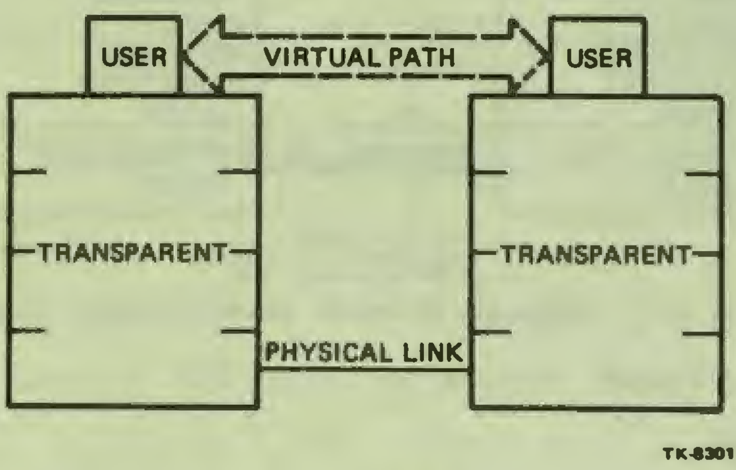


Figure 4-3 Network Functions Layer

Network functions also include the addressing, routing, queuing, and buffering of data for transmission. The modules refer to tables to find the best route available to send a message to any specified destination in the network. In packet switching networks, they also divide messages into segments of a specified size.

The Data Link Functions Layer

In the data link functions layer, software and hardware modules ensure that data transmissions over the available physical links are complete, correctly formatted and sequenced, and error free.

Data link functions include:

- Framing,
- Error checking and recovery,
- Sequencing,
- Data transparency,
- Line control,
- Synchronization, and
- Timeout and start-up control.

Framing -- Framing is the method that enables the receiver to determine which groups of bits in the entering stream make up characters, and which groups of characters make up messages.

Error Checking -- Error checking routines confirm the integrity of network data. In general, error checking involves redundant information that the transmitter adds to the data. This added information is a function of the data and is derived from it by the transmitter. The receiver tries to derive the same information and compare it with the added data sent with the message. If the results match, the receiver acknowledges the correct message. If the results differ, the receiver assumes that an error has occurred and requests another transmission. Three types of error checking are commonly used:

- Parity, or vertical redundancy checking (VRC),
- Longitudinal redundancy checking (LRC), and
- Cyclic redundancy checking (CRC).

In parity checking, the transmitter sends an added bit with each character. (In 7-bit ASCII code, this is the eighth bit in the byte.) In even parity, the added bit is used to confirm that the byte has an even number of ones. If the receiver finds an odd number of ones, it assumes an error. In odd parity, the added bit confirms that the byte contains an odd number of ones. In this case, the receiver assumes that any byte with an even number of ones is in error. Parity checking is also referred to as vertical redundancy checking.

In longitudinal redundancy checking, the transmitter generates a parity bit for each bit position in the stream of transmitted characters. That is, the transmitter keeps a running total of the number of ones that occur in the first bit of each character, another total for the second bit in each character, and so on. Then it computes a check character that is made up of the correct parity bits for all the bit positions. It transmits this character at the end of the message.

The receiver generates a corresponding check character and compares it with the transmitter's. Figure 4-4 shows a message that has an odd-parity bit for each character and an LRC check character.

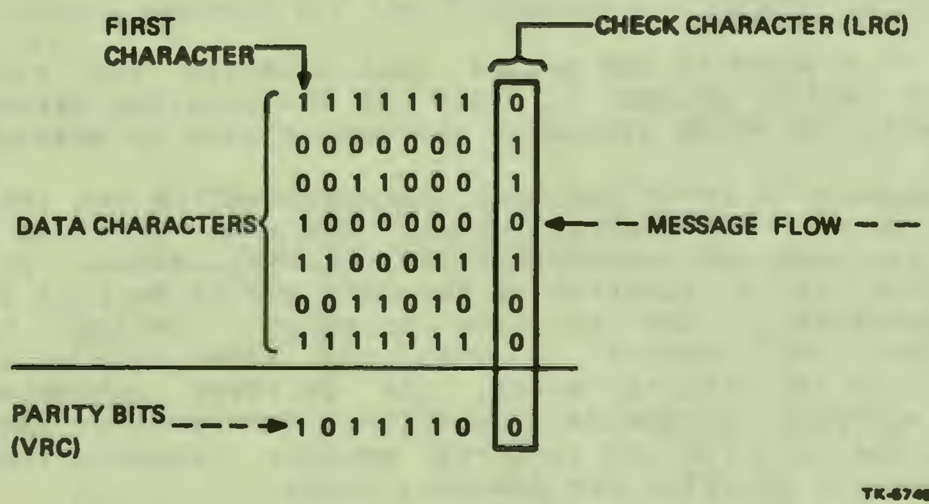


Figure 4-4 Longitudinal Redundancy Check

Both parity checking and longitudinal redundancy checking have one serious failing: neither can detect transmissions that contain an even number of wrong bits. If two bits of a word (or bit position) are in error, the second error disguises the first.

To overcome this drawback, a technique called cyclic redundancy checking (CRC) was developed. The transmitter uses the data flow as input for a numerical operation (basically repeated division by a constant) to the end of the message. The receiver performs the same operation on the data coming in, then compares the two check characters. Any difference indicates an error. There are three different CRCs:

- CRC-12, which generates a 12-bit character,
- CRC-16, which generates a 16-bit character, and
- CRC-CCITT, which also generates a 16-bit character.

CRC-16 and CRC-CCITT can each detect any set of wrong bits in strings up to 16-bits long, and over 99 percent of the errors in strings longer than 16 bits.

Sequencing -- A data link protocol must provide a method for keeping track of message sequences. Sequence control routines prevent unnecessary duplication of messages and loss of data. In the case of an error, they enable a receiver to request another transmission of a specific message.

Transparency -- A transparency procedure enables the transmitter to send a bit stream of non-character data. The procedure causes the receiver to turn off its character recognition procedure. Without this feature, the receiver would see certain sequences of bits as control characters and permit the wrong operation.

Line Control -- In networks with half-duplex or multipoint links, a line control procedure is needed to determine which node may transmit a message at any given time.

Synchronization -- On a synchronous link, the transmitter and receiver must be synchronized before a transmission can occur. Most synchronizing procedures use a special sync character, a certain pattern of bits, which the transmitter sends at the beginning of a message to tell the receiver to set its timing device.

Timeout and Start-Up Control -- The data link control protocol must specify the procedure that a node is to follow when messages stop coming in over the link. Usually, network devices maintain timers for each line. If the device does not receive a response to a transmission before the specified time runs out, it transmits again. If after many transmissions the transmitter continues to get no response, the system takes some prearranged action; for example, releasing the line and informing the operator.

Start-up controls start communications over a line that has been idle. The procedure includes a start-up message, for which idle receiving stations "listen," as well as a sequence of "handshaking" messages.

The Physical Link Layer

The physical link layer includes the software I/O driver and the communications equipment, including the hardware interface and the modem, which we examined in the Communications Fundamentals module of this course. The architecture specifies this equipment and the characteristics of the physical connection to the link.

Benefits of Layering

Not all architectures define functional layers. Layering, however, offers many benefits. It makes the relationship between communications functions easy to understand. It permits a modular approach to implementation that keeps a network flexible and capable of development and change. Modules, and complete layers of modules, can be removed and replaced, and new modules and layers added as needed.

INTERFACES

An interface is the mechanism that a module in one layer uses to communicate with a module in another layer. In some networks, interfaces are defined by the architecture and are, therefore, the same at every node. In other networks, interfaces are defined by users at each node and, therefore, may differ from node to node (see Figure 4-5).

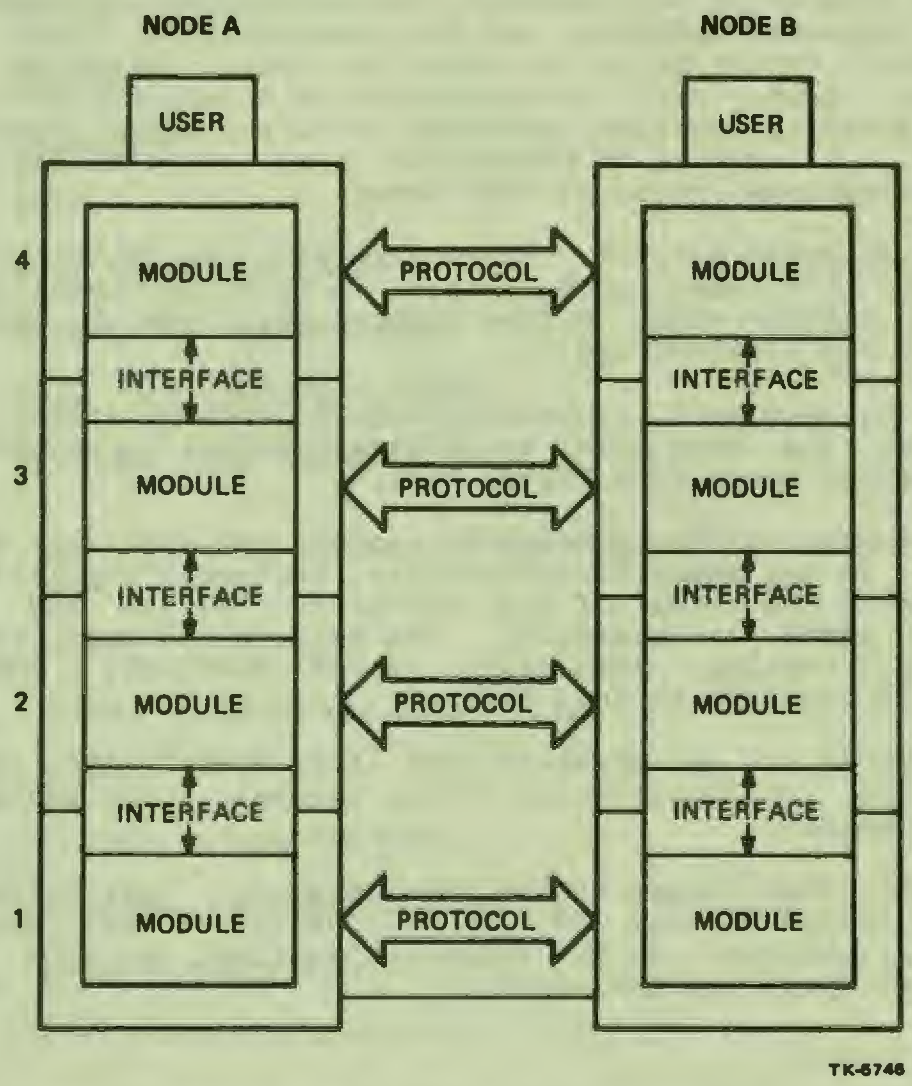


Figure 4-5 Network Interfaces

PROTOCOLS

When designing a network, a major consideration is the control of data from one computer to another within the network. What this entails is data control throughout the network which includes hardware, software, and the communications links. Almost every network design is on the order of a hierarchical or layered structure. Each layer is controlled by a separate set of rules designed to solve operating problems which may occur during data transfers. A protocol is essentially a set of rules for defining the communications system at each layer.

In a computer network, protocols are the rules that the modules in one node must follow in order to communicate with modules in another node. Unlike interfaces, protocols are always defined by the architecture.

For the most part, protocols control communications between modules at the same level in different nodes. Such modules are usually called peers (see Figure 4-5).

Peers communicate by means of headers and trailers added to the data. In our example architecture, protocols for levels 2, 3, and 4 specify the format of the different headers and trailers that are added to messages. The headers and trailers contain addressing, routing, sequencing, error checking, and other information specific to that layer.

Protocols for the physical link layer specify the electrical and physical characteristics of the communications equipment and their connections.

We will look closer at the protocols for user-oriented and network-oriented layers in our examination of DNA, SNA, and the other architectures. In the following section, we will look at some common data-link protocols.

EXERCISES

1. Match the following items and phrases. Each item may be used one or more times.

   A. Protocols

   B. Interface

   C. User

   D. User functions layer

   E. Network functions layer

   F. Data link functions layer

   G. Physical link layer

   ___ Mechanism for communication between layers in the same node.

   ___ Message routing.

   ___ Remote file access capability.

   ___ Modem.

   ___ Framing groups of bits.

   ___ Vertical redundancy checking.

   ___ Message addressing.

   ___ Peer communications.

2. Circle the letter of the statement or statements which are true.

   A. A serious problem with LRC and VRC is that they are not capable of detecting an even number of errors in a transmission.

   B. Transparency of communications functions is a goal of most networks.

   C. Protocols define communications procedures for peer modules.

   D. A network user may be a terminal operator or an applications program.

3. Circle the letter of the phrase or phrases which do not describe a function defined by data link protocols.

   A. Error checking and recovery.

   B. Message sequencing.

   C. File format conversion.

   D. Data transparency control.

4. Circle the letter of the error checking method that performs a parity check on each character.

   A. Character redundancy checking.

   B. Vertical redundancy checking.

   C. Longitudinal redundancy checking.

   D. Cyclic redundancy checking.

SOLUTIONS

1. Match the following items and phrases. Each item may be used one or more times.

   A. Protocols

   B. Interface

   C. User

   D. User functions layer

   E. Network functions layer

   F. Data link functions layer

   G. Physical link layer

   __B__ Mechanism for communication between layers in the same node.

   __E__ Message routing.

   __D__ Remote file access capability.

   __G__ Modem.

   __F__ Framing groups of bits.

   __F__ Vertical redundancy checking.

   __E__ Message addressing.

   __A__ Peer communications.

2. Circle the letter of the statement or statements which are true.

   A. A serious problem with LRC and VRC is that they are not capable of detecting an even number of errors in a transmission.

   B. Transparency of communications functions is a goal of most networks.

   C. Protocols define communications procedures for peer modules in the same node.

   D. A network user may be a terminal operator or an applications program.

3. Circle the letter of the phrase or phrases which do not describe a function defined by data link protocols.

   A. Error checking and recovery.

   B. Message sequencing.

   C. File format conversion.

   D. Data transparency control.

4. Circle the letter of the error checking method that performs a parity check on each character.

   A. Character redundancy checking.

   B. Vertical redundancy checking.

   C. Longitudinal redundancy checking.

   D. Cyclic redundancy checking.

DATA LINK PROTOCOLS

Data link protocol is another name for line protocol, a method for management of a data link. It provides a grammar by which nodes can converse with each other and serves as a vehicle for the transmission of messages. Were information transfer to always be between two locations as in a point-to-point dedicated data link, there would be no need for a complex data link protocol; a simple control procedure would suffice. However, since data communications use a variety of facilities and configurations, a complex data link protocol is required.

The message format of a data link protocol message consists of three basic parts or fields (refer to Figure 4-6):

- Header,
- Body, and
- Trailer.

The header of a data link protocol message usually contains control information pertaining to the message. Control information may consist of addressing, message sequence and response, message identification, message byte count, and any other information which may be used to control the message.

The body of a data link protocol message consists of the data that is to be transmitted.

The trailer of a data link protocol message contains the error checking information (commonly referred to as BCC, block check character).

Three types of protocols are used to specify the way functions are performed at the data link level:

- Character-oriented protocols,
- Byte-count oriented protocols, and
- Bit-oriented protocols.

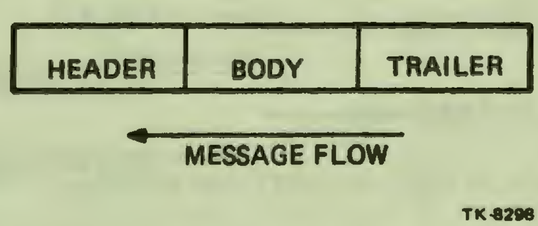


Figure 4-6 Standard Parts of a Data Link Protocol Message

Character-Oriented Protocols (BISYNC)

One of the most widely used character-oriented protocols in the industry today is IBM's BInary SYNChronous Communications Protocols, known as BISYNC (or BSC). BISYNC is a somewhat sophisticated data link control for half-duplex operations. It uses a large number of control characters and permits synchronous operation. BISYNC uses its control characters to delineate the various fields of a message and to control the necessary protocol functions.

With BISYNC and other character-oriented protocols, **handshaking** is the term commonly used to describe the interaction between nodes. Typically the following information is exchanged:

- Message available for transmission
- Start of text transmission
- Acknowledgment or rejection of the text
- Detection of errors
- Retransmission after error detection
- End of transmission

A summary of a simplified handshaking sequence is given in Figure 4-7. In this sequence, a terminal is talking to a computer. Notice that when the computer detects an error in a message, it notifies the terminal of the error and the terminal simply retransmits the message. When the terminal is done transmitting all of its messages, it sends a message indicating the end of its transmitting sequence and allows the computer to begin sending its own messages or to disconnect the link (this is know as line turnaround).

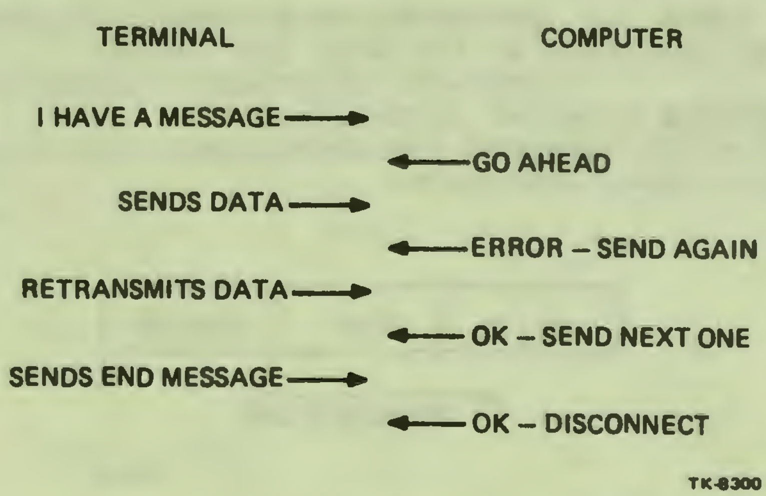


Figure 4-7 Typical Data Link Protocol Communication Process

Synchronization Sequence -- Before transmitting any control/message block (a block is usually 256 bytes), the sending node must begin each message by transmitting a minimum of two sync characters (SYN) to synchronize the two nodes. A message exchange is initiated when a node sends an enquiry (ENQ) to the node it wishes to talk to. If the receiving node can accept a message, it acknowledges (ACK) the enquiry. Throughout the handshaking sequence, each ACK is alternately numbered one and zero; for example, ACK0 followed by ACK1, which in turn is followed by ACK0, and so on.

As shown in Figure 4-8, SYN followed by ENQ is transmitted to synchronize and initiate the exchange, and the response is ACK0. The sending node then transmits SYN characters, followed by the message block.

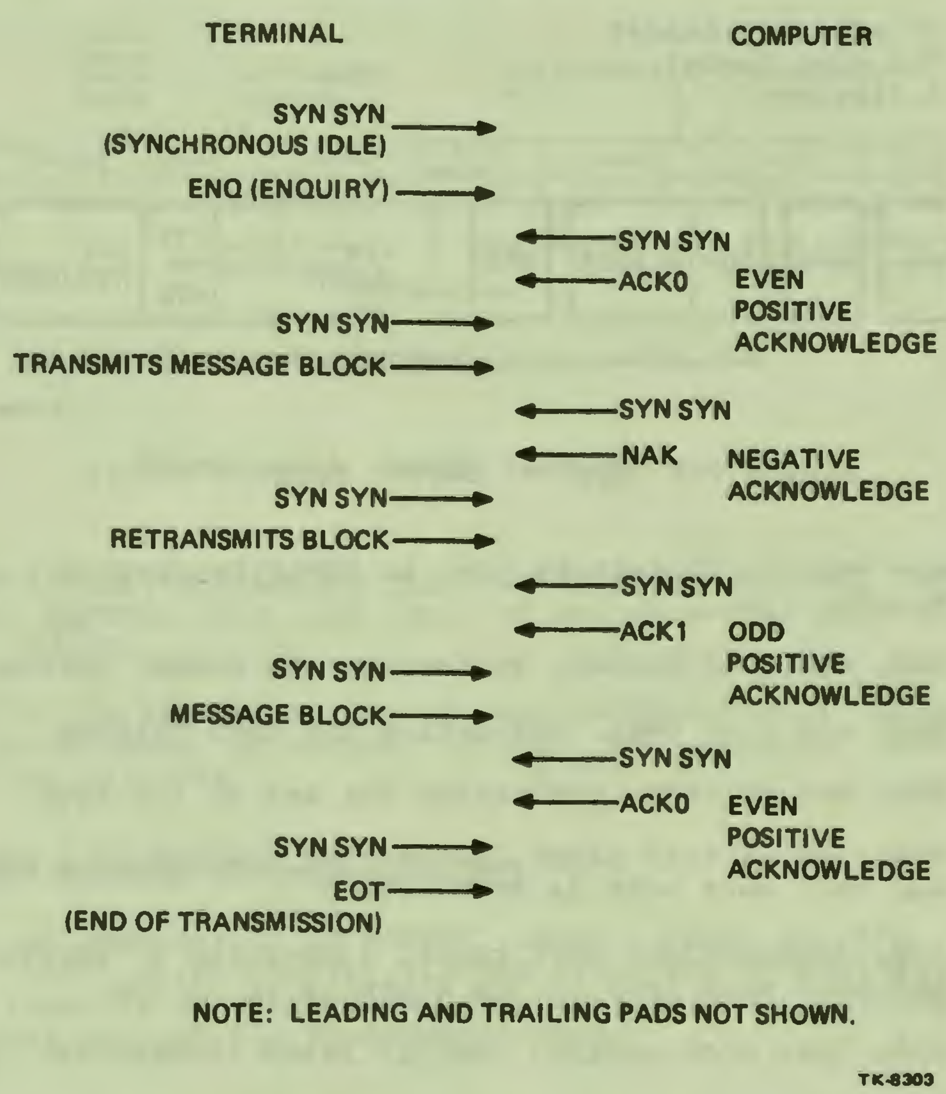


Figure 4-8 BISYNC Structured Sequence

When a node looks at a received message block and detects an error in it, it will respond to the transmitting node with a **negative acknowledgment** (NAK) message. Upon recognition of a NAK message, the transmitting node will retransmit the message. This process continues until the receiving node receives the message error-free and responds with a positive acknowledgment, ACK. To turn around the line, the transmitting node simply transmits an end of transmission (EOT) message after it has transmitted all of its messages.

Message Fields -- A BISYNC message format is shown in Figure 4-9. Each transmission can contain up to three elements:

- An optional header
- The text (body)
- A trailer

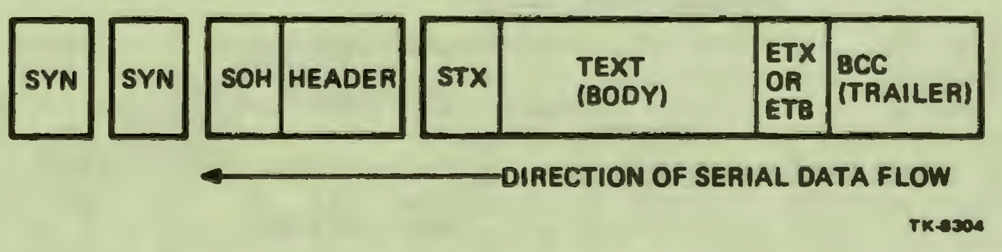


Figure 4-9 Typical BISYNC Message Format

Common control characters used to identify various fields of a message are:

- SOH, start of header, indicating the header follows
- STX, start of text, indicating the text follows
- ETX, end of text, indicating the end of the text
- ETB, end of text block, indicating the end of a text block and that more text is to follow
- ITB, intermediate text block, indicating a partial block and that more text of the block is to follow
- DLE, data link escape, used to frame transparent data

Transparency -- In the previous discussion of control characters, no text could have included control characters because the receiver would detect them as control characters rather than text. To be able to include control characters as part of text, BISYNC must be able to make them transparent to the receiver. This is accomplished by transmitting the message in the BISYNC transparent mode.

Messages transmitted in the transparent mode will have all control characters prefixed by the data link escape character DLE (character stuffing). See Figure 4-10. Only when the DLE prefix is present will the receiver recognize a control character.

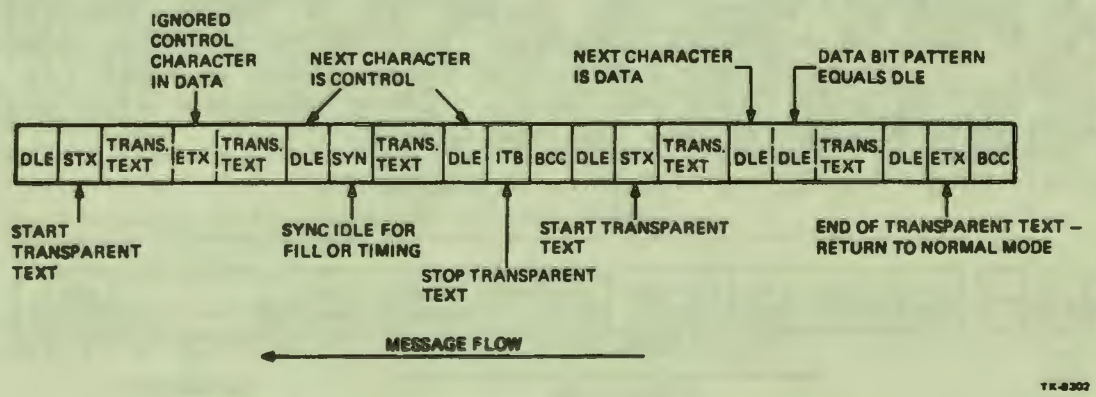


Figure 4-10 BISYNC Transparent Mode

The control characters referred to here, as well as others used by BISYNC, are all part of the character sets for ACSII, EBCDIC, and SBT.

BISYNC is limited to serial, synchronous lines. Although it is a half-duplex protocol, it may be operated over full or half-duplex circuits.

Byte-Count Oriented Protocols (DDCMP)

As our examination of BISYNC shows, many of the problems arising from a character-oriented protocol are a result of the special sequence of characters, and the character stuffing needed to transmit and receive transparent data.

Byte-count oriented protocols solve the problem of transparency without the use of DLE or other control characters. This is accomplished by having the transmitting end identify the total number of characters (bytes) that are contained in the body of the message. The receiving end can now accept data and treat the data accordingly because it knows exactly how many bytes of data it can expect in the body. Such a character count is often called the byte count because many computers store one character per 8-bit byte.

One of the most widely used byte-count protocols is the Digital Data Communications Message Protocol (DDCMP). DDCMP is a general purpose protocol in that it can be used on synchronous or asynchronous, half-duplex or full-duplex; serial or parallel; and point-to-point or multipoint lines.

As Figure 4-11 shows, the format of a DDCMP message is similar to that of a BISYNC message. In a DDCMP message, however, the header is required and a separate error check is made for the header and the body (CRC1 and CRC2). DDCMP messages also must be preceded by a minimum of two SYN characters.

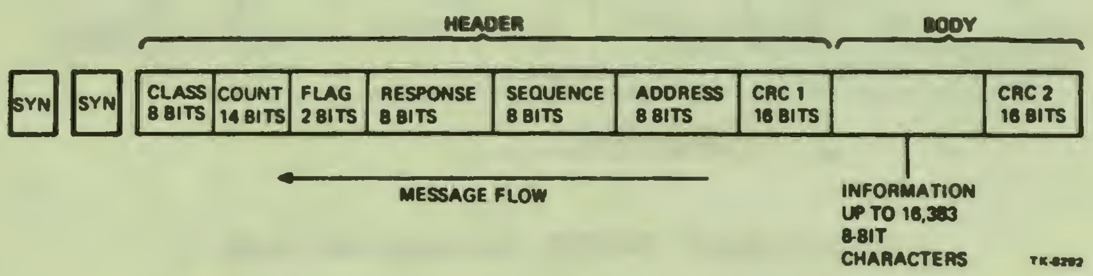


Figure 4-11 Typical DDCMP Message Format

Classification Field -- This 8-bit field is used to tell the receiver which of the following three message types it is receiving:

- Data message
- Control message (information)
- Maintenance message (used for down-line loading)

The header of a DDCMP message is made up of the following seven fields totaling eight bytes:

- Classification field (1 byte)
- Count field (14 bits)
- Flag field (2 bits)
- Sequence field (1 byte)
- Response field (1 byte)
- Address field (1 byte)
- Error check field (2 bytes)

Count Field -- When used for a data or maintenance message, this 14-bit field contains the exact count of bytes that may be found in the body of the message. When this field is used in a control message, it is used as an extension of the classification field to identify the type of control message being received.

Flag Field -- This field contains two bits; one bit represents quick sync and the other is the select bit. The quick sync bit is used whenever resynchronization is desired. The select bit is used when operating in a half-duplex environment, and is treated similarly to the EOT message used in BISYNC. The select bit is used to create the line turnaround needed in half-duplex.

Sequence and Response Fields -- These byte size fields are normally used in conjunction with each other. The sequence field identifies the sequence number of the current message, and the response field identifies the last good sequenced message. In DDCMP the receiver is prohibited from receiving messages out of sequence. With that in mind, the response field count will represent the number of messages already received, and indicate the message it next expects to receive.

Address Field -- This 8-bit field is used to identify a tributary address in a multipoint network configuration.

CRC 1 and 2 -- Both CRC 1 and 2 are 16-bit fields used for error checking. CRC 1 is used as an error check for the header information of the message, and CRC 2 is used as an error check for the text information of the message. With this dual error check method, errors pertaining to the header and text portions of the message can be singled out.

The body of a DDCMP message consists of an information field and also has its own error check referred to as CRC 2. A separate trailer is not needed because of CRC 1 and 2.

DDCMP Synchronization -- Before transmitting any messages across the link, a synchronization handshaking sequence is required to prepare each end. Figure 4-12 illustrates a typical DDCMP synchronization sequence. Notice that both ends are required to issue a **DDCMP Start** message. The proper acknowledgment to a DDCMP Start message is a Stack or Start acknowledgment. From this point on, normal message exchanges can occur (see Figure 4-13). Please make note of the use of the sequence and response fields as messages are exchanged.

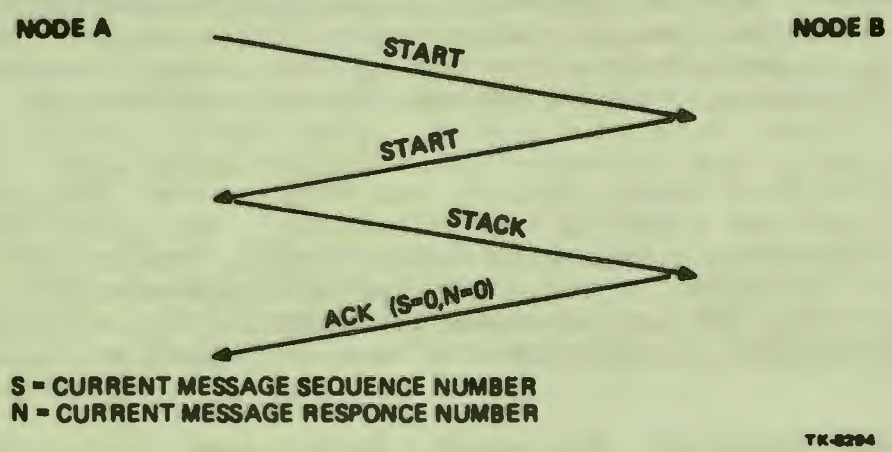


Figure 4-12 Typical DDCMP Start-Up Sequence

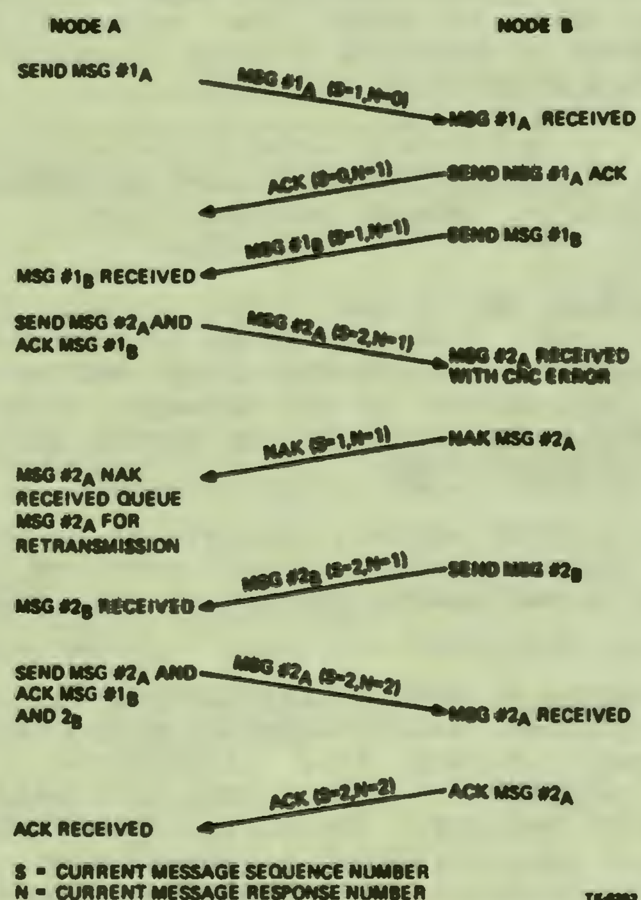


Figure 4-13 DDCMP Structured Handshaking Process

Bit-Oriented Protocols (HDLC and SDLC)

The third class of protocols, the bit-oriented or bit-stuff protocols, enclose messages and control information in frames. Two of the most commonly used bit-oriented protocols are High-Level Data Link Control (HDLC), proposed by the International Standards Organization (ISO), and IBM's Synchronous Data Link Control (SDLC). A typical SDLC message format (See Figure 4-14) consists of the following six fields:

- Beginning flag
- Address
- Control
- Information
- Frame check
- Ending flag

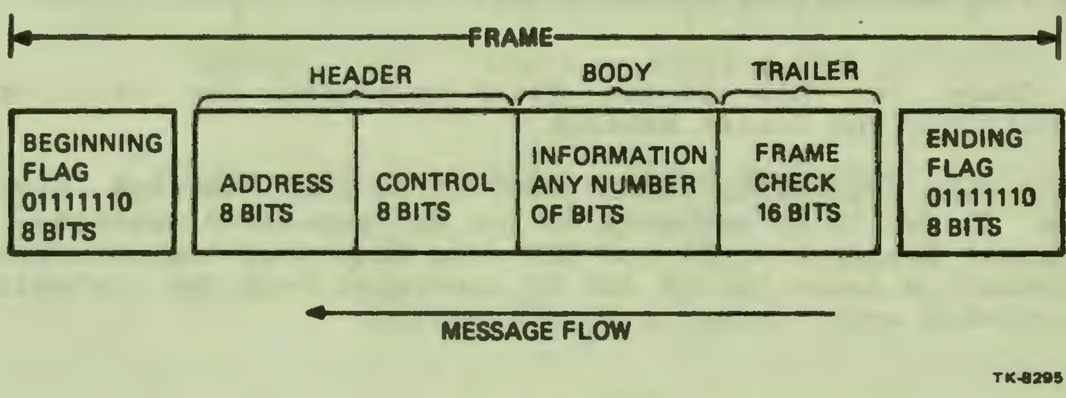


Figure 4-14 Typical SDLC Message Format

Beginning and Ending Flag -- Each SDLC message is framed by a special flag character (beginning and ending flag) which has a bit pattern characteristic of Ø111111Ø. These byte size flags are used by the receiving end to determine the beginning and ending of the received message.

Address Field -- This 8-bit field is used in multipoint environments to identify a tributary address. In point-to-point environments, this field is not used for any purpose and will be transmitted as a null character.

Control Field -- This 8-bit field serves the same purpose as the sequence field, the response field, and the classification field of a DDCMP message. This field can consists of:

- A 3-bit sequence count referred to as Ns
- A 3-bit response count referred to as Nr
- A 1-bit classification field identifying a sequenced message, or data and a non-sequenced message, or control
- A 1-bit line turnaround bit referred to as a poll or final bit.

Information Field -- This field is used to contain the body of the message (data). A count field is not used because the receiver can determine the end of the information field by locating the ending flag and backing up two bytes.

Frame Check -- This 16-bit field contains the error check characters for the entire message.

The only sequencing requirement of an SDLC message exchange is one in which an exchange of two non-sequenced messages, SNRM (Set Normal Response Mode) and NSA (Non-Sequenced Acknowledgment), are issued to reset the Ns and Nr counts at both the transmitting and receiving ends (refer to Figure 4-15).

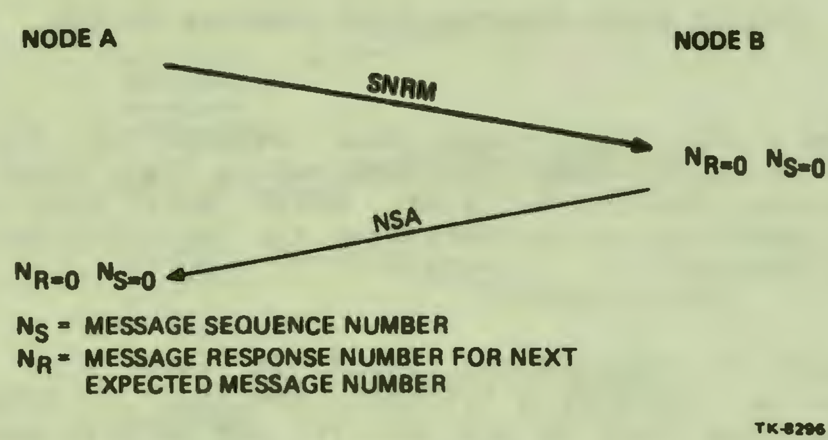


Figure 4-15 Typical SDLC Start-Up Sequence

To prevent the receiver from interpreting a bit sequence which contains six ones in a row as a flag character, the sending node bit-stuffs a binary zero after every sequence of five ones in a row. Bit-stuffing is not done to the beginning and ending flag characters. Now, once a message is received, the receiving end can recognize a flag character and will not misinterpret any of the bit sequences in the message. Because the transmitting end bit-stuffed its message, the receiving end must remove all the zeros which were stuffed prior to running any error checking sequence (refer to Figure 4-16).

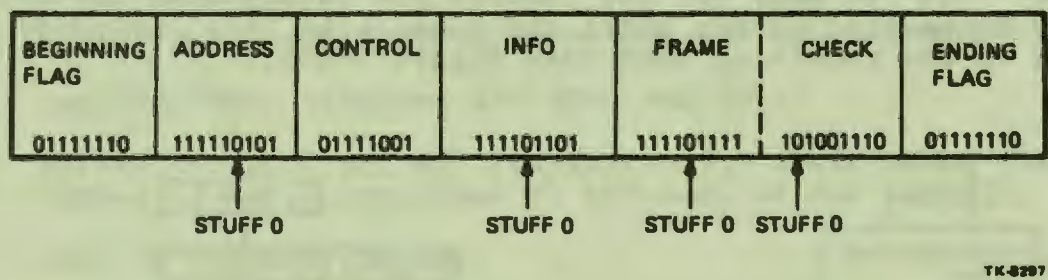


Figure 4-16 SDLC Bit-Stuffing Technique

MESSAGE FLOW

In general, modules in any given layer depend on the services provided by the layers below it. The user passes exiting data to the user functions layer. Modules in this layer pass the message "down" to the next layer. This procedure is repeated until the message reaches the physical link layer.

As the message goes down the hierarchy, it gets control information at each layer in the form of headers and trailers. These headers and trailers make up an envelope for the data. The message (within its envelope) then moves over the link to the destination node, and then moves "up" through the corresponding layers. As it goes up, the headers are interpreted and stripped off by succeeding layers until the message in its original form passes to the receiving user (see Figure 4-17).

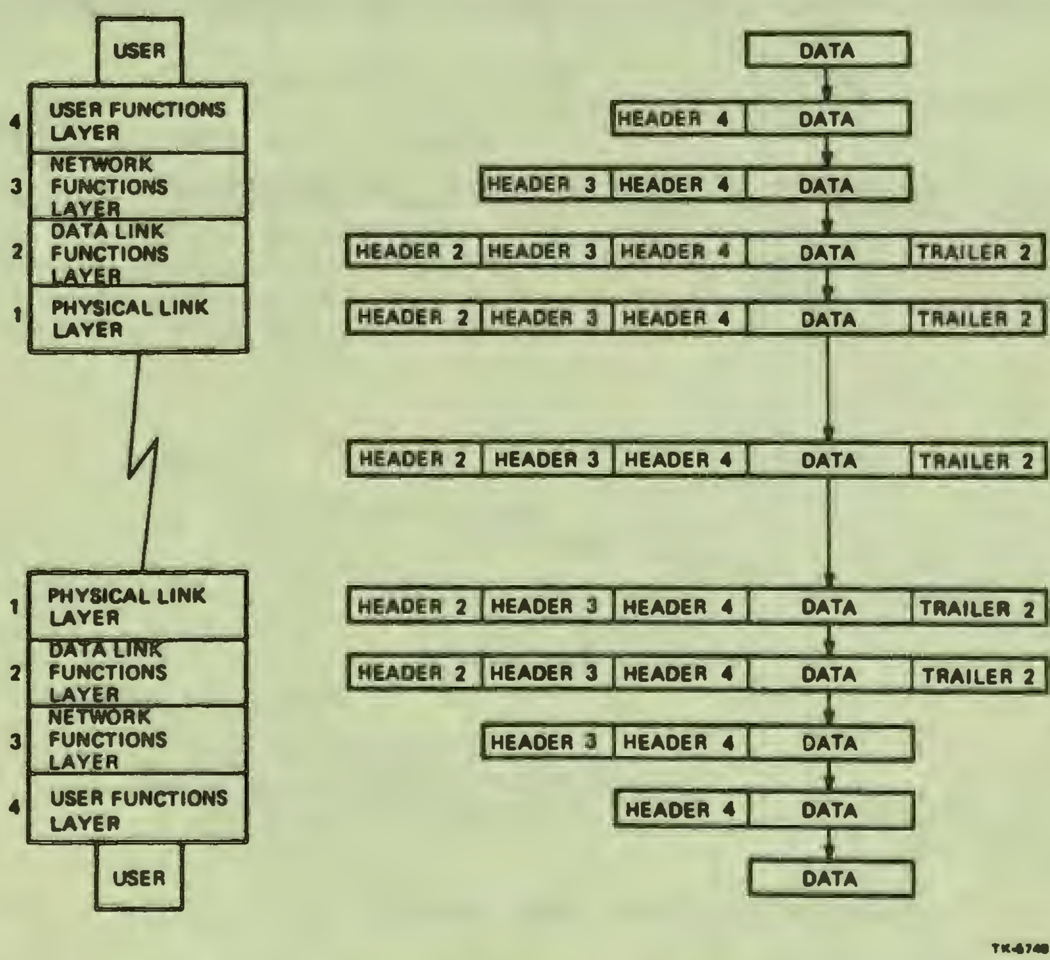


Figure 4-17 Typical Message Flow Between Nodes

EXERCISES

Match the following items and phrases. Each item will be used at least once.

A. Character-oriented protocol

B. Byte-count oriented protocol

C. Bit-oriented protocol

___ DLE STX tells the receiver to enter transparent mode.

___ The receiver automatically removes any stuffed DLEs.

___ Two important examples are HDLC and SDLC.

___ Solves the problem of transparency, indicating the total number of bytes contained in the body of the message.

___ Uses an optional header.

___ Uses bit-stuffing to make sure that the flag character does not occur during the data part of the message.

___ Eight-bit classification field tells the receiver that one of three types of DDCMP messages is coming.

___ SOH character tells the receiver that the following data is a header.

___ Header includes the character count.

SOLUTIONS

Match the following items and phrases. Each item will be used at least once.

A. Character-oriented protocol

B. Byte-count oriented protocol

C. Bit-oriented protocol

_A_ DLE STX tells the receiver to enter transparent mode.

_A_ The receiver automatically removes any stuffed DLEs.

_C_ Two important examples are HDLC and SDLC.

_B_ Solves the problem of transparency, indicating the total number of bytes contained in the body of the message.

_A_ Uses an optional header.

_C_ Uses bit-stuffing to make sure that the flag character does not occur during the data part of the message.

_B_ Eight-bit classification field tells the receiver that one of three types of DDCMP messages is coming.

_A_ SOH character tells the receiver that the following data is a header.

_B_ Header includes the character count.

DIGITAL NETWORK ARCHITECTURE AND DECnet

5

INTRODUCTION

DECnet is the collective name for the set of software products that extend the facilities of various DIGITAL operating systems so that they can be interconnected to form computer networks. The design of all DECnet software products derives from the DIGITAL Network Architecture (DNA).

DNA is the specification of a layered hierarchy in which each layer contains modules that perform defined functions. The architecture defines the functions of each module and the relationships between modules. DNA achieves the following design goals:

- Creates a common user interface
- Supports a broad spectrum of applications
- Supports a wide range of communications facilities
- Is cost effective
- Supports a wide range of topologies
- Is highly available
- Is extensible
- Is easily implemented
- Is highly distributed
- Implements network control and maintenance functions at a user level
- Allows for security

DECnet is the software package which extends various operating systems for PDP-11s, VAX-11s, DECsystem-10s, and DECsystem-20s. DECnet provides a set of facilities that allows the following:

- **Device sharing** - Device sharing allows the user to connect to and use the peripheral devices of a remotely located system.
- **File sharing** - File sharing allows the user to open, read, write, and close files stored on a remote system.
- **Program sharing** - Program sharing allows the user to transfer an executable program to a remote system, and to load and execute the program on the remote system.
- **Intertask communication** - Intertask communication allows two tasks in the same system or in remotely located systems to create a virtual path for transfer of information between them.

To provide these functions, DECnet is designed as a layered structure of protocols, which are formal sets of conventions governing the format, sequencing, and control of message exchange. In this module we will examine the features of DNA and its implementation of DECnet.

OBJECTIVES

1. Identify the network architecture features which make up the DIGITAL Network Architecture
2. Describe the following features of DECnet:
   - Layering Protocols
   - Architecture Goals
   - Functions that are performed

RESOURCES

1. DECnet DIGITAL Network Architecture General Description, Order No. AA-K179A-TK.

2. DNA Data Access Protocol (DAP) Functional Specification, Version 5.6.Ø, Order No. AA-K177A-TK.

3. DNA Digital Communications Message Protocol (DDCMP) Functional Specification, Version 4.1.Ø, Order No. AA-K175A-TK.

4. DNA Maintenance Operations Protocol (MOP) Functional Specification, Version 2.1.Ø, Order No. AA-K178A-TK.

5. DNA Network Management Functional Specification, Version 2.Ø.Ø, Order No. AA-K181A-TK.

6. DNA Network Services (NSP) Functional Specification, Version 3.2.Ø, Order No. AA-K176A-TK.

7. DNA Session Control Functional Specification, Version 1.Ø.Ø, Order No. AA-K182A-TK.

8. DNA Transport Functional Specification, Version 1.3.Ø, Order No. AA-K18ØA-TK.

9. Introduction to DECnet, Order No. AA-JØ55A-TK.

DECnet FUNCTIONS

DECnet offers a wide range of networking functions. Some, like task-to-task communications, are provided throughout DECnet, while others can only be supplied by a subset of DECnet implementations. For example, loading a system image down-line to a satellite node may be a function of some operating systems, but not all. The following list identifies DECnet's major functions:

- Task-to-task communications
- Remote file access
- Terminal-to-terminal communications
- Remote terminal facilities
- Network management facilities
- Down-line loading
- Up-line dumping
- Loopback testing
- Routing
- Multipoint

Task-to-Task Communications

All DECnet implementations allow two programs within a network to perform task-to-task communication; that is, to exchange data over a virtual path. In most DECnet implementations, performing task-to-task communication is similar to performing I/O. The virtual path (DECnet calls this a **logical link**) between two programs is like an I/O channel over which both programs can send and receive data.

A logical link is a virtual path or circuit (as opposed to a physical one) on which information flows in two directions. For example, in Figure 5-1, node Kermit and node Piggy have established a physical link between them and communications can exist. For processing information between task A in Kermit and task B in Piggy, a logical link has been established. A physical link has been established between nodes Kermit and Piggy, as well as between nodes Piggy and Grover in Figure 5-2. To process information between task A in Kermit and task C in Grover, a logical link must be established between them.

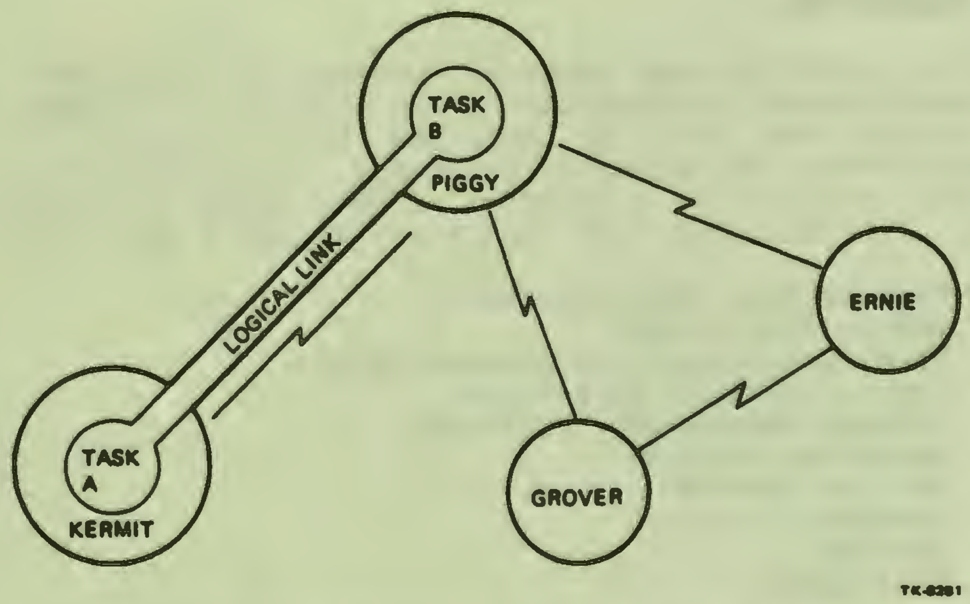


Figure 5-1 Physical Link Between Nodes

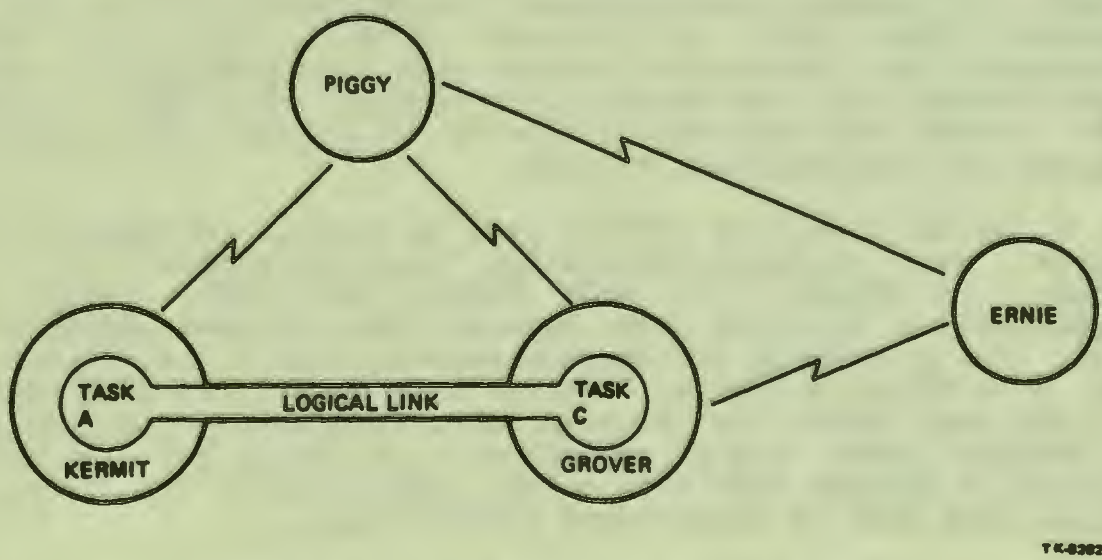


Figure 5-2 Logical Link Between Nodes

Remote File Access

Using DECnet, a program in one node can access a file in another node, despite differences in the two node's operating and file systems. This remote file access capability has the following applications:

- A user-written program can incorporate DECnet calls that allow it to perform record-level operations on remote files.
- A terminal user can run a DECnet utility or issue a command to manipulate remote files.
- A terminal user can run a DECnet utility or issue a command to execute a command file in a remote node.

Like other DECnet functions, remote file access requires the cooperation of two network programs: a program in one node issues a remote file access request, and, in the target node, a DECnet program receives the file access request and translates it into a form recognizable to the local file system. Before the first program can issue the remote access request, the two programs establish a logical link between them by exchanging handshake messages. The program making the request is the source; the program receiving the request is the target.

Terminal Facilities

DECnet provides several terminal facilities for interactive access to the network. These include:

- Terminal-to-terminal communications, provided by the TLK utility
- Direct access to a remote node's operating system
- Remote file access from a terminal

A simple example of a terminal facility would be that of the TLK utility. This terminal-to-terminal communications utility is very helpful to system managers and operators who oversee network operations. A user can invoke TLK at a local terminal to exchange messages with other terminal users in the network, as shown in Figure 5-3.

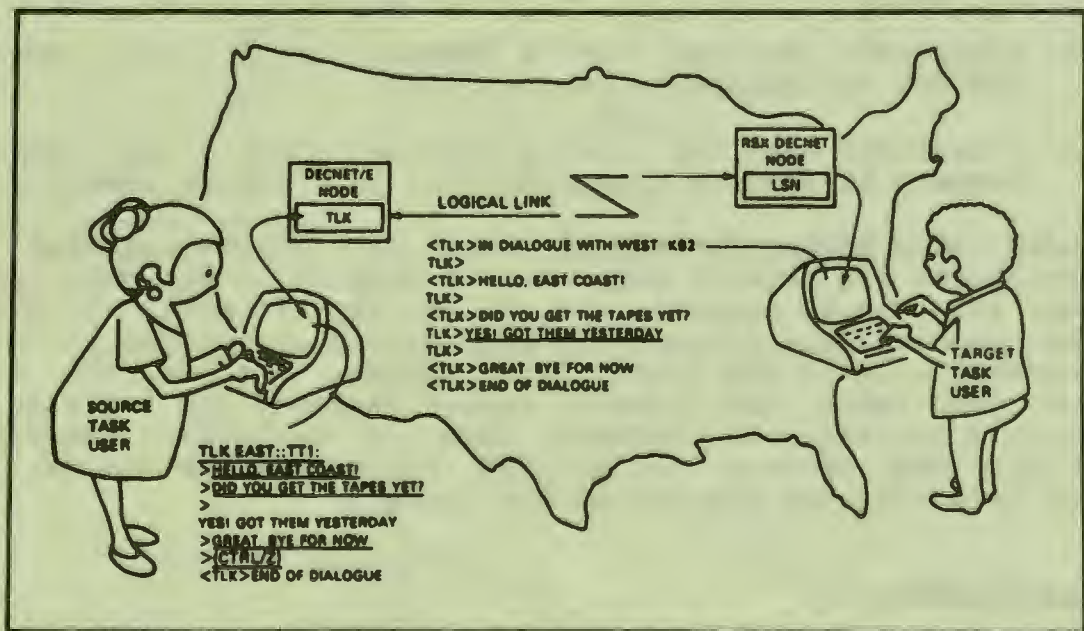


Figure 5-3 TLK Utility

Remote Terminal Facilities

Several DECnet products have a utility that logically connects a local terminal to a remote node's operating system. Utilizing the various utilities available, connections can be made between nodes that run the same operating system.

By running the applicable DECnet utilities, a terminal user can temporarily become a "local" user of a specific remote node. Figure 5-4 illustrates this process, making use of the Remote Command Terminal (RMT) utility.

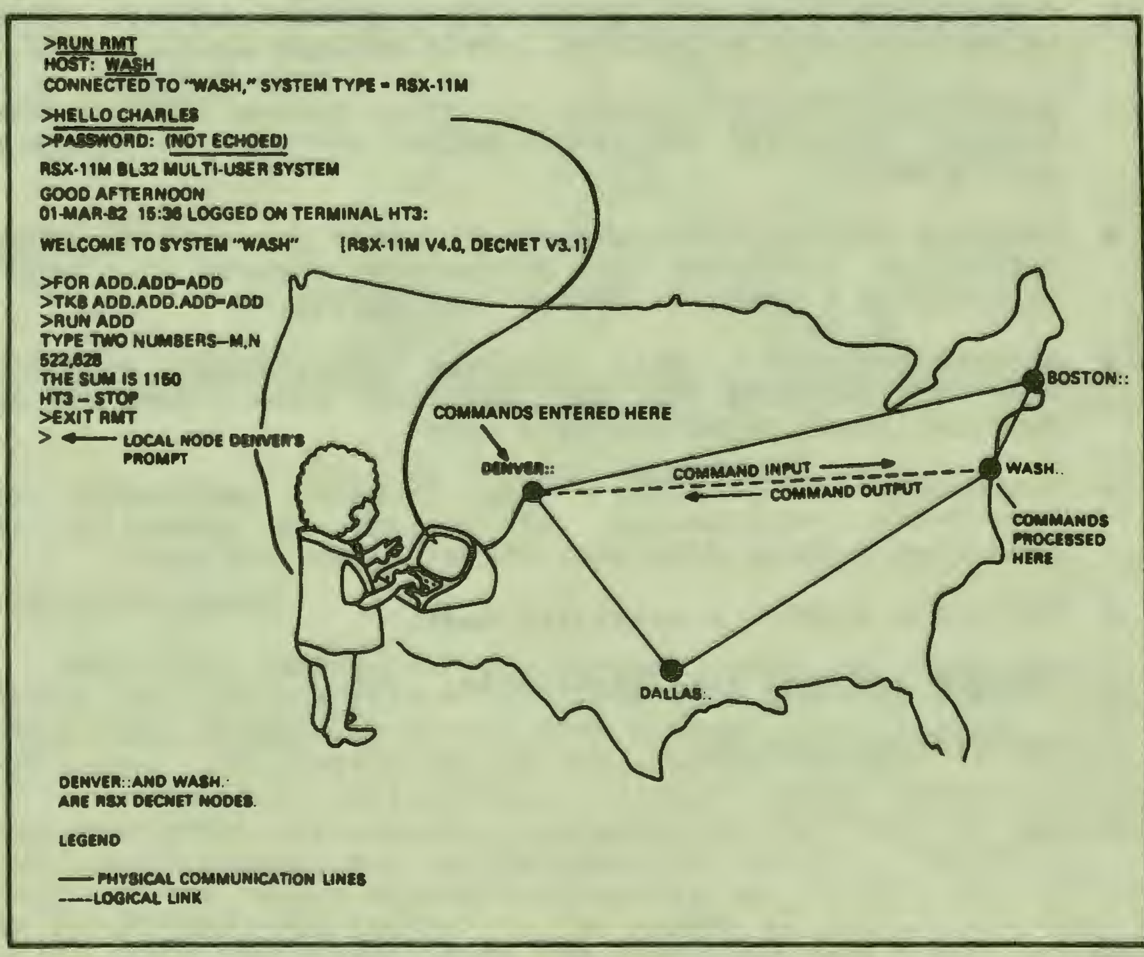


Figure 5-4 Remote Command Terminal Utility (RMT)

Network Management

Those who manage network nodes are responsible both to local and network users and should be aware that while local applications usually demand the greatest share of system resources, the remote users, a potentially very large group, must be sure of each node's response to network applications.

Network system management covers the following:

- DECnet management utilities - These are the network manager's principal tools for controlling and monitoring node activity.
- Planning for node generation - This process tailors DECnet software to suit a specific node's network application.
- Generating network software - This allows the network manager to build and tailor DECnet software to create an active node.
- Defining configuration and other static parameters - These parameters are used to define many aspects of a node's role within a specific DECnet configuration.
- Operating a node - This involves operational functions such as starting up and shutting down a node and the physical lines connected to a node.
- Monitoring node activity - This involves monitoring the day-to-day performance of a node by gathering and analyzing logging data that DECnet makes available.
- Down-line loading a satellite node.
- Perform loopback test procedures.
- Performance testing.

An example of network management responsibilities is illustrated in Figure 5-5.

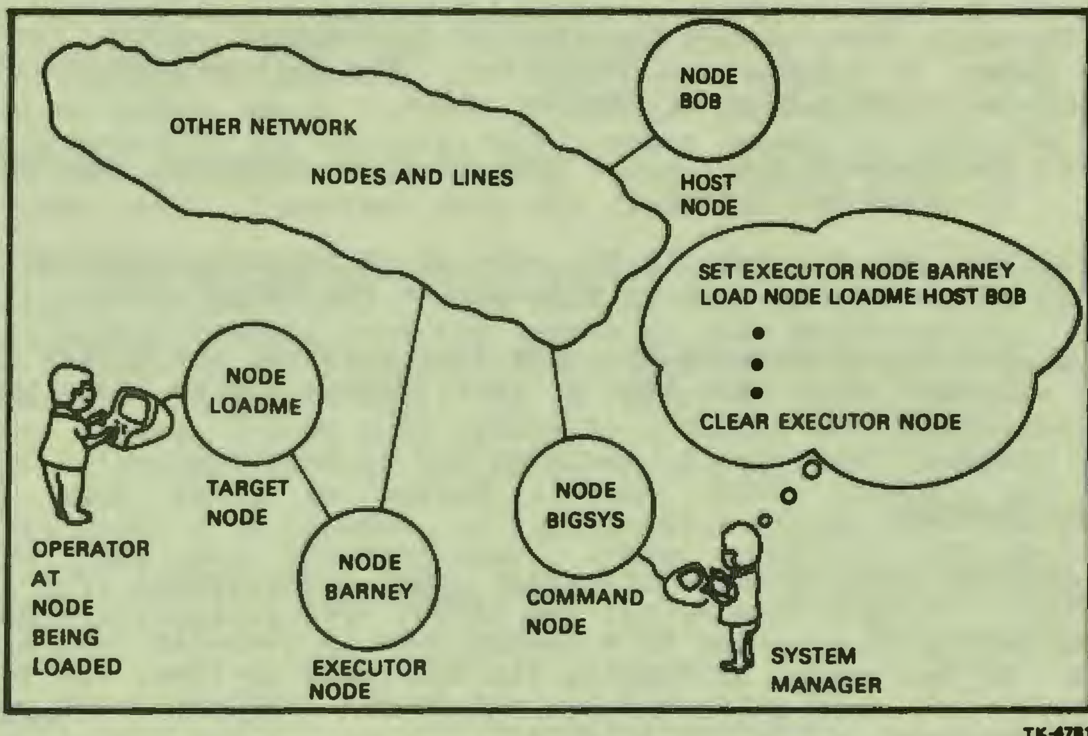


Figure 5-5 Network Management Layer

Down-Line Loading

Down-line loading is the process used to load memory or system image from a file at one node to a separate target node. A target node is usually a node with a memory-only system, with no disk-based file storage of its own. As an example, it is possible that in a hospital environment, a single large computer can be used to control the everyday operation of the hospital and smaller mini- or microcomputers can be used to monitor patients. Even though these mini- and microcomputers may be very powerful, they may be limited with respect to the number of tasks that they can do at any one time. With the down-line load capabilities, the larger computer can transfer the appropriate programs or routines to the smaller mini- or microcomputers when needed. This enhances the ability of the smaller computers to be able to perform many different types of tasks.

The down-load procedure can be initiated in one of two ways. An operator can issue a special command to load the image down-line to the target node, or an operator at the target node can initiate a request for a down-line load.

The down-line loading function is distributed among two or more nodes in a DECnet configuration. The following definitions clarify the roles played by various nodes.

- The command node is the node at which commands requesting the load are issued.
- The executor node is the node which actually executes the commands; it must be adjacent to the target node.
- The target node is the node that receives the system image loaded down the line or that dumps a system image up the line.

Up-Line Dumping

Up-line dumping is a function that complements down-line loading. In up-line dumping, the target node copies the contents of its memory up the line to a remote node in response to a system crash. To be capable of dumping its own image up-line, the target node must have been generated to include the routines which permit this operation; it is not automatic.

Loopback Testing

Loopback tests are procedures that exercise network software and hardware by repeatedly sending data through a number of network components and then returning the data to its source. If a test succeeds, that data loops back to its source without being corrupted. If a test fails, the data does not return to its source, or it returns in a corrupted state. A system manager can run variations of the loopback tests to isolate the network component responsible for losing or corrupting the data. DIGITAL Software Services personnel routinely run loopback tests after installing DECnet software at a node. Successful tests verify that both the software modules and hardware equipment within a node are operating correctly.

Routing

Routing is the network function that determines the physical path or "route" along which data travels to its destination. In a DECnet network, two types of nodes are included; **routing nodes**, which are nodes that can forward packets to other nodes in the network and can be adjacent to all other types of node, and **nonrouting nodes**, which are nodes that can send or receive packets to other nodes in the network, but packets cannot be forwarded or routed through them. Nonrouting nodes can be adjacent to one other node only; therefore they are always an end node.

Two important factors in the execution of full routing are **path** length and **path** cost. (The term path is used to identify the route a packet travels from its source to its destination.) Path length refers to the distance from source to destination measured in **hops**, where a hop is equal to a physical line between two adjacent nodes. Path cost refers to a sum of arbitrary positive integer values assigned to the physical lines that compose the path: each value is called a line cost. Ideally, the most suitable paths from source to destination would be those of minimum hops and minimum cost. (This concept of path cost and length will be discussed again later, for now it might be helpful to refer to Figure 5-9.)

Multipoint

In DECnet, two types of multipoint concepts are used; those which do not require polling **algorithms** (a set of rules or procedures), and those which do. In multipoint environments where DECnet does not use a polling algorithm, point-to-point connections are made between nodes (up to sixteen) by means of a parallel time division multiplexing (TDM) bus.

In multipoint environments where a polling algorithm is required, it is implemented by either software or **firmware** (internal hardware code) at the DDCMP level. In this implementation there is a control station which contains the polling algorithm code. This station is responsible for selecting the Tributary station for communication. Each station monitors the link and waits for a message from the control station. All messages which are addressed to other stations are discarded. If the control station does not have any messages to send to the Tributaries, it then polls each station one at a time to see if the Tributary stations have any data to send to the control station.

THE DNA LAYERS

Modules in a specific layer will typically use services from modules in the layer immediately below it (see Figure 5-6). DNA defines the following layers, described in order from highest to the lowest:

- User layer
- Network Management layer
- Network Applications layer
- Session Control layer
- Network Services layer
- Transport layer
- Data Link layer
- Physical Link layer

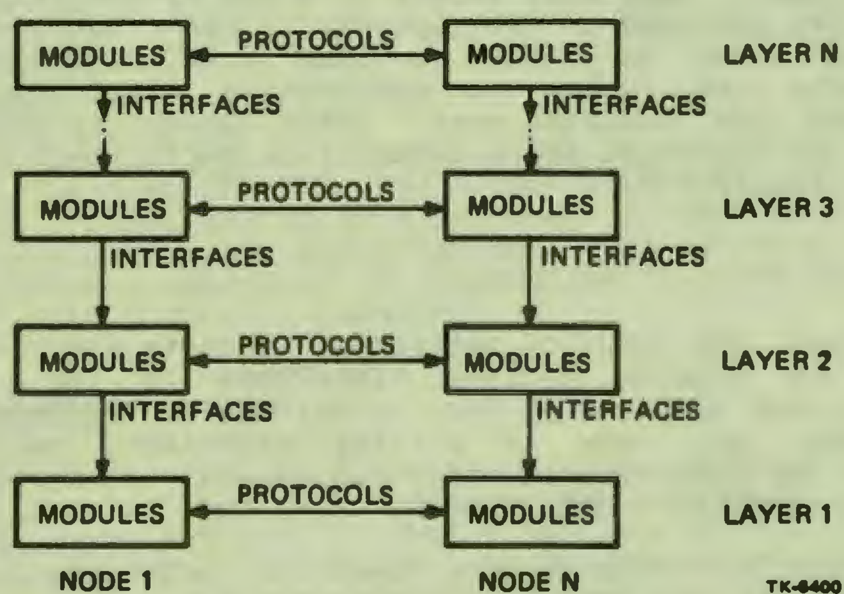


Figure 5-6 DNA Module Services Handling

User Layer

The User layer encompasses user-written programs and services that access the network. It is the highest layer in the architecture. The User layer contains most user-supplied functions. It also contains the Network Control Program, a Network Management module that gives system managers access to lower layers of the architecture.

Network Management

Network Management allows system managers to control and monitor network operation. Network Management also provides information for use in planning the evolution of a network and correcting network problems.

The Network Management design has three outstanding characteristics:

- Both programs and terminals can access and control a DECnet network via a set of functionally discrete calls and commands.
- Control over a DECnet network can be either distributed or central.
- Network Management is a set of "tools." The system or network managers can fashion them into a management system that meets their specific needs. This allows managers to construct their own network management philosophy.

Most of the Network Management modules reside in the Network Management layer. In addition to these, however, there are Network Management modules in the User and Network Applications layers. Since Network Management is modular, a DECnet system is not required to implement the architecture in its entirety.

Network Management performs the following functions:

- Loading and dumping remote stations (such as down-line loading an operating system into an unattended remote system)
- Changing and examining network parameters
- Examining network counters and events that indicate how the network is performing
- Testing links at both the data link and logical link level
- Setting and displaying the states of lines and nodes
- Performance testing

The Network Applications Layer

The Network Applications layer is designed to contain a number of separate, commonly-used modules that access data and provide other often-used services to users.

The Network Applications layer supports the various user services and programs that utilize the network facilities. These services and programs must utilize the network communications mechanism provided by the Session Control and Network Services layers.

The Session Control Layer

The Session Control layer, residing immediately above the Network Services layer, provides system-dependent, task-to-task communications functions. These functions bridge the gap between the Network Services layer and the logical link functions required by processes running under an operating system. Thus, Session Control is the point at which DECnet is integrated with an operating system.

Session Control functions include:

- Mapping node names to node addresses
- Identifying end users
- Activating or creating processes
- Validating incoming connect requests

Session Control defines two data bases that are not implementation-specific:

- A node-name mapping data base

- A data base containing the states of Session Control and optional default connection timers

This layer also defines the "system-dependent" aspects of logical link communications. Figure 5-7 shows a model of Session Control operating within a network node.

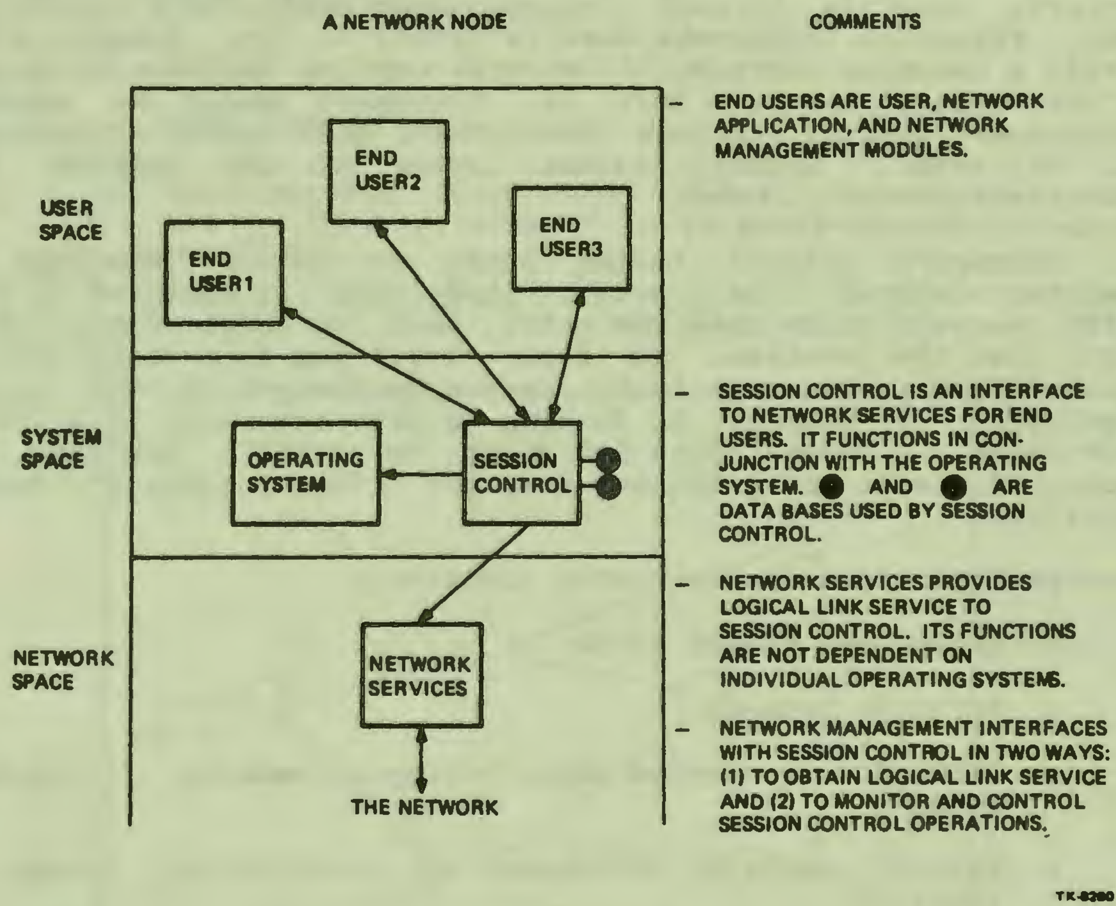


Figure 5-7 Session Control Layer

The Network Services Layer

The Network Services layer, residing immediately above the Transport layer, provides a system-independent process-to-process communication service that allows two processes to exchange data reliably and sequentially, regardless of their locations in a network. Communication is on a connection basis. A connection between two processes, the logical link permits two-way simultaneous transmission of normal data messages, and independent two-way simultaneous transmission of interrupt messages.

The Transport Layer

Transport is a message delivery service. Transport accepts messages, called packets within the context of Transport from the Network Services layer in a source node; and forwards the packets, possibly through intermediate nodes, to a destination node. Transport implements what is termed in the communications field, a Datagram Service. A Datagram Service delivers packets on a "best effort" basis. That is, Transport makes no absolute guarantees against packets being lost, duplicated, or delivered out of order. Rather, higher layers of DNA provide such guarantees.

Transport selects routes based on network topology and operator-assigned line costs. (Low cost is assigned to lines which you want to be used the most, such as high speed lines; high cost is assigned to lines which you want to be used the least.) Transport automatically adapts to changes in the network topology; for example, by finding an alternate path if a line or node fails. Transport does not adapt to traffic loading: The amount of traffic on a channel does not affect Transport's routing algorithms.

Transport provides the following functions:

- Determines packet paths
- Forwards packets
- Periodically updates other Transport modules on adjacent nodes
- Returns packets addressed to unreachable nodes, if required
- Delivers packets between nodes
- Manages buffers
- Limits the number of nodes a packet can visit
- Limits the amount of time a packet can spend in a node
- Performs node verification (exchange of passwords), if necessary
- Monitors errors detected by the Data Link layer
- Maintains counters and keeps track of events for network management purposes

Transport allows for a functional subset so that in terms of routing there can be two types of DECnet implementations:

1. **Routing** - Sometimes called full routing, this is the full complement of Transport components (refer to Figure 5-8). A routing node can:

   - Receive packets from any other node.
   - Send packets to any other node.
   - Route packets from other nodes, through to other nodes. This is referred to as route-through or packet switching.

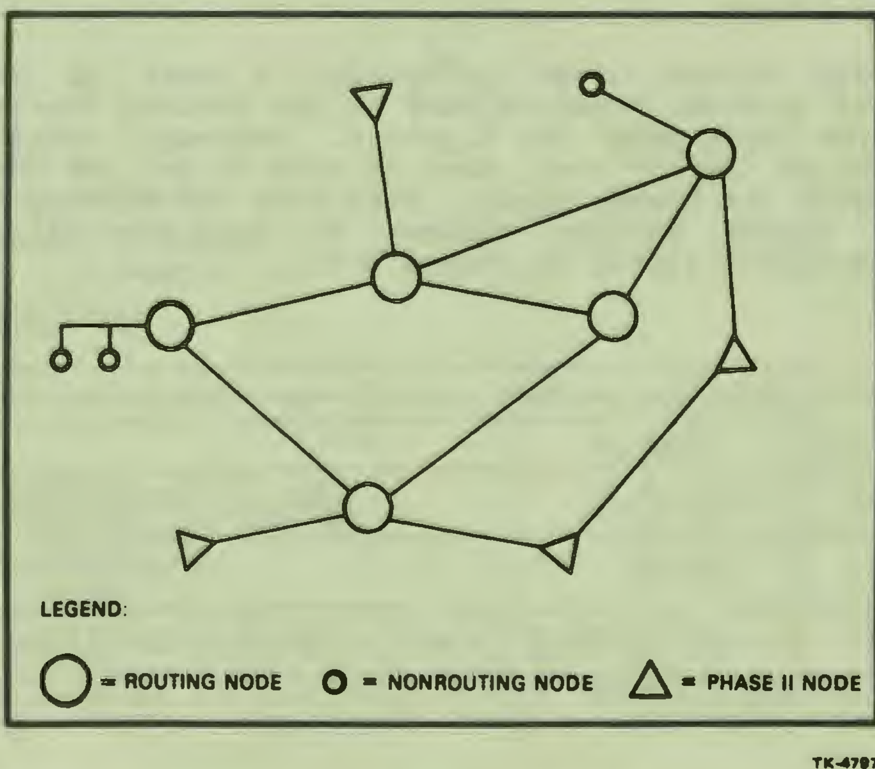


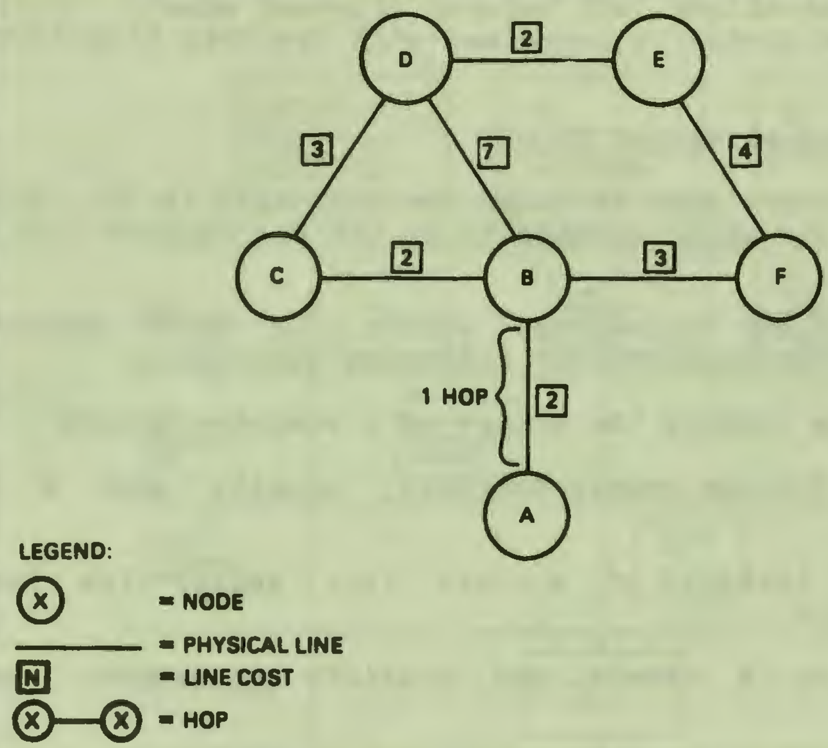
Figure 5-8 Routing Node

2. Nonrouting - This is a subset of the Transport components (refer to Figure 5-8). A nonrouting node can:

   - Send packets from itself to any other node.
   - Receive packets addressed to itself from any other node.
   - Only be placed as end nodes in a network.
   - May be physically connected to a network by only one physical link.

The logical distance from one node to another in a network is called a hop. The complete distance that a message travels from source to destination is called the path length. The maximum number of hops that the routing algorithm will forward packets is called maximum hops.

Transport allows users to assign a cost to each line connected to a node. Cost is used in the routing algorithm that determines the "best path" for a packet. Transport routes packets on the path of "least" cost, even if this is not the most direct path (path with the fewest hops). Both cost and maximum hops are values the system manager assigns at each node using Network Management software (refer to Figure 5-9).



NODE A WANTS TO SEND A PACKET TO NODE D. THERE ARE THREE POSSIBLE PATHS.		
	PATH COST	PATH LENGTH
A TO B, B TO C, C TO D	2 + 2 + 3 = 7*	3 HOPS
A TO B, B TO D	2 + 7 = 9	2 HOPS
A TO B, B TO F, F TO E, E TO D	2 + 3 + 4 + 2 = 11	4 HOPS

*7 IS THE LOWEST PATH COST; NODE A THEREFORE ROUTES THE PACKET TO NODE D VIA THIS PATH.

TK-4784

Figure 5-9 Nonrouting Node

The Data Link Layer

As previously discussed, the Data Link layer resides immediately above the Physical Link layer and is responsible for creating a communications path between adjacent nodes. Currently there are two DNA protocols concerned with the Data Link layer:

- DDCMP
- MOP (operates within DDCMP)

Since the normal mode of DDCMP was discussed in the previous module, we will now only concentrate on the maintenance mode (MOP) of DDCMP.

MOP messages are transmitted within the DDCMP maintenance mode envelope. MOP performs the following functions:

- Down-line loading the memory of a computer system

- Up-line dumping memory contents, usually upon a system failure

- Loopback testing of a data link and/or its hardware components

- Restarting a remote and possibly unattended computer system

At the request of a higher-level module, MOP sends an appropriate message within the DDCMP envelope. MOP messages and functions handle all message acknowledgments, time-out, and retransmission functions.

It is possible for a node on a multipoint link to be in the MOP mode, executing MOP protocol while the other nodes are on-line executing DDMCP protocol.

DECnet MODULE INTERFACES

DNA determines how DECnet software modules interact vertically with one another; the interfaces between modules in separate layers of the same node are precisely defined. Reflecting the structure of DNA, DECnet modules are like building blocks: within each node, a layer contains only those modules required to support modules in higher layers. Depending on the layer, one or more modules may be required to provide the necessary support. Each module requests and provides services according to the rules defined by DNA.

Figure 5-10 illustrates a collection of modules residing in a typical DECnet node. The arrows represent the interfaces between modules. The arrows point down because each module uses the services provided by a module in a lower layer; a module cannot use services provided by a higher-level module.

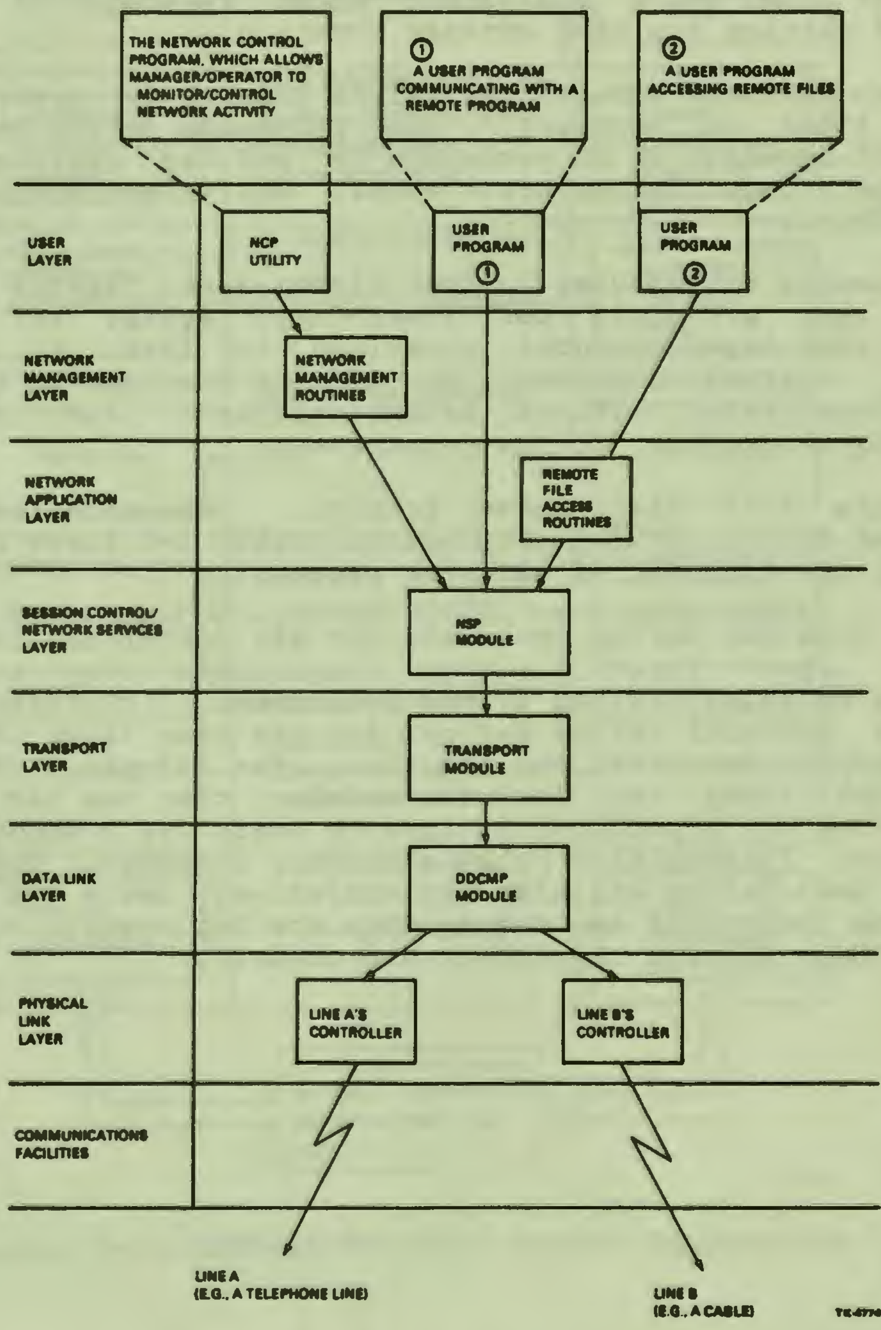


Figure 5-10 Transport Layer

DNA PROTOCOLS

So far, DNA has been viewed in the context of an individual node. However, in addition to defining vertical interfaces, DNA also defines the relationship between modules in separate nodes: a module in one node communicates only with an equivalent module in another node, where "equivalent" means resident in the same layer and serving the same network function.

Communications between equivalent modules is governed by a set of rules of protocol. Most protocols define the form and content of messages to be exchanged by modules residing in the same layer, but in separate nodes. Equivalent modules use the same protocol.

Protocols for modules in the higher-level layers are more complex than protocols for lower-level layers. For example, a Physical Link layer protocol is defined in terms of electrical signals; whereas a protocol for modules resident in the Network Applications layer defines message formats and rules for exchanging messages.

Figure 5-11 illustrates protocol communication between equivalent modules in separate nodes. Table 5-1 lists and briefly describes the function of each DNA protocol.

DNA does not define protocols for all functional layers. For example, User layer programs communicate over the network according to rules defined by the programmer. Furthermore, more than one protocol can be defined for the same layer because some layers support more than one function. For example, the Network Application layer can include modules that use DAP as well as modules that use a protocol defined by users for a specfic network application (transaction processing, for example). The protocols that DNA does define are also not exclusive; users can substitute their own protocols as long as they are implemented consistently by equivalent modules throughout the network.

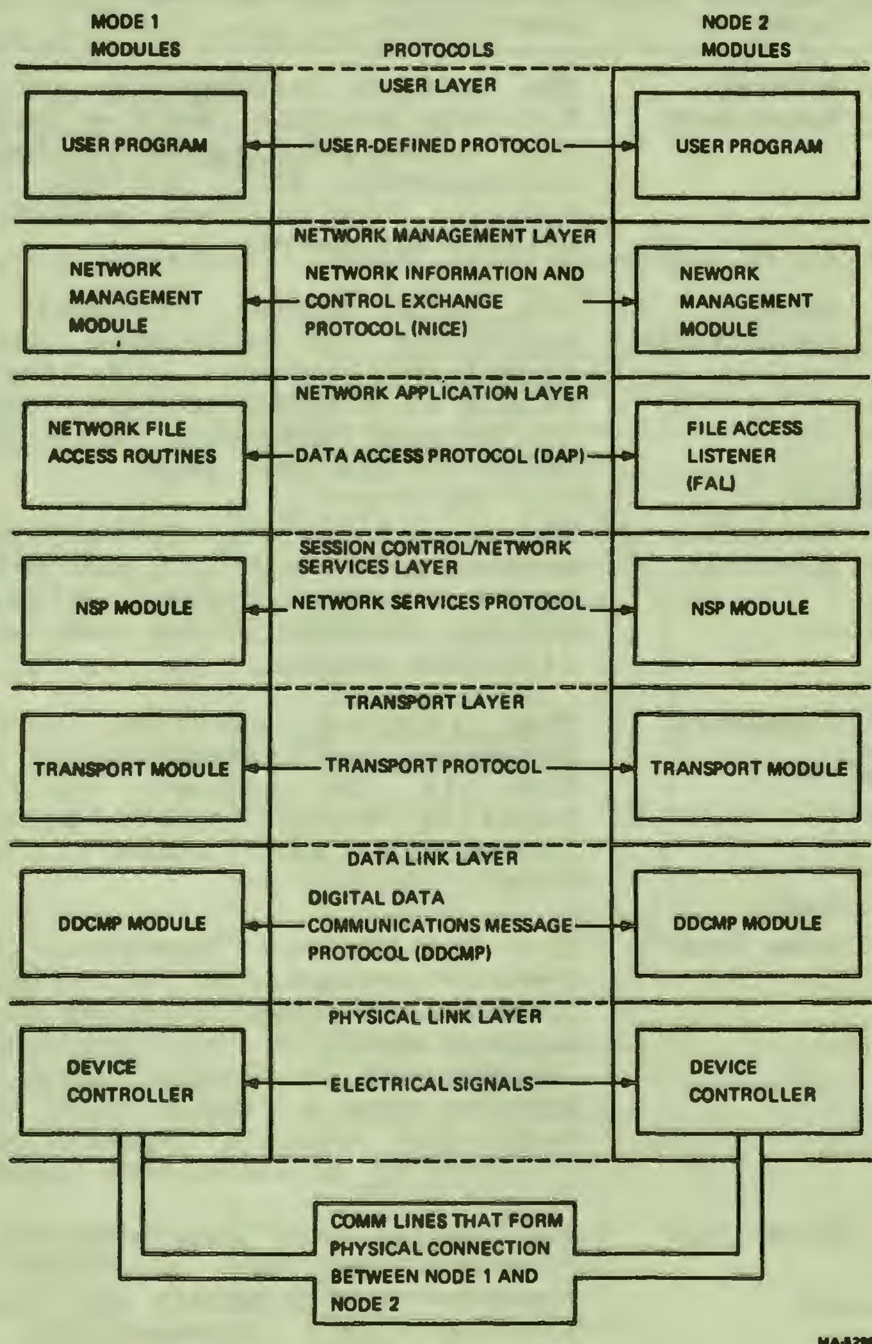


Figure 5-11 DECnet Software Module Interaction

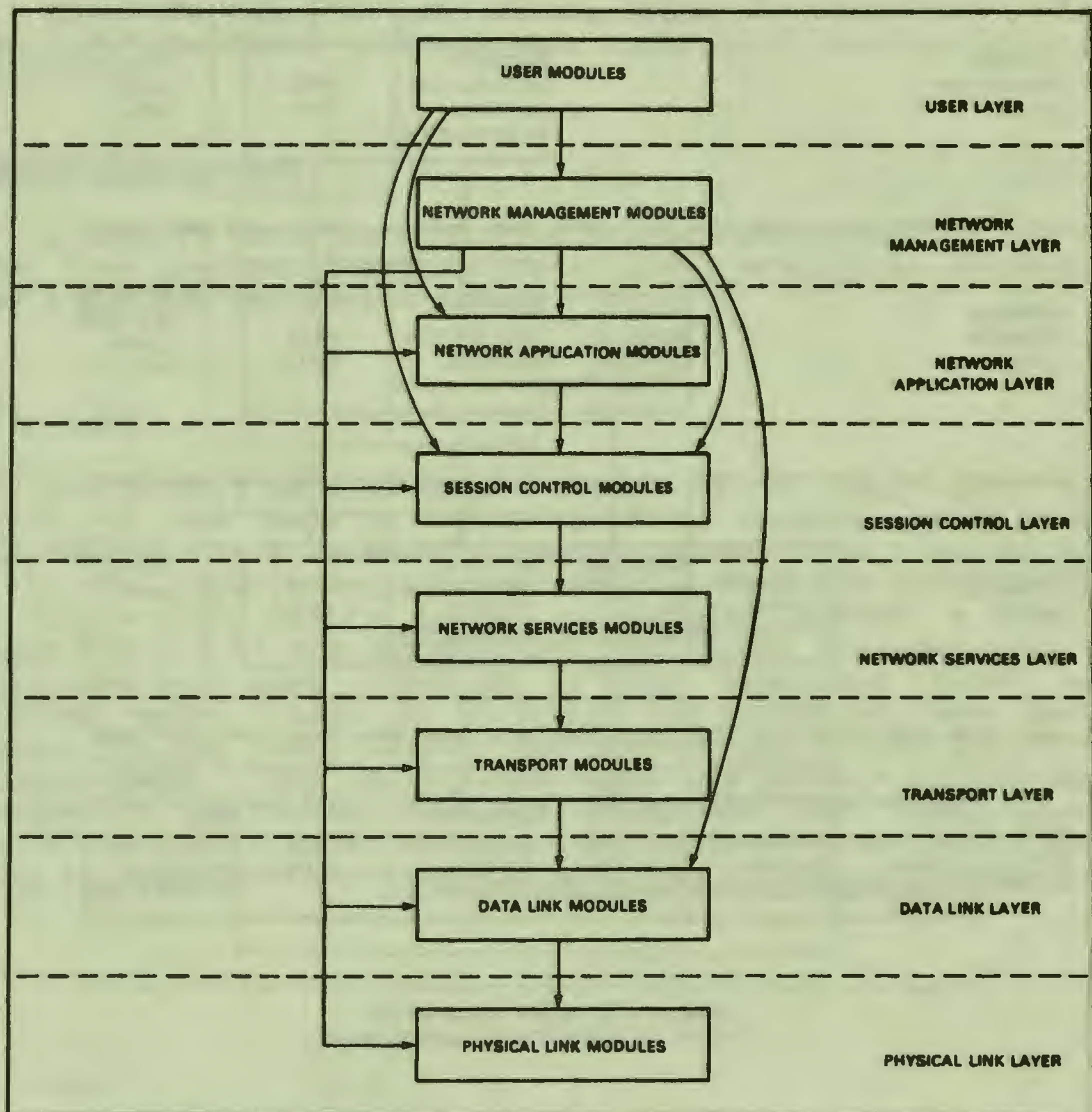
Table 5-1 DNA Protocols

Protocol	Layer	Description
NICE	Network Management	The Network Information and Control Exchange protocol defines mechanisms for exchanging network, node, and configuration data, and for servicing requests from modules residing in the Network Management layer.
DAP	Network Application	The Data Access Protocol defines mechanisms for performing remote file access and remote file transfer on behalf of software modules residing in the Network Management layer (Phase III only) and the User layer.
NSP	Network Services	The Network Services Protocol defines a mechanism for creating and maintaining logical links between higher-level modules residing in the same node or in different nodes.
Routing	Transport	The routing protocol defines a mechanism for dispatching data to any node in the network by the best possible route. This protocol is implemented with Phase III products only.
MOP	Data Link	The Maintenance Operation Protocol defines mechanisms for transmitting data over a communications channel to achieve specific functions: down-line loading of a remote node; up-line dumping from a remote node; testing a node and network connections; and starting up an unattended remote node.
DDCMP	Data Link	The Digital Data Communications Message Protocol defines a mechanism for ensuring the integrity and sequentiality of data transmitted over a communications channel.

USER SERVICES

As shown in Figures 5-12 and 5-13, there are different ways in which modules use the services of other modules in a system. DNA allows the following services:

- User-to-user
- Remote application
- Network management



HORIZONTAL ARROWS SHOW DIRECT ACCESS FOR CONTROL AND EXAMINATION OF PARAMETERS, COUNTERS, ETC. VERTICAL AND CURVED ARROWS SHOW INTERFACES BETWEEN LAYERS FOR NORMAL USER OPERATION SUCH AS FILE ACCESS, DOWN-LINE LOAD, UP-LINE DUMP, END TO END LOOPING, AND LOGICAL LINK USAGE.

Figure 5-12 DNA Protocol Usage

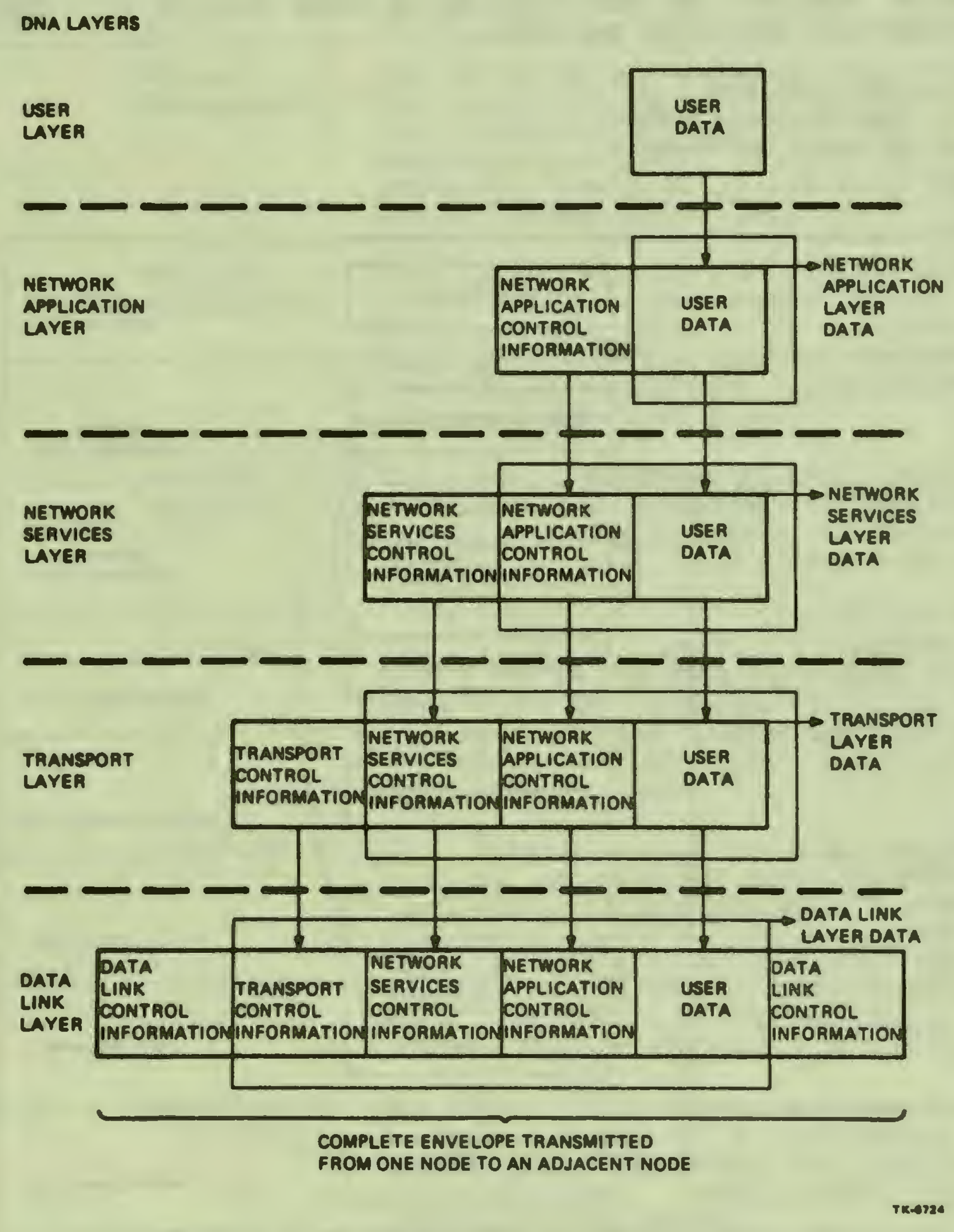


Figure 5-13 DNA Layers and Interfaces

User-to-User

With this services a user level process in one node can communicate via a logical link with a user process in another node.

Remote Application

Using the Network Applications module to establish a logical link, a user-level process can perform a function at a remote node.

Network Management

A Network Management service can be performed at a remote node from the user-level by performing a command or process which uses a Network Management module.

DATA FLOW

The primary purpose of a network is to pass data from a source in one node to a destination in another. Data traveling from one node in a network to another passes from a source process in the user layer, down through each layer of the DNA hierarchy of the source node, before being transmitted across a line (see Figure 5-13). If the destination node is not adjacent (see Figure 5-14), the data must travel up to the transport layer of the adjacent node, where it is routed (or switched), sent back down through the two lower layers, and transmitted across the next line in the path. The data keeps traveling in this manner until it reaches its destination node. At this node, the data passes up the hierarchy of layers to the destination process. The numbered steps on Figure 5-14 correspond with the following explanation:

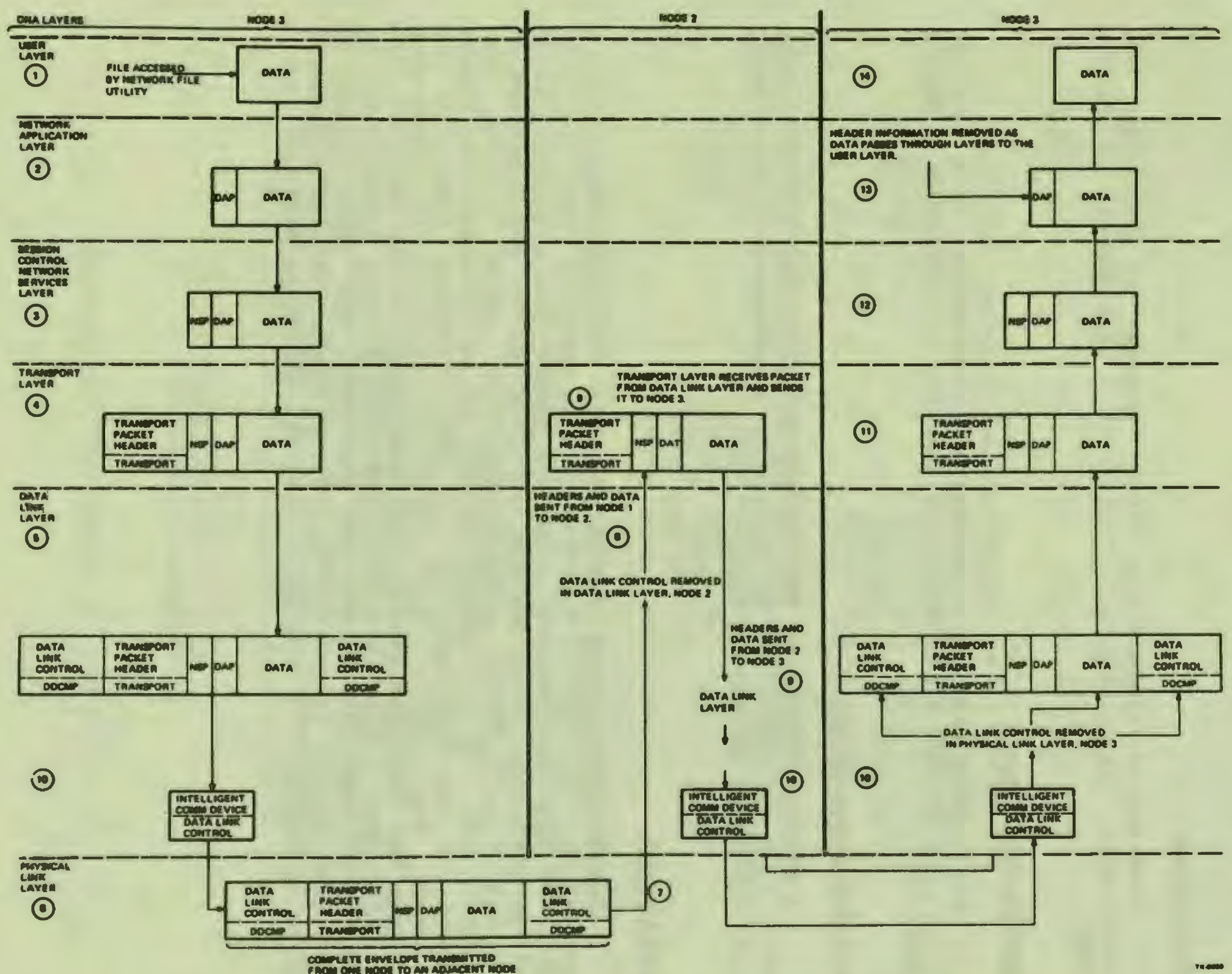


Figure 5-14 DNA Message Building and Processing

Data Flow at the Source Node

1. The source user requests a connection to the destination user and passes connect data.

2. The Session Control module receives the data, determines a destination address and places the data in a transmit buffer, adding control information to the message.

3. The Network Services modules adds its own control information (including a logical link identification), and passes the message (called a datagram) to the Transport layer.

4. The Transport module adds a header consisting of the destination and source node addresses. The Transport module then selects an outgoing channel for the message based on routing information. The Transport module then passes the message (now called a packet) to the Data Link layer.

5. The Data Link module adds its protocol header information and prepares the enveloped message for transmission.

6. The Physical Link module transmits the enveloped message over the data link.

Data Flow Across the Network to the Destination Node

7. The message is received by the Physical Link module and is passed to the Data Link layer.

8. The Data Link module checks the packet for errors. If errors are detected, the appropriate response is made and the message is retransmitted. If the message was received without error, the Data Link header information is removed and the message is passed on to the Transport layer.

9. The Transport module checks the destination address in the Transport header and determines the address. If the node address is different then its own (which it is in this example), the message is passed along to the Data Link module where it is prepared for further transmission. The message proceeds as in steps 5 and 6 above.

10. The message proceeds through the network, switching at routing nodes, until it arrives at the Transport layer with the same address as the destination address in the message.

Data Flow at the Destination Node

11. The packet passes to the Transport layer of the destination node as described in steps 7 and 8 above. The destination Transport module removes the Transport header, and passes the datagram to Network Services.

12. The Network Services module examines the Network Services header on the datagram. If it has resources to form a new logical link, Network Services passes the connect data, without the Network Services control information, to Session Control.

13. The Session Control module performs any necessary access control functions and passes the message to the appropriate process in the User layer after removing Session Control header information.

14. The destination process interprets the data according to whatever higher-level protocol is being used.

EXERCISES

1. Circle the letter of the phrase which best describes the purpose of the Data Link layer.

   A. Provides the network with logical links.

   B. Provides the functions used to plan, control, and maintain the operation of the network.

   C. Provides error-free, physical links between adjacent nodes.

   D. None of the above.

2. Circle the letter of the phrase which correctly completes the statement. The Transport layer interfaces with the following layers:

   A. Network Services layer and Network Management layer.

   B. Session Control layer, Data Link layer, and Network Management layer.

   C. Network Services layer, Data Link layer, and Network Management layer.

   D. Data Link layer and Session Control layer.

3. Circle the letter of the protocol used by the User layer.

   A. DAP

   B. DDCMP

   C. NSP

   D. None of the above.

4. Circle the letter of the phrase which correctly completes the statement. The network user at the User level has interfaces to the following layers:

   A. Session Control layer and Network Management layer.

   B. Session Control layer, Network Applications layer, and Network Management layer.

   C. Session Control layer, Transport layer, and Network Applications layer.

   D. None of the above.

5. In the spaces provided, match the following questions with the answers below. (Each answer may be used one or more times, or not at all.)

   ___ Which DNA layers are made transparent to the network user by the DNA user layers?

   ___ What is the purpose of the Network Applications layer?

   ___ Which DNA layers provide the interfaces to DNA?

   ___ What is the purpose of the Network Management layer?

   ___ How are task-to-task communications defined?

   A. User layer, Network Management layer, and Network Applications layer.

   B. Session Control layer, Network Services layer, Transport layer, Data Link layer, and Physical Link layer.

   C. It provides a layer in the DNA model for generalized applications software packages that can be used by all users.

   D. It provides the functions to plan, control, and maintain the network.

   E. Network Management layer and Session Control layer.

   F. The ability of two user programs to exchange data over a logical link.

   G. The ability of two user programs to exchange data over a physical link.

6. The Data Access Protocol (DAP) is the only protocol that can ever be in a Network Applications layer. (Circle A, B, C, or D.)

   A. True, with the possible exception of a transaction processing protocol.

   B. False, DEC or any DECnet user can implement other protocols that could be with DAP in the Network Applications layer.

   C. True, DAP is needed by the Network Applications layer.

   D. False, DAP is needed by the Network Services layer.

7. The Session Control layer is where DECnet products interface with the operating system. (Circle A or B).

   A. False, the Network Applications layer provides this interface.

   B. True, the Session Control layer is system dependent.

8. Circle the letter of the phrase that best describes the main purpose of the Network Services layer.

   A. To provide functions to control the network.

   B. To create logical links and transmit data over these links.

   C. To provide physical links to adjacent nodes.

   D. To create data file functions.

9. Match the following items with the phrases that best describe them. Each item may be used one or more times, or not at all.

   A. Datagram

   B. Route-through

   C. End node

   D. Transparent node

   E. Hop

   F. Nonrouting node

   G. Line cost

   H. Remote node

   I. Routing node

   J. Message

   K. Jump

   ___ Unit of data passed between Transport and Network Services layers. Becomes a packet when a route header is added.

   ___ Topological definition of a nonrouting node.

   ___ The logical distance between two adjacent nodes in a network.

   ___ A positive integer value connected with using a line. Used to determine message routes.

   ___ A DECnet node, connected to the network by a single line. Can send and receive packets.

   ___ To one node, any other node in the network.

   ___ Directing packets from source node to destination node through one or more intervening nodes.

   ___ Another term for packet switching.

   ___ A DECnet node that can deliver, receive, and route packets through.

SOLUTIONS

1. Circle the letter of the phrase which best describes the purpose of the Data Link layer.

   A. Provides the network with logical links.

   B. Provides the functions used to plan, control, and maintain the operation of the network.

   (C.) Provides error-free, physical links between adjacent nodes.

   D. None of the above.

2. Circle the letter of the phrase which correctly completes the statement. The Transport layer interfaces with the following layers:

   A. Network Services layer and Network Management layer.

   B. Session Control Layer, Data Link layer, and Network Management layer.

   (C.) Network Services layer, Data Link layer, and Network Management layer.

   D. Data Link layer and Session Control layer.

3. Circle the letter of the protocol used by the User layer.

   A. DAP

   B. DDCMP

   C. NSP

   (D.) None of the above.

4. Circle the letter of the phrase which correctly completes the statement. The network user at the User level has interfaces to the following layers:

   A. Session Control layer and Network Management layer.

   (B.) Session Control layer, Network Applications layer, and Network Management layer.

   C. Session Control layer, Transport layer, and Network Applications layer.

   D. None of the above.

5. In the spaces provided, match the following questions with the answers below. (Each answer may be used one or more times, or not at all.)

   __B__ Which DNA layers are made transparent to the network user by the DNA user layers?

   __C__ What is the purpose of the Network Applications layer?

   __A__ Which DNA layers provide the interfaces to DNA?

   __D__ What is the purpose of the Network Management layer?

   __F__ How are task-to-task communications defined?

   A. User layer, Network Management layer, and Network Applications layer.

   B. Session Control layer, Network Services layer, Transport layer, Data Link layer, and Physical Link layer.

   C. It provides a layer in the DNA model for generalized applications software packages that can be used by all users.

   D. It provides the functions to plan, control, and maintain the network.

   E. Network Management layer and Session Control layer.

   F. The ability of two user programs to exchange data over a logical link.

   G. The ability of two user programs to exchange data over a physical link.

6. The Data Access Protocol (DAP) is the only protocol that can ever be in a Network Applications layer. (Circle A, B, C, or D.)

   A. True, with the possible exception of a transaction processing protocol.

   B. False, DEC or any DECnet user can implement other protocols that could be with DAP in the Network Applications layer.

   C. True, DAP is needed by the Network Applications layer.

   D. False, DAP is needed by the Network Services layer.

7. The Session Control layer is where DECnet products interface with the operating system. (Circle A or B).

   A. False, the Network Applications layer provides this interface.

   B. True, the Session Control layer is system dependent.

8. Circle the letter of the phrase that best describes the main purpose of the Network Services layer.

   A. To provide functions to control the network.

   B. To create logical links and transmit data over these links.

   C. To provide physical links to adjacent nodes.

   D. To create data file functions.

9. Match the following items with the phrases that best describe them. Each item may be used one or more times, or not at all.

   A. Datagram

   B. Route-through

   C. End node

   D. Transparent node

   E. Hop

   F. Nonrouting node

   G. Line cost

   H. Remote node

   I. Routing node

   J. Message

   K. Jump

   __A__ Unit of data passed between Transport and Network Services layer. Becomes a packet when a route header is added.

   __C__ Topological definition of a nonrouting node.

   __E__ The logical distance between two adjacent nodes in a network.

   __G__ A positive integer value connected with using a line. Used to determine message routes.

   __F,C__ A DECnet node, connected to the network by a single line. Can send and receive packets.

   __H__ To one node, any other node in the network.

   __B__ Directing packets from source node to destination node through one or more intervening nodes.

   __B__ Another term for packet switching.

   __I__ A DECnet node that can deliver, receive, and route packets through.

SYSTEM NETWORK ARCHITECTURE

6

INTRODUCTION

The System Network Architecture (SNA) is a set of definitions and specifications for an IBM computer network. Under IBM's definition, a network can:

> "range all the way from a single small processor that supports one or two terminals to complicated interconnections in which tens of processing units of various sizes are interconnected to one another and to thousands of terminals."*

The Systems Network Architecture (SNA) defines all the functions required for IBM processors, controllers, and terminals to exchange messages over telecommunications links. The SNA architects have three main problems to solve.

First, they must define an entity that can exert central control over the activities of the network, where desired. As a manufacturer of mainframe computers, IBM continues to support its centralized approach to data processing and data communications. At the same time, the architects must define a network flexible enough to allow for the increasing distribution of functions to remote sites.

Second, the architects must define a way to control the operation of the physical components in the network (to turn them on and off, and so on) from a central location. Physical components include communications controllers, cluster controllers, terminals, and physical links.

Finally, the network architects must define a mechanism by which end users can exchange messages without concern for the communication functions performed by the network. One of the principal goals of SNA is to make communication functions transparent to the user.

Note that SNA, in contrast to the DIGITAL Network Architecture (DNA), includes terminal networks; that is, networks which are made up of a single processor connected to one or more remote job entry devices or interactive terminals. A DNA network must contain at least two communicating processors.

*P.E. Green, "An Introduction to Network Architectures and Protocols," IBM Systems Journal, 18:2, 1979, p. 202.

To see how SNA solves these problems, one must understand the following concepts:

- End users
- Network addressable units
- Sessions
- Functional layers
- Peer protocols
- Function subsets
- Domains

OBJECTIVES

Describe the following features of IBM's System Network Architecture (SNA):

- SNA layers
- Network Addressable Units (NAU)
- SNA nodes
- Software products
- SNA capabilities

RESOURCES

1. IBM Systems Network Architecture (SNA) Format and Protocol Reference; Architecture Logic, (SC30-3112).

2. IBM Systems Network Architecture (SNA) General Information, (GA-3102).

3. Cypser, R.J. Communications Architecture for Distributed Systems. Reading, Massachusetts, 1978.

OVERVIEW

IBM applies the term SNA to both the architecture itself and the software and hardware products that implement the architecture. "SNA communications products can perform functions that were formerly done by the main processor. The functions distributed to these products may include the management of communication lines, device control, data formatting, and in some cases, execution of application programs."*

Here again, we can see the difference between IBM's and DIGITAL's definition of a network. In an SNA implementation, nodes may in some cases be able to execute applications programs. In a DECnet implementation, on the other hand, all nodes without exception have the ability to execute applications programs. This ability is basic to distributed processing.

The difference between SNA and DNA reflects the differences between the two companies that developed them. DIGITAL developed DNA as a way to connect many stand-alone minicomputer systems. IBM created SNA to connect many terminal users to a central IBM central processor. IBM's original goal in designing SNA was not distributed processing, but instead distributed access (through different terminals) to centralized processing. This approach is consistent with the needs of IBM's traditional customers: large commercial organizations, such as banks and insurance companies, which must maintain control over their central data bases.

An SNA terminal user is provided with all the functions needed to perform a specific job, for example, running a bank teller's window, shipping materials from a warehouse, or entering a new insurance policy. Network control and all data is centralized at the host. Program development and direct access to the system's resources are reserved for the system programmers at the central data processing center.

IBM offers special terminal systems for supermarkets, banks, and retail stores. These systems store local transaction data and send it to the host central processor at the end of each working day. The host maintains a central company data base and provides all data processing for the local work areas.

Recently, IBM has started to offer general-purpose processors to meet the increasing demand for equipment that can provide local processing capabilities.

*<u>Systems Network Architecture General Information</u>, IBM (GA27-2102-C), 1975.

END USERS

A network provides a mechanism to reliably transfer data between end users. An application program running in the host is one type of end user. Terminal operators and physical devices are other types of end users.

End users are not part of the network itself. Rather, they are the ultimate sources and destinations of the data transmitted over the network.

NETWORK ADDRESSABLE UNITS

A Network Addressable Unit (NAU) is an intelligent component (software or hardware logic) to which a message can be sent. Each NAU is assigned a unique address. Within the network, all communication takes place between NAUs.

SNA defines three types of NAU:

- Logical Units (LU)
- System Service Control Point (SSCP)
- Physical Units (PU)

Logical Units

A **Logical Unit** (LU) provides an end user with access to a data communications network. Each LU represents an end user: a program, a device, or a terminal operator. The end user passes outgoing messages to the LU, and receives incoming messages from the LU. The network uses LU addresses to route messages from their sources to their proper destinations.

The LUs in a host node are system programs. The users they represent are applications programs or IBM applications subsystems. Certain terminals include hardware logic capable of handling some LU functions for terminal users. For low-function, nonintelligent terminals, LU functions can be performed by the cluster controller to which the terminals are attached.

The System Service Control Point

The **System Service Control Point** (SSCP) is a collection of software modules residing in the host. Each domain must include an SSCP.

The SSCP totally manages the network. It is aware of everything in the host's domain: all physical components, and names and addresses of all users who want to send and receive messages. Users who want to communicate must first make a request to the SSCP.

SSCP functions include bringing up the network, establishing logical connections between addressable units (called sessions, described below), and performing recovery procedures when a component in the network fails to maintain contact.

Users communicate with the SSCP through their LUs, over a special virtual path reserved for control messages. Network operators may communicate with the SSCP by means of a special interface in the SSCP itself.

The Physical Unit

A Physical Unit (PU) is an NAU that controls a physical network component (turning it on and off, and so on) according to instructions received from the SSCP.

It is important to see that in SNA terms, a PU is not literally a physical device. It is an intelligent component (programming or hardware logic) that performs control functions for a device. PUs are involved in activation and deactivation, error recovery, resynchronization, tests, and gathering statistics on the operation of network devices.

The PU for a device may exist either within the device itself or in an attached controlling device. Hosts, communications controllers, and cluster control units contain their own PUs. Some high-function terminals contain their own PU. For PU functions, low-function terminals rely on the terminal control unit, cluster controller, or communications controller to which they are attached.

Distribution of NAUs

Figure 6-1 illustrates the distribution of NAUs. Note that the cluster controller provides LU and PU functions for the attached terminals.

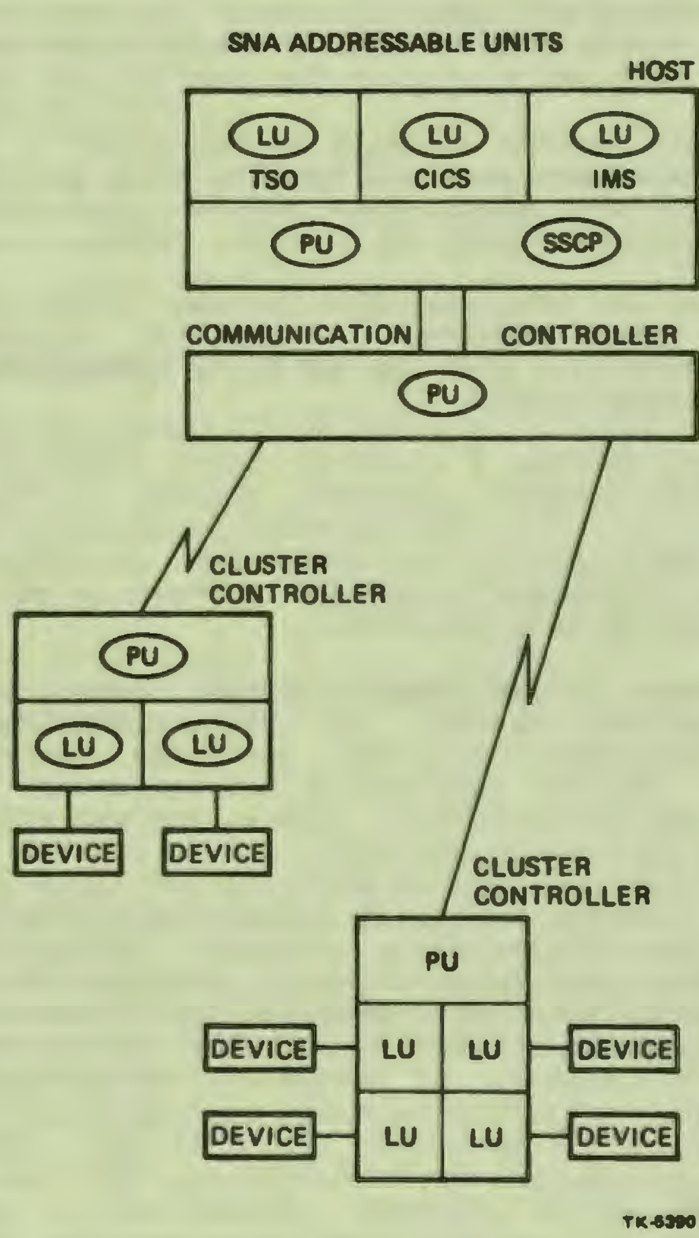


Figure 6-1 LU Modules

Boundary Functions

In many implementations, the host or communications controller performs functions for cluster controllers and terminals in the network. A PU or LU function performed by a host or a communications controller on behalf of another node is called a boundary function.

SESSIONS

A session is a temporary logical connection between two NAUs for the purpose of exchanging messages. (The equivalent to DNA's virtual link.)

In **single-domain networks** (networks developed around a single SSCP) the following types of session are established:

- SSCP to PU (established with each PU as part of the bring-up process)
- SSCP to LU (established with each LU as part of the bring-up process)
- LU to LU (to establish communications between end users)

As part of the process of bringing up the network, the SSCP establishes a session with each PU. The SSCP uses these sessions to control the resources at the various nodes in its domain, and to collect various maintenance and operational statistics.

As part of the bring-up process, the SSCP also establishes a session with each LU in the network. Each LU can support a minimum of two concurrent sessions, one with the SSCP and one with another LU. Some LUs can support concurrent sessions with multiple LUs.

LU-LU sessions are the virtual communication paths by which users communicate. In current implementations, LUs establish sessions through the SSCP, not directly. Figure 6-2 illustrates the different types of session.

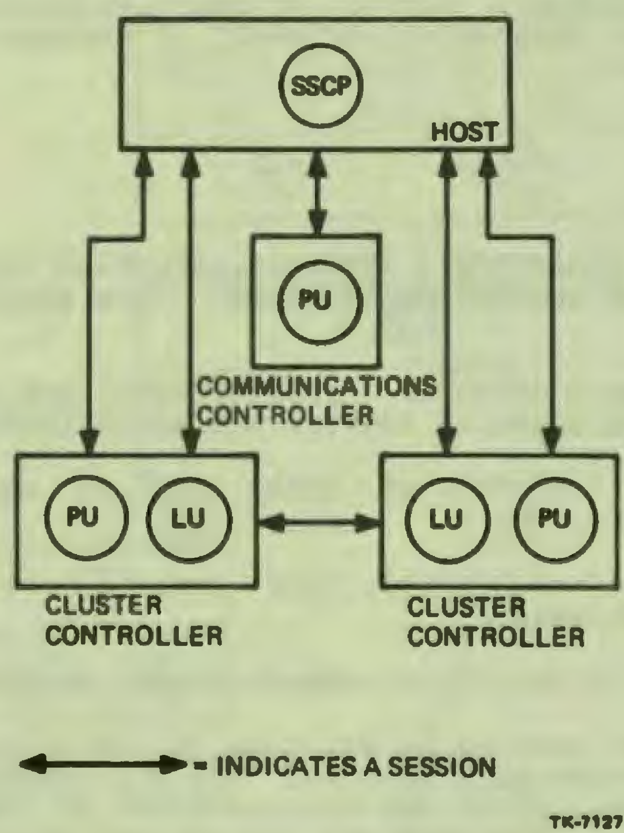


Figure 6-2 Simple SNA Nodes

SNA LAYERS

SNA, like DNA, groups communications functions into logical layers (Figure 6-3) of related functions. Messages are passed "up and down" from layer to layer, across cleanly defined layer boundaries. (In the IBM program products that implement SNA, the boundary lines between layers are not always so clearly marked.)

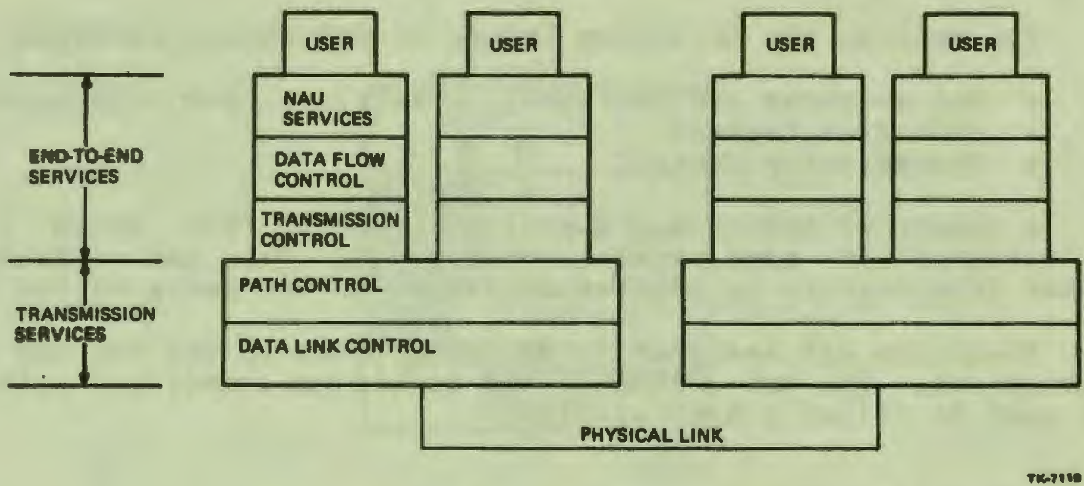


Figure 6-3 SNA Communication Functions

As far as possible, each layer, and each component within a layer, is designed to be self-contained and independent. This enables layers and parts of layers to be modified or even removed and replaced, without needing to change other layers.

This independence also enables designers to distribute communications functions among the various nodes in the network. For example, layers of functions, or parts of layers, residing in the host can be moved out to a communications controller.

The functions performed by a layer should be transparent to "higher" layers. In this sense, each layer can be defined as a user of the services provided by lower layers.

Communication functions fall into two broad categories:

- End-to-end services
- Transmission services

End-to-End Services

End-to-end services include all functions required to establish sessions and maintain communications between two users.

End-to-end services are generally distributed as a matched pair between the source node and the destination node. From the complete set of end-to-end services available, a subset is selected and tailored to meet the needs of each user. These services are not shared among users.

SNA defines the following layers of end-to-end services:

- NAU Services (LU Services, PU Services, and SSCP Services)
- Data Flow Control
- Transmission Control

A subset of end-to-end functions selected from these layers is assigned to each session established. This set of functions varies from session to session according to the needs of the user.

Functions are assigned in matching pairs to the two end users in sessions. The set of end-to-end functions associated with each end user is called a half-session.

Transmission Services

Transmission services include all functions needed to route messages over the network between two half-sessions.

Transmission services are shared by all active sessions. They are distributed among all nodes (source, destination, and intermediate) and generally require complementary functions in adjacent nodes.

SNA defines two layers of transmission functions:

- Path Control
- Data Link Control

These two layers make up a common transmission network that is shared by all communicating half-sessions. (The common network also includes a physical "layer" made up of links and other communications equipment. This physical layer is not defined by SNA.)

User Access to the Network

As we have seen, the user typically gains access to the network through the LU services layer. User access is also provided as needed by the Data Flow Control and Transmission Control layers (Figure 6-4).

Different users may require different interfaces for different sessions.

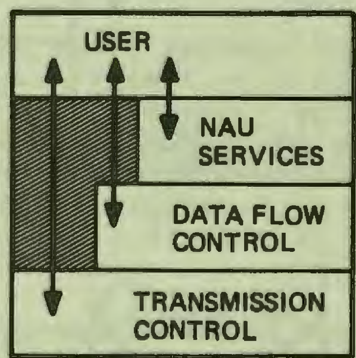


Figure 6-4 User Access to a Network

Interface Between Layers

In SNA, an interface is a mechanism that a software module in one layer uses to pass a message to a module in another, usually adjacent, layer. SNA specifies the messages that must be passed between layers. However, it does not define a particular mechanism, for example macro calls, used to pass these messages (see Figure 6-5).

PEER PROTOCOLS

Modules in the same logical layer are considered peers. Network communications involve the exchange of messages between peer components at all layers. In most cases, these peers, while residing in the same logical layer, are physically located in different nodes.

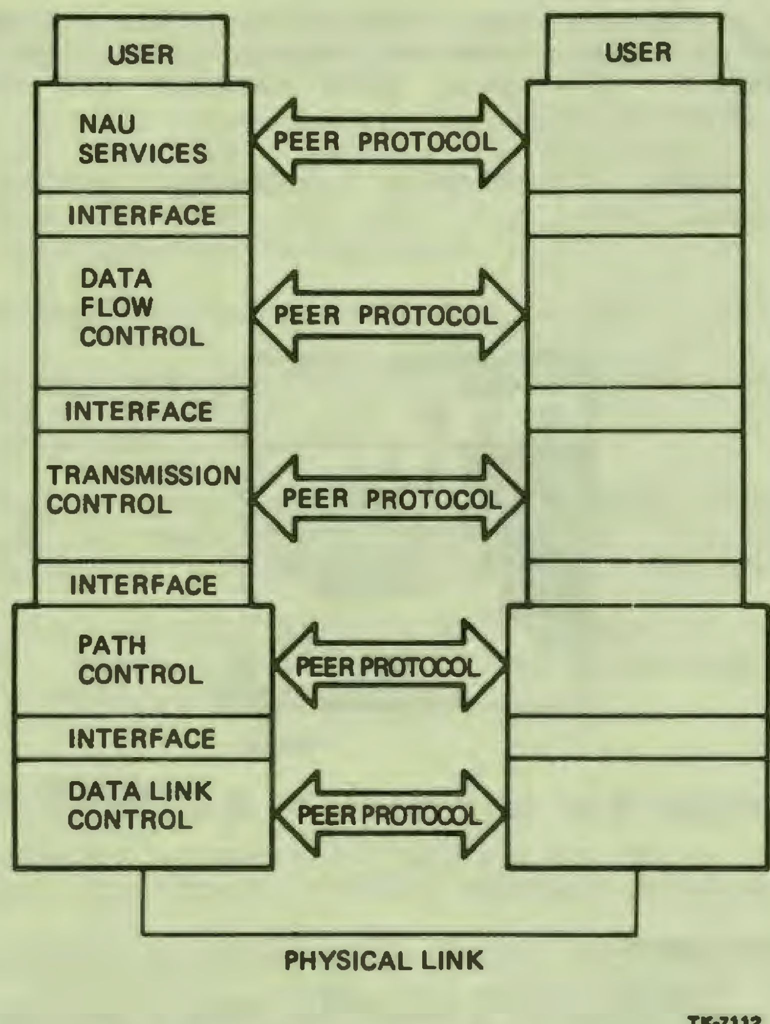


Figure 6-5 SNA Protocols and Peer Modules

As defined in previous modules, the rules that govern such peer-to-peer communications between nodes are called protocols (Figure 6-5). SNA protocols, as well as others, define:

- The syntax of commands and headers.
- The meaning of commands and headers and the actions they initiate.
- The sequence in which commands and headers are sent.

Peers communicate in two ways: by exchanging special control messages and by adding headers and trailers to the user's data. Control messages, headers, and trailers are discussed in the following sections.

Request/Response Units

The network handles two types of messages: data messages and control messages. Data messages are exchanged by network users. Control messages are exchanged by peer modules at all functional levels of the network.

The network considers every message it handles, both data and control, to be either a request or a response. A request is a directive which causes a data transfer or related operation to occur. When a module sends a request to one of its peers, it may or may not receive a response. Ordinarily, a response simply acknowledges receipt of the request. It is not a reply. When a module sends a reply to a peer, it usually puts it in the form of another request.

Requests and responses are known collectively as **request/response** units (RUs). Data RUs, created by a component in the NAU Services layer, contain user messages.

Control RUs contain commands and information exchanged between peer modules. In the NAU Services layer, peer modules called service managers exchange RUs which contain control messages for session establishment, bringing up or shutting down network components, and so on. Peer modules in the Data Flow Control and Transmission Control layers also exchange control RUs.

Depending on their function, RUs move in the normal flow of network traffic or in an expedited flow. Most RUs move in the normal flow. Expedited RUs control the normal flow and remain independent of it. Both types are usually processed sequentially within a particular data path, though expedited traffic may move ahead of the normal flow if required.

Headers

In addition to exchanging control RUs of their own, peer modules in the various layers also add headers and trailers to the RU containing the user's data.

In the sending node, these headers and trailers are added as the user's message "descends" through the layers (see Figure 6-6).

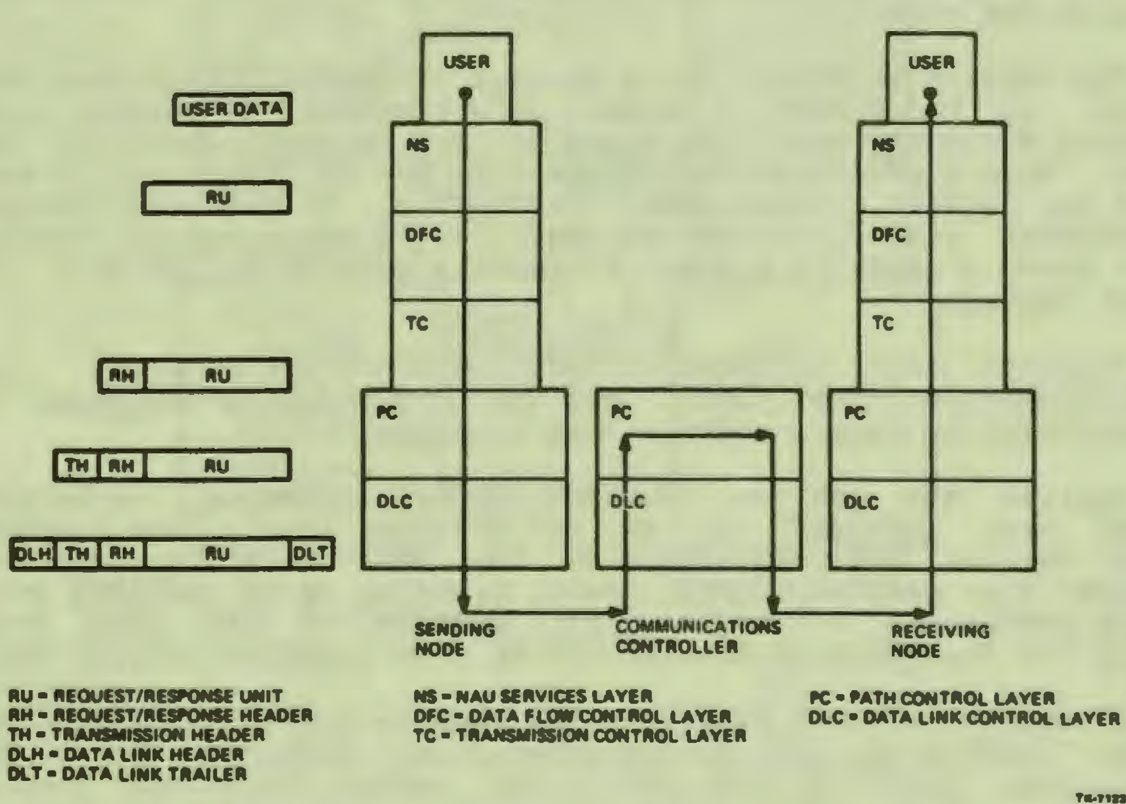


Figure 6-6 Data Transmissions in SNA

Creating a Data RU -- A component in the NAU Services layer takes the user's data in the user's format and converts it into an RU format that can be transmitted over the network.

The Request/Response Header -- As the RU passes through the transmission control element assigned to the half-session, a module call the Connection Point Manager adds a Request/Response Header (RH). The RH contains indicators for data flow control and such transmission control functions as pacing.

Transmission Header -- The Path Control layer adds its own header, called transmission header (TH). The TH contains message blocking and segmenting information, and specifies the origin and destination of the message. In segmenting, a long message is broken into separate units for transmission. In blocking, several short messages are combined into a larger message for transmission.

Data Link Header and Trailer -- The data link control generates its own headers and trailers according to the Synchronous Data Link Control (SDLC) protocol. From this layer, the message (within its envelope of headers and trailers) is sent out onto the link to the first intermediary node.

Data Flow

As the message (within its envelope) travels through the network, its headers and trailers are examined by the Data Link Control and Path Control layers in each intermediary node. At the destination node, the message "rises" through the layers to the receiving half-session, and finally to the destination end user.

FUNCTION SUBSETS

A function subset is a family of options available within each layer. Different users may employ different subsets of the available functions at every level, according to their needs. Or, a user may require a different subset for each session in which he or she is involved.

In some cases, the subset may include a null function, enabling a user to skip an entire layer of end-to-end functions (for example, presentation services) not needed for a particular half-session.

LU Types and Function Profiles

A commonly-used subset of functions selected from an end-to-end layer is called a function profile.

As part of establishing a session, a user can choose subsets from Presentation Services, Data Flow Control functions, and Transmission Control functions.

Different LUs have different function profiles. A particular combination of profiles identifies an LU-type.

PU Types

A subset of path control and data link control functions is typically built into a product when it is designed. A physical network component can be defined according to its particular built-in subset of transmission facilities. Products using the same subset are classified as members of the same PU type.

- PU Type 1 Terminal
- PU Type 2 Cluster Controller
- PU Type 3 (undefined)
- PU Type 4 Communications Controller
- PU Type 5 Host

Type 1 Terminal -- A terminal node provides attachment for one or more I/O devices.

Terminals include a PU and up to 64 logical units. The LUs serve devices. Each LU has one session with the host and one with another LU.

PU Type 1 nodes permit segmenting but not blocking. The terminal can neither send nor receiver blocked RUs.

Type 2 Cluster Controller -- A cluster controller provides control functions for various I/O devices and may have a data-processing capability.

Cluster controllers can have up to 256 NAUs. LUs support both devices and applications programs.

PU Type 2 nodes permit segmenting but not blocking.

Type 4 Communications Controller -- A communications controller handles transmission services for a subarea of the network. It controls lines, maintains buffers, and provides both intermediate and boundary functions for attached cluster controllers and terminals.

Currently, there are no LUs in any communications controller; SNA, however, imposes no restrictions against them. A communications controller, for example, might perform LU boundary functions on behalf of attached terminals and devices incapable of housing their own LU.

PU Type 4 nodes permit both segmenting and blocking.

Type 5 Host -- A network host provides data processing for remote users and in some cases intermediate and boundary functions.

The SSCP for a domain is located in the host telecommunication access method. In addition, hosts include LUs that represent either programs or devices.

PU Type 5 nodes permit both segmenting and blocking.

Domains

A **domain** is the portion of a network controlled by an SSCP in a host computer. SNA defines single- and multi-domain networks.

In a single-domain network, one host computer with one SSCP controls all communications. In such a system, the terms "domain" and "network" are synonomous.

A multi-domain network contains more than one host computer with SSCP capabilities. Each SSCP controls its own domain.

Domains can be joined by either a cross-domain link between two local communications controllers, or a communications controller that is channel-attached to more than one host computer.

An application running in one domain can establish sessions with logical units in its own and other domains. In cross-domain communications, neither end of the session need be aware that the other end is in a different domain.

IBM SOFTWARE PRODUCTS

The software products that IBM offers to implement SNA do not always correspond to the layers that SNA defines. Some software products may represent all of one or more SNA-defined layers, and parts of others. Another software package may complete the partial layers.

SNA communications functions are divided among a series of software products that run on different hardware devices. These software products include:

- Virtual Telecommunications Access Method (VTAM)
- Telecommunications Access Method (TCAM)
- Extended Telecommunications Modules (EXTM)
- Customer Information Control System (CICS)
- Information Management System (IMS)
- Time Sharing Option (TSO)
- Network Control Program (NCP)

VTAM, TCAM, and EXTM

IBM offers software packages that interface applications software to a network. The only two that offer full SNA functionality are VTAM and TCAM.

Early versions of VTAM and TCAM supported only single host networks. It was not until the addition of the Advanced Communications Function (ACF) that multiple hosts were supported. Therefore, the current software is called ACF/VTAM and ACF/TCAM.

EXTM is designed to run on smaller systems in conjunction with CICS. EXTM offers a subset of SNA capabilities. VTAM, TCAM, and EXTM support different peripheral devices.

The different access methods offered by IBM are designed to be used in conjunction with data management or communications software packages such as CICS, IMS, and TSO. These are described in the next section.

Depending on the access method used and the other software present in the system, the user may be responsible for writing some parts of the network software and applications programs.

CICS, IMS, and TSO

CICS is a transaction processing system. IMS is a data base control system that controls hierarchically designed data bases. IMS can also provide communications functions. TSO is simply a general timesharing offering.

The Network Control Program (NCP)

NCP runs on the 370X series of communications controllers. These are programmable minicomputer-like devices. NCP corresponds to the Path Control and Data Link Control layers of SNA. The current version of NCP is called ACF/NCP/VS and needs a 3705 controller. The NCP controller is controlled by the host and, therefore, does not operate as a true distributed processing system.

It is possible to run a special program in a communications controller that translates the BISYNC protocol into SNA protocol. This program, called the Partitioned Emulator Program (PEP), can work with NCP in a single 3705 controller and allows non-SNA terminals to be connected to an SNA network (see Figure 6-7).

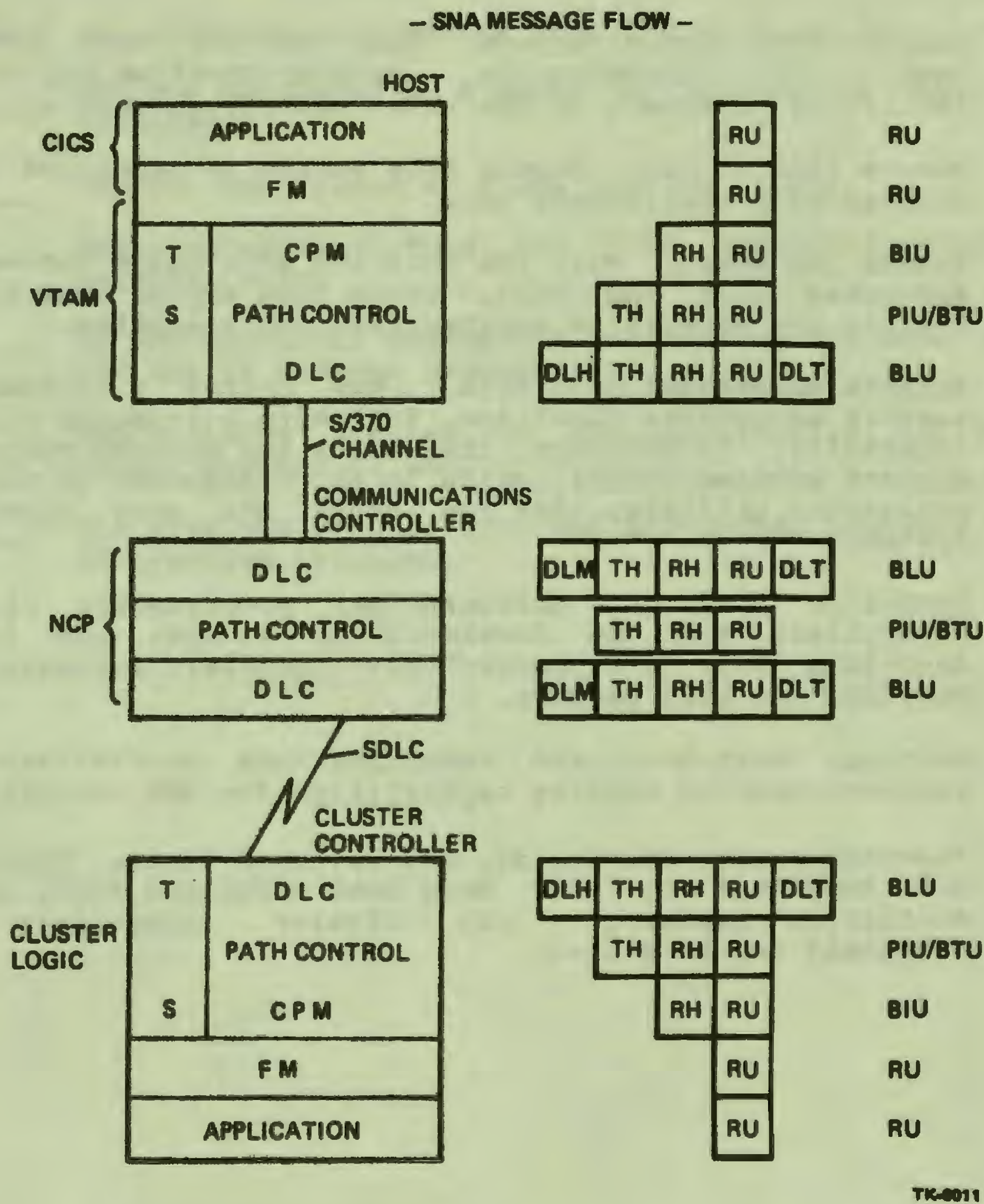


Figure 6-7 Partioned Emulator Program (PEP)

SNA CAPABILITIES

SNA supports the following network functions:

- Task-to-task communications. Both VTAM and TCAM provide task-to-task communication. Neither provides all of the facilities necessary to use this capability, however.

- Remote file access. Remote file access is supported in a CICS-to-CICS environment only.

- Remote job entry. With the addition of a batch job entry subsystem (JES and RES), batch jobs may be sent by one host in the network to another host for execution.

- Network management utilities. SNA offers a number of network management functions, including a loopback testing capability. In addition, IBM offers the Network Operation Support Program (NOSP), which is an arrangement of network management utilities that run under the host operating system.

- Down-line loading. Software for programmable cluster controllers must be developed on a host and loaded down-line into the controller. Special software is provided for this purpose.

- Routing. Both host and communications controllers may provide adaptive routing capabilities for SNA networks.

- Point-to-point, multipoint, and switched links. These are all supported by SNA. Only host nodes and 3705s may be multipoint masters. Only cluster controllers and terminals may be slaves.

EXERCISES

1. Mark the following statements true (T) or false (F).

   ___ SNA was developed as a means of connecting many small computers.

   ___ The SNA equivalent of a DNA task is an NAU.

   ___ Early versions of TCAM and VTAM supported single-host networks only.

   ___ In SNA networks, addressing and routing are done without the use of message headers.

   ___ SNA software modules do not necessarily reflect the layers defined by SNA.

   ___ The simplest node in both SNA and DECnet networks is an interactive terminal.

2. Match the following items with the phrases that best describe them. Each item may be used one or more times, or not at all.

   A. Domain

   B. CICS

   C. Communications controller

   D. Connection point manager

   E. Cluster controller

   F. BIU

   G. Logical unit

   H. TCAM

   I. NAU

   ___ The user's interface to the network.

   ___ Any element in the network to which a message may be addressed and sent.

   ___ Operates as a terminal concentrator and front end processor.

   ___ Transaction processing system.

   ___ The part of a network over which a specific host has control.

SOLUTIONS

1. Mark the following statements true (T) or false (F).

   F SNA was developed as a means of connecting many small computers.

   T The SNA equivalent of a DNA task is an NAU.

   T Early versions of TCAM and VTAM supported single-host networks only.

   F In SNA networks, addressing and routing are done without the use of message headers.

   T SNA software modules do not necessarily reflect the layers defined by SNA.

   F The simplest node in both SNA and DECnet networks is an interactive terminal.

2. Match the following items with the phrases that best describe them. Each item may be used one or more times, or not at all.

   A. Domain

   B. CICS

   C. Communications controller

   D. Connection point manager

   E. Cluster controller

   F. BIU

   G. Logical unit

   H. TCAM

   I. NAU

   __G__ The user's interface to the network.

   __I__ Any element in the network to which a message may be addressed and sent.

   __C__ Operates as a terminal concentrator and front end processor.

   __B__ Transaction processing system.

   __A__ The part of a network over which a specific host has control.

X.25

7

INTRODUCTION

X.25 is a protocol developed by the Comite Consultatif Internationale de Telegraphie et Telephonie (CCITT) for communications between a user and a public packet-switching network (PPSN). Public packet-switching networks will be discussed in the Common Carrier module of this course.

As the "X" indicates, X.25 is a recommendation only. The international committee itself can not agree on all of the details of the protocol. Therefore, X.25 is flexible, with the result that different implementations of X.25 in different countries and companies are not always compatible.

In contrast to DNA and SNA, X.25 is not a network architecture for end-to-end communications. Instead, it is a set of rules that users must follow to interface their equipment to the PPSN and pass data to the PPSN. Once the data has been accepted by the PPSN, the way it is routed and the way errors are detected are no longer the user's (or X.25's) concern. After the PPSN has sent the error-free message to the proper receiving center within the network, the X.25 protocol is used again, this time to transfer the message to a user outside the PPSN.

In this module we will discuss the characteristics and features of the X.25 protocol, and provide an overview of how it is implemented in a public packet-switching network.

OBJECTIVES

Describe the following features of CCITT's X.25 proposal:

- The interactive terminal interface (X.28, X.29, and X.3)
- X.25 virtual circuits
- X.25 levels

RESOURCES

1. DATAPRO Research Corporation, A McGraw-Hill Company, 1978.

2. Provisional Recommendations X.3, X.25, X.28 and X.29 on Packet-Switched Data Transmissions Services, CCITT, Geneva, 1978.

X.25 STANDARD

X.25 specifies the interface between data terminal equipment (DTE) and data circuit termination equipment (DCE) on the PPSN. The DCE is a node processor that serves as an entry and an exit point to the PPSN. The DCE is the network side of the user/network interface.

X.25 defines communication procedures between DTEs and DCEs only. The way in which DCEs talk to one another within the PPSN is up to the designers. DTEs are usually connected to modems and communicate with the nearest DCE over a telephone line. DTEs are programmable devices that are the user side of the user/network interface.

DTE and DCE

The DTE is located at the user site only for users that have installed X.25. At such installations, the DTE can be defined as any one of several items (see Figure 7-1):

- A computer
- A front-end processor
- An intelligent terminal
- A series of multiplexed intelligent terminals
- A processor acting on calls received from multiple locations communicating over the PPSN

Regardless of the device or application, all DTEs present to the DCE (over communication facilities) standard formatted data and control information, and employ standardized communications facilities.

The connection between two DTEs is called a virtual circuit. Although no two packets of a message may follow the same path through the PPSN, the transmitter and receiver appear to be connected by a single, error-free physical link. Essentially, the user is able to cause the network to establish a logical circuit connection with the receiving station for transmission of multiple, contiguous packets.

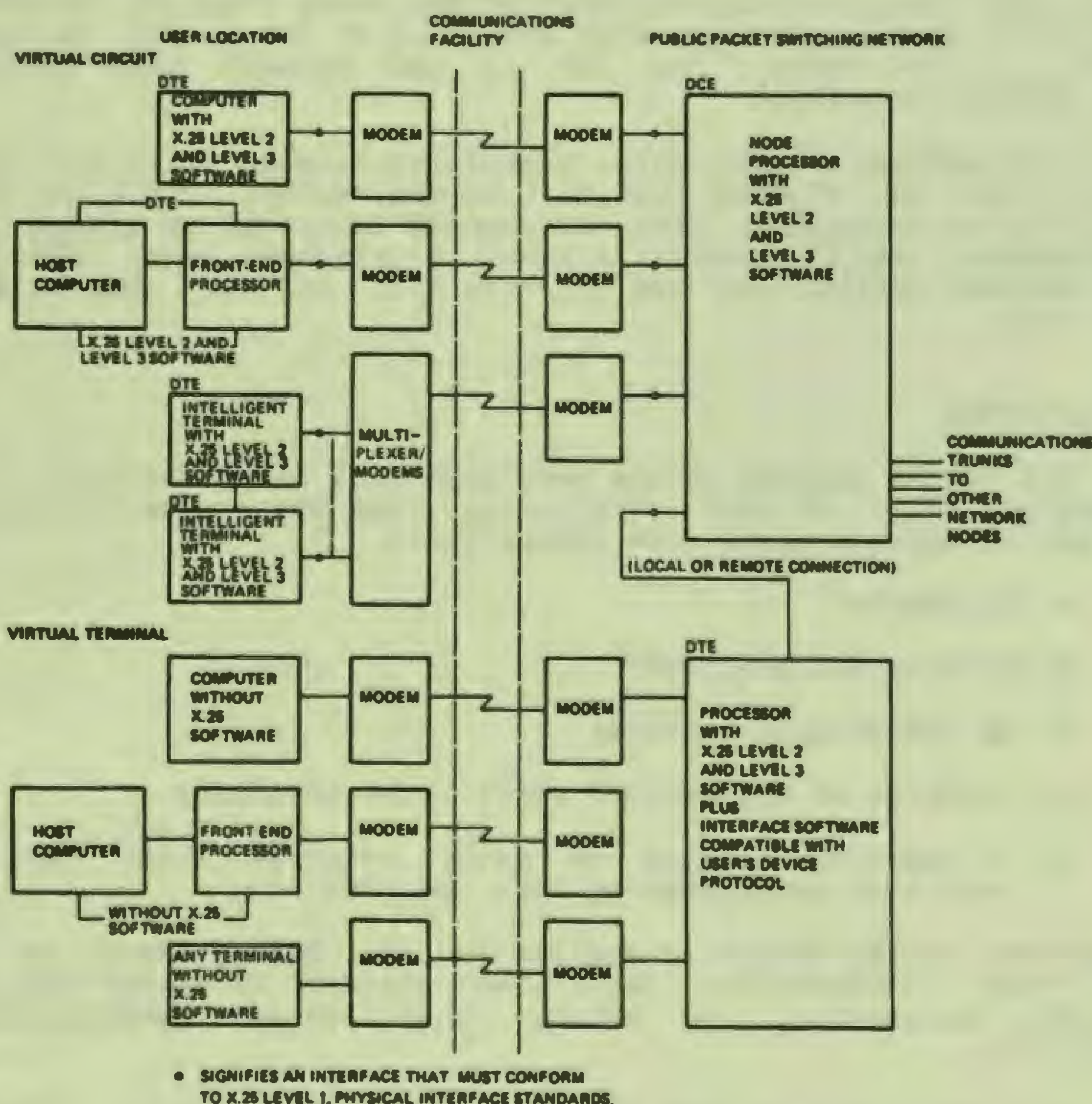


Figure 7-1 X.25 User Terminal Configuration

PPSNs offer both switched virtual circuits and permanent virtual circuits. Switched virtual circuits are created for the length of one exchange. Permanent virtual circuits are not set up and taken down after each exchange, but are always ready when there is a message to send. Users can rent a permanent virtual circuit in the same way that they lease a telephone line.

Users that are not using X.25 can also use the PPSN for data transmissions. However, they cannot transmit to a DCE. These users must transmit to a special network-operated DTE. This special DTE is programmed and configured to properly interface with the protocol and physical characteristics of the user's device. This special DTE will receive data reformatted into X.25 format and forward it to the DCE.

CCITT's X.3, X.28, and X.29 recommendations define an interactive terminal interface (ITI) for devices such as asynchronous hardwired terminals that want to communicate over a PPSN, but cannot be programmed according to the X.25 format.

Packet Assembly/Disassembly (PAD) facilities of a PPSN permit the connection of terminals (start-stop mode DTEs) which are, typically, teletypewriters or CRT terminals. A terminal can be connected to the PAD facility of a packet switching exchange by dial-up or by a leased line.

A terminal sends data at low speed to a PAD, which assembles it into packets and forwards it at full network speed to the destination packet-mode DTE. In the reverse direction, the PAD receives data, formats it, and sends it at the appropriate speed to the terminal. In general, the PAD performs network-related functions in such a way that a terminal appears to its operator to be directly connected to the packet-mode DTE. From the view of a packet-mode DTE, the PAD and the terminal appear to be another packet-mode DTE operating as recommended by X.25. No facilities are provided for terminals to communicate directly with each other. A **permanent virtual circuit** (PVC) cannot be set up between a start-stop mode DTE and terminal.

The PAD maintains a conversion table for each type of terminal that it services. The form of this table is defined by CCITT recommendation X.3. Another recommendation, X.28, defines the manner in which a virtual terminal communicates with a PAD. A third recommendation, X.29, defines the interface between the PAD and a packet-mode DTE (Figure 7-2).

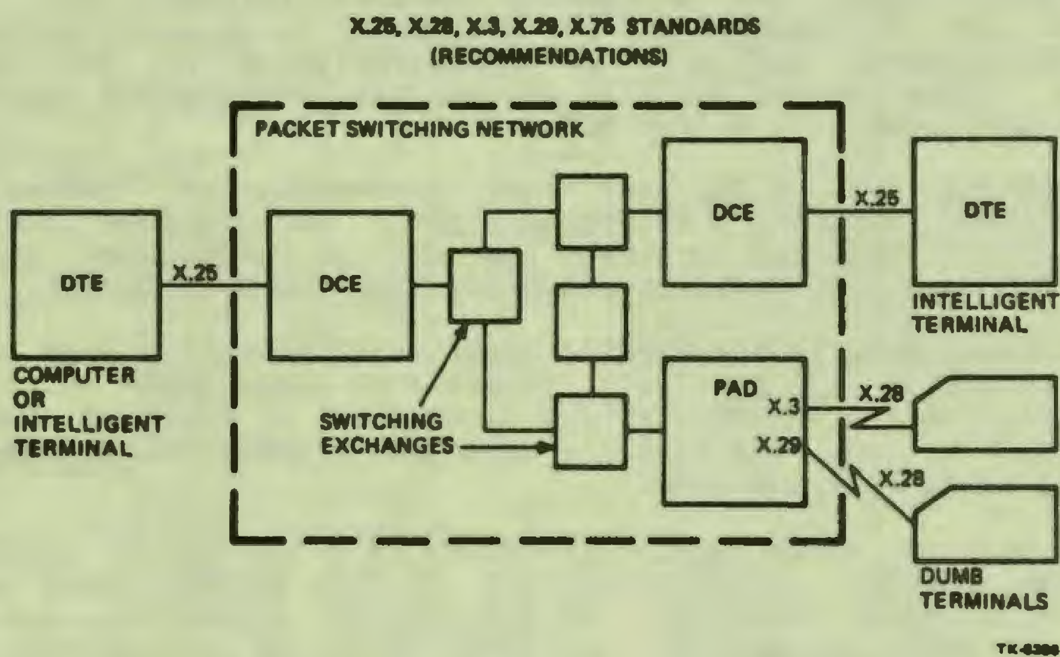


Figure 7-2 CCITT's Recommendations X.3, X.28, and X.29

X.25 LEVELS

X.25 defines three levels; in function, these levels are similar to RS-232-C and to the two lower levels of DNA and SNA.

- Level 1 - Physical Interface
- Level 2 - Frame Level
- Level 3 - Packet Level

Level 1 defines the physical and electrical characteristics of the connection between the DTE and the DCE (the points specified in Figure 7-1). Level 2 defines the rules for transmitting data between the DTE and DCE. Level 3 defines the rules for transmitting and controlling the packet.

Level 1 - Physical Interface

This level specifies the connection between DTEs and DCEs and modems. Level 1 is described fully in CCITT standard X.21. X.21 defines the physical, electrical, functional, and procedural characteristics to establish, maintain and disconnect the physical link between the DTE and the DCE. X.21 is an equivalent to EIA's RS-232-C interface standard and its replacement, RS-449. Currently there is a temporary standard, X.21-BIS, which is essentially the same as RS-232-C.

Level 2 - Frame Level

This level describes the Link Access Procedure to be used for data exchanges between DCE and DTE. This procedure calls for a full-duplex facility so that the DTE and DCE can conduct two-way simultaneous, independent transmissions. The protocol used at this level is called High Level Data Link Control Procedure (HDLC), which is basically the same as IBM's SNA SDLC, with a slightly different mechanism used to calculate its CRC characters. The major difference in HDLC and SDLC is in the way that the link is used. In SNA, the host always controls the link and the terminals are always slaves. In HDLC, any device can initiate a call. Also, the address field of HDLC is not used (except to indicate whether the message is a command or a response). Addressing information is handled in the Level 3 header. (See Figure 7-3 for a description of an HDLC message format.)

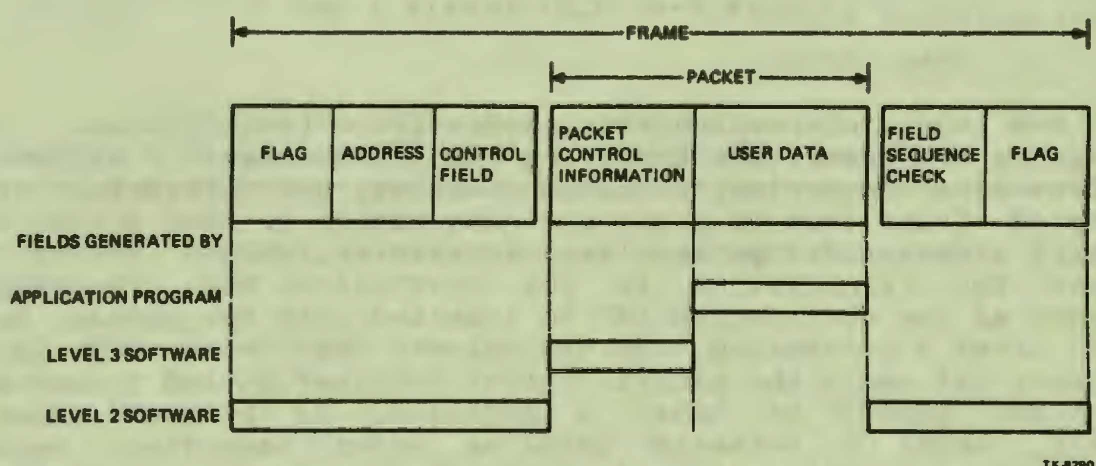


Figure 7-3 X.25 Message Format

Level 3 - Packet Level

Level 3 describes the packet format and the control procedures for the exchange of packets between the DTE and the DCE. In addition to the user data packet format, there are several formats for DTE/DCE administrative messages.

The software to format packets and to control packet exchange is the Level 3 software, resident in both the DCE and the DTE. Figure 7-4 shows the relationship of Level 3 and Level 2 software operating in the DTE and the DCE. In the DTE, a user application program normally presents the data to be transmitted to the operating system's communications access method. The access method invokes Level 3 software to append the packet header information (see Figure 7-3), then the access method invokes Level 2 software to create the frame.

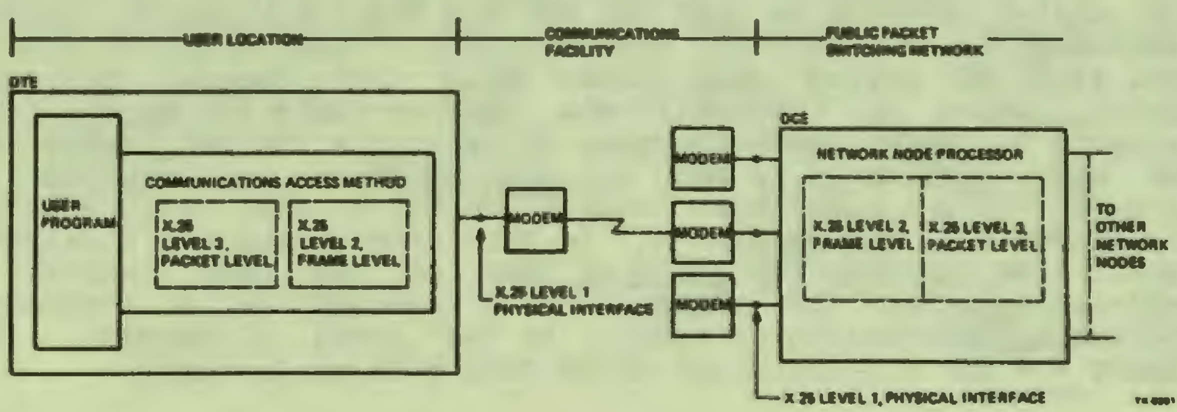


Figure 7-4 X.25 Levels 2 and 3

Now the information is ready for transmission. Upon receiving the frame, the receiving DCE invokes Level 2 software to perform error detection, sequence checking, and stripping of an accepted frame down to a packet. The packet is then presented to Level 3 software for packet level processing, and to prepare the packet for transmission to the destination DCE. The physical address of the destination DTE is inserted into the packet during this Level 3 processing. At the network destination DCE, Level 3 software reformats the packet control information and presents the resultant packet to Level 2 software. At the destination DTE (user), Level 2 software performs error detection, sequence checking, and stripping of accepted frames down to the packet. Level 3 software is then applied to the packet to strip the header information from the packet and, via the communications access method, pass the user information to the appropriate applications program. Table 7-1 summarizes the packets exchanged between terminals during a virtual call.

Table 7-1 Summary of X.25 Packets

Event	Calling DTE	Called DTE
Call initiation	Call Request/Packet is sent	Incoming Call paoket is received
		Call Accepted packet is sent
	Call Connected packet is received	
Data transfer	Data packet sent	
		Data packet received
		Ready Receive packet sent
	Ready Receive packet received	
	Data packet A sent*	Data packet B sent*
	Data packet B received*	Data packet A received*
Disconnect	Call Request packet sent	Clear Indication packet received
		Clear Confirmation packet sent
	Clear Confirmation packet received	

* Two-way (full-duplex) transmission of Data packets between terminal equipment.

X.25 USER RESPONSIBILITIES

X.25 defines only the lowest levels of a network architecture. It is up to the user to implement all higher functions. These include:

- Internal addressing and protection within a DTE,
- End-to-end error control and sequencing, and
- All higher-level network functionality.

The PPSN is concerned only with getting a message to the right DTE. It does not care if the DTE is a computer system with a hundred users. It is up to the user to make sure that a message gets to the right place within the system and that the message is not interrupted by an unauthorized receiver.

The PPSN is responsible for making sure that each packet makes it from one packet switching exchange to another without error. However, it is not responsible for making sure that a complete message gets from one user to another. A message may be lost in the system, or the sequence of messages (not packets) may be changed because they take different paths through the network. It is even possible for a memory error in the sending computer to direct a message to the wrong address. It is up to the receiving user system, not the PPSN, to detect such errors.

PPSNs provide only the simplest type of task-to-task communication and remote terminal capability. More extended functions, such as remote file transfer and conversion, remote submission, and so forth, must be implemented by the user.

EXERCISES

Match the following items with the phrases that best describe them.

A. PAD

B. DTE

C. DLE

D. DCE

E. Virtual circuit

F. Frame level

G. Packet level

H. Physical path

___ Uses the HDLC protocol.

___ The connection between two DTEs.

___ Handles addressing and flow control.

___ Accepts data from devices that cannot conform to X.25 standards.

___ Defined by X.25 as the user's X.25 compatible equipment.

___ Includes all processors within the PPSN.

SOLUTIONS

Match the following items with the phrases that best describe them.

A. PAD

B. DTE

C. DLE

D. DCE

E. Virtual circuit

F. Frame level

G. Packet level

H. Physical path

__F__ Uses the HDLC protocol.

__E__ The connection between two DTEs.

__G__ Handles addressing and flow control.

__A__ Accepts data from devices that cannot conform to X.25 standards.

__B__ Defined by X.25 as the user's X.25 compatible equipment.

__D__ Includes all processors within the PPSN.

LOCAL AREA NETWORKS

8

INTRODUCTION

Local area networks (LANs) are most often described as privately owned networks that offer reliable high-speed communication channels optimized for connecting information processing equipment in a limited geographic area, namely, an office, building, complex of buildings, or campus.

LANs elude precise definition because they can be designed with a rich variety of technologies and arranged in different configurations. Consequently, they vary with respect to their transmission speeds, the distances they can span, their operating and performance characteristics, and the capabilities and services they offer.

Local area networks are often hailed as revolutionary. In terms of their potential for changing the way we work and communicate at work, this may be true. However, they are also **revolutionary**, having emerged from existing means of communications as a result of technological advances in communications hardware and software.

OBJECTIVES

Describe the following local area network features:

- Topologies
- Transmission techniques
- Media
- Software and hardware considerations
- Relationship to the automated office

RESOURCES

1. Cravis, Howard. "Local Networks Prove Practical for Datacom Systems in Close Proximity", Datamation, 27:2, March 1981, p. 98.

2. DATAPRO Research Corporation, A McGraw-Hill Company, 1978.

3. Introduction to Local Area Networks, Digital Equipment Corporation (EB-22714-18).

4. Kinnucan, Paul. "Local Networks Battle for Billion-Dollar Market", High Technology, 1981.

5. Sideris, George; Contributing Editor. "Network Hardware is Key Element in Connection Cost", Sept. 1981, pp. SS-53.

6. Sideris, George; Contributing Editor. "Software Helps Networks Grow With Compatibility", Electronic Design, Sept. 1981, pp. SS-39.

OVERVIEW

Local area networks (LANs) utilize concepts that are different than those used with traditional data communication networks. LANs eliminate switching by broadcasting messages to every device on the network, and allowing the message to be ignored by all devices but the device to which it is addressed.

Nearly all current LANs utilize packet techniques, and as a result, the slightest pause by a sending device can enable another device to seize the line and send one or more packets. This concept is utilized to give equal priority to all attached LAN devices and to avoid having a single device monopolize the network.

LANs require the use of some sort of communications facility, such as a coaxial cable of limited length to carry transmissions, plus interface units to interconnect computers and terminals to the facility. A few number of LANs permit only the supplier's terminals and processors to be connected, but the majority of LANs permit any manufacturer's equipment to be connected. The facility and the interface units may be purchased by the user and may be installed and maintained by either the user or the supplier.

Since the technology embodied in the LANs is quite unique, individual vendors may have their own terminology for the components used. Consequently, at this point in time, these LANs are not compatible with each other and may never be.

Currently, there are efforts going on by standards groups to establish standards for LAN protocols, connections, lines, speeds, etc., so that multiple manufacturers' devices can be attached to the same network. A company with a current requirement for a LAN had best not wait for the establishment of a standard in order to implement a network, because it is likely to take years before one can be established. If the LAN must communicate with a distant location, some sort of common carrier interface, such as the telephone network, will be required.

Current common carrier interfaces, called gateways, are available but many LANs do not provide for them. Using gateways in a LAN takes away from advantages provided by LANs, such as speed, accuracy, and simplified control, as soon as the data leaves the LAN facility and enters a common carrier facility. For the time being, the only way to interface between distant facilties is by common carrier.

LOCAL AREA NETWORK TOPOLOGIES

The most fundamental difference in LAN schemes lies in the network topology. Currently there are five basic topologies:

- Star
- Loop
- Ring
- Common bus
- Broadband bus

Most LANs utilize two types of topologies, ring and common bus. In the common bus topology (see Figure 8-1) a single cable, called a bus, is provided and all the stations are attached to it by means of cable taps. Signals from a station branch out from the tap in both directions to reach the other stations in the network. In the ring networks, a circular chain of signal repeaters with links between them is created to form the ring (see Figure 8-2). Network stations communicate with each other by sending messages into the ring via the repeaters. The repeaters will relay the message bit-by-bit around the ring until the destination is reached.

Both of these topologies share certain advantages:

- A common transmission medium is used to link the network stations.

- A common transmission medium allows messages to be broadcast among network stations.

- Network stations are allowed to directly communicate with other network stations.

- Decentralized control. A master controller is not necessary to control access to the network or route messages among stations.

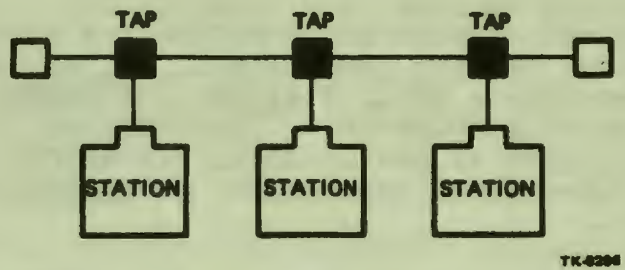


Figure 8-1 Common Bus Topology

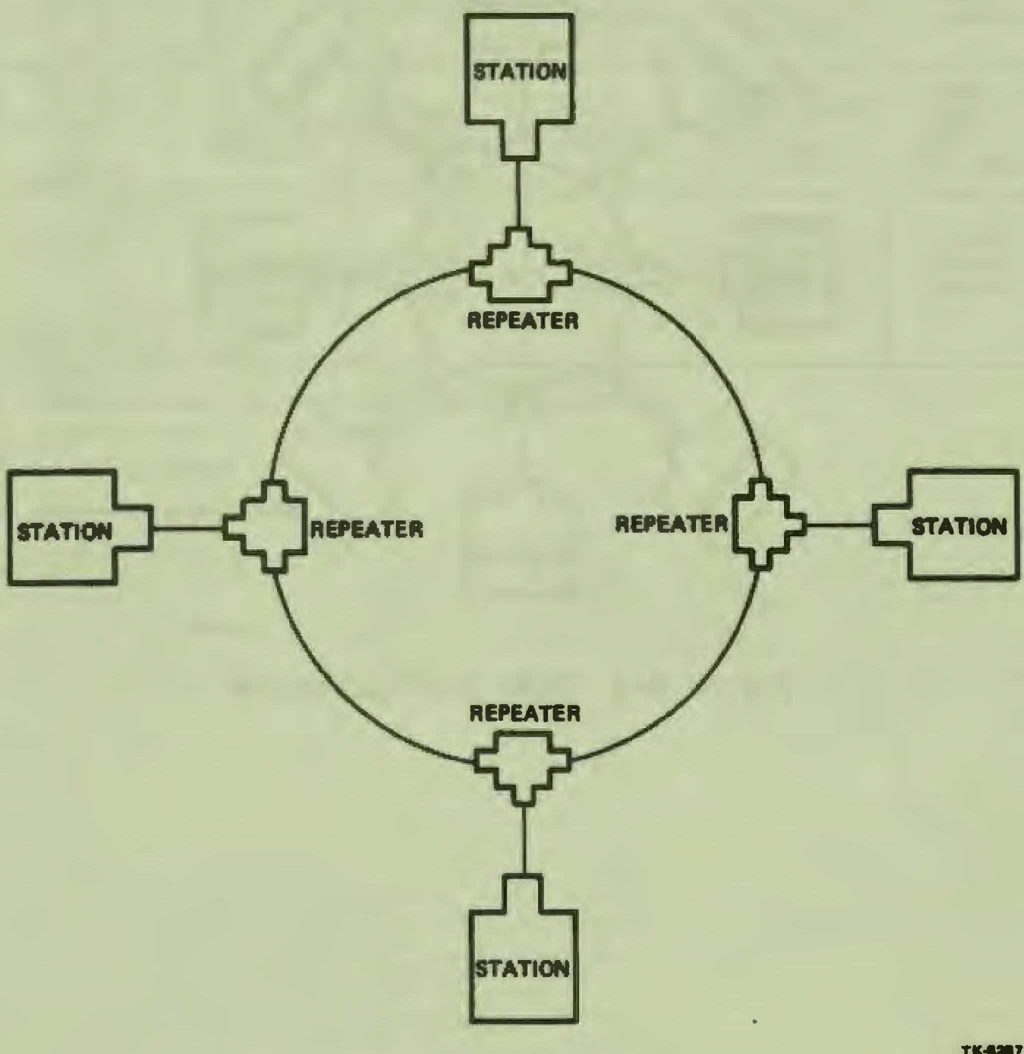


Figure 8-2 Ring Topology

Full connectivity and message broadcasting are more difficult to achieve with other network topologies. For example, in the star configuration (see Figure 8-3), stations can communicate directly only with a central computer. Interstation traffic must be switched through a central controller. For a brief overview and comparisons of the five topologies and their characteristics, refer to Table 8-1.

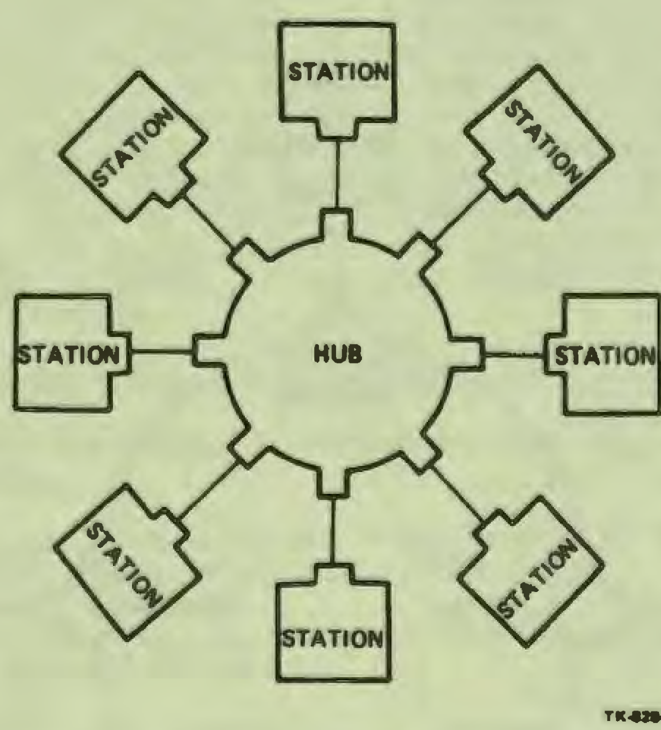
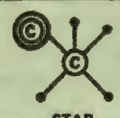
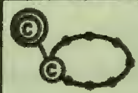
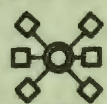
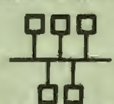
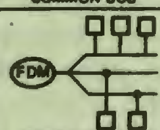


Figure 8-3 Star Configuration

Table 8-1 Basic Local Area Network Architectures

TOPOLOGY	TRANSMISSION MODE	TYPICAL PROTOCOLS	TYPICAL NO. OF NODES	ADVANTAGES	TYPICAL SYSTEMS
 STAR	POINT-TO-POINT VIA CHANNEL SWITCH OR COMPUTER MEMORY	RS-232C OR COMPUTER	TENS	WELL-KNOWN LARGE BASE OF USERS	PABX, COMPUTER µC CLUSTERS
 LOOP	MESSAGE ROUTING VIA LOOP CONTROLLER	SDLC	TENS	WELL-KNOWN LARGE BASE OF USERS	IBM 3600/3700, µC CLUSTERS
 RING	PACKET TRANSMISSION AROUND RINGS	HDLC (TOKEN PASSING)	TENS TO HUNDREDS PER CHANNEL	DISTRIBUTED CONTROL, NO CONTENTION, POPULAR FOR COMPUTER NETS	PRIMENET, DOMAIN, OMNILINK µC CLUSTERS
 COMMON BUS	BROADCAST ALONG SERIAL BUS	CSMA/CD OR CSMA WITH ACKNOWLEDGMENT	TENS TO HUNDREDS PER SEGMENT	DISTRIBUTED CONTROL POPULAR FOR OFFICE NETWORKS AND COMPUTER NETS	ETHERNET, NET/ONE, OMNINET, Z-NET µC CLUSTERS
 OTHER SERVICES BROADBAND BUS	PACKET BROADCAST BUS WITH DEDICATED OR PRIORITIZED CHANNELS	CSMA/CD RS-232C & OTHERS	TWO TO HUNDREDS PER CHANNEL	DISTRIBUTED CONTROL, LARGE VARIETY OF USERS AND SERVICES	WANGNET, LOCALNET M/A-COM

- • TERMINAL
- ☐ TERMINAL WITH DISTRIBUTED CONTROLLERS AND/OR MULTIVENDOR INTERFACES
- ⓒ LOCAL CONTROLLER
- ⓒ (double circle) HIERARCHICAL NETWORK
- (FDM) FREQUENCY DIVISION MULTIPLEX

TRANSMISSION TECHNIQUES

Regardless of the topology used, any shared transmission medium must allow access in an equitable manner. LAN techniques differ greatly in this aspect. The most widely used approach is that of timesharing, known as time division multiple access (TDMA). With this technique, each station in the network is allowed to transmit across the medium at certain times under the rules governed by the network interface.

Different types of timesharing techiques include polling, reservation, contention and token passing. In polling, a master network controller provides time on the channel on a request basis. During idle periods, the master controller polls stations for channel requests on a round-robin basis. When using a reservation scheme, time on the channel is allocated according to a predetermined schedule. Each station is allocated a unique time period in which it is free to access the channel and transmit its messages. Contention schemes allow stations in the network to compete for time on the channel. With this technique, rules are established to ensure orderly access and an equitable distribution of channel usage stations. Token passing, used by most ring networks, is a form of contention control. In ring networks, a short message, or token, is circulated during idle periods. A station that wants to transmit a message acquires the token and transmits it's message. Afterwards, it recreates the token, which is then recirculated until another station has a message to transmit. Although token passing is more closely associated with ring networks, some bus networks will also use them.

Of the time sharing techniques described, contention is the most widely used because it does not require a master controller. Another important reason for using the contention scheme is that it is very well suited to the intermittent nature of most LAN traffic. In contrast, reservation and polling schemes are very inefficient when it comes to controlling intermittent traffic, because many periods of time go by unused between transmissions.

LAN SERVICES

Many organizations need to connect telephones and other voice-related equipment. They also need to connect such data devices as computers, terminals, printers, and a wide range of peripherals to exchange text, numeric, and graphic information.

Some organizations, particularly larger ones, also want to link video equipment -- monitors, cameras, and record/playback devices, for security, teleconferencing, and educational applications.

There are three popular technologies being used by organizations to implement LANs for interconnecting different varieties of equipment and exchanging different forms of information:

- Private branch exchange (PBX)
- Broadband
- Baseband

Each of these has a unique set of characteristics and, by and large, each is optimized for certain applications.

Private Branch Exchange (PBX)

The PBX was originaly used as a means of switching telephone calls within a business site, and from the site to outside lines. Later, the PBX was used for low-speed tranmission of data, in addition to voice, by converting digital data signals to analog form. Now, PBX systems are evolving toward a digital approach to serve voice and data applications by combining digital switching and microcomputer control.

Broadband

Broadband technology was developed for video transmission. It was originally used to bring improved television reception to remote areas via coaxial cable. Now, it is also used to deliver "cable TV" service. The broadband approach is being used for LANs in an attempt to transmit data, video, and voice over the same cable in a multifunction network.

Baseband

Baseband LANs, such as Ethernet, offer a solution for the connection of computers and data resources and for the connection of users to these resources. They provide what is called the information and resource backbone of the workplace. The baseband approach conbines developments in communications protocols and LSI and VLSI technology to offer the high-bandwidth and transmission speeds needed for interconnection of the growing number of intelligent devices in the local environment.

PROTOCOLS

As mentioned in the Architecture Overview module, ISO has proposed a seven-level architecture. In newer networks, the architecture levels may take a different look but will generally provide for the same features. This is particularly true of local nets, which might even separate the lowest levels and lump all the higher levels together (refer to Figure 8-4).

Eventually, as the use of LANs expands, the need to standardize at all seven levels will arise. Presently no such seven-level standardization for LANs exists. However, the Institute of Electrical and Electronic Engineers (IEEE) has drafted a standardization proposal for levels 1 and 2 (the physical and data link levels). The U.S. Department of Defense has standardized higher-level protocols, and the National Bureau of Standards is currently working on standards at the higher levels.

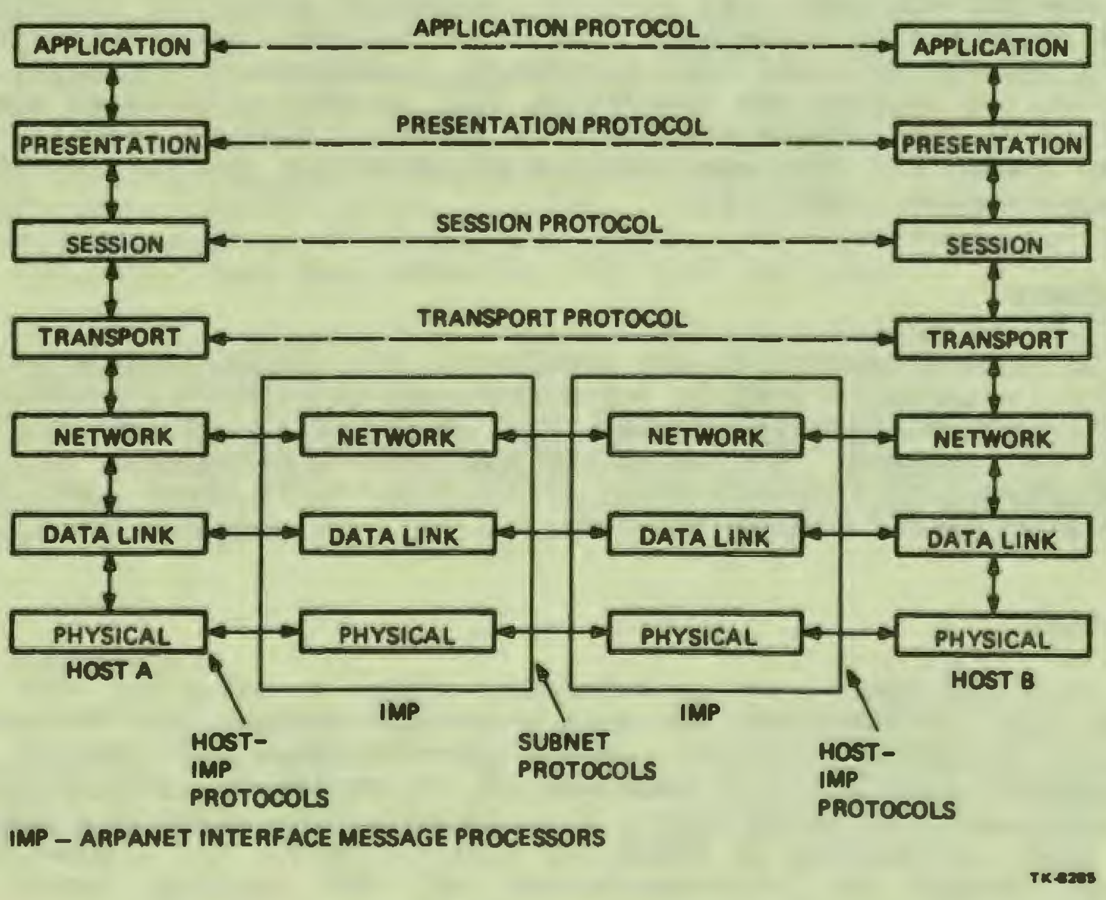


Figure 8-4 LAN Protocol

Currently some LANs are using such protocols as RS-232-C, HDLC and SDLC. Some LANs are based on the Multibus, IEEE-48 bus, and other standard bus structures. However, new protocols are coming into use with the newer ring and bus topologies to support distributed-control approaches.

Most bus networks use a contention protocol developed (and patented) by Xerox Corp. This protocol is called carrier sense **multiple** access with collision detection (CSMA/CD). CSMA/CD provides two basic rules for governing access to a shared channel:

- Rule one applies when the channel is being used by a station. In this situation, the rule states that all other stations should wait until the station using the channel is finished before attempting to transmit messages of their own.

- Rule two applies there when the channel is clear. In this situation, two or more stations may be waiting to transmit messages. When the channel clears, the stations will begin to transmit simultaneously, causing their messages to interfere, or collide. This rule states that if a **collision** occurs, the colliding stations should cut short their transmission (back off) and retransmit later at different times.

CSMA/CD derives its name from the technique used to enforce the two rules. The protocol is implemented in the interfaces that link stations to the transmission medium. A station interface detects a busy channel by listening for message signals, called **carriers**. Collisions are detected by listening to ones own transmission as it is being sent (listen-while-talk). When a message collides with another it will be garbled. Different techniques are used to randomize the back-off delay time to prevent colliding stations from retransmitting simultaneously. Some CSMA/CD schemes will automatically lengthen the back-off delay as collisions increase, to limit loading of the communications channel.

MEDIA

Of the LAN topologies previously discussed, none of them dictate the choice of media. In some cases a telephone wire medium is chosen to avoid a costly rewiring. Telephone systems, primarily PBX-based, are designed around the premise that it is most economical to use the same wiring system for voice and data.

Media choice heavily influences the costs of interface units. Costly emitter-coupled logic (ECL) assemblies (logic used to achieve higher speeds) are required to fully exploit the data rates achievable with base-band cable, while frequency-agile modems and other subsystems are needed to divide broadband cable into channels. Very high performance large-scale integration (LSI) devices are currently being used to reduce some LAN interfaces to fit onto a single board.

Telephone cabling systems with modem eliminators, twisted pairs with conventional protocols, or coaxial cable at a fraction of its traffic capacity, are currently being used in smaller networks that serve a relatively limited number of microcomputers. Fiber optics provide extremely high data rates, however, because low-cost components are only available for point-to-point connections; the topologies are very limited.

Increasingly, the choice of media and architecture will be made by users and their building architectures. New office, industrial and research centers are being built with cabling intended to form the backbone of LANs.

SOFTWARE AND HARDWARE CONSIDERATIONS

Most existing applications software will remain usable because of efforts made, on the one hand, by computer manufacturers and LAN vendors to keep new links compatible, and on the other hand, by operating systems venders to provide compatibility with new protocols. In between, agreement is growing on the intermediate protocols for the newer packet switching systems.

The common objective in designing LAN software is to establish reliable communications between user software processes. Current vendors of small LANs supply the software needed to share resources, to use the LAN as a front-end extension of a large computer, or both. This provides users with the basis for developing specialized applications without disrupting on-going applications. Popular operating systems are being enhanced to cater to LANs. Also becoming very popular is the idea of network software being supplied as firmware on an interface card for each microprocessor in the network.

A common rule of thumb in determining the cost of adding a work-station to a local network is that it should cost only a fraction of the cost of a stand-alone host, say somewhere between 10% and 20%. Today's LANs could easily satisfy this requirement. The cost per connection should be based on three major considerations:

- Type of topology
- Interconnectability (number of stations that can share resources concurrently)
- Cost of specialized networking components

A common problem in industrial and military systems is building up reliability in a centralized system without greatly increasing overhead or cost. Some of the basic approaches that are currently being used are to:

- Replace current overcrowded plant wiring with one or two broadband cables, using protocol-transparent RF modems (which help reduce radio frequency noise).
- Support rapid-fire polling and transmission with a combination of frequency and time multiplexing.
- Extend bus capabilities by using optical links.
- Utilize token-passing systems with optional redundant network switching.

RELATIONSHIP TO THE AUTOMATED OFFICE

Currently, no satisfactory definition of an automated office has been agreed upon; but one thing that is universally assumed, is that all systems and subsystems within an office will be able to communicate with each other. For example, any and all of the word processors in a company, in addition to being able to communicate with each other, will be able to communicate with all of the other computers in the office or company. Each of the computers in turn, will be able to communicate with each other, and with all of the word processors.

It should be evident that the supplier of a word processing system could not be expected to supply the interfaces to all of the other equipment with which that supplier's system might be expected to communicate. And going beyond a single company, the same word processing supplier cannot be expected to supply interfaces to every machine on the market.

As a result of this, the interfaces to various systems should be provided by the LAN as part of the network interface unit. Some of the more sophisticated LANs are providing this now for different manufacturers' computer systems. Agreements between Xerox, DIGITAL, and Intel to develop standard products and interfaces for the Ethernet network is typical of arrangements which will permit connecting a variety of equipment to a specific local network. You can expect to see more and more agreements in the near future.

EXERCISES

Match the following items with the phrases that best describe them. Each item may be used one or more times, or not at all.

A. Gateways

B. Loop

C. Ring

D. Common bus

E. Token

F. CSMA/CD

G. SDLC

H. HDLC

___ Typical data link protocol used in LAN ring topology.

___ Common carrier interface.

___ Most common LAN topology.

___ Uses a single cable and all stations are attached to it by cable taps.

___ Used to pass obtain control of the data link.

___ Most common LAN contention protocol.

SOLUTIONS

Match the following items with the phrases that best describe them. Each item may be used one or more times, or not at all.

A. Gateways

B. Loop

C. Ring

D. Common bus

E. Token

F. CSMA/CD

G. SDLC

H. HDLC

_H_ Typical data link protocol used in LAN ring topology.

_A_ Common carrier interface.

_C,D_ Most common LAN topology.

_D_ Uses a single cable and all stations are attached to it by cable taps.

_E_ Used to pass control of the data link.

_E_ Most common LAN contention protocol.

COMMON CARRIERS

9

INTRODUCTION

Any company that has government authorization to transport the property of others for a regulated charge is called a common carrier. This term applies to carriers of information as well as carriers of physical goods. The rate that a common carrier is allowed to charge for a specific service is called a tariff.

In the United States, interstate common carriers of information are regulated by the Federal Communications Commission (FCC). Intrastate carriers are usually regulated by a state Public Utilities Commission. In Europe, most governments have postal, telephone, and telegraph authorities (PTTs) that operate national communications systems.

In this module you will be provided with a short examination of widely-used common carriers of data.

OBJECTIVES

1. Describe the features of the following U.S. domestic common carriers:

   - The Bell System
   - Western Union
   - Satellite carriers

2. Describe the features of the following value-added and Public Packet-Switching Networks.

   - ARPANET
   - TYMNET
   - TELENET
   - DATAPAC
   - TRANSPAC
   - DATEX-P
   - PSS

3. Describe the features of the Public Packet-Switching Network (PPSN).

4. Describe the features of the following international common carriers:

   - RCA Global Communications
   - International Telephone and Telegraph World Communications
   - Western Union International

RESOURCES

1. Computer Networks, Andrew S. Tanenbaum, Prentice-Hall, 1981.

2. DATAPRO Reports on Data Communications, A McGraw-Hill Company, 1978.

3. Rowan, James T. Southern Bell, Atlanta, Georgia, "A Look at a Popular Switching Technique", Data Communications, March 1981, p. 117.

4. Telecommunications and The Computer, 2nd Edition, James Martin, Prentice-Hall, 1976.

U.S. DOMESTIC CARRIERS

The largest common carrier of data in the United States is the Bell System. In addition to the Bell System, there are over 2000 other telephone companies, providing various services; General Telephone ranks second.

Services provided by U.S. domestic carriers include telephone lines, leased lines, microwave links, and satellite links.

The Bell System

Services provided by the Bell System that apply to data communications include:

- Dataphone
- Wide Area Telephone Service (WATS)
- Leased Lines
- Dataphone Digital Service (DDS)

Dataphone Service -- Dataphone service provides the subscriber with a modem and a standard telephone line. The telephone can be used for normal voice communications as well as data transmissions. Dataphone users are charged the standard amount for a standard telephone line, plus a monthly charge for the modem.

Wide Area Telephone Service (WATS) -- WATS divides the United States into zones. Each user indicates the zone coverage needed. WATS charges each user a flat monthly rate for a specific number of hours of usage within the indicated zones. If the user exceeds the number of hours covered by the monthly rate, WATS adds an overtime charge. Both outward WATS (to place calls) and inward WATS (to receive calls) are available. WATS uses the normal switched public telephone service.

Leased Lines -- Leased line connections are set up and reserved for the user who pays for them. Charges for leased lines are made on a per-mile basis between defined rate centers, which correspond to major and minor cities. Rates between major cities (named grade-A rate centers) are somewhat lower than rates between other cities (named grade-B).

Bell charges a one-time installation fee for the line and a monthly charge after that. Added charges are included for line conditioning, voice service, and local four-wire connections (if they are used for full-duplex communications). If the usage exceeds a few hours each day, leased lines are probably more economical than public switched lines.

Dataphone Digital Service (DDS) -- The DDS system, offered only by Bell, transmits data in digital signals. The usual problems found in sending digital signals over long distance were solved by using special cables, and installing amplifiers that completely regenerate the signal every 6000 feet along the line. High-bandwidth microwave links are also used. DDS offers users full-duplex, synchronous communications at 2400, 4800, 9600, 56K and 1.54M bits per second (bps).

Western Union

Western Union data communications services include two subvoice-grade switched teletypewriter networks, worldwide Telex and TWX. Telex is a Western Union world-wide teletypewriter exchange service that uses the public telegraph network. TWX is a Western Union public teletypewriter exchange (switched) service in the United States and Canada. Western Union also offers voice-grade and broadband data communications services over conventional cables and microwave and satellite links.

Western Union connects over 300 cities in the United States with digital and voice lines ranging from 75 bps to 1M bps. Domestic Western Union links can be connected to links going to over 100 foreign countries.

Satellite Carriers

In 1973, RCA's Satcom began offering the first coast-to-coast domestic satellite communications service. Since then, a number of other private companies have put satellites into orbit. For example, Western Union's Westar, which is used by several carriers, provides facilities for voice, data, and television. Each Westar can handle up to 12 TV channels or 6000 voice channels.

Satellite data links, which can offer considerable savings over leased land lines, are available with capacities up to 1.2M bps.

VALUE-ADDED AND PUBLIC PACKET-SWITCHING NETWORKS

A value-added network (VAN) is a network that uses the communications facilities of a common carrier, but adds such services as packet switching (used with X.25 and local area networks), error control, or code conversion. Advantages of value-added networks include:

- Lower costs than DDS or WATS for the same application. The charges of most VANs do not increase with distance. Instead, rates are usually based on an hourly port charge, a charge for traffic volume, and charges for terminal equipment and other hardware.
- Lower error rates than normal leased lines. The VAN provides its own error-checking capability, which may reduce errors to less than one bit in every 100 billion.
- Adaptive routing. If a link goes down or becomes congested, messages are sent by another route.
- Single-source network management, which designs and installs the system and handles maintenance of communications equipment for the user.

Most VANs are packet-switched networks. Packet switching is a communication technique that allows you to send data to remote destinations over a Packet-Switching Network (PSN).

In packet switching, the data is divided into independent and standard-format units of data called packets. Each packet contains a header that consists of control and destination information. This information enables a PSN to interleave packets from many users over shared transmission lines and to deliver these packets in the correct order to their intended destinations. The route each packet takes is determined by the network, and cannot be influenced by either the sender or the receiver of the packets (refer to Figure 9-1).

Packet switching significantly improves the efficiency of a transmission line by sharing the line between many users, thus reducing the amount of time the line is idle. A PSN can also cater to different speeds and formats of data transfer and, by means of its error control mechanisms, a PSN virtually eliminates undetected errors.

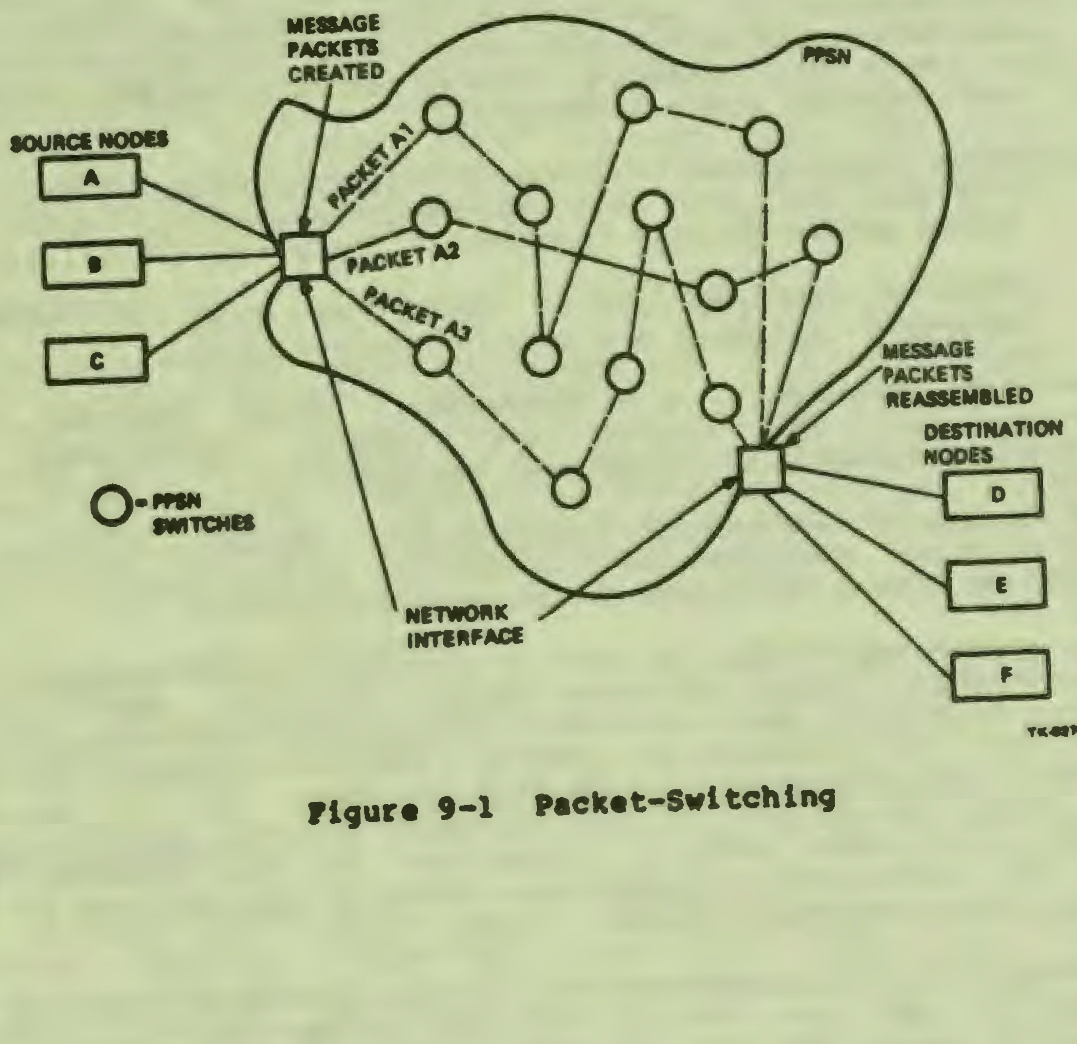


Figure 9-1 Packet-Switching

Public Packet-Switching Networks (PPSNs)

Public Packet-Switching Networks (PPSNs) are evolving to fill the need for a data communications service that is faster, more economical, and more dependable than can be provided by conventional communication circuits. The Postal Telephone and Telegraph Authorities (PTT) or common carriers of many countries now offer a PPSN.

The PPSNs are based on a number of geographically distributed switching nodes (or network interfaces), connected by high-speed links, to which you attach your computer or terminal. Any computer or terminal (that conforms to the PPSN interface) can communicate with one or more remote computers (that also conform to the interface and are also connected to the PPSN) as if they were directly connected. The PPSN provides buffering and speed changes to permit communication between computers or terminals of different types and makes and with different operating speeds. A leased circuit, provided by the country's PTT, connects the computer and the PPSN. Terminals are connected to the PPSN over leased circuits or dial-up lines (see Figure 9-2).

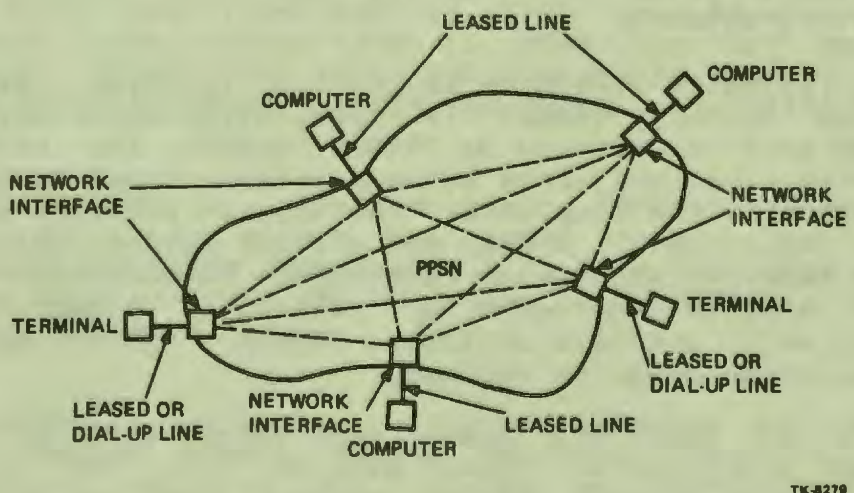


Figure 9-2 Public Packet-Switching Network

ARPANET -- The ARPANET was created by the Defense Advanced Research Projects Agency (DARPA) of the U.S. Department of Defense. ARPANET was first operational in December 1969 with only an experimental four-node network. It has been operating ever since, and has subsequently grown to well over 100 computers spanning half the globe, from Hawaii to Norway. Much of our present knowledge about networking is a direct result of this network.

The ARPANET interface message processors (IMPs) are mostly connected to 50K bps leased lines, except for those in Hawaii and Norway, which use 50 and 9.6K bps leased satellite channels, respectively. There are also some higher-speed (e.g., 230.4K bps) leased lines in use.

The ARPANET IMP-IMP protocol really corresponds to a mixture of the transport, network, and data link layers of protocol utilizing an elaborate routing mechanism. In addition, there is a mechanism that explicitly verifies the correct reception at the destination IMP of each and every packet sent by the source IMP. The ARPANET has not one, but two major transport protocols: the original NCP and the newer Transmission Control Protocol (TCP). There is no session layer or presentation layer, although some of the presentation layer functions are available through specific higher-level protocols.

TYMNET -- Originally developed as part of Tymshare Corporation's time-sharing system, TYMNET is now a value-added network. The value added services offered by TYMNET include the establishment of data communications links between terminals and computers, and between computers and computers. The network services consist of providing public dial access and private or dedicated access to the TYMNET network, and error-controlled communications to and from the requested computer or terminal. TYMNET also offers "electronic mail" services utilizing TYMNET's store and forward message-switching computer systems.

Currently, over 550 nodes are interconnected by leased voice-grade and digital lines ranging form 2400 to 9600 bps. TYMNET's features include dynamic routing, error correction, and a high-security coding technique. A unique feature among VANs is TYMNET's method of mixing data from many users into one 58-character packet. TYMNET offers data communications to many cities in North America, and also offers connections to Canada's DATAPAC network.

TELENET -- GTE's TELENET features error detection, code conversion between dissimilar devices, and dynamic routing of data in packets of up to 1024 bytes. TELENET was the first U.S. public network to offer X.25 user interface. TELENET offers service to all 50 states and servicing to over 80 cities in the U.S. TELENET services are also available to Canada's DATAPAC, to Mexico, and over 25 other countries worldwide.

Many privately-run data centers are connected to the TELENET network, which offers both public and private access to a variety of services. In effect, TELENET provides a common public/private network for both users and suppliers of time-sharing services and other process activities. TELENET also provides hardware and services for large private data networks.

DATAPAC -- DATAPAC is a Canadian packet-switching network implemented by the TransCanada Telephone System. Its features include dynamic routing, error control, collect call billing (and blocking), and several security options.

DATAPAC uses an X.25 based protocol called SNAP (for Standard Network Access Protocol), but will also accept BISYNC messages.

Line speeds range from 110 to 1200 bps for asynchronous transmissions, and from 1200 to 9600 bps for synchronous transmissions. DATAPAC has switching centers in several Canadian cities with connections to TELENET and TYMNET also available.

TRANSPAC -- TRANSPAC is a packet-switching network implemented by the French PTT. It offers two types of network services: a teletypwriter-compatible interface (50 to 1200 bps) and an X.25 interface (2400 to 48K bps). Access to TRANSPAC is via Telex, dialup line, or leased line.

DATEX-P -- DATEX-P is a packet-switching network implemented by the West German Public Packet Switching Network. DATEX-P packet-switching services are offered by Deutsche Bundespost (DBP), the German Federal Post office.

DATEX-P consists of a network of DATEX packet switching centers (DPVST's) connected by trunk lines. DATEX-P supports both intelligent terminals and simple character-mode terminals. DATEX-P intelligent terminals access the network through ports, with transmissions up to 9600 bps using a protocol equivalent to X.25. DATEX-P character-mode terminals are capable of transmitting at speeds up to 1200 bps, accessing the network' PAD facility through ports and the telephone network using a protocol that is equivalent to X.3, X.28, and X.29.

PSS -- The Packet Switching Service (PSS) is a packet-switching network implemented by the United Kingdom PPSN. PSS is provided by a network of Packet Switching Exchanges (PSEs). PSE is the PSS term for DCE. Two modes of operation are offered by the PSS - packet mode and character mode.

Packet-mode terminals are connected to the nearest PSS exchange by Datalines. Each Dataline consists of local and remote modems and the data line. Four Datalines are supported by PSS: 2400 bps, 4800 bps, 9600 bps, and 48k bps.

Character-mode terminals are connected to a PAD facility located within each PSS exchange by either a Dataline, or a dial-up using a Datel service. PSS supports two Datalines (300 bps and 1200 bps) and two Datel services (200 bps and 600 bps) for character-mode terminals.

INTERNATIONAL CARRIERS

The largest international carriers in order of size are:

- RCA Global Communications
- International Telephone and Telegraph World Communications
- Western Union International

All three offer largely equivalent services. Those related to data communications include:

- Leased lines to over 100 foreign countries, with speeds ranging from 12.5K to 56K bps. Depending on their bandwidth, these lines may also be used for voice communication.

- Datel, which allows a subscriber to connect to the switched telephone system of a foreign country.

- A choice of satellite or ground links for the same price.

EXERCISES

Circle the letter of the correct answer (either True or False).

1. Three largest international common carriers are RCA Global Communications, ITT World Communications, and the Bell System.

   A. True

   B. False

2. Only two common carriers provide coast-to-coast service -- the Bell System and Western Union.

   A. True

   B. False

3. A PTT is a private teletypewriter line.

   A. True

   B. False

4. ARPANET was the first large-scale, public packet-switching network.

   A. True

   B. False

5. Leased line rates do not depend upon distance.

   A. True

   B. False

6. Circle the letter of the company or companies that provide digital data transmission service.

   A. General Telephone

   B. Bell System

   C. Western Union

   D. Bell System and Western Union

   E. No companies currently provide this service

7. Circle the letter of the phrase that best describes a tariff.

   A. A document that specifies rates for common carrier services and facilities.

   B. A software overhead in data transmission that results from the use of CRC error checking.

   C. A contract that common carriers must sign to get permission to lay cables over private land.

   D. An agreement by which common carriers allow special equipment to be connected to their lines.

8. Circle the letter of the.X.25-compatible network or networks.

   A. TELENET

   B. DATAPAC

   C. TRANSPAC

   D. DATEX-P

   E. PSS

SOLUTIONS

Circle the letter of the correct answer (either True or False).

1. Three largest international common carriers are RCA Global Communications, ITT World Communications, and the Bell System.

   A. True

   (B.) False

2. Only two common carriers provide coast-to-coast service -- the Bell System and Western Union.

   (A.) True

   B. False

3. A PTT is a private teletypewriter line.

   A. True

   (B.) False

4. ARPANET was the first large-scale, public packet-switching network.

   A. True

   (B.) False

5. Leased line rates do not depend upon distance.

   A. True

   (B.) False

6. Circle the letter in front of the company or companies that provide digital data transmission service.

   A. General Telephone

   (B.) Bell System

   C. Western Union

   D. Bell System and Western Union

   E. No companies currently provide this service

7. Circle the letter of the phrase that best describes a tariff.

   (A.) A document that specifies rates for common carrier services and facilities.

   B. A software overhead in data transmission that results from the use of CRC error checking.

   C. A contract that common carriers must sign to get permission to lay cables over private land.

   D. An agreement by which common carriers allow special equipment to be connected to their lines.

8. Circle the letter(s) in front of the X.25-compatible network or networks.

   (A.) TELENET

   (B.) DATAPAC

   (C.) TRANSPAC

   (D.) DATEX-P

   (E.) PSS

GLOSSARY

GL

GLOSSARY

acoustic coupler - A device that converts electical signals into audio signals, enabling data to be transmitted over the public switched telephone network via a conventional telephone handset.

adjacent node - A node removed from the local node by a single physical line.

Ancillary Control Processor - A program that acts as an interface between user software and an I/O driver.

amplitude modulation (AM) - A method of transmission whereby the amplitude of the carrier wave signal is modified in accordance with the amplitude of the signal wave.

asynchronous transmission - Transmission in which time intervals between transmitted characters can be of unequal length. Transmission is controlled by start and stop elements at the beginning and end of each character. Also called Start-Stop transmission.

automatic calling unit (ACU) - A dialing device supplied by the communications common carrier. This device permits business machines to automatically dial calls over the communications network.

bandwidth - The range of frequencies assigned to a channel; the difference, expressed in Hertz, between the highest and lowest frequencies of a band. The higher the bandwidth, the greater the data throughput.

batch processing - A techinque of data processing in which jobs are collected and grouped before processing. Data thus is normally processed in a deferred mode.

baud - A unit of signaling speed equal to the number of discrete conditions or signal events per second. In asynchronous transmission, the unit of signaling speed corresponding to one interval is 20 milliseconds; that is, if the duration of the unit interval is 20 milliseconds, the signaling speed is 50 baud. Baud is the same as bit-per-second (bps) only if each signal event represents exactly one bit.

binary digit (bit) - In binary notation, either of the characters 0 or 1. "Bit" is the commonly used abbreviation for binary digit.

GLOSSARY

Binary Synchronous Protocol (BISYNC) - A data link protocol that uses a defined set of control characters and control characters sequences for synchronized transmission of binary coded data between stations in a data communications system.

bit - Abbreviation for binary digit.

block - Data transmitted as a unit, over which a coding procedure is usually applied for synchronization or error control purposes.

BPS (bits per second) - The commonly used measure for data transfer rate. (Other notations are bit(s), b.p.s., bit/sec., etc.)

byte - Assumed to be eight bits throughout, unless stated otherwise. Commonly equal to a character.

carrier - A continuous frequency capable of being modulated or impressed with a signal

centralized (computer) network - A computer network configuration in which a central node provides computing power, control, or other services. (Compare with decentralized network.)

channel - The data path joining two or more stations, including the communications control capability of the associated stations.

command node - The node where a Network Control Program (NCP) command originates.

common carrier - In data communications, a public utility company that is recognized by an appropriate regulatory agency as having a vested interest and responsibility in furnishing communication services to the general public; e.g., Western Union, The Bell System, General Telephone, etc.

component - An element in the network that can be controlled and monitored. Components include lines and nodes.

computer network - An interconnection of assemblies of computer systems, terminals, and communications facilities.

conditioning - The addition of equipment to leased voice-grade lines to provide specified minimum values of line characteristics required for data transmission; e.g., equalization and echo suppression.

control node station - the station on a network that supervises the network control procedures such as polling, selecting, and recovery. It is also responsible for establishing order on the line in the event of contention, or any other abnormal situation arising between any stations on the network. (Compare with **Tributary Station.**)

congestion - The condition that arises when there are too many packets to be queued.

congestion control - The transport component that manages buffers by limiting the maximum number of packets on a queue for a line. Also called transmit management.

data base - 1) The entire collection of information available to a computer system; 2) a structured collection of information as an entity or collection of related files treated as an entity.

data concentration - Collection of data at an intermediate point from several low and medium-speed lines for retransmission across high-speed lines.

data communication - The interchange of data messages from one point to another over communication channels. (See also **data** transmission.)

data flow - The movement of data from a source session control to a destination session control. NSP transforms data from session control transmit buffers to a network form before sending it across a logical link. NSP retransforms the data at the destination from its network form to its receive buffer form. Data flows in both directions (full-duplex) on a logical link.

datagram - A unit of data passed between Transport and the Network Services layer. When a route header is added, it becomes a packet.

data transmission - The sending of data from one place for reception elsewhere. (Compare with data communication.)

data type - ASCII data is subject to formatting conversion by the DECnet software, depending on the data's record attributes; image data is a stream of bits to which the software applies no interpretation.

decentralized (computer) network - A computer network, where some of the network control functions are distributed over several network nodes. (Compare with centralized network.)

demodulation - The process of retrieving an orignal signal from a modulated carrier wave. This techinque is used in data sets to make communication signals compatible with computer signals.

dial-up line - A communications circuit that is established by a switched circuit connection.

distributed network - A network configuration in which all node pairs are connected either directly, or through redundant paths through intermediate nodes.

down-line load - The process by which one node in a computer network transfers an entire system image, or a program (task) image, to another node and causes it to be executed.

echo suppressor - A device used to suppress the effects of an echo.

equalization - Compensation for the increase of attenuation with frequency. Its purpose is to produce a flat frequency response.

end node - A topological description of a nonrouting node. Since a nonrouting node cannot perform route-through and supports only a single line, it must be an end node. However, it is also possible for a routing node with a single line to be an end node.

end user module - A module that runs in the "user space" of a network node and communicates with session control to obtain logical link service.

error control - The NSP function that ensures the reliable, sequential delivery of NSP data messages. It consists of sequencing, acknowledgment, and retransmission mechanisms.

executor node - An active network node connected to one end of a line used for a load, dump, or line loop test; it is the node that executes the request.

flow control - The NSP function that coordinates the flow of data on a logical link in both directions, from transmit buffers to receive buffers to ensure that data is not lost, to prevent buffer deadlock, and to minimize communications overhead. The protocol mechanism that ensures the sending station does not overrun the receiving station with more data than it can accept.

framing - The DDCMP component that synchronizes data at the byte and message level.

frequency division multiplexing (FDM) - Dividing the available transmission frequency range into narrower bands each of which is used for a separate channel.

frequency modulation (FM) - A method of transmission whereby the frequency of the carrier wave is changed to correspond to changes in the signal wave.

front-end processor - A communications computer associated with a host computer. It can perform line control, message handling, code conversion, error control, and application functions such as control and operation of special-purpose terminals.

full-duplex line/channel - The line can transmit data in both directions simultaneously. A full-duplex line allows a node to send and receive data at the same time. The channel line services concurrent communications in both directions (to and from the station).

fully-connected network - A network in which each node is directly connected with every other node.

half-duplex line/channel - The line can transmit data in either direction, but only in one direction at any given time. In other words, the line cannot be used to send and receive data simultaneously. The channel permits two-way communications, but in only one direction at any instant.

hardware controller - The control hardware for a line. For a multiple line controller device, the controller is responsible for one or more units. The controller indentification is part of a line identification.

hierarchical network - A computer network in which processing control functions are performed at several levels by computers specially suited for the functions performed; for example, in a factory or laboratory automation.

hop - To the transport layer, the logical distance between two adjacent nodes in a network.

host node - Provides services for another node (for example, during a down-line task load).

interactive communication - A protocol that allows one system to interact with a connected system at the transaction level rather than at the file level.

interface - 1) A shared boundary defined by common physical interconnection characteristics and meanings of interchanger signals; 2) a device or equipment making possible interoperation between two systems; e.g., a hardware component or a common storage register; 3) a shared logical boundary between two software components.

leased-line - A line reserved for the exclusive use of a leasing customer without interexchange switching arrangments. Also called a private line.

line - A physical path. In the case of a multipoint line, each tributary is treated as a separate line.

line cost - An arbitrary integer value assigned to a line between two adjacent nodes. Each line has a separate cost. Packets are routed on paths with the least cost. Nodes on either end of a line can assign different costs to the same line.

line level loopback - Testing a specific data link by sending messages directly to the data link layer and over a wire to a device that returns the message to the source.

link - 1) Any specific relationship between two nodes in a network; 2) a communications path between two nodes; 3) a data link (refer to line).

logical link - A carrier of a single stream of full-duplex traffic between two user-level processes. A virtual channel between two end users in the same node or in separate nodes. Session control acts as an interface between an end user requiring logical link service and NSP, which actually creates, maintains, and destroys logical links.

local node - A frame of reference; the node at which the user is physically located. (Compare with remote node.)

master station - A station that has control of a channel at a given instant, for the purpose of sending data messages to a slave station (whether or not it actually does).

maximum address - The maximum number of nodes the local node can handle in its routing data base.

maximum cost - An operator-controlled transport parameter that defines the point where the routing decision algorithm in a node declares another node unreachable, because the cost of the least costly path to the other node is excessive. For correct operation, this parameter must not be less than the maximum path cost of the network.

maximum hops - An operator-controlled transport parameter that defines the point where the routing decision algorithm in a node declares another node unreachable, because the length of the shortest path between the two nodes is too long. For correct operation, this parameter must not be less than the network diameter.

maximum path cost - The routing cost between the two nodes of the network having the greatest routing cost, where routing cost is the cost of the least cost path between a given pair of nodes.

maximum path length - The routine distance between the two nodes of the network having the greatest routing distance, where routine distance is the length of the least cost path between a given pair of nodes.

maximum visits - An operator-controlled transport parameter that defines the point where the packet lifetime control algorithm discards a packet which has traversed too many nodes. For correct operation, this parameter must not be less than the maximum path length of the network.

message - The unit of communication as seen by the user; it can be segmented into several packets to traverse the network, or in some circumstances several messages can be carried in one packet.

modem - Modulator-demodulator. A device that modulates and demodulates signals transmitted and received over communications circuits. Often referred to as a dataset.

multiple line controller - A controller that can manage more than one unit. (DIGITAL multiple line controllers are also called multiplexers.)

multiplex - to simultaneously transmit two or more data streams on a single channel. In DNA, NSP is the only protocol that multiplexes.

multipoint connection - A network configuration in which more than two computers are attached to the same line. Use of this type of line normally requires some kind of polling mechanism, addressing each terminal with a unique ID. Also called multi-drop. (Compare with point-to-point connection.)

network - A configuration of two or more computers linked to share information and resources. A computer having the capacity to participate in a network is called a node.

network diameter - The distance between the two nodes of the network having the greatest reachability distance, which is the length of the shortest path between a given pair of nodes.

node - A network management component consisting of a DIGITAL system that supports DECnet software.

node address - The unique numeric identification of a specific node.

node level loopback - Testing a logical link using messages that flow with normal data traffic through the session control, network services, and transport layers within one node, or from one node to another and back. In some cases, node level loopback involves using a loopback node name associated with a particular line.

node name - An optional alphanumeric identification associated with a node address in a strict one-to-one mapping. No name may be used more than once in a node. The node name must contain at least one alpha character.

node name mapping table - Defines the correspondence between node names and node addresses or channel numbers. Session control uses the table to identify destination nodes for outgoing connect requests, and source nodes for incoming connect requests.

noise - Undesirable disturbances in a communication system. Noise can generate errors in tranmission.

nonrouting (end) node - Can send packets to other nodes in the network, but packets cannot be forwarded or routed through it. It can be adjacent to one other node only; therefore, it is always an end node in a Phase III configuration.

null modem - A device which interfaces between a local peripheral that normally requires a modem, and the computer near it that expects to drive a modem to interface to that device; an imitation modem in both directions.

object type - Numeric value that may be used instead of a process name for process or task adressing by DECnet processes.

operating system - An integrated collection of service routines for supervising the sequencing and processing of programs by a computer. An operating system provides access to the features of a central processor; also organizes and optimizes a central processor and peripheral equipment for a specific range of applications.

packets - (transport protocol) A group of bits including data and control elements which is switched and transmitted as a composite whole. The data and control elements, and possibly error control information, are arranged in a specified format.

packet lifetime control - The transport component that monitors lines to detect if a line has gone down, and prevents excessive looping of packets by discarding packets that exceed the maximum visit limit.

parallel data transmission - A data communication technique in which more than one code element (for example, bit) of each byte is sent or received at the same time.

path - The route a packet takes from source node to destination node. This can be a sequence of connected nodes between two nodes.

path cost - The sum of the line costs along a path between two nodes. Path cost is direction dependent; cost A to B is not necessarily equal to cost from B to A.

path length - The sum of the hops along a path between two nodes. Path length is the number of lines a packet must go through to reach its destination.

phase II node - Runs a Phase II implemenation of DECnet and therefore does not support routing. It can send packets only to adjacent nodes and cannot forward packets it receives onto other nodes in the network. It can be adjacent to one or more full routing nodes and/or to other Phase II nodes. Logically, it is an end node within a Phase III configuration.

phase III node - Runs a Phase III implementation of DECnet and supports routing as either a full routing or nonrouting (end) node. (Refer to routing node and nonrouting node.)

phase modulation (PM) - A method of transmission whereby the angle of phase of the carrier wave is varied in accordance with the signal.

physical link - An individual hardware addressable communications path. In terms of hardware, a physical link is a combination of either a channel and its controllers, or a channel, controllers, and units.

piggybacking - Sending an acknowledgment within a returned data message.

piplining - Sending messages without waiting for individual acknowledgment of each successive message.

point-to-point (link/channel) - The direct connection of two nodes by a single link or channel. A network configuration in which a connection is established between two (and only two) computers. (Compare with multi-point connection.)

polling - The process of inviting another station or node to transmit data. (Refer to control station and tributary station.)

protocol - A basic procedure or set of rules that govern and control the flow of messages between computers; also, a set of conventions between communicating processes on the format and content of messages to be exchanged. DIGITAL Network Architecture (DNA) uses three basic protocols in a layered structure as the framework for DECnet.

reachable node - A destination node to which the DECnet routing module has determined there is a usable path.

real time system - A system performing computation during the actual time the related physical process transpires, so that the results of the computation can be used in guiding the process.

reassembly - The placing of multiple, received data segments by NSP into a single session control receive buffer.

remote job entry (RJE) - 1) Submission of jobs through an input device that has access to a computer through a communications link; 2) the mode of operation that allows input of a batch job by a card reader at a remote site, and receipt of the output via a line printer or card punch at a remote site.

remote node - A frame of reference; any node other than the one at which the user is located in the network. (Compare with local node.)

retransmission - Resending data messages that have not been acknowledged within a specific period of time. This is usually part of a protocol's error control mechanisms.

route-through - Directing packets from source nodes to destination nodes by one or more intervening nodes. Routing nodes permit route-through. Also called packet switching.

routing - A network function that directs data message packets from a source node to a destination node.

routing node - A full routing node can forward packets to other node in the network and can be adjacent to all other types of nodes. A Phase III DECnet node that contains the complete set of transport modules, and can deliver, receive, and route packets through.

satellite (node) - A node that is dependent upon another node (host) for software loading and control. A satellite node has little or no peripheral equipment of its own.

segment - The data carried in a data segment message; NSP divides the data from session control transmit buffers into numbered segments for transmission by transport.

segmentation - The division of normal data from session control transmit buffers into numbered segments for transmission over logical links.

serial transmission - A method of transmission in which all information is sent sequentially on a single channel, rather than simultaneously, as in parallel transmission.

slave node (station) - A tributary station that can send data only when polled or requested to be a master control station. In some multiplex situations, a tributary can act as both a slave and a master.

solicited messages - Normal data messages which network taks explicitly send and receive.

star topology - A network configuration in which one central node is connected to more than one adjacent end node. A star can be a subset of a larger network.

station - With regard to the data link layer protocol, this is a termination on a data link. A station is a combination of the physical link (communication hardware) and the data link protocol implementation.

subchannel - A logical communications path within a logical link that handles a defined category of NSP data messages. Because data segment messages are handled differently from other data messages, the two types of messages travel in two different subchannels.

synchronous, serial data transmission - Transmission in which the data characters and bits are transmitted at a fixed rate, with the transmitter and receiver synchronized. This eliminates the need for start-stop elements, thus providing greater efficiency. (Compare with asynchronous transmission.)

switched line - A communications link for which the physical path can vary with each usage; e.g., the dial-up telephone network.

target node - The node that receives a memory image during a down-line load, generates an up-line dump, or loops back a test message.

telecommunications - Data transmission between a computing system and remote devices or another computing system.

time division multiplexing - A system of multiplexing in which channels are established by connecting terminals one at a time, at regular intervals, by means of an automatic distribution.

topology - the physical or logical placement of nodes in a computer network.

transparent data - Binary data transmitted with the recognition of most control characters suppressed. DDCMP provides data transparency because it can receive data containing bit patterns that resemble DDCMP control characters.

tributary node (station) - A station other than the control station on a centralized multipoint data communications system that can communicate only with the control station, when polled or selected by the control station.

unattended operation - The automatic features of a node's operation that permit the transmission and reception of messages on an unattended basis.

unit - The hardware controlling one channel on a multiple line controller. A unit, a controller, and associated data link modules form a station.

up-line dump - Used to send a copy of a target node's memory image up a line, to a file at the host node.

NETWORK CONCEPTS COURSE CRITIQUE

Please complete this course critique and either turn it in to the course administrator or send it to DIGITAL Equipment Corp., DDP Training and Development, Mail Stop TW/CØ5, Tewksbury, MA Ø1876.

Date:

1. How long did it take you to complete the course?

2. Do you feel that your student environment aided or hampered your learning?

   Aided

   Hampered

   Please explain:

3. Degree of difficulty:

   Very Difficult

   Average

   Simple

   Please explain:

4. Do you feel that the course taught you the skills stated in the objectives?

   Yes

   No

   If no, please elaborate:

NETWORK CONCEPTS COURSE CRITIQUE

5. What areas of the course were not presented clearly (modules, tests, exercises, resource documents, etc.)?

6. Do you feel that the information in this course was relevant to your needs in performing your job?

   Yes

   No

   Please explain your response:

7. What is your general feeling about this course?

8. What suggestions (additions/deletions) do you have which would improve this course?

digital

108714

Digital Equipment Corporation • Bedford, MA 01730