Cabletron Systems

# ETHERNET TECHNOLOGY GUIDE

# Notice

Cabletron Systems reserves the right to make changes in specifications and other information contained in this document without prior notice. The reader should in all cases consult Cabletron Systems to determine whether any such changes have been made.

The hardware, firmware, or software described in this manual is subject to change without notice.

IN NO EVENT SHALL CABLETRON SYSTEMS BE LIABLE FOR ANY INCIDENTAL, INDIRECT, SPECIAL, OR CONSEQUENTIAL DAMAGES WHATSOEVER (INCLUDING BUT NOT LIMITED TO LOST PROFITS) ARISING OUT OF OR RELATED TO THIS MANUAL OR THE INFORMATION CONTAINED IN IT, EVEN IF CABLETRON SYSTEMS HAS BEEN ADVISED OF, KNOWN, OR SHOULD HAVE KNOWN, THE POSSIBILITY OF SUCH DAMAGES.



Printed in the United States of America.

Order Number: 9031913-01 April 1997

Cabletron Systems, Inc.
P.O. Box 5005
Rochester, NH 03866-5005

# Contents

## CHAPTER 5 ETHERNET MEDIA ACCESS METHOD



## CHAPTER 6 ETHERNET DEVICES



## CHAPTER 7 ETHERNET NETWORK DESIGN

**CHAPTER 8 PROPAGATION DELAY**



**CHAPTER 9 ETHERNET BRIDGE OPERATION**



## Index

Chapter 1

# Overview

## Purpose of This Manual

Welcome to the Cabletron Systems Ethernet Technology Guide. This guide discusses the aspect of an Ethernet network known as the physical and datalink layer. Although there may be some mention of specific networking products and software, the primary focus is on the understanding, design and implementation of a generic Ethernet Local Area Network (LAN).

Throughout this document, references are made to both Ethernet and IEEE 802.3 (CSMA/CD). The differences between Ethernet Version 2 and 802.3 are relatively minor. Because of these minor differences you will find the text often refers to 802.3 as Ethernet. Throughout the industry, the most popular nomenclature for an IEEE 802.3 CSMA/CD network is Ethernet. Where there is a major difference between Ethernet and 802.3, it is noted. This manual provides a basic overview of Ethernet and IEEE standard 802.3 technology and is not meant to be a complete guide. The objective of this manual is to provide Cabletron Systems customers with information to understand why networks should be designed in a particular way and why the rules need to be followed.

## Who Should Use This Manual

This manual is intended for users of Cabletron Systems Ethernet products and should be used as a supplement to Cabletron Systems Ethernet Product User's Manuals.

# Structure of This Manual

This manual is organized as follows:

Chapter 1, **Overview** - Outlines the purpose of this manual, who should use it, and how it is structured.

Chapter 2, **Introduction** - Introduces Ethernet and gives a brief discussion of the features and characteristics of the Ethernet technology.

Chapter 3, **Ethernet LAN Standards** - Discusses the Open Systems Interconnect Model (OSI).

Chapter 4, **Ethernet Media Access Method** - Explains the different Ethernet data frames and describes the Ethernet data encoding technique.

Chapter 5, **Ethernet Media Access Method** - Provides a basic explanation of Carrier Sense Multiple Access with Collision Detection (CSMA/CD) operation.

Chapter 6, **Ethernet Devices** - Describes the different devices used on an Ethernet network, and provides examples of each.

Chapter 7, **Ethernet Network Design** - Discusses the design considerations of a single segment 10BASE5 Ethernet network and builds it into a multi-segment Ethernet network. The chapter then continues with the design of a 10BASE2 Ethernet network, a Fiber Optic Ethernet network, and finally a 10BASE-T Ethernet network.

Chapter 8, **Propagation Delay** - Provides an example of the step-by-step process used in calculating the propagation delay of an Ethernet network.

Chapter 9, **Ethernet Bridge Operation** - Provides a step-by-step explanation of the operation and learning process of an Ethernet Bridge. It also provides a detailed explanation on the Spanning Tree Process.

Chapter 2

# Introduction

*This chapter introduces Ethernet features and describes characteristics that distinguish Ethernet from other Local Area Network (LAN) technologies such as Token Ring or FDDI.*

## Ethernet History

Ethernet was developed by Xerox Corporation's Palo Alto Research Center (PARC) in the mid-1970s. Ethernet was the technological basis for the IEEE 802.3 specification, which was initially released in 1980. Shortly thereafter, Digital Equipment Corporation, Intel Corporation, and Xerox Corporation jointly developed and released an Ethernet specification (Version 2.0) that is compatible with IEEE 802.3. Together, Ethernet and IEEE 802.3 currently maintain the greatest market share of any Local Area Network (LAN) protocol. Today, the term Ethernet is often used to refer to all Carrier Sense Multiple Access/Collision Detection (CSMA/CD) LANs that generally conform to Ethernet specifications, including IEEE 802.3.

At the time of its creation, Ethernet was designed to fill the middle ground between long-distance, low-speed networks carrying data at high speeds for very limited distances. Today, Ethernet is well-suited to applications where a local communication medium must carry sporadic, occasionally heavy traffic at high peak data rates.

# Ethernet Features

The Institute of Electrical and Electronic Engineers (IEEE) is a standards organization that establishes standards for many different technical areas. This broad standards responsibility includes computer networking. IEEE project 802 is responsible for networking standards for all network access methods while project 802.3 specifically defines the Carrier Sense Multiple Access/Collision Detection (CSMA/CD) access method, or Ethernet.

The following sections detail specific features involved with the CSMA/CD media access method.

## Media Access Method

As mentioned above, Ethernet is a Carrier Sense Multiple Access/Collision Detection (CSMA/CD) LAN technology. Stations on an Ethernet LAN can access the network at any time. Before sending data, Ethernet stations "listen" to the network to see if it is already in use. If so, the station wishing to transmit waits. If the network is not in use, the station transmits. A collision occurs when two stations listen for network traffic, "hear" none, then transmit simultaneously. In this case, both transmissions are damaged and the stations, sensing this collision, must retransmit at some later time. Backoff algorithms determine when the colliding stations retransmit.

Ethernet is a broadcast network. In other words, all stations see all frames, regardless of whether they represent an intended destination. Each station must examine received frames to determine if it is the destination. If so, the frame is passed to a higher protocol layer for appropriate processing.

## Bandwidth

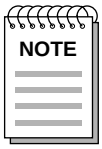
Ethernet bandwidth is 10 megabits per second although later developments have produced Fast-Ethernet bandwidths of 100 megabits per second.

## Transmission Medium

Ethernet transmits data frames over a physical medium of coaxial, fiber optic, or twisted pair cable. The coaxial and fiber optic cable typically represents the backbone of an Ethernet LAN while twisted pair is used as a low cost connection from the backbone to the desktop.

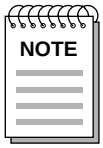
Ethernet LANs have the following media restrictions in order to adhere to IEEE standards:

- Bus Length: The maximum bus length for an Ethernet LAN for all media types is the following:
  - 500 m for 10BASE5 coaxial cable
  - 185 m for 10BASE2 coaxial cable
  - 2,000 m for multimode fiber optic (10BASE-F) cable (5,000 m for single mode)
  - 100 m for twisted pair (10BASE-T) cable.




These media lengths are not precise values. Actual maximum cable lengths are strongly dependent on the physical cable characteristics.

- AUI Length: The maximum Attachment Unit Interface (AUI) cable length is 50 m for connections from a transceiver to an Ethernet device and 16.5 m for office AUI.
- Number of Stations per Network: IEEE standards specify that the maximum allowable number of stations per un-bridged network is 1,024, regardless of media type. 10BASE5 networks are allowed 100 taps per segment while 10BASE2 networks are allowed 30 taps per segment with a maximum of 64 devices per tap each (Fiber optic and twisted pair cable are point-to-point media which do not allow taps or branches).




If it becomes necessary to extend the network beyond the IEEE limit of 1,024 devices, a bridge can be used to connect another full specification Ethernet network.

- Maximum Signal Path: The maximum allowable signal path is 4 repeaters, 5 segments (with at least 2 segments being unpopulated Inter-Repeater Links), and 7 bridges for all media types.

## Frame Transmission

Ethernet stations encode their data into groups using a method known as Manchester Encoding. These data groups, called frames, can be one of four basic Ethernet frame types:

- 802.3 “raw” frame
- Ethernet II (DIX) frame
- Ethernet 802.2 frame
- Ethernet SNAP frame

The Ethernet 802.2 and Ethernet SNAP frames are extensions of the 802.3 “raw” frame format, while the Ethernet II frame is formatted differently. More about Ethernet frame types and Manchester Encoding will be discussed in Chapter 4, **Ethernet Data Frames**.

## Ethernet Topologies

The physical topology of the network defines its shape. Various network topologies exist in the form of stars, trees, rings or buses. Complex networks may employ several of the above topologies to form a hybrid topology. A bus of stars or a ring of buses are two such examples. Figure 2-1 shows some examples of several topology types. The boxes indicate equipment of some type and the lines indicate cabling.

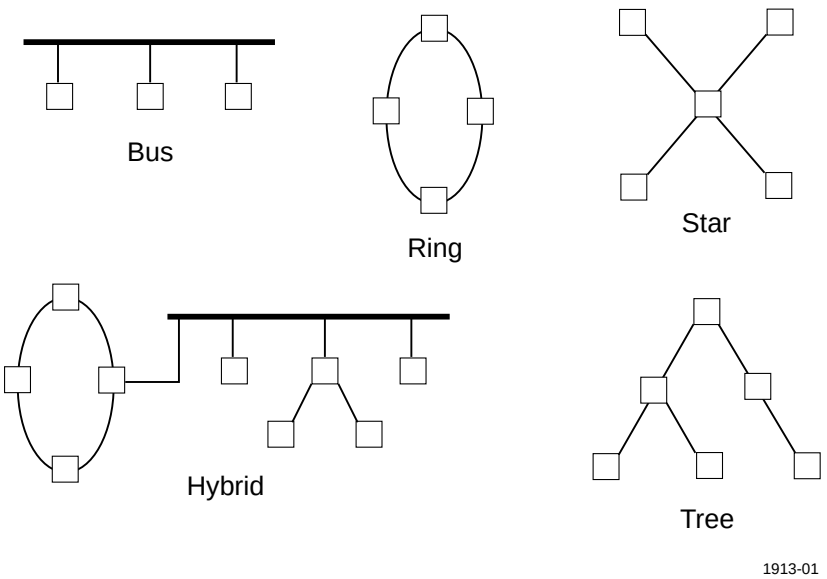



**Figure 2-1. Various Network Topologies**

The most popular topologies used in Ethernet are the bus, star and tree. Even though this discussion is about Ethernet, it is worth spending a few moments on topologies not commonly used with Ethernet. A couple of definitions at this point will assist with understanding the following descriptions.

- Point-to-Point: A point-to-point connection is a connection between two and only two network devices (computers, servers, printers, etc.). No taps or daisy chains are allowed. The most common point-to-point medias are twisted pair and fiber optics.
- Multi-point: A multi-point connection utilizes a single cable to connect more than two network devices. A cable that has several devices connected, one after another, (also known as daisy chaining) is an example of a multi-point connection. The most common multi-point media is coaxial cable.

Today's networks employ various media topologies. The following sections look at the general characteristics of the three most popular topologies: bus, ring, and star.

## Bus Topology

Employed in most older networks, the bus topology is a multi-point network topology is which all devices are connected by a single common cable or communications link. Taps are used to get the signal from the coaxial cable to the device.

### General Characteristics

Ethernet uses what is known as a contention bus topology. Any station on the network can talk as long as no other station is talking. The more stations that are on the network and want to talk, the worse the overall performance. As a general rule, bus topologies are fairly straightforward and easy to expand. Often, it is possible to expand the network without affecting the network operation.

### Vulnerability

Because all stations in the network share a common transmission media, a failure of that media interrupts the exchange of signals between stations. In a properly designed and constructed network however, such failures are uncommon. Failure of a single workstation will not usually affect the entire network.

## Ring Topology

A ring topology is a point-to-point topology in which the network devices are connected, device to device, in an unbroken circle. Each signal to be transmitted on the network must be processed by each station on the ring before it is passed (or repeated) to the next station.

### General Characteristics

Ring topologies commonly use an access method that is called token passing. No station may talk unless that station has a free token, or specialized signal code designated to determine which station on the network is allowed to transmit. The token is passed from station to station on the network along with the data being transmitted until it is released by the receiving station.

Ring topologies can be complex in nature. They are easy to expand but may involve calculations of cable lengths to keep the network within specification. Most modern ring topologies resemble a physical star but careful examination will reveal a logical ring cabled in a star configuration. The use of networking hardware such as modular hubs takes care of maintaining the logical ring in the wiring closet.

### Vulnerability

Adding or removing network stations is simple and can, in most cases, be done while the network is in operation. High level software takes care of the recognition of problem nodes and also, in most cases, will remove the problem nodes from the network and automatically reconfigure the ring.

## Star Topology

A star topology is a point-to-point network in which the network devices are connected through a central concentrator or controller. Two types of access methods are employed: polling and contention.

### Polling Star Topology

On a polling network, devices cannot talk or send messages unless they are given permission (or polled) by a central computer or controller. A device must wait to transmit until the controller asks for the information. Performance of a polled network is dependent on the performance of the controller and the number of devices attached to the controller.

Failure of the controller in a star topology network will bring the network down. Failure of an individual node typically will not affect the remainder of the network.

### Contention Star Topology

The contention star is the access method used with Ethernet. Workstations are connected to a hub or concentrator located in a wiring closet. Contention rules dictate that only one station can transmit data at any given time and any station may talk providing the network is quiet. This access method eliminates the need for polling and vastly improves throughput and performance. Hubs can be expanded to handle hundreds of devices without performance degradation. Expansion is easily accomplished by simply plugging in a connection at the concentrator.

Failure of the hub can bring that section of the network down. Some manufacturers allow for redundant backup of the hub and multiple load sharing power supplies to reduce the possibility of hub failure and minimize the impact of any such failure. The failure of a node will not normally affect network operation.

## Hybrid Network Topology

A hybrid topology is a combination of any of the three major topologies. Examples include a ring of stars or a bus of stars or trees. Hybrid networks may use a combination of point-to-point and multi-point connection techniques.

For any of the above networks to function reliably and to allow multiple vendors' equipment to interoperate on the same network, a set of standards have been developed. The most notable standards organizations affecting data communications are the International Organization for Standardization (ISO), the Institute of Electrical and Electronic Engineers (IEEE), the Telecommunications Industry Association (TIA), the American National Standards Institute (ANSI), and the Consultative Committee on International Telegraphy and Telephony (CCITT). In the following chapter we will look at the ISO standard as well as the IEEE project 802 standards, specifically 802.3.

Chapter 3

# Ethernet LAN Standards

*Standards play an important role in modern local area networks. Without standards, users are forced to buy proprietary networking equipment from a single vendor. Companies come and go, and product lines are changed or discontinued. This leads to increased network costs to the users, network down time and network equipment that does not inter-operate if standards are not in place.*

*Standards allow for easy integration of multiple vendor equipment into a common network. If one company disappears, a purchaser has the flexibility to purchase another vendor's product, being confident that the two standards based products inter-operate.*

---

## The Open Systems Interconnect (OSI) Model

The International Organization for Standardization (ISO) Open Systems Interconnect (OSI) Model provides a framework for the development of system connection standards by using a consistent hierarchy of rules. The OSI model defines where the needed tasks for system interconnection are performed but not how they are performed. How tasks are performed on a given layer is defined by the protocols, or rules, written for that particular network based on the OSI model. The layers may be implemented in hardware, software or both. Each layer in a network based on the OSI Model performs specific functions required for proper system interconnection.

As shown in Figure 3-1, there are seven layers in the OSI Model. They begin with the Application Layer and finish with the Physical Layer.

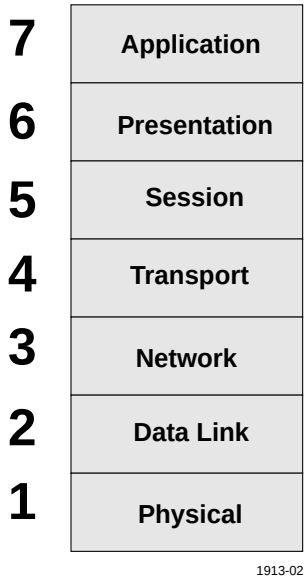



**Figure 3-1. Open Systems Interconnect (OSI) Model**

## Application of the OSI Model

The perception of network operation appears as a direct peer-to-peer communication to the user. The user message appears to go from the sending application layer directly to the receiving application layer as if the devices were directly attached. In actuality, the user message is routed from the sending application layer down through the other layers of the system. Each layer adds to or modifies the message according to the network operating system's protocol for each layer. The message passes through all the layers of the system before appearing on the data channel (cable or communications media) at the physical layer.

From the data channel the message passes upward through the same layers at the destination device. As the message progresses from layer to layer, each layer strips off information that was added by its counterpart in the transmitting station. The result is the same message as was originally sent, arriving at the top of the destination application layer.

Each layer performs a specific function with respect to the complete communications process. The functions of each layer are as follows:

- LAYER SEVEN: **Application Layer**–The application layer is the user's interface with the network. This layer directly interacts with user application programs to provide access to the network. All other layers exist to support the requirements of this layer. The application layer is usually involved with tasks such as electronic mail and file transfer.

- LAYER SIX: **Presentation Layer**–The presentation layer deals with data translation and code conversion between devices with different data formats (e.g., ASCII to EBCDIC). This layer also handles translation between differing device types and file formats, as well as data encryption and decrypting services.

- LAYER FIVE: **Session Layer**–The session layer manages the communication dialogue (the "session") between two communicating devices. The session layer establishes rules for initiating and terminating communications between devices and provides error recovery as well. If an error or communications failure is detected, the session layer retransmits data to complete the communications process. The session layer requests a certain level of service from the transport layer such as one way transmission that doesn't require a reply, or a two way conversation that requires a lot of monitoring and feedback.

- LAYER FOUR: **Transport Layer**–The transport layer deals with the optimization of data transfer from source to destination by managing network data flow and implementing the quality of service requested by the session layer. The transport layer determines the packet size requirements based on the amount of data to be sent and the maximum packet size allowed on the communications media. If the data to be sent is larger than the maximum packet size allowed on the network, the transport layer is responsible for dividing the data into acceptable sizes and sequencing each packet for transmission. During the dividing and sequencing process, this layer adds information such as sequence number and error control information to the data portion of the packet.

  When receiving data from the network layer, the transport layer ensures that the data is received in order and checks for duplicate and lost frames. If data is received out of order, which is possible in a larger, routed network, the transport layer correctly orders the data and passes the data up to the session layer for additional processing. A popular protocol that uses the transport layer is Transmission Control Protocol (TCP) used in TCP/IP.

- LAYER THREE: **Network Layer**–The network layer accepts data from the transport layer and adds the appropriate information to the packet to provide proper network routing and some level of error control. Data is formatted for the appropriate communications method such as local area network, wide area network such as T1, or packet switched technology such as X.25. A popular protocol that uses the network layer is the Internet Protocol (IP) used by TCP/IP.

- LAYER TWO: **Data Link Layer**–The data link layer is involved with transmission, error detection and flow control of the data. The major function of the data link layer is to act as a shield for the higher layers of the network model, controlling the actual transmission and reception process. Error detection and control of the physical layer are the primary functions of this layer ensuring the upper layers that any data received from the network is error free. The IEEE 802 model divides the data link layer into two sub-layers: Logical Link Control (LLC) and Media Access Control (MAC).

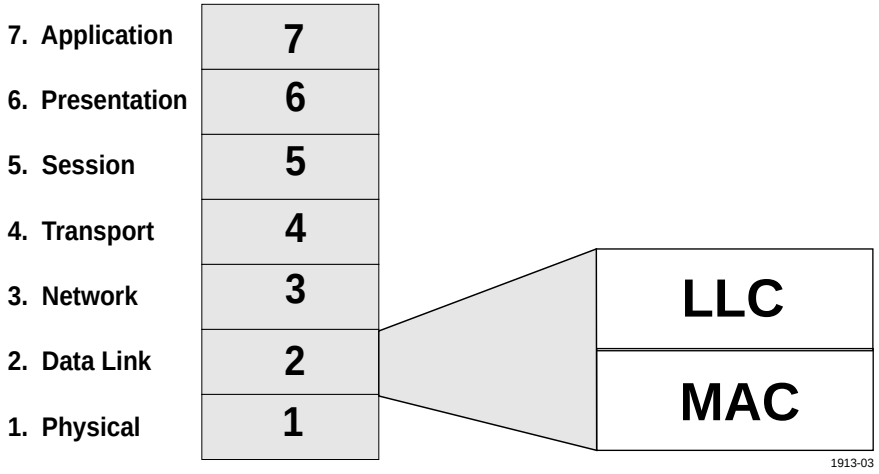



**Figure 3-2. Data Link Layer of the OSI Model**

- **Logical Link Control**–The LLC layer is responsible for shielding the upper layers from any particular access method or media. The upper layers need not worry about whether they are connected to a Token Ring or Ethernet network because the logical link control handles the interface. The LLC is a defined standard (IEEE 802.2) that provides for a common interface of the layers above to any physical network implementation.
- **Media Access Control**–The MAC layer is responsible for several areas of operation. On the transmit side the MAC layer is responsible for receiving data from the Logical Link Control layer and encapsulating it into a packet ready for transmission. The MAC layer is also responsible for determining if the communications channel is available. If the channel is available, the MAC layer transmits the data onto the cable through the physical layer and monitors the physical layer status for an indication of a collision (more than one station transmitting at the same time). If there is a collision, the MAC layer also handles the backoff and retransmission function.

On the receive side, the MAC layer follows the reverse of the above steps. It checks the frame for errors, strips control information then passes the remainder of the packet to the upper layers by way of logical link control.

- LAYER ONE: **Physical Layer**–At this layer, the transmission of data between devices is defined. The definition includes cables and connectors, connector pinouts, voltage levels that represent digital logic levels, bit timing, and the actual network interface device called a Transceiver (transmitter/receiver). The IEEE 802 model divides the physical layer into four sub-layers: Physical Layer Signaling (PLS), Attachment Unit Interface (AUI), Physical Medium Attachment (PMA), and Medium Dependent Interface (MDI).

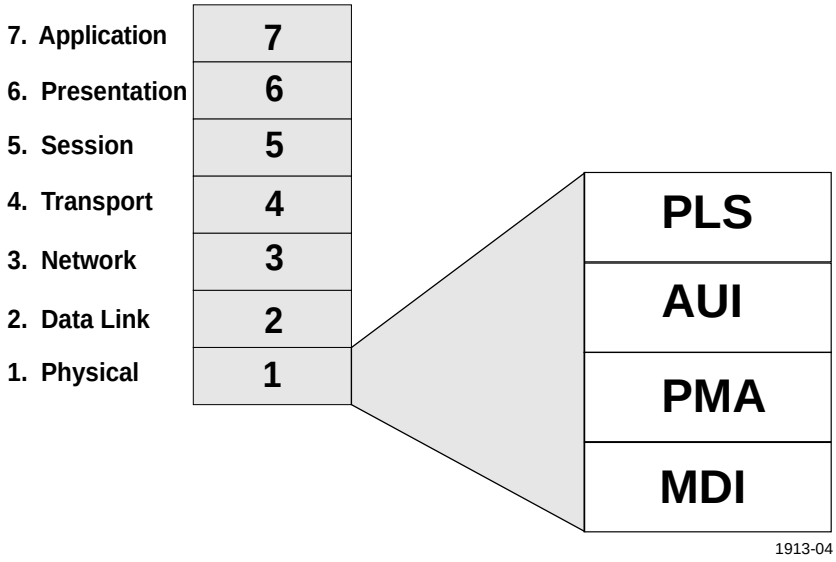



**Figure 3-3. Physical Layer of the OSI Model**

- **PLS (Physical Layer Signaling)**–Defines the signaling and the interface to the transceiver cable.
- **AUI (Attachment Unit Interface)**–Defines the transceiver cable specifications.
- **PMA (Physical Medium Attachment)**–Defines the transceiver operation and specifications.
- **MDI (Medium Dependent Interface)**–Defines the specifications for the portion of the transceiver that connects to specific cable types such as 10BASE5 coaxial cable.

The remaining chapters will concentrate primarily on the Physical and Data Link layers of the OSI Model.

Chapter 4

# Ethernet Data Frames

*Most computer networks require that information transmitted between two stations be divided into blocks called frames. For these frames to be sent successfully to other devices on the network, certain protocol and routing information must be added to the data. In addition, the way this information is arranged inside the frame must conform to a specific format. The way an Ethernet device places the data bits into frames before it is placed on the LAN is called Manchester Encoding.*

*The following chapter discusses Manchester Encoding and describes the four different Ethernet frame types.*

---

## Manchester Encoding

The information that is to be transmitted on the cable is in the form of a constantly changing voltage signal. This signal is electronically transformed into a D. C. signal with a value of either 0 volts or -1.2 volts. The value of the D. C. signal is found by periodically sampling the voltage value of the original signal and assigning it one of the two values, depending on the value found at the time it was checked. The result is to change a non-constant analog signal into a digital signal with a value toggling between either 0 or -1.2 volts.

In Ethernet, the digital D.C. signal is transformed into discrete time segments called bits by a method called Manchester Encoding. With Manchester Encoding, the incoming digital signal is checked at specific time intervals for its change of state. In other words, the signal is checked to see if it is changing from 0 volts to -1.2 volts or from -1.2 volts to 0 volts during a certain time period. Depending on its change of state in this specific time interval, or bit time, the signal is assigned a logic “1” or a logic “0” for that time interval. The resulting signal is a steady stream of bit times (or bits) with values of either logic “1” corresponding to a change from -1.2 v to 0 v or a logic “0” corresponding to a change from 0 v to -1.2 v. If there is no change of state during a certain bit time, that bit time is assigned a value corresponding to the value of the bit time preceding it.

Manchester Encoding also provides the digital signal with a method of alignment by ensuring that the transitions to the logic levels happen only during the center of the bit time. Figure 4-1 shows how Manchester Encoding works on a random digital signal.

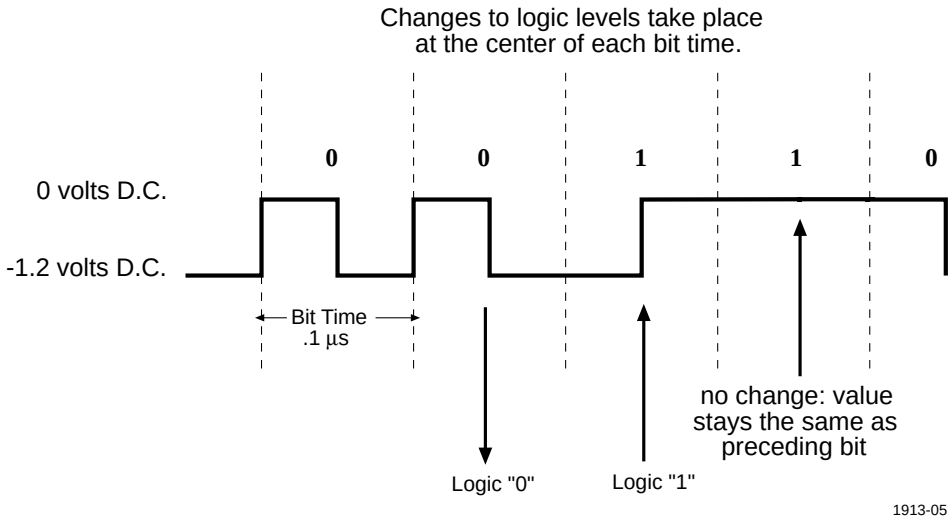



**Figure 4-1. Manchester Encoding**

# Ethernet Data Frames

Previously it was shown how Ethernet data signals are transformed into bits. Before these bits are sent onto an Ethernet network, they must be formatted into specific groups called data frames. Data frames are strings of bytes (eight bits equal one byte) which contain addressing, timing, protocol, and error correction information as well as the data being sent. The packet structure used in IEEE 802.3 and Ethernet is shown in Figure 4-2.

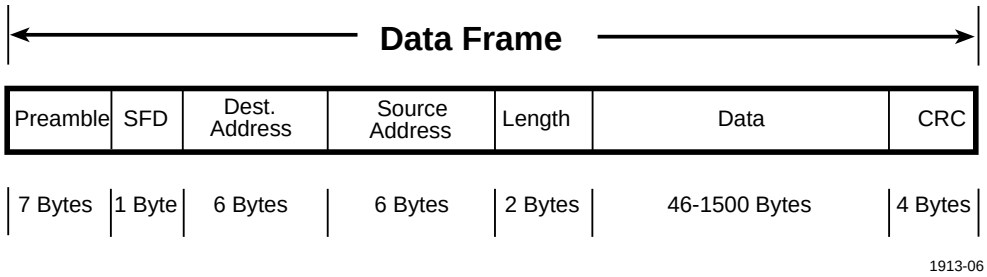



**Figure 4-2. Ethernet Data Frames**

Each section of the frame is described as follows:

- Preamble: The preamble indicates the beginning of frame transmission. The preamble allows for frame timing at the receiving station. The signal pattern is a repeating pattern of alternating ones and zeroes for a total of 56 bits (7 bytes).
- Start Frame Delimiter (SFD): The SFD signal pattern is 10101011 for a total of 8 bits (1 byte). It follows the preamble and indicates the start of information by the last two bits, 11.
- Destination Address: The address of the station, or stations, that the data frame is intended for. It follows the SFD and is 48 bits (6 bytes) in length.
- Source Address: Follows the destination address and indicates the address of the station initiating the transmission. The source address is 48 bits (6 bytes) in length.
- Length Field: The length field follows the source address and indicates the length of the data field. The length field is 16 bits (2 bytes) long. In Ethernet version 1.0 or version 2.0, this field is called a type field. The type field will usually indicate the packet protocol (e.g., TCP/IP, XNS, DECNet, Novell IPX, etc.).

- Data Field: The data field follows the length field. It is 46 bytes minimum to a maximum of 1500 bytes in length. This field contains the actual data being sent across the network along with some control information. If the data to be sent is less than the minimum 46-byte packet size, a special bit pattern called PAD is used to fill in up to the 46-byte minimum. The minimum packet size set by the IEEE 802.3 specification is explained below.
- Cyclic Redundancy Check (CRC): The CRC follows the data field and is 32 bits (4 bytes) in length. Also knows as the Frame Check Sequence (FCS), this field is used to check the integrity of the frame. Before placing a frame out on the wire, the sending station takes all the bytes within the frame, performs a mathematical calculation, and places the result at the end of the frame. When the frame arrives at the destination, the receiving station performs the same mathematical calculation and should receive the same result. If not, it assumes something has been corrupted and discards the frame.

## Data Frame Size

IEEE defines both a minimum and a maximum frame size. The minimum frame size is 64 bytes (12 address bytes, 2 length bytes, 46 data bytes and 4 CRC bytes). The maximum frame size is 1,518 bytes (same as above with 1,500 byte data field).

The minimum frame size has been determined to give the best bridge switch speed on heavily used networks by minimizing the amount of time a station must defer to other transmissions. It also increases the amount of overhead involved in completing a transmission. The minimum frame size will move quickly because of its size, but two full size frames move as much information as 66 minimum size frames containing only 46 bytes of data each. Therefore, the two large frames require only 36 bytes of overhead (Preamble, SFD, addresses, etc.) while the small frames require 1,188 bytes of overhead: nearly half the size of the original transmission. This also doesn't include the problems involved in trying to transmit 66 times in an operating network with collisions. The minimum frame size also plays an important role in the detection of collisions and determining the maximum network size.

In an Ethernet network, a station must still be transmitting its data to detect that it was involved in a collision. We know that a minimum size frame is 64 bytes in length which equates to 512 bits (8 bits per byte). We also know that each bit time in an Ethernet network is defined as 0.1 µs. Multiplying 512 bits by 0.1 µs yields 51.2 µs to transmit a 64-byte minimum size frame.

Due to inherent propagation delays in electronics and cabling it would make sense that within 25.6 µs (half of 51.2 µs) our transmitted signal should have reached the farthest point on the network. If a collision were to happen at the farthest point on the network the collision signal will have the remaining 25.6 µs to travel back to the transmitting node thus alerting the node that its transmission needs to be re-sent. The 25.6 µs one way propagation window is also called the collision domain.

## Data Frame Types

Ethernet data frames are packaged one of four ways:

- Ethernet II (DIX)
- 802.3 “raw”
- Ethernet 802.2
- Ethernet SNAP

Figure 4-3 shows each of the Ethernet frame types. The Ethernet 802.2 and Ethernet SNAP frames are extensions of the 802.3 “raw” frame format, while the Ethernet II frame is formatted differently. The following sections describe each frame type.

**Ethernet II Frame**

| Preamble | Destination Address | Source Address | Type | Data | FCS |
|---|---|---|---|---|---|
| 8 bytes | 6 bytes | 6 bytes | 2 bytes | 46 - 1,500 bytes | 4 bytes |

**802.3 "Raw" Frame**

| Preamble | Start Frame Delimiter | Destination Address | Source Address | Length | Data | FCS |
|---|---|---|---|---|---|---|
| 7 bytes | 1 byte | 6 bytes | 6 bytes | 2 bytes | 46 - 1,500 bytes | 4 bytes |

**Ethernet 802.2 Frame**

| Preamble | Start Frame Delimiter | Destination Address | Source Address | Length | DSAP | SSAP | Control | Data | FCS |
|---|---|---|---|---|---|---|---|---|---|
| 7 bytes | 1 byte | 6 bytes | 6 bytes | 2 bytes | 1 byte | 1 byte | 1 byte | 43 - 1,497 bytes | 4 bytes |

**Ethernet SNAP Frame**

| Preamble | Start Frame Delimiter | Destination Address | Source Address | Length | DSAP | SSAP | Control | Protocol Identifier | Data | FCS |
|---|---|---|---|---|---|---|---|---|---|---|
| 7 bytes | 1 byte | 6 bytes | 6 bytes | 2 bytes | 1 byte | 1 byte | 1 byte | 5 bytes | 37 - 1,492 bytes | 4 bytes |

1913-07

**Figure 4-3. Ethernet Frame Types**

## Ethernet II Frame Type

In the early days of computer networks, Digital, Intel, and Xerox got together and specified a networking standard that they called Ethernet. This standard included the definition of a data link level access method and a packet format that shared the Ethernet name (it is now called Ethernet II because it is in its second revision). Table 4-1 shows the fields defined for Ethernet II data frames.

**Table 4-1. Ethernet II Frame Type**

| Field Name | Field Size | Field Definition |
|---|---|---|
| Preamble | 8 bytes | Signals beginning of the packet. |
| Destination Address | 6 bytes | Contains address of the destination of the frame. |
| Source Address | 6 bytes | Contains address of the packet's origin. |
| Type Field | 2 bytes | Specifies the upper layer protocol used. |
| Data | 46–1500 bytes | Contains the data to be transferred. |
| Frame Check Sequence | 4 bytes | Verifies the integrity of the frame. |

The Ethernet II standard specified that a header (consisting of the Preamble, Source and Destination addresses, and frame type) be added to the data before sending it on the network medium. The frame format follows the rules to access a network using the CSMA/CD access method.

## Ethernet "Raw" Frame Type

Eventually, both the Ethernet media and packet format were pursued by the standards committees of the IEEE. Working from the original DIX specification, IEEE proposed its own Ethernet standard which they called 802.3 (named after the committee that worked on it). Table 4-2 describes each of the Ethernet 802.3 frame fields (also known as "raw" frames).

**Table 4-2. Ethernet "Raw" Frame Type**

| Field Name | Field Size | Field Definition |
|---|---|---|
| Preamble | 7 bytes | Signals beginning of the frame. |
| Start Frame Delimiter | 1 byte | Signals start of data. |
| Destination Address | 6 bytes | Contains the address of the destination of the frame. |
| Source Address | 6 bytes | Contains the address of the frame's origin. |
| Length Field | 2 bytes | Specifies the length of the data field. |
| Data | 46–1500 bytes | Contains the data to be transferred. |
| Frame Check Sequence | 4 bytes | Determines the integrity of the frame. |

The IEEE 802.3 frame format is almost identical to the Ethernet II format. The only difference is that a length field is used in place of the type field. This field indicates the length of the data portion of the 802.3 frame, with the maximum length being 1,518 decimal.

A network device can decipher the difference between an 802.3 "raw" frame and an Ethernet II frame by looking at this portion of the packet (the length or type field). As it turns out, the assigned values for the Ethernet II type field are always greater than 1,500 decimal. Since the maximum frame size for Ethernet is 1,518 bytes, with 18 bytes of overhead, the length field always contains a value less than that.

## Ethernet 802.2 Frame Type

Without a protocol type field, it is impossible to determine what protocol to use for interpreting the encapsulated data in an 802.3 "raw" frame. If more than one upper-layer protocol exists on the network, the packet may be incorrectly routed. Therefore, sometime after the 802.3 standard was released, IEEE came out with the 802.2 standard. Table 4-3 describes each of the Ethernet 802.2 frame fields.

**Table 4-3. Ethernet 802.2 Frame Type**

| Field Name | Field Size | Field Definition |
|---|---|---|
| Preamble | 7 bytes | Signals beginning of the frame. |
| Start Frame Delimiter | 1 byte | Signals start of data. |
| Destination Address | 6 bytes | Contains address of the destination frame. |
| Source Address | 6 bytes | Contains address of the frame's origin. |
| Length Field | 2 bytes | Indicates length of the Data plus LLC fields. |
| Destination Service Access Point (DSAP) | 1 byte | Shows first byte of 2 byte value indicating the frame's upper layer protocol destination. |
| (Source Service Access Point) SSAP | 1 byte | Shows second byte of 2 byte value indicating the frame's upper layer protocol destination. |
| Control | 1 byte | Indicates the type of LLC frame. |
| Data | 43–1,497 bytes | Contains the data to be transferred. |
| Frame Check Sequence | 4 bytes | Determines the integrity of the frame. |

The Ethernet 802.2 header envelopes the data before it is encapsulated within an IEEE 802.3 header. This frame adds several fields to the header; a destination service access point (DSAP), a source service access point (SSAP), and a control field.

- Service Access Point: This denotes the point of service the packet is intended for, or what upper layer protocol is supposed to use the data. Both the DASP and the SSAP fields contain values that identify the upper layer protocol type of the frame.
- Control Field: This is used by certain protocols for administrative purposes.

## Ethernet SNAP Frame Type

After the 802.2 frame was defined, there was some concern that the one byte DSAP and SSAP fields were not adequate for the number of protocols that eventually needed to be identified. In response from Apple Computer and the TCP/IP community, another frame standard was defined for both Ethernet and Token Ring. It was called the Sub-Network Access Protocol, shown in Table 4-4 below.

**Table 4-4. Ethernet SNAP Frame Type**

| Field Name | Field Size | Field Definition |
|---|---|---|
| Preamble | 7 bytes | Signals beginning of the frame. |
| Start Frame Delimiter | 1 byte | Signals start of data. |
| Destination Address | 6 bytes | Contains the address of the destination of the frame. |
| Source Address | 6 bytes | Contains the address of the frame's origin. |
| Length Field | 2 bytes | Contains the length of the Data plus LLC fields. |
| Destination Service Access Point (DSAP) | 1 byte | Set to AA (hex) and 10101010 (binary). |
| Source Service Access Point (SSAP) | 1 byte | Set to AA (hex) and 10101010 (binary). |
| Control Field | 1 byte | Set to 03 (hex) and 00000011 (binary). |
| Protocol Identifier | 5 bytes | Specifies the upper layer protocol. |
| Data | 38– 1,492 bytes | Contains the data to be transferred. |
| Frame Check Sequence | 4 bytes | Determines integrity of the frame. |

This frame type adds a five-byte protocol identification field at the end of the 802.2 header, where the protocol is identified. To distinguish an IEEE 802.2 SNAP frame, the value of the DSAP and SSAP fields in the 802.2 header are both set to AA. If a network device finds AA in the DSAP and SSAP fields, it knows this is a SNAP-based frame and it should look for the protocol type in the protocol identification field.

## Ethernet Addressing Schemes

There are three types of addressing schemes used in Ethernet networks. Each address type serves a different purpose. They are as follows:

1. Specific Addressing
2. Multicast Addressing
3. Broadcast Addressing

### Specific Addressing

The IEEE specifies that each addressable network device will have a unique hardware address that is made up of 6 bytes of information. The address is either hard-coded into every network interface controller card during manufacturing or assigned out of a group of addresses given to a particular corporation. The availability of addresses is strictly controlled by the IEEE.

The IEEE assigns each network hardware manufacturer a unique manufacturer identifier and a block of numbers that the manufacturer usually assigns sequentially to each piece of hardware. The combination of the manufacturer ID and the sequential number makes up the common 48-bit Ethernet address.

The first 3 bytes of the address contains the manufacturer identifier and the last 3 bytes contain the sequential numbering. The numbering scheme is given in hexadecimal format. An example of a typical Ethernet address is shown in Figure 4-4.

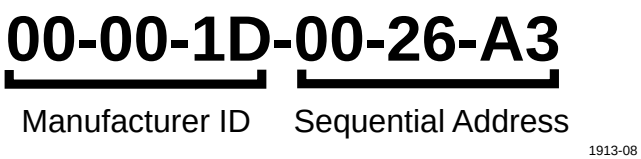



**Figure 4-4. A Typical Ethernet Address**

The 00-00-1D manufacturer ID belongs to Cabletron Systems. When a specific Ethernet address is used as the destination address in a packet, that packet will be decoded only by the station with that specific address.

### Multicast Addressing

At times it is necessary to communicate with many devices on a network simultaneously. For instance, a network management station might poll or query a group of devices to determine their status. Instead of keeping a list or table of unique addresses a group address can be used. This type of group addressing is accomplished using a multicast address.

A multicast address is formed by modifying the manufacturer ID portion of the destination address. As shown in Figure 4-5, the least significant bit of the first byte is changed from a 0 to a 1. The net result is to turn a Cabletron Systems address of 00-00-1D... into a multicast address of 01-00-1D... A multicast address is destined for devices from a particular manufacturer.

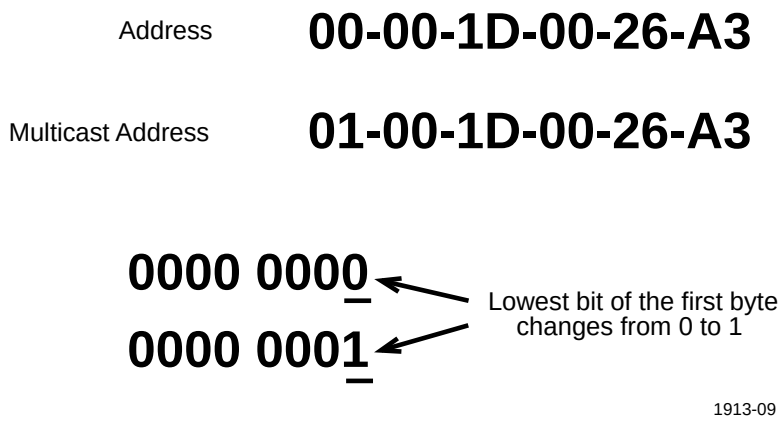



**Figure 4-5. Ethernet Multicast Address**

### Broadcast Addressing

A broadcast address is an address that is meant to be heard by all stations on the network. Certain protocols running on workstations will occasionally send out broadcast messages to servers on the network to let the servers know that the node is on-line.

A broadcast address contains all “F” hexadecimal characters which is equivalent to all bits being set to logic 1 in both the manufacturer ID and sequential number area of the address (see Figure 4-6).


**Figure 4-6. Ethernet Broadcast Address**

In the following chapter we will look at how the packet is transmitted onto the network and the rules that must be followed to ensure a successful transmission.

Chapter 5

# Ethernet Media Access Method

*Ethernet, as stated in Chapter 1, uses a method of access control known as Carrier Sense Multiple Access with Collision Detection, or CSMA/CD. Access to the network media is controlled by the lower half of the Data Link Layer called Media Access Control, or MAC. The following chapter describes the operation of the Ethernet MAC.*

## Clean Frame Transmission

The following determines the process that an Ethernet station goes through in a clean frame transmission over the network.

A station wanting to transmit first listens to the communication channel to see if any other station is transmitting. If a carrier is sensed (another station is transmitting), the station waits a random length of time, and then listens to the communication channel again. If no other station is transmitting, the station begins frame transmission.

During frame transmission, the station continuously monitors the bus. As long as the monitoring shows there is no other station transmitting on the cable, the frame transmission continues until the transmission is complete.

Once the frame transmission is complete, the station is quiet for 9.6 µs to allow for the required interframe gap. After 0.6 µs into the interframe gap, the transceiver is given a 1.4 µs window to test its collision detect circuitry (see Figure 5-1).

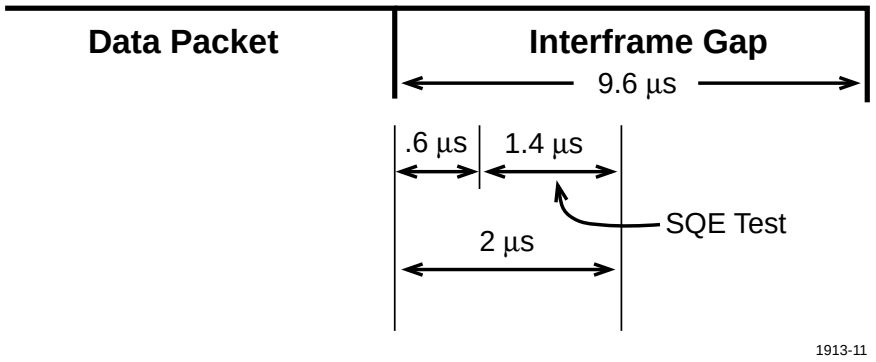



**Figure 5-1. Ethernet Interframe Spacing**

During this time, the station will see the Signal Quality Error (SQE) test signal on its collision detection. When the station sees this signal during the 1.4 µs window, it is informed that the transceiver collision detect circuits are working properly and if a collision occurred, it would be notified.

After the remainder of the 9.6 µs interframe gap is complete, the station is able to start the transmission process again by first listening to see if the cable is available.

### Signal Quality Error Test

The SQE (Heartbeat) is only generated by a transceiver and is only seen by the host device that is connected to that specific transceiver. The SQE does not appear on the network bus. It is a signal from a transceiver to its associated station simply to inform the station that the transceiver's collision detection is working properly.

# Packet Involved in a Collision

Because of propagation delays in the network, it is possible for two stations to simultaneously find the bus available, in which case both stations will begin transmitting frames. When these signals meet on the cable, a collision occurs. These two signal voltages add together and increase the voltage level on the cable, which is sensed by the transmitting transceivers. The transceiver then sends a collision signal to the host station, all while it is still transmitting the packet. If the station is not still transmitting the packet, there is a problem in the design of the network, and it does not meet IEEE 802.3 specifications.

When a collision is detected, both stations will transmit a jam signal that is long enough to ensure that the collision is detected by all stations on the network. Then, each station involved in the collision will wait for a random period of time and then attempt transmission again. The station will attempt again transmission up to 16 consecutive times before an error is sent to the upper layer protocols notifying the station of a serious communication problem.

## Collision Detection on Point-to-Point Media

On Point-to-Point media, it is not possible to detect a collision by listening to the transmission as with multi-point media. Point-to-point media transceivers use a method called Transmit Mode Collision Detection. With this method, the transceivers will monitor their receive ports while transmitting. If they receive a signal while transmitting, then a collision has occurred.

## Out-Of-Window Collision

As mentioned in Chapter 3, from any station on the network, a transmitted frame has 25.6 µs to get to the end of the collision domain. If a collision were to happen at the farthest point from the transmitting station (25.6 µs away), the collision signal will take an additional 25.6 µs to propagate back to the transmitting station for a total of 51.2 µs round trip time (or the time it takes to transmit a minimum size frame). If a station is able to transmit for 51.2 µs without detecting a collision, the station should have acquired the communications channel and its signal should be the only one using the network. If a collision is detected after the station has transmitted the required minimum frame size, an Out-Of-Window (OOW) collision has occurred. In other words, the station has transmitted for 51.2 µs without a collision but senses a collision after 51.2µs has passed. Out- Of-Window collisions indicate abnormal network operation. They are usually caused by the network being too long where the round trip propagation delay is greater than 51.2 µs, a station somewhere on the network is transmitting at will, or a cable somewhere on the network failed during the transmission of the frame.

In the following chapters we will look at the importance of propagation behavior. Propagation delay is discussed in more detail in Chapter 8, **Propagation Delay**.

Chapter 6

# Ethernet Devices

*This chapter describes devices that are common to an Ethernet network. All devices attached to an Ethernet bus must comply with the IEEE 802.3 Standard. Typical Ethernet devices include stations, transceivers, repeaters, bridges and routers. Figure 6-1 shows various devices attached to an Ethernet LAN. The following sections provide a description of each of these devices and their network functions.*

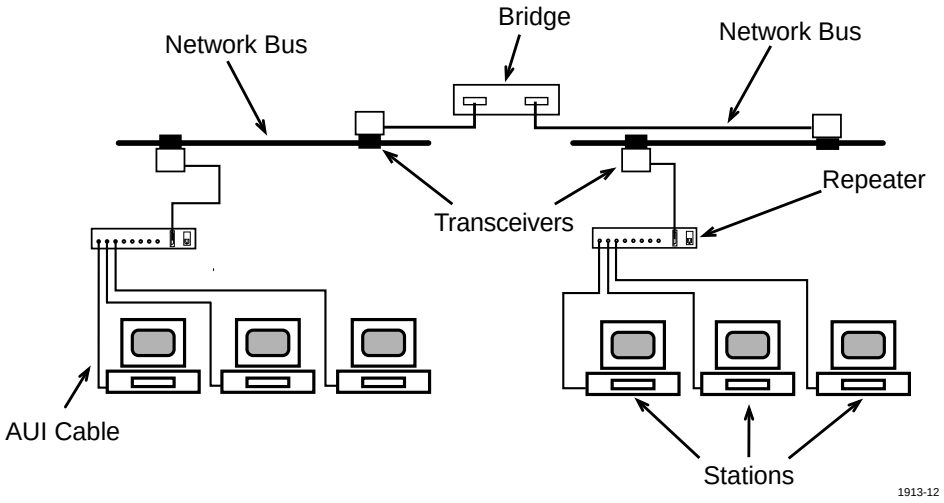



**Figure 6-1. Devices on an Ethernet LAN**

## Ethernet Stations

Ethernet stations are addressable nodes on an Ethernet network capable of transmitting, receiving, and repeating information. Workstations, file servers, and printers (shown in Figure 6-2) are some examples of these types of devices. Stations connect to the Ethernet bus through devices called transceivers and are discussed in the next section.

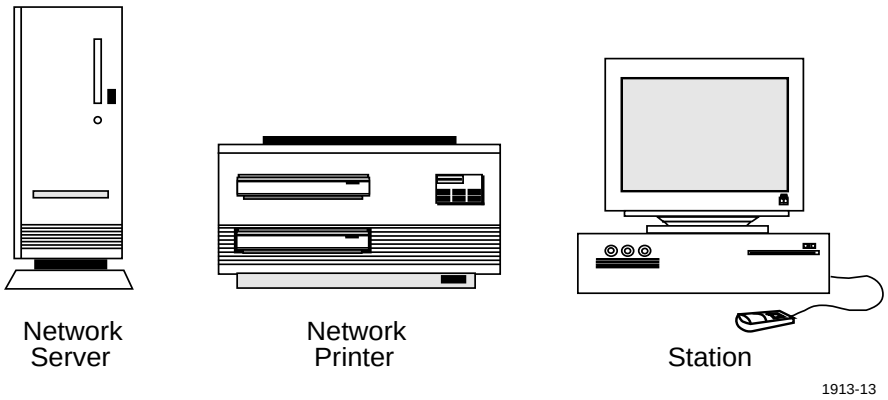



**Figure 6-2. Ethernet Stations**

# Ethernet Transceivers

A transceiver (transmitter/receiver) is the device that connects workstations, servers, and other equipment to the Ethernet cabling media being used for network transmissions. For full descriptions of Ethernet networking media, refer to the *Cabletron Systems Cabling Guide.* The transceiver (Figure 6-3) is responsible for listening to the Ethernet bus to determine if the bus is currently in use by another station. If a collision occurs during transmission, the transceiver is responsible for alerting its connected device by sending a collision signal down the AUI cable.

IEEE 802.3 compliant transceivers employ special circuitry to perform a watchdog function on the transceiver transmitter. If a workstation starts to continuously transmit data, the transmitter shuts down so the network is not taken over by one device.

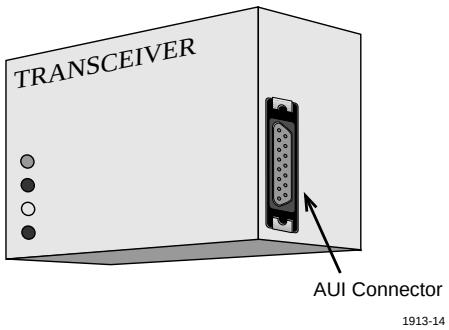



**Figure 6-3. Standard Ethernet Transceiver**

## Multi-port Transceivers

A multi-port transceiver or fanout is a transceiver that has one port to connect to a regular transceiver and up to fifteen AUI ports to connect through AUI cables to individual devices. This allows you to connect several addressable devices to one cable tap. IEEE 802.3 standards for transceiver placement and tap spacing specify that only 100 taps will be allowed on a 10BASE5 segment with a distance of 2.5 m between them. If it becomes necessary to concentrate a number of workstations in one physical location, a multi-port transceiver can be used.

Figure 6-4 shows a typical multi-port transceiver.

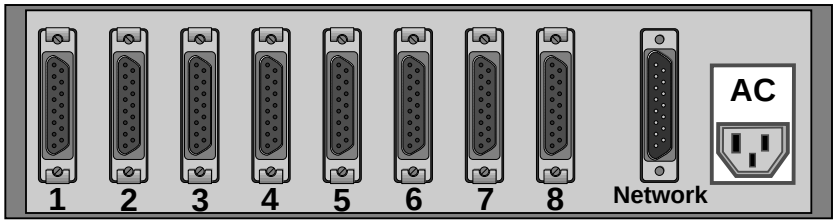



**Figure 6-4. Standard Ethernet Multi-port Transceiver**

# Ethernet Repeaters

If it is necessary to add additional taps beyond the 100 tap limit, another coaxial segment must be added. If the new segment uses the same architecture at the Physical Layer of the OSI model as the old segment, a repeater can be used to join the two segments. A repeater regenerates the preamble, and amplifies and retimes the signal from one cable segment to the other.

Figure 6-5 shows a Cabletron Systems LR2000 two-port local Ethernet repeater.


**Figure 6-5. Cabletron Systems Two-Port Local Ethernet Repeater**

## Repeaters and Collisions

A collision happens when more than one station transmits on the network at one time. Since the repeater physically separates the two coaxial segments, a collision on one segment cannot be seen by devices on the other segment connected to the repeater. Therefore, the repeater is responsible for ensuring that the collision signal is propagated to all segments attached to it. To force a collision, the repeater sends out a special bit pattern called a Jam signal to all segments attached to it. This signal notifies all stations on the network that a collision has occurred.

## Auto Partition

When the repeater detects 32 consecutive collisions on one port it will logically turn off or segment the port that it detected the problem on, thus allowing the rest of the network to function properly. When the repeater detects a collision on the segmented port, the collision will not be forwarded to the other segments of the network, leaving the port in segmented condition. When the repeater receives a packet on a good port, it attempts to transmit the packet to the segmented port. If the packet transmits successfully, the repeater will turn the segmented port back on, bringing it out of segmentation.

## Multi-port Repeaters

A multi-port repeater is a device which has more than two ports that connect to full-length Ethernet Segments. Generally the repeaters are very similar in appearance to the two-port local repeater shown in Figure 6-5, with the exception of the number of ports. These repeaters regenerate the preamble and amplify and re-time a signal from one cable segment to the others.

## Inter-Repeater Links (IRLs)

Up to 3 repeaters can be used to connect separate coaxial segments before special considerations need to be taken into account. If the segments are physically located a distance from each other, the use of an Inter-Repeater Link (IRL) comes into play. As shown in Figure 6-6, an IRL is a segment that connects only two repeaters. It can be made up of thin coaxial cable (185 m), thick coaxial cable (500 m), twisted pair cable (200 m), or Fiber optic cable (2/5 km). With the use of IRLs it is possible to build a network consisting of up to three populated segments with two IRLs and four repeaters joining them.

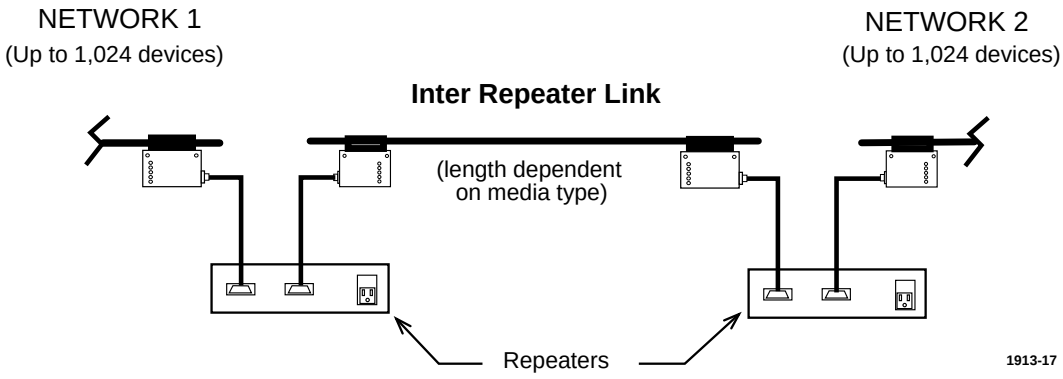



**Figure 6-6. Using an IRL**

# Ethernet Bridges

A bridge is a device that can be added to a network to allow expansion beyond the limitations of 802.3. If a network has a repeater hop of four repeaters or a propagation delay near the 51.2 µs maximum, a bridge can be used to accomplish the addition of the new full specification Ethernet.

Unlike a repeater, which sends all frames it receives to all segments it is connected to, a bridge reads the frames it receives and decides whether to filter or forward the frame based on the addressing information contained within it. Bridges can also be used to connect similar networks (networks with the same upper five layers of the OSI model) such as Ethernet, Token Ring, and Fiber Distributed Data Interface (FDDI) together. Because bridges work at layer 2 of the OSI Model, they are protocol independent. They have a longer response time than repeaters because a bridge must read the complete data frame, check for errors, and make forward or filter decisions based on recognized addresses stored in its source address table.

Bridges are discussed further in Chapter 9, **Ethernet Bridge Operation**.

# Routers

A router works much like a bridge, except that a router pays attention to the upper network layer protocols (OSI model level 3) rather than just Physical Layer protocols like a bridge. A router will decide whether to forward a packet by looking at the protocol level addresses rather than the MAC address. Because routers transfer packets between different media types, many routers can also function as bridges.

A detailed discussion of routing operations is beyond the scope of this guide.

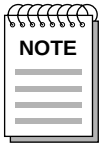
The following chapter explains the use of each of these devices while considering your Ethernet network design.

Chapter 7

# Ethernet Network Design

*Designing an Ethernet network must be approached with care. Many aspects of the network must be considered before actual implementation can begin. A detailed plan must be laid out to ensure that all the goals and obstacles are identified. In this chapter we will discuss many aspects of network design by beginning with a single segment 10BASE5 Ethernet network and gradually building the single segment into a large, multiple segment network. We will then discuss the design of a 10BASE2 network, a Fiber Optic network, and finally a 10BASE-T network.*

## 10BASE5 Ethernet Network Design




The characteristics and test requirements of Ethernet 10BASE5 cables are presented in the *Cabletron Systems Cabling Guide*.

### Single Segment 10BASE5 Ethernet Network

A single segment of thick Ethernet cable can be a maximum of 500 m in length. The cable must be terminated at both ends with N-type connectors and 50 ohm N-type terminators. The cable should be marked with annular rings by the cable manufacturer every 2.5 m indicating potential transceiver tap points (see Figure 7-1).

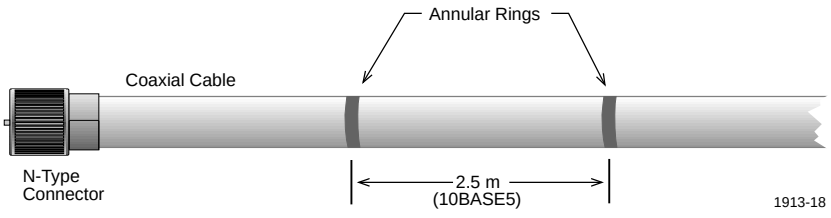



**Figure 7-1. Coaxial Annular Rings**

The coaxial cable can be run in one continuous length or in sections, joined using N-type connectors and N-type barrel connectors. If the cable is installed in segments and connected together, IEEE recommends that the segments should be an odd multiple of 23.4 m in length. These special lengths of cable are used to minimize signal reflections caused by the insertion of connectors and barrel splices. The segment lengths are optimum lengths and it is sometimes difficult to adhere to these specifications in the real world. If it is not possible to adhere to the above cable lengths, attempt to connect the cable segments at annular rings.

To further reduce signal reflections, the cable segments should come from one manufacturer to minimize cable discontinuity problems such as differences in propagation speed or impedance.

### Transceiver Placement

The terminated coaxial cable is now ready for the placement of the transceivers. Transceiver taps can be made into the coaxial backbone by one of two ways:

1. Intrusive: By cutting the coaxial cable and installing connectors that screw into the transceiver. This method of tapping is known as an intrusive tap because the coaxial cable must be severed for the transceiver to be connected.

2. Non-intrusive: By drilling a hole into the coaxial cable with a special tool and clamping the transceiver assembly in place. This method is known as a non-intrusive tap because the coring process, when done properly, allows the tap to be made without interruption of network traffic on the coaxial segment (see Figure 7-2).

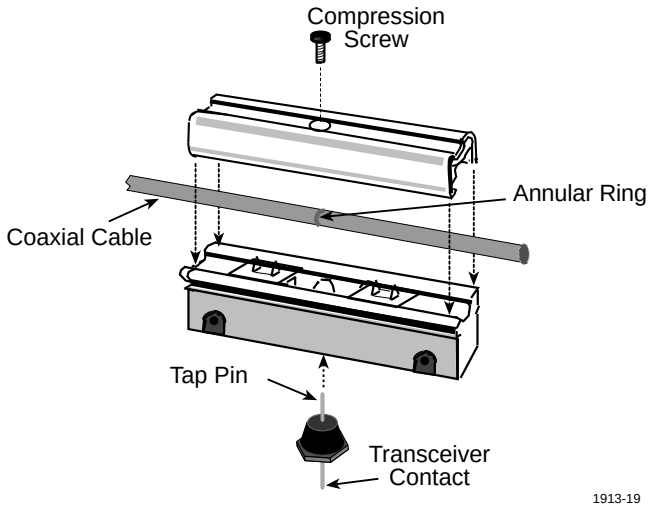



**Figure 7-2. A Non-intrusive Coaxial Cable Tap**

A maximum length 10BASE5 coaxial cable has 199 annular rings marked off at 2.5 m intervals. Because each transceiver tap introduces noise onto the coaxial cable in the form of a small impedance discontinuity, and contributes to the overall attenuation of the cable, IEEE has specified that only 100 taps will be allowed on a 10BASE5 segment, with each tap separated by a minimum of 2.5 m.

Once the coaxial taps are in place, they can be attached directly to the transceivers (see Figure 7-3). An Attachment Unit Interface (AUI) cable is used to connect devices such as workstations or file servers to the transceivers. The AUI cable, which can be a maximum of 50 m in length, is made up of four shielded twisted pairs that carry the transmit, receive and collision signals between the transceiver and its connected equipment.

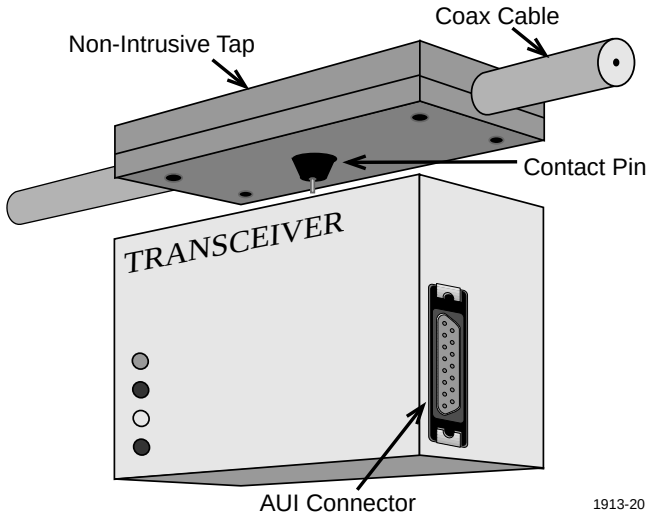



**Figure 7-3. Transceiver Attachment**

### Multi-port Transceivers

As you recall from the previous Chapter 6, **Ethernet Network Design**, a multi-port transceiver is a device that is used to connect several network devices at a single tap point. This is useful when it is necessary to concentrate a number of devices at one physical location or to add more than 100 devices on a single coaxial backbone. A typical multi-port transceiver has eight AUI ports for connection to workstations and one AUI port for connection to the backbone segment.

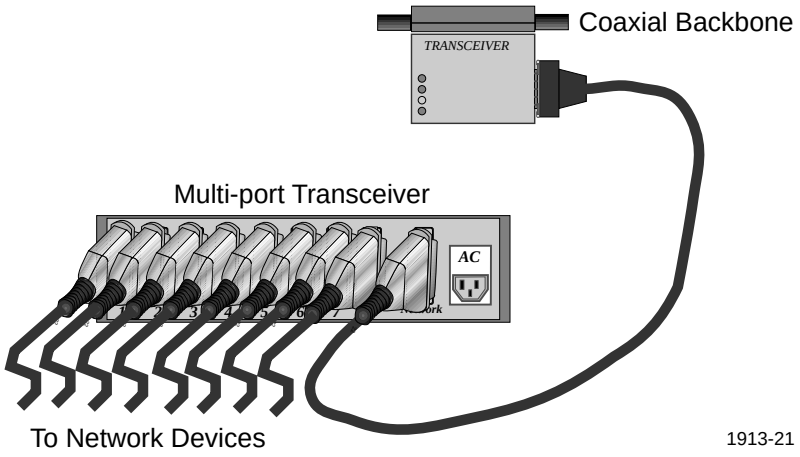



**Figure 7-4. Multi-port Transceiver**

### Multi-port Transceiver Rules

When multi-port transceivers are used, two rules must be observed:

1. Multi-port transceivers can be cascaded by connecting the male connector of one multi-port transceiver to the female connector of a second multi-port transceiver. However, cascading more than two multi-port transceivers will result in unacceptable amounts of jitter, causing alignment errors on the network to occur.

2. The maximum length of a standard AUI cable is 50 m. This maximum length is reduced by 10 m for each multi-port transceiver that the signal must pass through. In other words, if two multi-port transceivers are cascaded, only 30 m total of AUI cable between them, the device, and the coaxial tap, is allowed.

You can now see how it would be possible to construct a 10BASE5 Ethernet network that would support the IEEE maximum of 1,024 devices. By using cascaded multi-port transceivers it is possible to have up to a maximum of 64 user devices per coaxial tap point. See Figure 7-5.

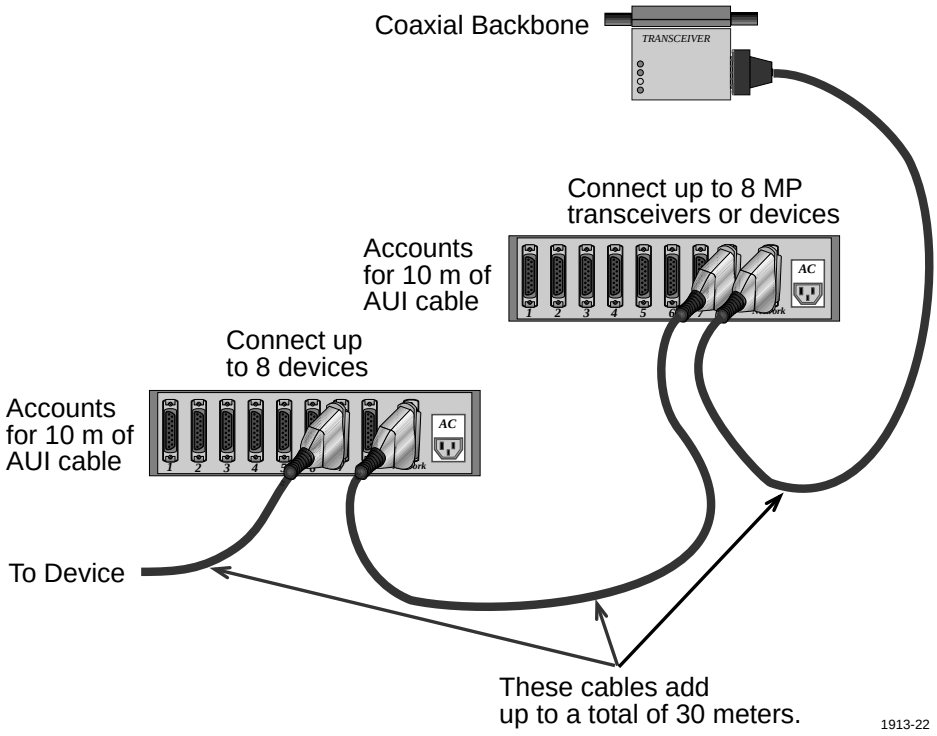



**Figure 7-5. Cascaded Multi-port Transceivers**

### Grounding and Insulation

The only issue that remains before we have a functioning single segment 10BASE5 Ethernet network is grounding the coaxial cable and insulating the connectors. The backbone cable must be connected to a reliable earth ground at only one point. The actual connection to ground can be made at any point on the cable, but is usually accomplished at an N-type connector, which allows for convenient attachment to the coaxial shield through the connector body.

All connections, other than the grounded connection, must be insulated from any other metallic surface to avoid inadvertent grounding and creation of ground loops (multiple paths to ground).

Once the connectors are insulated, the coaxial cable is grounded and the workstations are connected to the transceivers, the single segment network is ready for operation.

## Multiple Segment 10BASE5 Ethernet Network

We have seen how we can build a single segment 10BASE5 Ethernet network. This is adequate if we only want to span a distance of 500 m. If it is necessary to cover a greater area or to add additional coaxial taps beyond the 100 tap limit, more coaxial cable must be added. To connect the new coaxial segment to the existing backbone, a repeater must be used.

As you recall from Chapter 6, **Ethernet Devices**, the repeater is a Physical Layer device that has the capability to forward frames at up to full Ethernet bandwidth. It regenerates the preamble and amplifies and retimes the signal from one backbone cable to another. It is connected to the coaxial cable through transceivers in the same fashion as any other node on the network and requires external power to operate. The repeater also extends fragmented frames and will auto-partition ports in the event of excess collisions.

### Repeater Use

When a local repeater is used to connect two 10BASE5 coaxial segments, each segment may be a maximum of 500 m in length with up to 100 taps, including taps made to connect the repeater. The maximum of 1,024 devices allowed and the AUI cable length limitation of 50 m apply to both segments. It is not necessary that repeaters be connected at the ends of the coaxial segment. Refer to Figure 7-6.

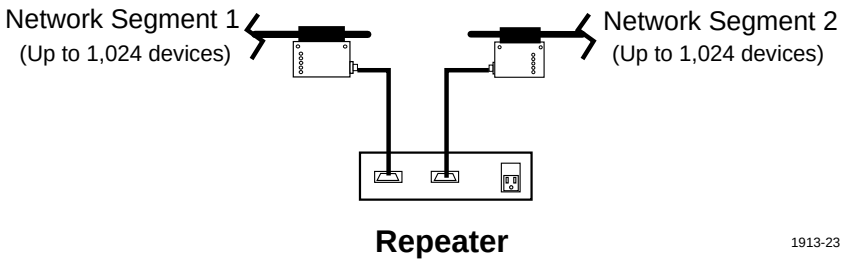



**Figure 7-6. Repeater Use**

### Inter-Repeater Link (IRL)

Repeaters can be used to connect up to 3 coaxial segments before special considerations need to be taken into account. If it is necessary to connect more than 3 coaxial segments together, then an Inter-Repeater Link (IRL) must be used (see Figure 7-7). An IRL is a segment that spans between two repeaters with no other devices attached to it. It can be made up of thin coaxial (185 m), thick coaxial (500 m), fiber optic cable (2/5 km) or twisted pair cable (200 m) with the appropriate media limitations being observed.

IEEE states that if IRLs are used, the maximum network size can be up to 4 repeaters with 5 segments, giving a total linear distance of 2,500 m (if just 10BASE5 coaxial is used), not including AUI cables. IEEE has determined that this is the maximum network size that will allow the round trip propagation delay budget to be met. Propagation delay is discussed in Chapter 8, **Propagation Delay**.

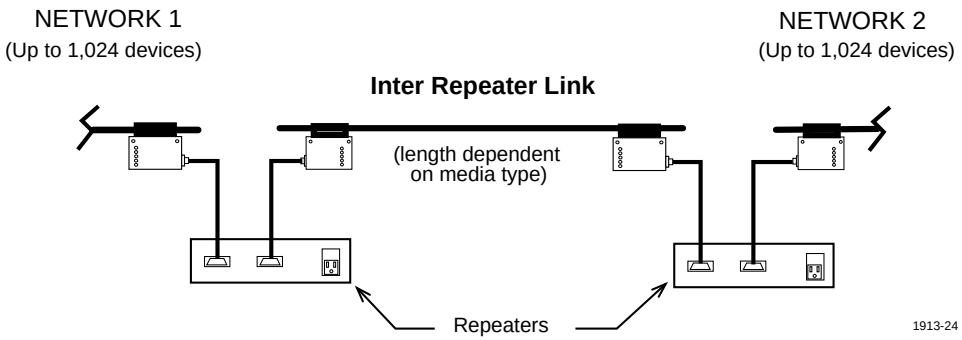
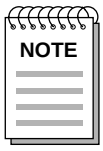



**Figure 7-7. Using Inter-Repeater Links**




In multiple segment networks, each coaxial segment must be grounded at only one point. All exposed metallic connectors and terminators must be insulated from ground to prevent ground loops.

# 10BASE2 Ethernet Network Design

NOTE

The characteristics and test requirements of 10BASE2 cables are presented in the *Cabletron Cabling Guide*.

## Single Segment 10BASE2 Ethernet Network

10BASE2 is the IEEE specification for Ethernet running on RG58 A/U coaxial cable. 10BASE2 coaxial cable is more flexible and less expensive than 10BASE5 coaxial cable while still maintaining the required 50 ohm nominal impedance. The maximum length of a 10BASE2 cable segment is 185 m. The reduced distance is due to the higher loss characteristics of the RG58 A/U compared with 10BASE5 coaxial cable. Only 30 taps are allowed on a 10BASE2 segment and the cable is connected using BNC connectors and terminated using 50 ohm BNC terminators.

Tapping into a 10BASE2 cable is accomplished by cutting the cable and attaching a BNC T-connector (see Figure 7-8). One end of the T-connector is either attached to the network interface card in the workstation or to a wall plate and the other two ends are attached to the coaxial cable backbone. An AUI cable is used to go from the wall plate to the device or workstation. The minimum spacing between T-connectors or splices is 0.5 m.

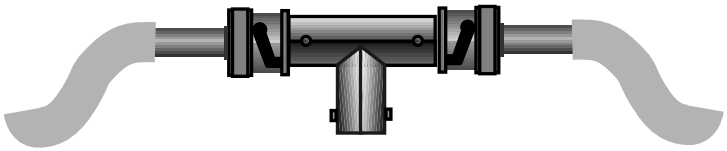
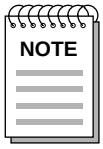



**Figure 7-8. Typical BNC T-Connector**

### Workstation Connections

Workstations may be connected to the BNC T-connector in one of the following two ways:

1. By connecting a transceiver with a BNC connection directly to the T-connector then running up to 50 m of standard AUI cable to the workstation.
2. By connecting the T-connector to the internal transceiver that is built into most network interface cards on the device itself. This is the most popular way to connect a device to the coaxial backbone in a 10BASE2 network.




An AUI cable is used to go between the transceiver and the workstation, not between the tap and the transceiver. If the internal transceiver on the network interface card is used in a 10BASE2 network, the use of an AUI cable is not allowed.

### Grounding and Insulation

As with 10BASE5 networks, the 10BASE2 segment must be grounded at only one point and all remaining connectors must be insulated from contact with ground.

## Multiple Segment 10BASE2 Ethernet Network

Since 10BASE2 has a physical limitation of only 30 devices and 185 meters of coaxial cable, it is easy to see how a network might quickly grow beyond specifications. As with 10BASE5 Ethernet, 10BASE2 can also be expanded using repeaters. Although a standard 10BASE5 repeater and BNC style transceivers may be used, the more common approach is to use a multi-port repeater such as a Cabletron Systems MR9000C. The MR9000C has eight BNC connections to allow the connection of up to 8 full specification 10BASE2 segments. A multi-port repeater counts as only one repeater when calculating your maximum repeater path. Many also have an AUI port to allow for connection to a standard transceiver on a 10BASE5 network.

By using repeaters, multi-port repeaters, or a combination of both, it is possible to design a 10BASE2 network that has 4 multi-port repeaters and five interconnected segments that span a distance of 5x185 m = 925 m, with a maximum of 1,024 connected devices. As long as the longest possible signal path does not pass through more than four repeaters and five segments, with only three of the segments being populated with devices, the network is within IEEE 802.3 specifications (see Figure 7-9).

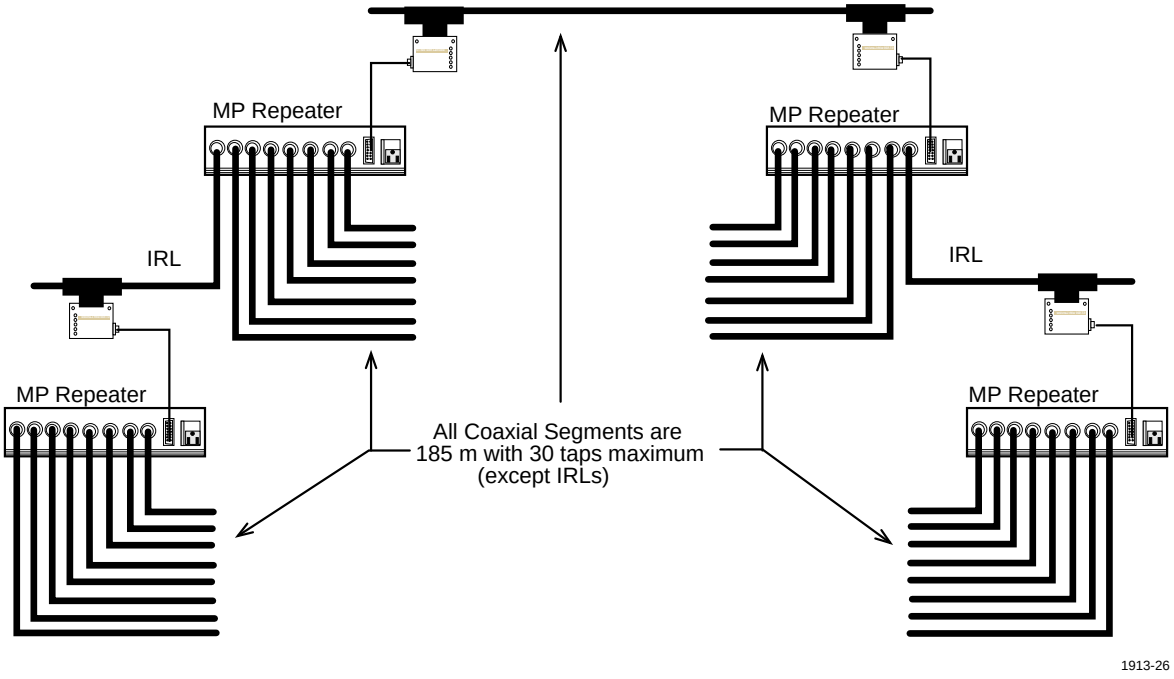



**Figure 7-9. Maximum Size 10BASE2 Ethernet Network**

### Grounding and Insulation

When cascading multi-port repeaters be careful to avoid the creation of ground loops (or multiple paths to ground). If two multi-port repeaters that perform internal grounding are connected using the BNC ports, a ground loop will result (see Figure 7-10). In other words, if one repeater is connected using a BNC port, the other must be connected using the AUI port.

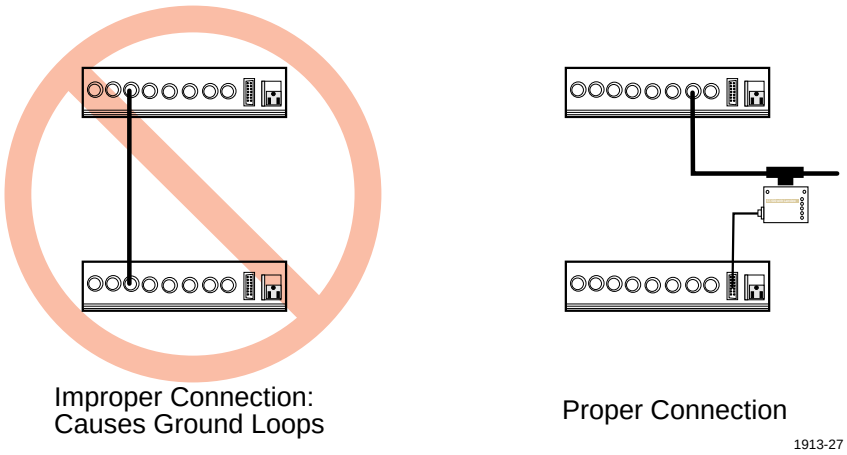



**Figure 7-10. Multi-port Repeater Connection**

# Fiber Optic Ethernet Network Design

Communication over fiber optics is done with pulses of light transmitted on glass instead of pulses of electricity transmitted on copper. Because of the low loss and high noise immunity of fiber, it is the media of choice when designing an extended distance LAN. A fiber optic segment can be up to 5 km in length and is used as a point-to-point media where no taps or branches are allowed.

Fiber optics is typically used in Ethernet networks to connect repeaters between buildings and to traverse long distances as an IRL. It can also be used with transceivers or fiber optic network interface cards to connect network nodes.

A completely fiber optic network may be built as long as the 4 repeater, 5 segment and 1,024 devices rules are not broken. To connect multiple fiber optic cable runs, a fiber optic multi-port repeater is used. Fiber is used as an IRL to span between repeaters and multiple fiber drops to workstations can be implemented (see Figure 7-11).

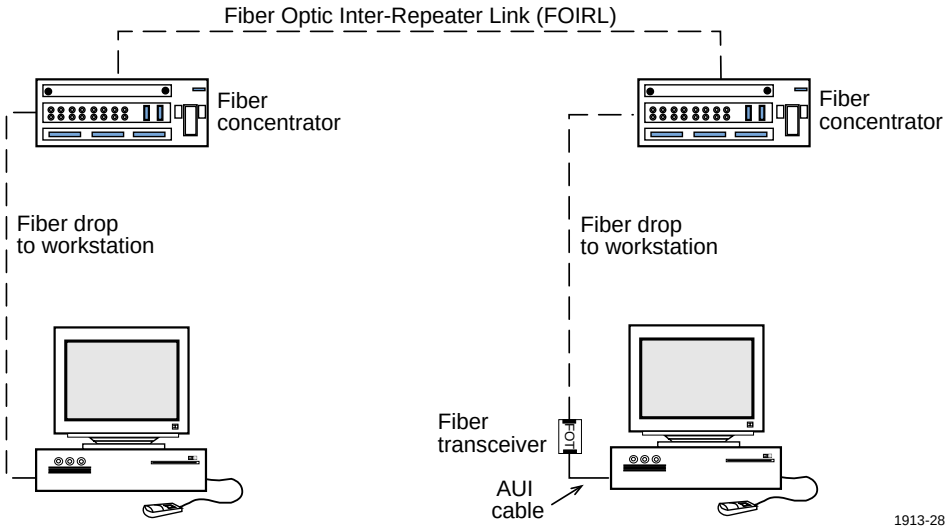



**Figure 7-11. A Complete Fiber Optic Ethernet**

# 10BASE-T Twisted Pair Ethernet Network Design

Unshielded Twisted Pair (UTP) wiring is found in most business environments. For this reason, IEEE adopted a set of standards for implementing this cost effective wiring into its own Ethernet category.

Designing a 10BASE-T Ethernet is much like designing a Fiber Optic network, as both media are point-to-point media. A 10BASE-T network is configured in such a way that the resulting topology is a star. A multi-port repeater is used in a central location, such as a wiring closet, with twisted pair segments going to the workstation locations. Each segment is a maximum of 200 m in length, with no other taps or branches allowed. As with all un-bridged Ethernet networks, the maximum signal path is 4 repeater hops and 5 segments. Refer to Figure 7-12.

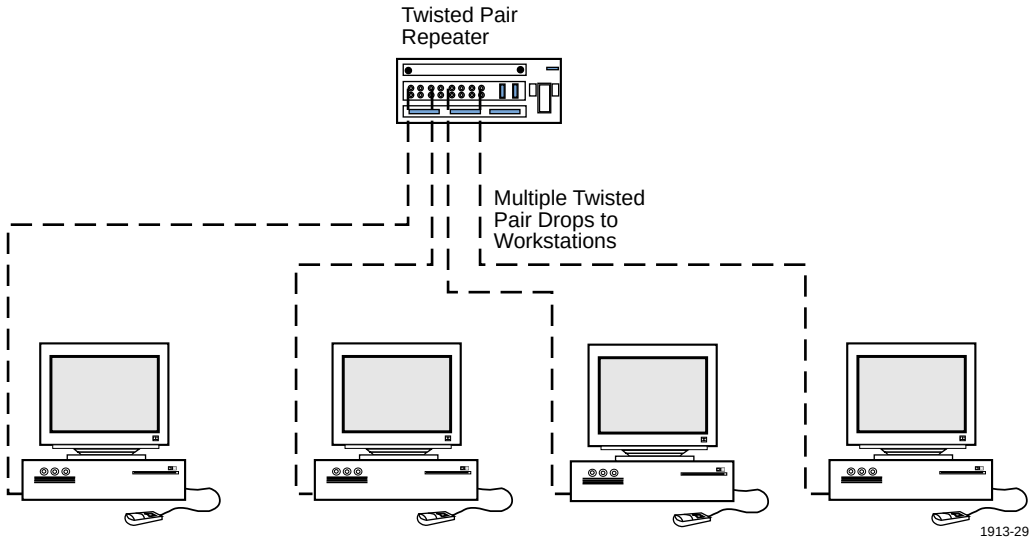



**Figure 7-12. 10BASE-T Ethernet**

Chapter 8

# Propagation Delay

*From the preceding, Chapter 7,* **Ethernet Network Design***, we have seen how it is possible to build a maximum size network using each of the available media. When designing any Ethernet network, it is wise to calculate the maximum round trip propagation delay for the proposed design. The maximum allowable round-trip propagation delay of 51.2 μs, governed by the minimum frame size of 64 bytes, is a very important consideration when it comes to accurate collision detection and data integrity.*

*Everything that lies in the signal path will contribute to the overall propagation delay. The items that add delay are transceivers, repeaters, active hubs, passive hubs and cables. Bridges, which effectively reset propagation delay, are not considered in the calculation of propagation delay. They will be discussed in Chapter 9,* **Propagation Delay***.*

---

## Calculating the Delay

As you design your network, you should sketch the overall topology to determine rough equipment locations, cable segment loading and maximum repeater hops. Calculating the overall round trip network propagation delay is an easy task that should only take a few minutes. The most difficult part, by far, is finding the proper delay times for individual pieces of hardware. These numbers should be obtainable from the original hardware manufacturer.

The total propagation delay is found by simply adding up the delays of the individual components and cabling in the longest signal path and multiplying the result by 2 to get the round trip delay time. The longest signal path is found by identifying the longest path between any two network devices, even though the two devices may not directly communicate with each other. Examples of network devices that could be considered an end point on the network would be PCs, workstations, file servers, bridges, routers, gateways, and printers.

# Propagation Delay Example

To clarify the methods used to calculate the propagation delay of a given network, refer to Figure 8-1 and complete the following steps to calculate the propagation delay of the network shown.

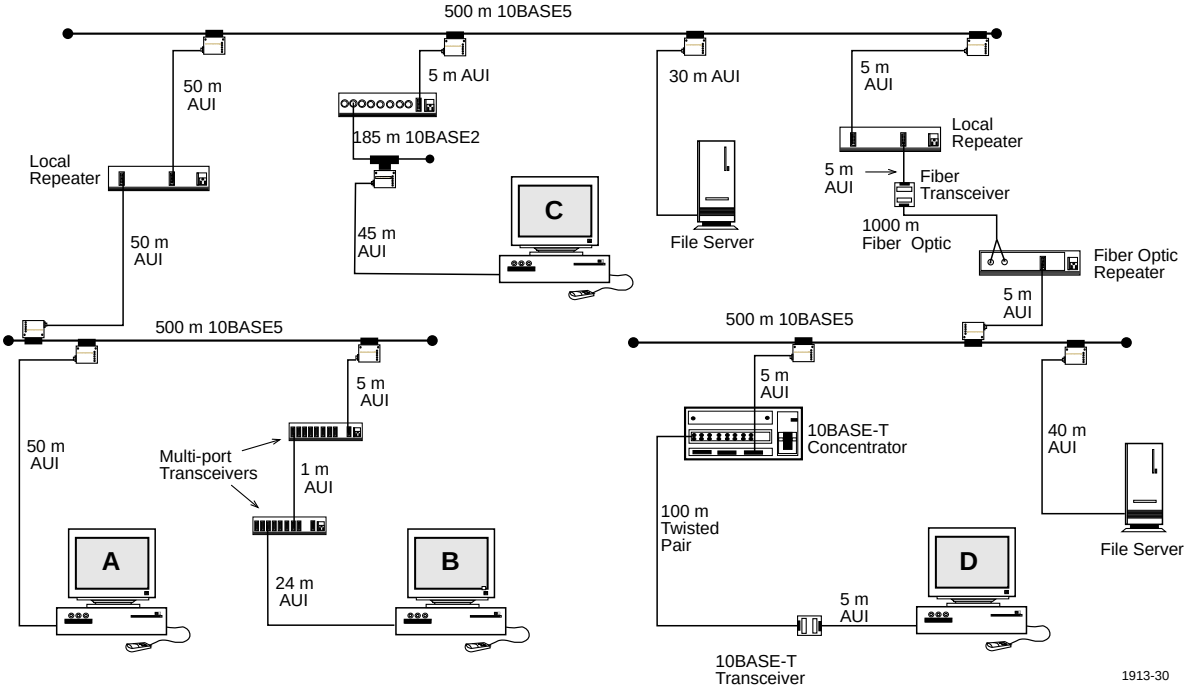



**Figure 8-1. Propagation Delay Example**

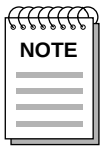
1. Carefully examine Figure 8-1 and identify the longest signal path between any two end devices. Remember, the longest path is not necessarily the longest physical path, but the path with the longest signal delay.

You should have determined that the longest path is the path between Workstation B and Workstation D.

2. Using the equipment delay table, Table 8-1, make a list of the equipment the signal must pass through in this path and the appropriate delay time for each piece of equipment.


The equipment delay times in Table 8-1 are Cabletron Systems specific delay times. Other manufacturer's equipment may have different delays. Refer to the specific manufacturer's delay times of the equipment in your network design to calculate your propagation delay.

3. Now that you have the equipment delay times, you must calculate the total delay for each media type. To do this, make a list of each media type found in the signal path and add up the total length, in meters, of each. Now, multiply this number by the appropriate media delay time for each media type found in Table 8-1, to get the delay time for each length of media found in the signal path.




The values contained in Table 8-1 are maximum values. It is possible that certain grades of cable will have a higher velocity of propagation which will result in a smaller delay per meter.

4. Add up the equipment delay times, calculated in step 2, to find the total equipment delay for this signal path.

5. Add up the media delay times, calculated in step 3, to find the total media delay for this signal path.

6. Add the numbers found in step 4 and step 5. This is the total one-way propagation delay time for this network. Multiply this number by two to get the round trip propagation delay time.

By IEEE definition, the one way propagation delay time must be less than or equal to 25.6 µs. Your result from this exercise should be 23.45 µs. Since this value is less than the IEEE maximum, the network is within specification. If you didn't come up with 23.45 µs, refer to Table 8-2 and Table 8-3 to see where you went wrong.

**Table 8-1. Equipment and Cable Propagation Delay Times**

| Equipment Type | Delay |
|---|---|
| Local Repeater | 0.65 µs |
| Fiber Optic Repeater | 1.55 µs |
| Multi-port Repeater | 1.55 µs |
| Multi-port Transceiver | 0.10 µs |
| Standard Transceiver | 0.86 µs |
| Fiber Optic Transceiver | 0.20 µs |
| Twisted Pair Transceiver | 0.27 µs |
| Concentrator | 1.90 µs |

| Media Type | Delay per Meter |
|---|---|
| 10BASE5 coaxial | 0.00433 µs/m |
| 10BASE2 coaxial | 0.00514 µs/m |
| Shielded Twisted Pair (STP) | 0.0057 µs/m |
| Unshielded Twisted Pair (UTP) | 0.0057 µs/m |
| Fiber Optic | 0.005 µs/m |
| AUI | 0.00514 µs/m |

**Table 8-2. Equipment Propagation Delay Worksheet**

| Equipment Type | | A Delay | B Quantity | (A*B) Total |
|---|---|---|---|---|
| 1 | Local Repeater | 0.65 µs | 2 | 1.30 µs |
| 2 | Fiber Optic Repeater | 1.55 µs | 1 | 1.55 µs |
| 3 | Multi-port Repeater | 1.55 µs | - | 0 |
| 4 | Multi-port Transceiver | 0.10 µs | 2 | 0.20 µs |
| 5 | Standard Transceiver | 0.86 µs | 6 | 5.16 µs |
| 6 | Fiber Optic Transceiver | 0.20 µs | 1 | 0.20 µs |
| 7 | Twisted Pair Transceiver | 0.27 µs | 1 | 0.27 µs |
| 8 | Concentrator | 1.90 µs | 1 | 1.90 µs |
| **Equipment Delay Total (Add lines 1–8)** | | | | **10.58 µs** |

**Table 8-3. Cable Delay Worksheet**

| Cable Type | | A Delay/Meter | B Quantity | (A*B) Total |
|---|---|---|---|---|
| 1 | 10BASE5 | 0.0043 µs/m | 1500 m | 6.5 µs |
| 2 | 10BASE2 | 0.00514 µs/m | - | 0 |
| 3 | Shielded Twisted Pair (STP) | 0.0057 µs/m | - | 0 |
| 4 | Unshielded Twisted Pair (UTP) | 0.0057 µs/m | 100 m | 0.57 µs |
| 5 | Fiber Optic | 0.005 µs/m | 1000 m | 5.00 µs |
| 6 | AUI | 0.00514 µs/m | 155 m | 0.80 µs |
| **Cable Delay Total (Add lines 1–6)** | | | | **12.87 µs** |

To calculate the One Way propagation delay time, add the values in the shaded bottom right hand corner of Table 8-2 and Table 8-3 together.

| | |
|---|---|
| Equipment Delay: | 10.58 µs |
| Cable Delay | 12.87 µs |
| Total One Way Delay | 23.45 µs |

As with designing anything, it is not advisable to design to the limits of the specification. As an electronic device ages, its internal propagation delay may change. Also, as devices such as repeaters and transceivers are changed due to upgrades or changing from one vendor to another, the propagation delay times may change as well. By using the worst case values for the delay times you should have a comfortable amount of design buffer built into your network.

Chapter 9

# Ethernet Bridge Operation

*Bridges are devices that are added to a network to allow expansion beyond the limits of the IEEE 802.3 specification. They are also used to connect two similar networks together, allowing communication between them. Bridges accomplish this by reading in frames and deciding to either filter or forward the frame based on the destination address of the frame. The following sections detail the operation of bridges and their functions.*

## Filtering and Forwarding

Bridges decide whether to forward or filter a frame based on the location of the destination with respect to the source. They dynamically learn the location of devices by logging the source address of each frame and the port it was received on in their Source Address Table (SAT). Refer to Figure 9-1, which shows two Ethernet LANs connected by a bridge, and the following explanation on the filtering and forwarding process.

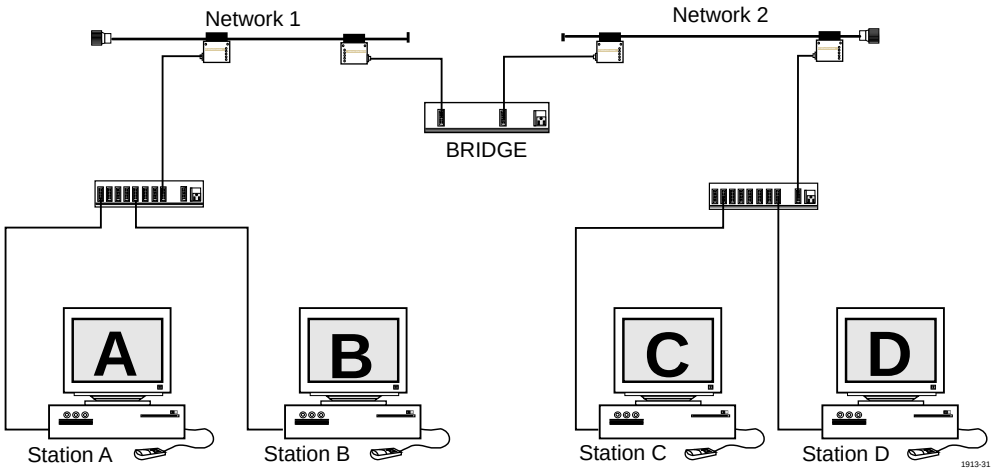



**Figure 9-1. Use of Ethernet Bridges**

When the bridge is first powered up, its SAT is empty:

| Network 1 | Network 2 |
|---|---|
| | |
| | |

Assume Station A wants to transmit a frame to Station B. The bridge receives the frame and checks the CRC (Cyclic Redundancy Check) of the frame. The bridge then looks at the source address of the frame and puts that address in the source address table of Network 1 as shown below:

| Network 1 | Network 2 |
|---|---|
| **A** | |
| | |

The bridge then checks the destination of the frame to see if it is located on Network 1. Since Station B has not transmitted anything yet, the bridge has no idea where it is located so the bridge forwards the frame to Network 2.

Station B now sends its response to Station A. The bridge receives the frame and checks the CRC. If the CRC is good, the bridge looks at the source address and sees that the address is B. The Station B address is placed in the SAT as shown below:

| Network 1 | Network 2 |
|---|---|
| **A** | |
| **B** | |

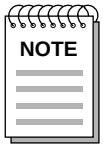
Next, the bridge checks to see if the destination address (Station A) is listed in the SAT. In this case, Station A is listed as being on Network 1 so the bridge blocks the frame from Network 2.

Assume that station A wants to transmit to Station D. Station A sends the frame to Station D. The bridge reads in the frame and checks the CRC. The bridge reads the source address of the frame (Station A) and makes sure Station A is still in the SAT. The bridge checks the SAT for the destination address, Station D. It is not found to reside on the Network 1 side of the bridge so the frame is forwarded to Network 2.

Next, Station D sends its response to Station A. The bridge reads in the frame, and after checking the CRC, updates the SAT with the Station D address as being located on Network 2. The SAT now shows the following:

| Network 1 | Network 2 |
|---|---|
| A | D |
| B | |

The bridge then inspects the SAT looking for Station A on the Network 2 side of the bridge. It was not found on the Network 2 side so the frame is forwarded to Network 1.




It is important to realize that the bridge did not make the forwarding decision because Station A was on the Network 1 side, but because Station A was not on the Network 2 side.

This process continues until all stations are logged on the SAT. The bridge will now isolate different network traffic, as well as extend the maximum size of each individual network.

# Spanning Tree Algorithm

Considering the important role bridges play in the transfer of data from one network to another, it is a good idea to set up a redundant bridge that commences operation if the primary bridge should fail. Therefore, IEEE chose to build some fault tolerance into the bridge specification. The 802.1d specification defines bridge operation, redundancy and a process called the Spanning Tree Algorithm (STA) which allows bridges to be connected in such a way as to create standby redundant paths without creating data loops. This same algorithm activates a redundant path in case of a failure in the active path.

## Configuration BPDU

When a bridge is powered up, it goes through a series of self tests to check its internal operation. During this time the bridge is in a standby condition and does not forward traffic. Also during this standby period, the bridge sends out special bridge management frames called Configuration Bridge Protocol Data Units (BPDU). Bridges use the BPDU frames as a way of communicating with each other. The configuration BPDU is 64 bytes long and contains the following fields:

- Destination Address: A specific 6-byte Ethernet multicast address that denotes the bridge group.
- Source Address: The standard 6-byte Ethernet hardware address of the bridge transmitting the BPDU.
- Length Field: A standard, 2-byte, IEEE 802.3 data field length.
- Data Field: Contains 35 bytes for BPDU data and 13 bytes of pad to equal the minimum data field size of 48 bytes. The BPDU data field contains the following information:
  - Protocol Identifier: A reserved, 2-byte, protocol identifier defined by IEEE.
  - Protocol Version Identifier: Identifies the version of the bridge protocol being used and is 1 byte in length.
  - BPDU Type: 1 byte to identify the BPDU type as either a configuration or topology change BPDU.
  - Flags: A 1-byte field that contains topology change and topology change acknowledgment flags.
  - Root Identifier: An 8-byte identification number derived from the Ethernet address of a bridge and its unique port addresses. This field contains the ID for the perceived root bridge.
  - Root Path Cost: A 4-byte field that contains a “cost” value composed of individual bridge port costs along a data path. Bridges use this information to determine the optimum frame transmission path.
  - Bridge Identifier: An 8-byte identification number that is derived from the Ethernet address of a bridge and its unique port addresses.
  - Port Identifier: A 2-byte field that contains port priority information based on unique bridge port addresses. The port with the lowest address has the highest priority.

- Message Age: A 2-byte field that contains the age of the configuration BPDU. This parameter allows a bridge to determine if the BPDU is too old and needs to be discarded.
- Max Age: A 2-byte field that contains a time-out value initially set by the root bridge. This value is compared to the Message Age to determine the validity of the BPDU.
- Hello Time: A 2-byte field that contains the value for the time interval used to generate configuration BPDU's by the root bridge. If a bridge does not hear from the root bridge within the time defined by the Hello Time, the bridge will initiate a topology change and attempt to become the new root.
- Forward Delay: A 2-byte field that contains a value used by all bridges as a delay value when a port changes state to the forwarding condition. This delay is necessary to prevent data loops and duplicate data that could be caused by an instantaneous change of port state. This delay value is also used as the age time for the source address table whenever a topology change has been detected by a bridge. This value temporarily replaces the default age time in order to quickly flush the source address table so network addresses will be properly relearned after the topology change. The Forward Delay time value is established by the root bridge.

The purpose of the configuration BPDU is to notify other bridges on all of the connected networks of the current topology. Based on the bridge priority and address, the other bridges automatically detect loops and negotiate a single path. The bridge or bridges involved in this primary data path then come on-line and the bridges with lower priority involved in the backup path(s) remain in standby.

## Topology Change BPDU

A topology change BPDU is made up of only 4 data bytes (plus pad) that contain the Protocol Identifier, Protocol Version Identifier and BPDU Type. The purpose of the topology change BPDU is to notify other bridges that a change has taken place. The other bridges then re-span to form a legal topology.

# Spanning Tree Operation

In the following explanation we take Spanning Tree through the network shown in Figure 9-2, which consists of two-port bridges. You must understand that the Spanning Tree process is a single operation, combining both root bridge determination and data loop detection and resolution. For the purpose of explanation, we split the process into two individual discussions:

- Root Bridge determination
- Data Loop detection and resolution

Also, to clarify our referencing, the bridges are named as shown:

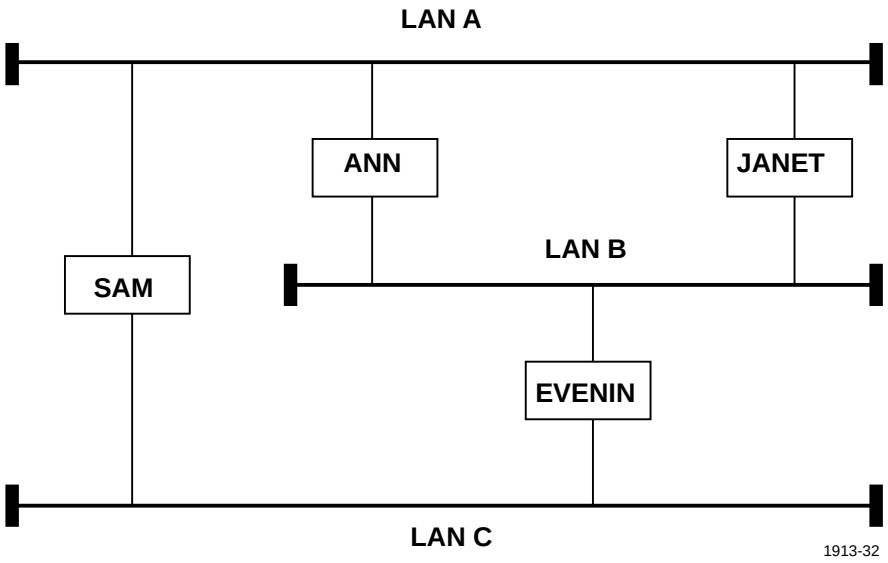



**Figure 9-2. Network Before Spanning Tree**

The Bridge Parameters for the network above are:

| Bridge | Bridge ID | Path Cost |
|---|---|---|
| SAM | 80-00-00-00-1d-23-56-a2 | 100 |
| ANN | 80-00-00-00-1d-56-d4-f4 | 100 |
| JANET | 80-00-00-00-1d-f4-67-2a | 100 |
| EVENIN | 80-00-00-00-1d-4f-94-a1 | 100 |

The primary function of the Spanning Tree Algorithm is to ensure that there is only one data path between any two end stations within the bridged Local Area Network. All computations are geared towards the fact that a bridge wants to be considered as the Designated Bridge for any LAN that it is connected to. Upon power up, BPDUs are sent out as multicast frames. A bridge directly connected to the same LAN that the BPDUs are sent out on will accept these BPDUs and make decisions on their contents.

For ease of explanation we will represent the individual bridge's BPDUs in a shortened version consisting of four parameters as shown below:

| **BPDU** |
|---|
| Root Bridge ID |
| Path Cost |
| Bridge ID |
| Port ID |

- Bridge ID: A 64-bit value that is comprised of two individual components:
  - Bridge Priority: A 16-bit configured value
  - MAC Address: A 48-bit value which is the hardware Ethernet address of the bridge, e.g., 00-00-1d-e2-45-a2
- Root Bridge ID: Same as the Bridge ID but is the ID of the designated root bridge.
- Path Cost: A value representing the contributing cost of passing through this bridge. The formula used to determine the default is 1000/network Mbps per sec. The default for Ethernet is 1000/10=100.
- Port ID: A 16-bit value made up of two components:
  - Port Priority: 8 bits in length and is the most significant byte of the Port ID.
  - Port ID: 8 bits in length numbered sequentially on a bridge from 1 to infinity (in theory).

The default Port ID for port 1 resembles the following: 8000-0001.

With the understanding of our shortened BPDU, we now begin our explanation of the Spanning Tree Operation. In Figure 9-3, SAM is sending out BPDUs across all LANs to which it is connected. SAM sends BPDUs out through its port 2 onto LAN C and through port 1 onto LAN A.

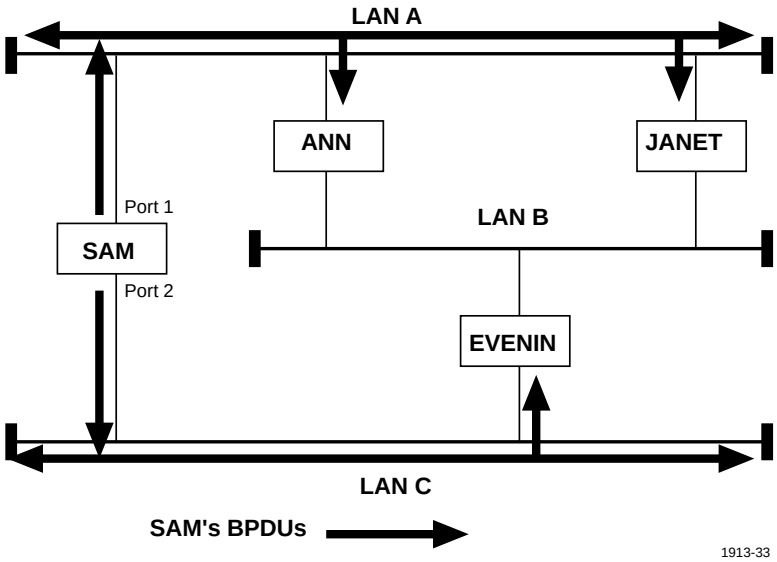



**Figure 9-3. SAM's Initial BPDUs**

SAM's BPDUs are represented by the following examples:

| | BPDU (Port 1) | BPDU (Port 2) |
|---|---|---|
| Root ID | 80-00-00-00-1d-23-56-a2 | 80-00-00-00-1d-23-56-a2 |
| Path Cost | 0 | 0 |
| Bridge ID | 80-00-00-00-1d-23-56-a2 | 80-00-00-00-1d-23-56-a2 |
| Port ID | 8001 | 8002 |

Each of these BPDUs represent SAM as their root bridge. This is shown by the Path Cost to the root from LANs A & C = 0 (if SAM was root, then there would be no bridge hops going from LAN A to the root, hence a cost of 0). They also indicate that SAM was the transmitting bridge (Bridge ID). The only difference between the BPDUs is the Port Identifier; 8001 for port 1 and 8002 for port 2.

Now let's look at one of the other bridges, in this case ANN. Figure 9-4 shows ANN generating BPDUs from all of its ports, considering itself as root until it finds out differently.

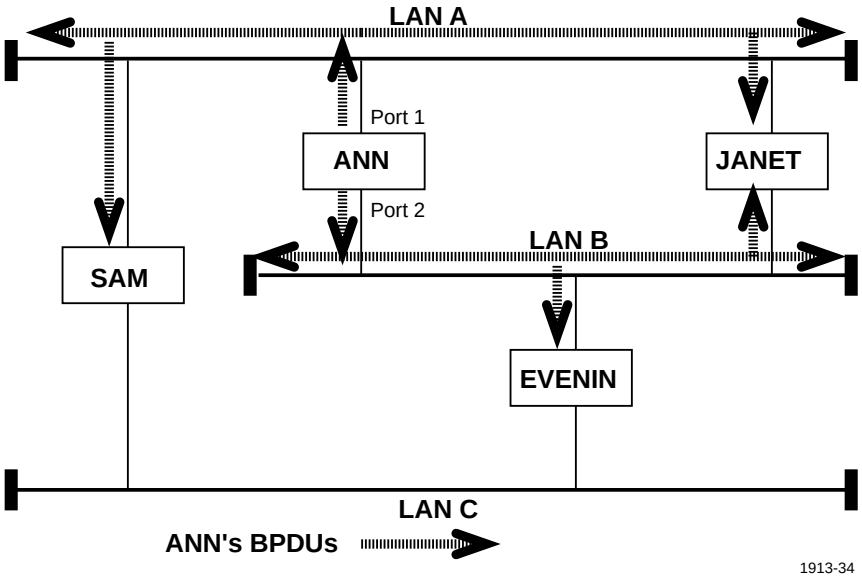



**Figure 9-4. ANN's Initial BPDUs**

The BPDUs generated by ANN are shown below:

| | BPDU (Port 1) | BPDU (Port 2) |
|---|---|---|
| Root ID | 80-00-00-00-1d-56-d4-f4 | 80-00-00-00-1d-56-d4-f4 |
| Path Cost | 0 | 0 |
| Bridge ID | 80-00-00-00-1d-56-d4-f4 | 80-00-00-00-1d-56-d4-f4 |
| Port ID | 8001 | 8002 |

Here again, this bridge thinks that it is the root, indicated by the root ID value within the BPDU, and the cost to the root of 0. It also indicates by the Bridge ID that ANN is the bridge that has transmitted this BPDU. Again, the Port Identifier reflects the transmitting port of the BPDU on ANN. We now assume that both SAM and ANN receive each other's BPDUs from LAN A.

SAM checks the incoming BPDU from ANN and it reflects a different root ID. In the process of checking the incoming data against that which is current at that port, SAM realizes that it has the higher priority root ID (lower number) and does not forward ANN's BPDU through port 2; it continues to send its own.

ANN checks the incoming BPDU from SAM and senses that the BPDU carries a higher priority BPDU (lower number) than its own. ANN now stops transmitting its own BPDUs and begins to modify and retransmit those received from SAM. These retransmissions by ANN are transmitted out of port 2, but not port 1. This procedure continues until the only BPDUs being generated are originating at the root bridge.

Before we jump to the root bridge, however, let's continue the BPDU trail to the finish. ANN relinquished its bid to become root and agreed that SAM is better qualified. ANN propagates SAM's BPDUs to any other LAN that it is directly connected to. Here are ANN's new BPDUs:

| | **BPDU IN (Port 1)** | **BPDU OUT (Port 2)** |
|---|---|---|
| Root ID | 80-00-00-00-1d-23-56-a2 | 80-00-00-00-1d-23-56-a2 |
| Path Cost | 0 | 100 |
| Bridge ID | 80-00-00-00-1d-23-56-a2 | 80-00-00-00-1d-56-d4-f4 |
| Port ID | 8001 | 8002 |

You should recognize the BPDU coming in on port 1 (SAM's BPDU). Now look at the BPDU coming out of port 2. It indicates that SAM is the root bridge (root ID) and the Bridge Identifier indicates that this BPDU was transmitted from ANN. Notice that the root Path Cost is updated to 100–meaning it costs 100 to get to the root through ANN.

A similar case can be made for EVENIN, the bridge that spans LAN B and LAN C shown in Figure 9-5.

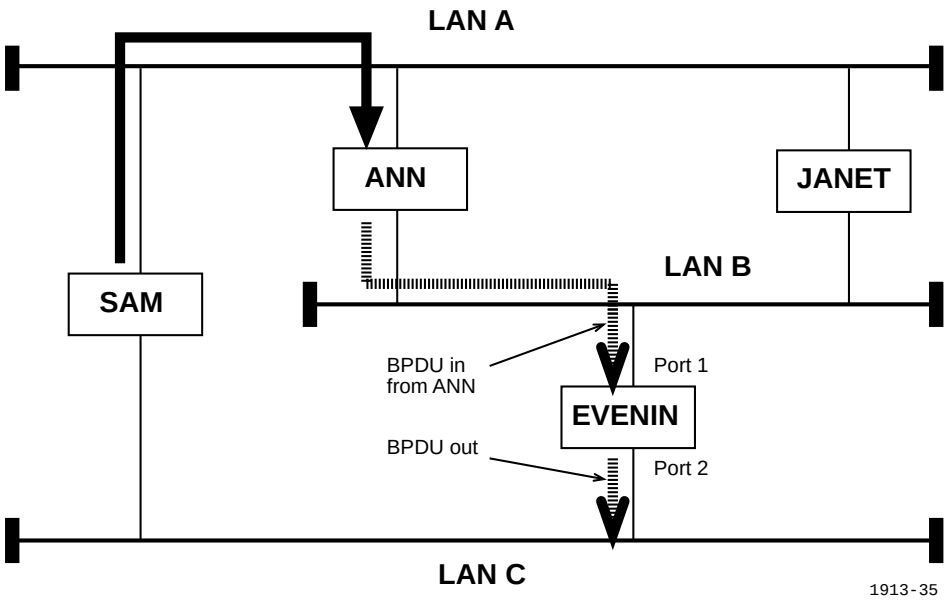



**Figure 9-5. BPDUs in from ANN to EVENIN**

EVENIN would have also thought that it was root until it found out otherwise. Let's look at the progression of BPDUs from ANN. Here are EVENIN's BPDUs:

| | BPDU IN (Port 1) | BPDU OUT (Port 2) |
|---|---|---|
| Root ID | 80-00-00-00-1d-23-56-a2 | 80-00-00-00-1d-23-56-a2 |
| Path Cost | 100 | 200 |
| Bridge ID | 80-00-00-00-1d-56-d4-f4 | 80-00-00-00-1d-4f-94-a1 |
| Port ID | 8002 | 8002 |

The BPDU coming in to port 1 on EVENIN indicates that SAM has a higher priority than it does, so it also stops generating its own BPDUs and forwards those of the higher priority. The BPDU coming out of port 2 of EVENIN reflects SAM as the root (root ID). The incoming BPDU reflects that ANN can get to the root from LAN B at a cost of 100. The outgoing BPDU onto LAN C represents the fact to any Bridge on LAN C that EVENIN can get to the root for a cost of 200 (root path cost).

## Data Loop Resolution

The center of our discussion here will be within EVENIN. It received BPDUs from both of its ports. A BPDU from SAM came in on port 2, and a BPDU from ANN came in on port 1. With the reception of these two frames from different ports, each identifying SAM as the root, EVENIN realizes that there is a data loop present.

Looking at this objectively, we can see that the root bridge is directly connected to both LANs A and C so the root bridge is the designated bridge for both of these LANs. However, the designated bridge for LAN B is still undecided.

The BPDU transmitted from EVENIN onto LAN B and the BPDU transmitted from ANN onto LAN B are shown next:

| | **EVENINs BPDU OUT (port 1) to ANN** | **ANNs BPDU OUT (port 2) to EVENIN** |
|---|---|---|
| Root ID | 80-00-00-1d-4f-94-a1 | 80-00-00-00-1d-23-56-a2 |
| Path Cost | 100 | 100 |
| Bridge ID | 80-00-00-00-1d-4f-94-a1 | 80-00-00-00-1d-56-d4-f4 |
| port ID | 8001 | 8001 |

At this point we see that SAM is considered to be the root bridge by all other bridges (root ID). There is no conflict concerning which bridge is the designated bridge for LANs A and C, as the root is directly connected to these two LANs. However we have two bridges trying to service LAN B: ANN and EVENIN. This problem is resolved by the Bridge Entities residing in both of these bridges.

Each bridge entity maintains a list of current port parameters for each bridge port. The bridge entity receives BPDUs coming in from a port and compare the BPDU information against the current port parameters. In the case of a data loop, the bridge entity makes a decision as to whether it should put the receiving port in the BLOCKING state or FORWARDING state. It is this periodic dialogue that eventually settles our Bridged Local Area Network in a single Spanning Tree.

In this scenario ANN retransmits SAM's BPDUs onto LAN B and EVENIN retransmits SAM's BPDUs onto LAN B.

The information that both the bridges have in relation to LAN B is as follows:

| Port Parameters | ANNs Port 2 | EVENINs Port 1 |
|---|---|---|
| Port ID | 8002 | 8001 |
| Designated Root | SAM | SAM |
| Designated Bridge | ANN | EVENIN |
| Designated Port | 8002 | 8001 |
| Root Path Cost | 100 | 100 |

The algorithm to determine who breaks the identified data loop is in the following order:

a. lowest root path cost

b. highest priority designated bridge ID

c. highest priority designated port ID

d. highest priority port ID

Each bridge compares the incoming BPDU data up against its current port parameters, and determines the active topology based upon the algorithm above.

1. The root path cost is the first parameter checked. Both ANN and EVENIN offer a cost of 100 from LAN B to the root so this is a deadlock.

2. The designated bridge parameter is the next parameter checked:

   - ANN's designated bridge is itself: 80-00-00-00-1d-56-d4-f4

   - EVENIN's designated bridge is itself: 80-00-00-00-1d-4f-94-a1

Here, it appears that we have broken the tie. The bridge that holds the highest priority designated bridge ID will win out. EVENIN sees an inferior bridge ID coming in compared to its current parameters, so it will go to the FORWARDING state. ANN sees a superior bridge ID coming in so it will go to the BLOCKING State. EVENIN becomes the designated bridge to LAN B.

There is one final data loop present. JANET is bridging from LAN A to LAN B. The bridge entities recognize the data loop condition by monitoring the incoming BPDUs. Upon seeing BPDUs coming in through both of its ports, all originating from the root, it realizes that there is more than one path to the root bridge. By using the order of comparisons shown above, the bridge entity can make an intelligent decision as to whether the port in question should be made part of the active topology or should be sent to the BLOCKING state.

Figure 9-6 shows the resulting topology. There is only one data path between any two end stations within this bridged LAN. All data loops have been identified and resolved. We now have built-in redundancy that can be used if one or more of the active bridge components fail.

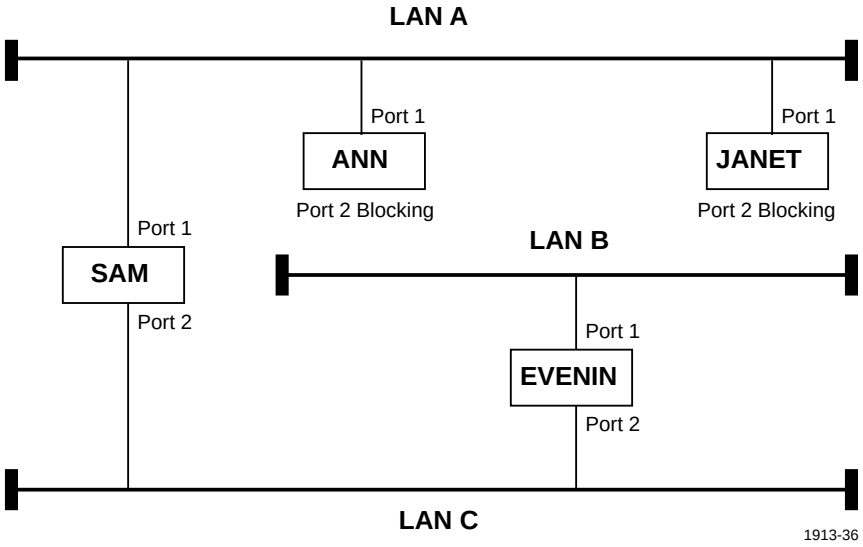



**Figure 9-6. Resulting Topology after Spanning Tree**

# Index

## Numerics



## A



## B



## C



## D



## E



## F



## G



## H



## I

## L



## M



## N



## O



## P



## R



## S



## T