

13.1 ETHERNET PROTOCOL

In Chapter 1, we mentioned that the TCP/IP protocol suite does not define any protocol for the data-link or the physical layer. In other words, TCP/IP accepts any protocol at these two layers that can provide services to the network layer. The data-link layer and the physical layer are actually the territory of the local and wide area networks. This means that when we discuss these two layers, we are talking about networks that are using them. As we see in this and the following two chapters, we can have wired or wireless networks. We discuss wired networks in this chapter and the next and postpone the discussion of wireless networks to Chapter 15.

In Chapter 1, we learned that a local area network (LAN) is a computer network that is designed for a limited geographic area such as a building or a campus. Although a LAN can be used as an isolated network to connect computers in an organization for the sole purpose of sharing resources, most LANs today are also linked to a wide area network (WAN) or the Internet.

In the 1980s and 1990s several different types of LANs were used. All of these LANs used a media-access method to solve the problem of sharing the media. The Ethernet used the CSMA/CD approach. The Token Ring, Token Bus, and FDDI (Fiber Distribution Data Interface) used the token-passing approach. During this period, another LAN technology, ATM LAN, which deployed the high speed WAN technology (ATM), appeared in the market.

Almost every LAN except Ethernet has disappeared from the marketplace because Ethernet was able to update itself to meet the needs of the time. Several reasons for this success have been mentioned in the literature, but we believe that the Ethernet protocol was designed so that it could evolve with the demand for higher transmission rates. It is natural that an organization that has used an Ethernet LAN in the past and now needs a higher data rate would update to the new generation instead of switching to another technology, which might cost more. This means that we confine our discussion of wired LANs to the discussion of Ethernet.

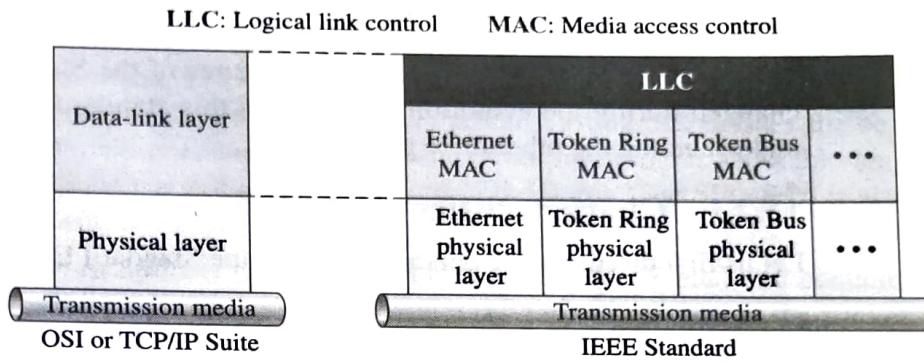
13.1.1 IEEE Project 802

Before we discuss the Ethernet protocol and all its generations, we need to briefly discuss the IEEE standard that we often encounter in text or real life. In 1985, the Computer Society of the IEEE started a project, called **Project 802**, to set standards to enable intercommunication among equipment from a variety of manufacturers. Project 802 does not seek to replace any part of the OSI model or TCP/IP protocol suite. Instead, it is a way of specifying functions of the physical layer and the data-link layer of major LAN protocols.

The relationship of the 802 Standard to the TCP/IP protocol suite is shown in Figure 13.1. The IEEE has subdivided the data-link layer into two sublayers: **logical link control (LLC)** and **media access control (MAC)**. IEEE has also created several physical-layer standards for different LAN protocols.

Logical Link Control (LLC)

Earlier we discussed *data link control*. We said that data link control handles framing, flow control, and error control. In IEEE Project 802, flow control, error control, and

Figure 13.1 IEEE standard for LANs

part of the framing duties are collected into one sublayer called the *logical link control* (LLC). Framing is handled in both the LLC sublayer and the MAC sublayer.

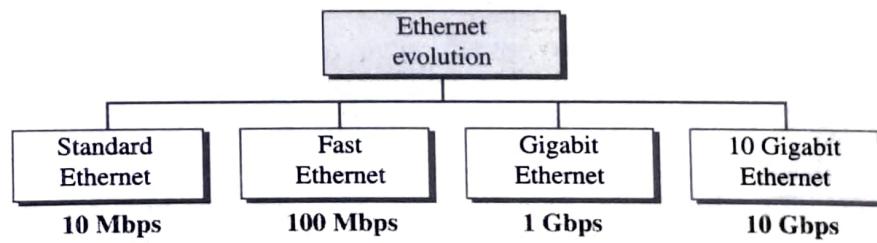
The LLC provides a single link-layer control protocol for all IEEE LANs. This means LLC protocol can provide interconnectivity between different LANs because it makes the MAC sublayer transparent.

Media Access Control (MAC)

Earlier we discussed multiple access methods including random access, controlled access, and channelization. IEEE Project 802 has created a sublayer called **media access control** that defines the specific access method for each LAN. For example, it defines CSMA/CD as the media access method for Ethernet LANs and defines the token-passing method for Token Ring and Token Bus LANs. As we mentioned in the previous section, part of the framing function is also handled by the MAC layer.

13.1.2 Ethernet Evolution

The Ethernet LAN was developed in the 1970s by Robert Metcalfe and David Boggs. Since then, it has gone through four generations: **Standard Ethernet** (10 Mbps), **Fast Ethernet** (100 Mbps), **Gigabit Ethernet** (1 Gbps), and **10 Gigabit Ethernet** (10 Gbps), as shown in Figure 13.2. We briefly discuss all these generations.

Figure 13.2 Ethernet evolution through four generations

13.2 STANDARD ETHERNET

We refer to the original Ethernet technology with the data rate of 10 Mbps as the *Standard Ethernet*. Although most implementations have moved to other technologies in the Ethernet evolution, there are some features of the Standard Ethernet that have not changed during the evolution. We discuss this standard version to pave the way for understanding the other three technologies.

13.2.1 Characteristics

Let us first discuss some characteristics of the Standard Ethernet.

Connectionless and Unreliable Service

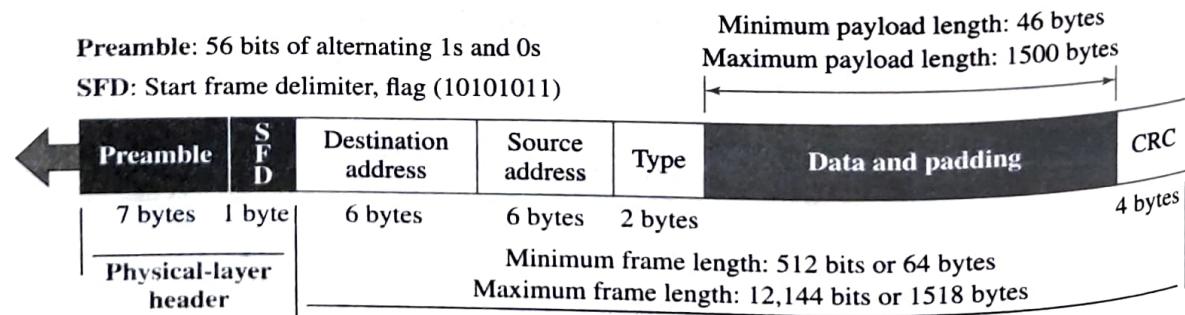
Ethernet provides a connectionless service, which means each frame sent is independent of the previous or next frame. Ethernet has no connection establishment or connection termination phases. The sender sends a frame whenever it has it; the receiver may or may not be ready for it. The sender may overwhelm the receiver with frames, which may result in dropping frames. If a frame drops, the sender will not know about it. Since IP, which is using the service of Ethernet, is also connectionless, it will not know about it either. If the transport layer is also a connectionless protocol, such as UDP, the frame is lost and salvation may only come from the application layer. However, if the transport layer is TCP, the sender TCP does not receive acknowledgment for its segment and sends it again.

Ethernet is also unreliable like IP and UDP. If a frame is corrupted during transmission and the receiver finds out about the corruption, which has a high level of probability of happening because of the CRC-32, the receiver drops the frame silently. It is the duty of high-level protocols to find out about it.

Frame Format

The Ethernet frame contains seven fields, as shown in Figure 13.3.

Figure 13.3 Ethernet frame



- ❑ **Preamble.** This field contains 7 bytes (56 bits) of alternating 0s and 1s that alert the receiving system to the coming frame and enable it to synchronize its clock if it's out of synchronization. The pattern provides only an alert and a timing pulse. The 56-bit

pattern allows the stations to miss some bits at the beginning of the frame. The *preamble* is actually added at the physical layer and is not (formally) part of the frame.

- **Start frame delimiter (SFD).** This field (1 byte: 10101011) signals the beginning of the frame. The SFD warns the station or stations that this is the last chance for synchronization. The last 2 bits are $(11)_2$ and alert the receiver that the next field is the destination address. This field is actually a flag that defines the beginning of the frame. We need to remember that an Ethernet frame is a variable-length frame. It needs a flag to define the beginning of the frame. The SFD field is also added at the physical layer.
- **Destination address (DA).** This field is six bytes (48 bits) and contains the link-layer address of the destination station or stations to receive the packet. We will discuss addressing shortly. When the receiver sees its own link-layer address, or a multicast address for a group that the receiver is a member of, or a broadcast address, it decapsulates the data from the frame and passes the data to the upper-layer protocol defined by the value of the type field.
- **Source address (SA).** This field is also six bytes and contains the link-layer address of the sender of the packet. We will discuss addressing shortly.
- **Type.** This field defines the upper-layer protocol whose packet is encapsulated in the frame. This protocol can be IP, ARP, OSPF, and so on. In other words, it serves the same purpose as the protocol field in a datagram and the port number in a segment or user datagram. It is used for multiplexing and demultiplexing.
- **Data.** This field carries data encapsulated from the upper-layer protocols. It is a minimum of 46 and a maximum of 1500 bytes. We discuss the reason for these minimum and maximum values shortly. If the data coming from the upper layer is more than 1500 bytes, it should be fragmented and encapsulated in more than one frame. If it is less than 46 bytes, it needs to be padded with extra 0s. A padded data frame is delivered to the upper-layer protocol as it is (without removing the padding), which means that it is the responsibility of the upper layer to remove or, in the case of the sender, to add the padding. The upper-layer protocol needs to know the length of its data. For example, a datagram has a field that defines the length of the data.
- **CRC.** The last field contains error detection information, in this case a CRC-32. The CRC is calculated over the addresses, types, and data field. If the receiver calculates the CRC and finds that it is not zero (corruption in transmission), it discards the frame.

Frame Length

Ethernet has imposed restrictions on both the minimum and maximum lengths of a frame. The minimum length restriction is required for the correct operation of CSMA/CD, as we will see shortly. An Ethernet frame needs to have a minimum length of 512 bits or 64 bytes. Part of this length is the header and the trailer. If we count 18 bytes of header and trailer (6 bytes of source address, 6 bytes of destination address, 2 bytes of length or type, and 4 bytes of CRC), then the minimum length of data from the upper layer is $64 - 18 = 46$ bytes. If the upper-layer packet is less than 46 bytes, padding is added to make up the difference.

The standard defines the maximum length of a frame (without preamble and SFD field) as 1518 bytes. If we subtract the 18 bytes of header and trailer, the maximum length of the payload is 1500 bytes. The maximum length restriction has two historical reasons. First, memory was very expensive when Ethernet was designed; a maximum length restriction helped to reduce the size of the buffer. Second, the maximum length restriction prevents one station from monopolizing the shared medium, blocking other stations that have data to send.

Minimum frame length: 64 bytes
Maximum frame length: 1518 bytes

Minimum data length: 46 bytes
Maximum data length: 1500 bytes

13.2.2 Addressing

Each station on an Ethernet network (such as a PC, workstation, or printer) has its own **network interface card (NIC)**. The NIC fits inside the station and provides the station with a link-layer address. The Ethernet address is 6 bytes (48 bits), normally written in **hexadecimal notation**, with a colon between the bytes. For example, the following shows an Ethernet MAC address:

4A:30:10:21:10:1A

Transmission of Address Bits

The way the addresses are sent out online is different from the way they are written in hexadecimal notation. The transmission is left to right, byte by byte; however, for each byte, the least significant bit is sent first and the most significant bit is sent last. This means that the bit that defines an address as unicast or multicast arrives first at the receiver. This helps the receiver to immediately know if the packet is unicast or multicast.

Example 13.1

Show how the address 47:20:1B:2E:08:EE is sent out online.

Solution

The address is sent left to right, byte by byte; for each byte, it is sent right to left, bit by bit, as shown below:

Hexadecimal	47	20	1B	2E	08	EE
Binary	01000111	00100000	00011011	00101110	00001000	11101110
Transmitted ←	11100010	00000100	11011000	01110100	00010000	01110111

Unicast, Multicast, and Broadcast Addresses

A source address is always a *unicast address*—the frame comes from only one station. The destination address, however, can be *unicast*, *multicast*, or *broadcast*. Figure 13.4 shows how to distinguish a unicast address from a multicast address. If the least significant bit of the first byte in a destination address is 0, the address is unicast; otherwise, it is multicast.

Note that with the way the bits are transmitted, the unicast/multicast bit is the first bit which is transmitted or received. The broadcast address is a special case of the

Figure 13.4 Unicast and multicast addresses

multicast address: the recipients are all the stations on the LAN. A broadcast destination address is forty-eight 1s.

Example 13.2

Define the type of the following destination addresses:

- a. 4A:30:10:21:10:1A
- b. 47:20:1B:2E:08:EE
- c. FF:FF:FF:FF:FF:FF

Solution

To find the type of the address, we need to look at the second hexadecimal digit from the left. If it is even, the address is unicast. If it is odd, the address is multicast. If all digits are Fs, the address is broadcast. Therefore, we have the following:

- a. This is a unicast address because A in binary is 1010 (even).
- b. This is a multicast address because 7 in binary is 0111 (odd).
- c. This is a broadcast address because all digits are Fs in hexadecimal.

Distinguish Between Unicast, Multicast, and Broadcast Transmission

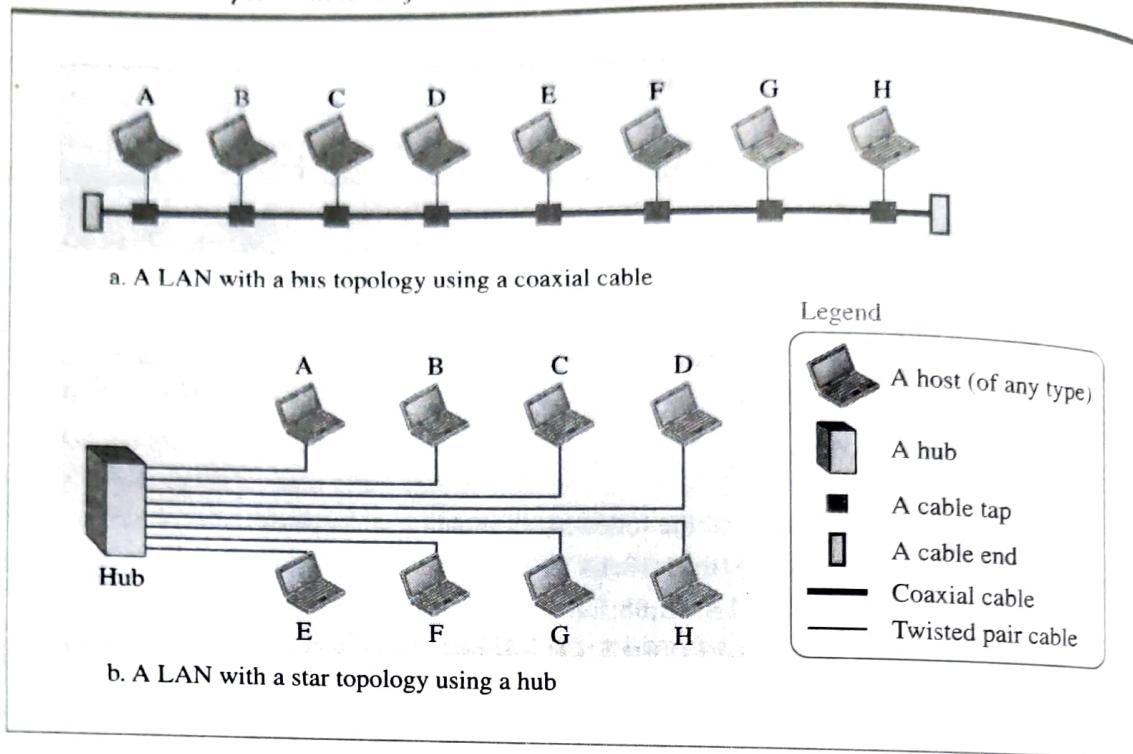
Standard Ethernet uses a coaxial cable (bus topology) or a set of twisted-pair cables with a hub (star topology) as shown in Figure 13.5.

We need to know that transmission in the standard Ethernet is always broadcast, no matter if the intention is unicast, multicast, or broadcast. In the bus topology, when station A sends a frame to station B, all stations will receive it. In the star topology, when station A sends a frame to station B, the hub will receive it. Since the hub is a passive element, it does not check the destination address of the frame; it regenerates the bits (if they have been weakened) and sends them to all stations except station A. In fact, it floods the network with the frame.

The question is, then, how the actual unicast, multicast, and broadcast transmissions are distinguished from each other. The answer is in the way the frames are kept or dropped.

- In a unicast transmission, all stations will receive the frame, the intended recipient keeps and handles the frame; the rest discard it.
- In a multicast transmission, all stations will receive the frame, the stations that are members of the group keep and handle it; the rest discard it.

Figure 13.5 Implementation of standard Ethernet



- In a broadcast transmission, all stations (except the sender) will receive the frame and all stations (except the sender) keep and handle it.

13.2.3 Access Method

Since the network that uses the standard Ethernet protocol is a broadcast network, we need to use an access method to control access to the sharing medium. The standard Ethernet chose CSMA/CD with 1-persistent method, discussed earlier in Chapter 12, Section 1.3. Let us use a scenario to see how this method works for the Ethernet protocol.

- Assume station A in Figure 13.5 has a frame to send to station D. Station A first should check whether any other station is sending (carrier sense). Station A measures the level of energy on the medium (for a short period of time, normally less than 100 μ s). If there is no signal energy on the medium, it means that no station is sending (or the signal has not reached station A). Station A interprets this situation as idle medium. It starts sending its frame. On the other hand, if the signal energy level is not zero, it means that the medium is being used by another station. Station A continuously monitors the medium until it becomes idle for 100 μ s. It then starts sending the frame. However, station A needs to keep a copy of the frame in its buffer until it is sure that there is no collision. When station A is sure of this is the subject we discuss next.
- The medium sensing does not stop after station A has started sending the frame. Station A needs to send and receive continuously. Two cases may occur:

- a. Station A has sent 512 bits and no collision is sensed (the energy level did not go above the regular energy level), the station then is sure that the frame will go through and stops sensing the medium. Where does the number 512 bits come from? If we consider the transmission rate of the Ethernet as 10 Mbps, this means that it takes the station $512/(10 \text{ Mbps}) = 51.2 \mu\text{s}$ to send out 512 bits. With the speed of propagation in a cable ($2 \times 10^8 \text{ meters}$), the first bit could have gone 10,240 meters (one way) or only 5120 meters (round trip), have collided with a bit from the last station on the cable, and have gone back. In other words, if a collision were to occur, it should occur by the time the sender has sent out 512 bits (worst case) and the first bit has made a round trip of 5120 meters. We should know that if the collision happens in the middle of the cable, not at the end, station A hears the collision earlier and aborts the transmission. We also need to mention another issue. The above assumption is that the length of the cable is 5120 meters. The designer of the standard Ethernet actually put a restriction of 2500 meters because we need to consider the delays encountered throughout the journey. It means that they considered the worst case. The whole idea is that if station A does not sense the collision before sending 512 bits, there must have been no collision, because during this time, the first bit has reached the end of the line and all other stations know that a station is sending and refrain from sending. In other words, the problem occurs when another station (for example, the last station) starts sending before the first bit of station A has reached it. The other station mistakenly thinks that the line is free because the first bit has not yet reached it. The reader should notice that the restriction of 512 bits actually helps the sending station: The sending station is certain that no collision will occur if it is not heard during the first 512 bits, so it can discard the copy of the frame in its buffer.
- b. Station A has sensed a collision before sending 512 bits. This means that one of the previous bits has collided with a bit sent by another station. In this case both stations should refrain from sending and keep the frame in their buffer for resending when the line becomes available. However, to inform other stations that there is a collision in the network, the station sends a 48-bit jam signal. The jam signal is to create enough signal (even if the collision happens after a few bits) to alert other stations about the collision. After sending the jam signal, the stations need to increment the value of K (number of attempts). If after increment $K = 15$, the experience has shown that the network is too busy, the station needs to abort its effort and try again. If $K < 15$, the station can wait a backoff time (T_B in Figure 12.13) and restart the process. As Figure 12.13 shows, the station creates a random number between 0 and $2^K - 1$, which means each time the collision occurs, the range of the random number increases exponentially. After the first collision ($K = 1$) the random number is in the range (0, 1). After the second collision ($K = 2$) it is in the range (0, 1, 2, 3). After the third collision ($K = 3$) it is in the range (0, 1, 2, 3, 4, 5, 6, 7). So after each collision, the probability increases that the backoff time becomes longer. This is due to the fact that if the collision happens even after the third or fourth attempt, it means that the network is really busy; a longer backoff time is needed.

13.2.4 Efficiency of Standard Ethernet

The efficiency of the Ethernet is defined as the ratio of the time used by a station to send data to the time the medium is occupied by this station. The practical efficiency of standard Ethernet has been measured to be

$$\text{Efficiency} = 1 / (1 + 6.4 \times a)$$

in which the parameter “ a ” is the number of frames that can fit on the medium. It can be calculated as $a = (\text{propagation delay}) / (\text{transmission delay})$ because the transmission delay is the time it takes a frame of average size to be sent out and the propagation delay is the time it takes to reach the end of the medium. Note that as the value of parameter a decreases, the efficiency increases. This means that if the length of the media is shorter or the frame size longer, the efficiency increases. In the ideal case, $a = 0$ and the efficiency is 1. We ask to calculate this efficiency in problems at the end of the chapter.

Example 13.3

In the Standard Ethernet with the transmission rate of 10 Mbps, we assume that the length of the medium is 2500 m and the size of the frame is 512 bits. The propagation speed of a signal in a cable is normally 2×10^8 m/s.

Propagation delay = $2500 / (2 \times 10^8) = 12.5 \mu\text{s}$	Transmission delay = $512 / (10^7) = 51.2 \mu\text{s}$
$a = 12.5 / 51.2 = 0.24$	Efficiency = 39%

The example shows that $a = 0.24$, which means only 0.24 of a frame occupies the whole medium in this case. The efficiency is 39 percent, which is considered moderate; it means that only 61 percent of the time the medium is occupied but not used by a station.

13.2.5 Implementation

The Standard Ethernet defined several implementations, but only four of them became popular during the 1980s. Table 13.1 shows a summary of Standard Ethernet implementations.

Table 13.1 Summary of Standard Ethernet implementations

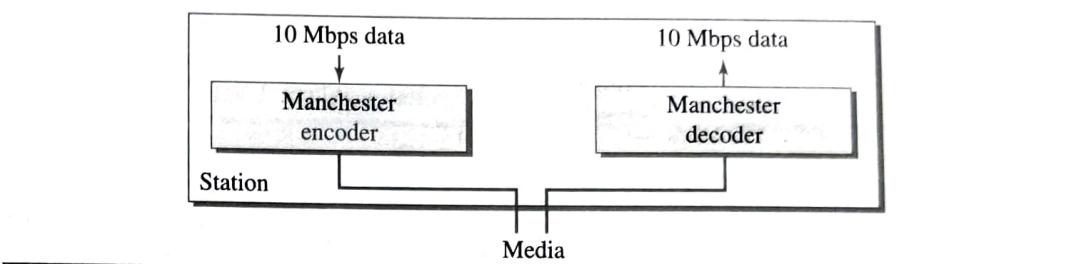
Implementation	Medium	Medium Length	Encoding
10Base5	Thick coax	500 m	Manchester
10Base2	Thin coax	185 m	Manchester
10Base-T	2 UTP	100 m	Manchester
10Base-F	2 Fiber	2000 m	Manchester

In the nomenclature 10BaseX, the number defines the data rate (10 Mbps), the term *Base* means baseband (digital) signal, and X approximately defines either the maximum size of the cable in 100 meters (for example 5 for 500 or 2 for 185 meters) or the type of cable, T for unshielded twisted pair cable (UTP) and F for fiber-optic. The standard Ethernet uses a baseband signal, which means that the bits are changed to a digital signal and directly sent on the line.

Encoding and Decoding

All standard implementations use digital signaling (baseband) at 10 Mbps. At the sender, data are converted to a digital signal using the Manchester scheme; at the receiver, the received signal is interpreted as Manchester and decoded into data. As we saw in Chapter 4, Manchester encoding is self-synchronous, providing a transition at each bit interval. Figure 13.6 shows the encoding scheme for Standard Ethernet.

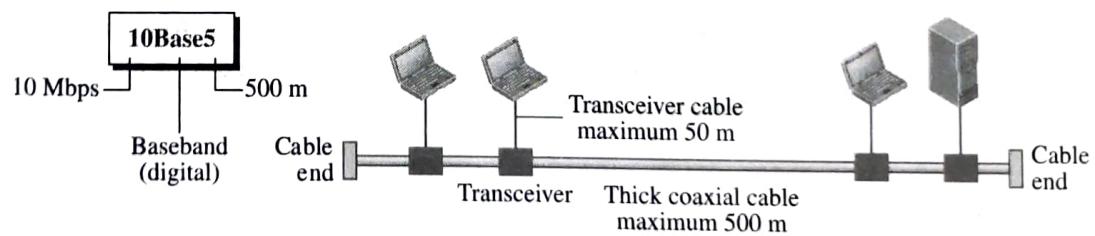
Figure 13.6 Encoding in a Standard Ethernet implementation



10Base5: Thick Ethernet

The first implementation is called **10Base5, thick Ethernet**, or **Thicknet**. The nickname derives from the size of the cable, which is roughly the size of a garden hose and too stiff to bend with your hands. 10Base5 was the first Ethernet specification to use a bus topology with an external **transceiver** (transmitter/receiver) connected via a tap to a thick coaxial cable. Figure 13.7 shows a schematic diagram of a 10Base5 implementation.

Figure 13.7 10Base5 implementation



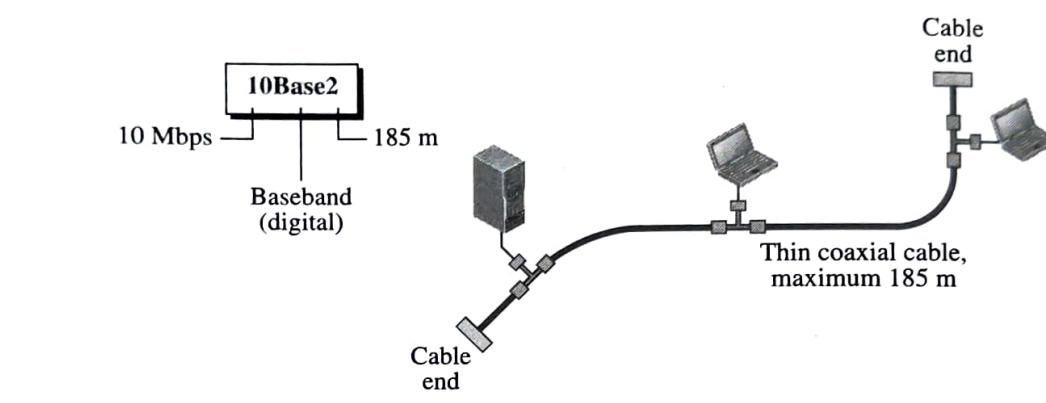
The transceiver is responsible for transmitting, receiving, and detecting collisions. The transceiver is connected to the station via a transceiver cable that provides separate paths for sending and receiving. This means that collision can only happen in the coaxial cable.

The maximum length of the coaxial cable must not exceed 500 m, otherwise, there is excessive degradation of the signal. If a length of more than 500 m is needed, up to five segments, each a maximum of 500 meters, can be connected using repeaters. Repeaters will be discussed in Chapter 17.

10Base2: Thin Ethernet

The second implementation is called **10Base2, thin Ethernet, or Cheapernet**. 10Base2 also uses a bus topology, but the cable is much thinner and more flexible. The cable can be bent to pass very close to the stations. In this case, the transceiver is normally part of the network interface card (NIC), which is installed inside the station. Figure 13.8 shows the schematic diagram of a 10Base2 implementation.

Figure 13.8 10Base2 implementation

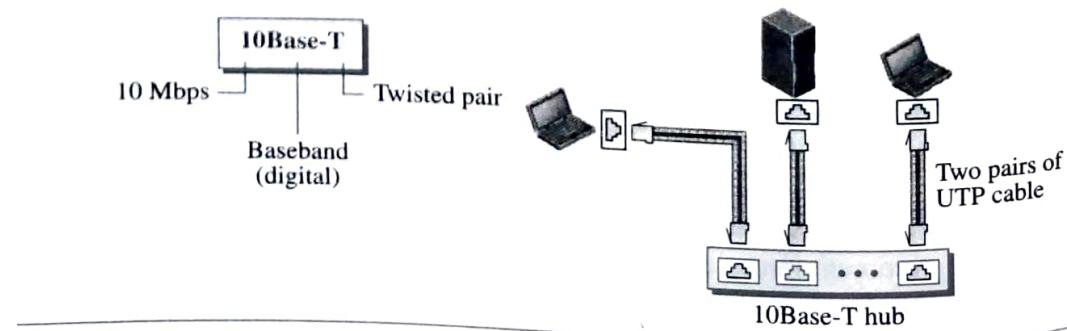


Note that the collision here occurs in the thin coaxial cable. This implementation is more cost effective than 10Base5 because thin coaxial cable is less expensive than thick coaxial and the tee connections are much cheaper than taps. Installation is simpler because the thin coaxial cable is very flexible. However, the length of each segment cannot exceed 185 m (close to 200 m) due to the high level of attenuation in thin coaxial cable.

10Base-T: Twisted-Pair Ethernet

The third implementation is called **10Base-T or twisted-pair Ethernet**. 10Base-T uses a physical star topology. The stations are connected to a hub via two pairs of twisted cable, as shown in Figure 13.9.

Figure 13.9 10Base-T implementation

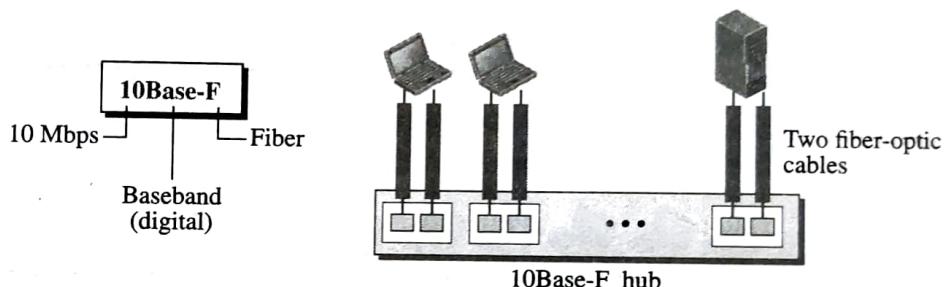


Note that two pairs of twisted cable create two paths (one for sending and one for receiving) between the station and the hub. Any collision here happens in the hub. Compared to 10Base5 or 10Base2, we can see that the hub actually replaces the coaxial cable as far as a collision is concerned. The maximum length of the twisted cable here is defined as 100 m, to minimize the effect of attenuation in the twisted cable.

10Base-F: Fiber Ethernet

Although there are several types of optical fiber 10-Mbps Ethernet, the most common is called **10Base-F**. 10Base-F uses a star topology to connect stations to a hub. The stations are connected to the hub using two fiber-optic cables, as shown in Figure 13.10.

Figure 13.10 10Base-F implementation



13.2.6 Changes in the Standard

Before we discuss higher-rate Ethernet protocols, we need to discuss the changes that occurred to the 10-Mbps Standard Ethernet. These changes actually opened the road to the evolution of the Ethernet to become compatible with other high-data-rate LANs.

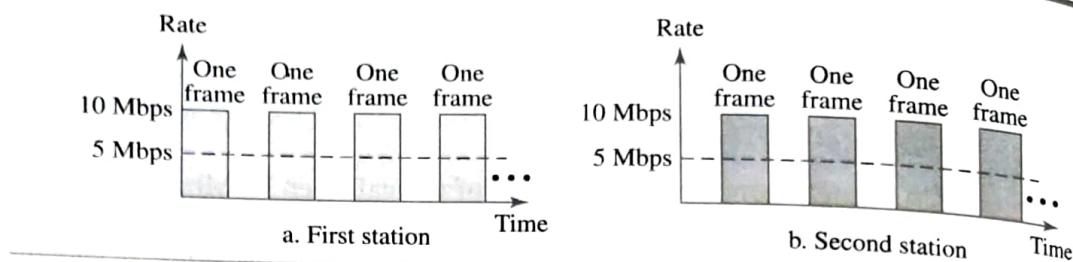
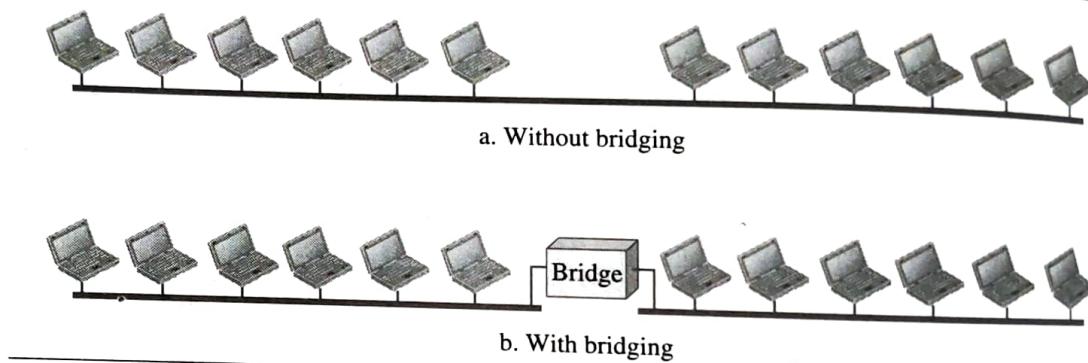
Bridged Ethernet

The first step in the Ethernet evolution was the division of a LAN by **bridges**. Bridges have two effects on an Ethernet LAN: They raise the bandwidth and they separate collision domains. We discuss bridges in Chapter 17.

Raising the Bandwidth

In an unbridged Ethernet network, the total capacity (10 Mbps) is shared among all stations with a frame to send; the stations share the bandwidth of the network. If only one station has frames to send, it benefits from the total capacity (10 Mbps). But if more than one station needs to use the network, the capacity is shared. For example, if two stations have a lot of frames to send, they probably alternate in usage. When one station is sending, the other one refrains from sending. We can say that, in this case, each station on average sends at a rate of 5 Mbps. Figure 13.11 shows the situation.

The bridge, as we will learn in Chapter 17, can help here. A bridge divides the network into two or more networks. Bandwidthwise, each network is independent. For example, in Figure 13.12, a network with 12 stations is divided into two networks, each with 6 stations. Now each network has a capacity of 10 Mbps. The 10-Mbps capacity in each segment is now shared between 6 stations (actually 7 because the bridge acts as a

Figure 13.11 Sharing bandwidth**Figure 13.12** A network with and without a bridge

station in each segment), not 12 stations. In a network with a heavy load, each station theoretically is offered $10/7$ Mbps instead of $10/12$ Mbps.

It is obvious that if we further divide the network, we can gain more bandwidth for each segment. For example, if we use a four-port bridge, each station is now offered $10/4$ Mbps, which is 3 times more than an unbridged network.

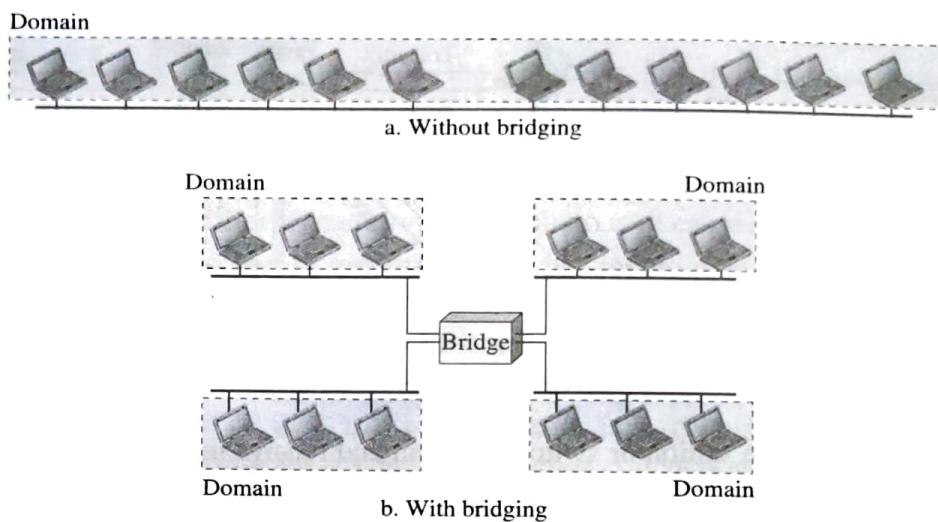
Separating Collision Domains

Another advantage of a bridge is the separation of the **collision domain**. Figure 13.13 shows the collision domains for an unbridged and a bridged network. You can see that the collision domain becomes much smaller and the probability of collision is reduced tremendously. Without bridging, 12 stations contend for access to the medium; with bridging only 3 stations contend for access to the medium.

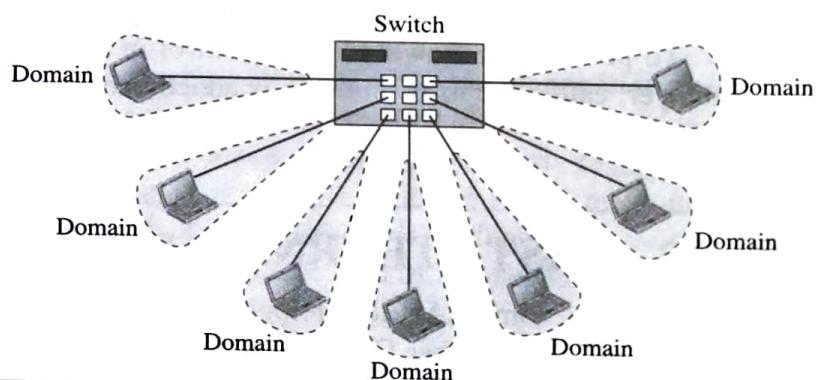
Switched Ethernet

The idea of a bridged LAN can be extended to a switched LAN. Instead of having two to four networks, why not have N networks, where N is the number of stations on the LAN? In other words, if we can have a multiple-port bridge, why not have an N -port switch? In this way, the bandwidth is shared only between the station and the switch (5 Mbps each). In addition, the collision domain is divided into N domains.

A layer-2 **switch** is an N -port bridge with additional sophistication that allows faster handling of the packets. Evolution from a bridged Ethernet to a **switched Ethernet** was

Figure 13.13 Collision domains in an unbridged network and a bridged network

a big step that opened the way to an even faster Ethernet, as we will see. Figure 13.14 shows a switched LAN.

Figure 13.14 Switched Ethernet

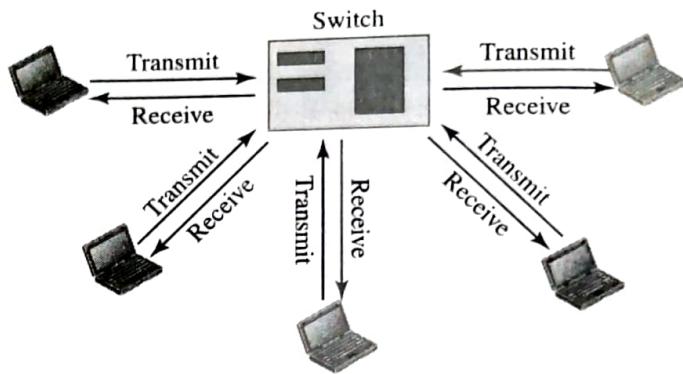
Full-Duplex Ethernet

One of the limitations of 10Base5 and 10Base2 is that communication is half-duplex (10Base-T is always full-duplex); a station can either send or receive, but may not do both at the same time. The next step in the evolution was to move from switched Ethernet to **full-duplex switched Ethernet**. The full-duplex mode increases the capacity of each domain from 10 to 20 Mbps. Figure 13.15 shows a switched Ethernet in full-duplex mode. Note that instead of using one link between the station and the switch, the configuration uses two links: one to transmit and one to receive.

No Need for CSMA/CD

In full-duplex switched Ethernet, there is no need for the CSMA/CD method. In a full-duplex switched Ethernet, each station is connected to the switch via two separate links.

Figure 13.15 Full-duplex switched Ethernet



Each station or switch can send and receive independently without worrying about collision. Each link is a point-to-point dedicated path between the station and the switch. There is no longer a need for carrier sensing; there is no longer a need for collision detection. The job of the MAC layer becomes much easier. The carrier sensing and collision detection functionalities of the MAC sublayer can be turned off.

MAC Control Layer

Standard Ethernet was designed as a connectionless protocol at the MAC sublayer. There is no explicit flow control or error control to inform the sender that the frame has arrived at the destination without error. When the receiver receives the frame, it does not send any positive or negative acknowledgment.

To provide for flow and error control in full-duplex switched Ethernet, a new sublayer, called the *MAC control*, is added between the LLC sublayer and the MAC sublayer.

13.3 FAST ETHERNET (100 MBPS)

In the 1990s, some LAN technologies with transmission rates higher than 10 Mbps, such as FDDI and Fiber Channel, appeared on the market. If the Standard Ethernet wanted to survive, it had to compete with these technologies. Ethernet made a big jump by increasing the transmission rate to 100 Mbps, and the new generation was called the *Fast Ethernet*. The designers of the Fast Ethernet needed to make it compatible with the Standard Ethernet. The MAC sublayer was left unchanged, which meant the frame format and the maximum and minimum size could also remain unchanged. By increasing the transmission rate, features of the Standard Ethernet that depend on the transmission rate, access method, and implementation had to be reconsidered. The goals of Fast Ethernet can be summarized as follows:

1. Upgrade the data rate to 100 Mbps.
2. Make it compatible with Standard Ethernet.
3. Keep the same 48-bit address.
4. Keep the same frame format.

13.3.1 Access Method

We remember that the proper operation of the CSMA/CD depends on the transmission rate, the minimum size of the frame, and the maximum network length. If we want to keep the minimum size of the frame, the maximum length of the network should be changed. In other words, if the minimum frame size is still 512 bits, and it is transmitted 10 times faster, the collision needs to be detected 10 times sooner, which means the maximum length of the network should be 10 times shorter (the propagation speed does not change). So the Fast Ethernet came with two solutions (it can work with either choice):

1. The first solution was to totally drop the bus topology and use a passive hub and star topology but make the maximum size of the network 250 meters instead of 2500 meters as in the Standard Ethernet. This approach is kept for compatibility with the Standard Ethernet.
2. The second solution is to use a link-layer switch (discussed later in the chapter) with a buffer to store frames and a full-duplex connection to each host to make the transmission medium private for each host. In this case, there is no need for CSMA/CD because the hosts are not competing with each other. The link-layer switch receives a frame from a source host and stores it in the buffer (queue) waiting for processing. It then checks the destination address and sends the frame out of the corresponding interface. Since the connection to the switch is full-duplex, the destination address can even send a frame to another station at the same time that it is receiving a frame. In other words, the shared medium is changed to many point-to-point media, and there is no need for contention.

Autonegotiation

A new feature added to Fast Ethernet is called ***autonegotiation***. It allows a station or a hub a range of capabilities. Autonegotiation allows two devices to negotiate the mode or data rate of operation. It was designed particularly to allow incompatible devices to connect to one another. For example, a device with a maximum data rate of 10 Mbps can communicate with a device with a 100 Mbps data rate (but which can work at a lower rate). We can summarize the goal of autonegotiation as follows. It was designed particularly for these purposes:

- To allow incompatible devices to connect to one another. For example, a device with a maximum capacity of 10 Mbps can communicate with a device with a 100 Mbps capacity (but which can work at a lower rate).
- To allow one device to have multiple capabilities.
- To allow a station to check a hub's capabilities.

13.3.2 Physical Layer

To be able to handle a 100 Mbps data rate, several changes need to be made at the physical layer.

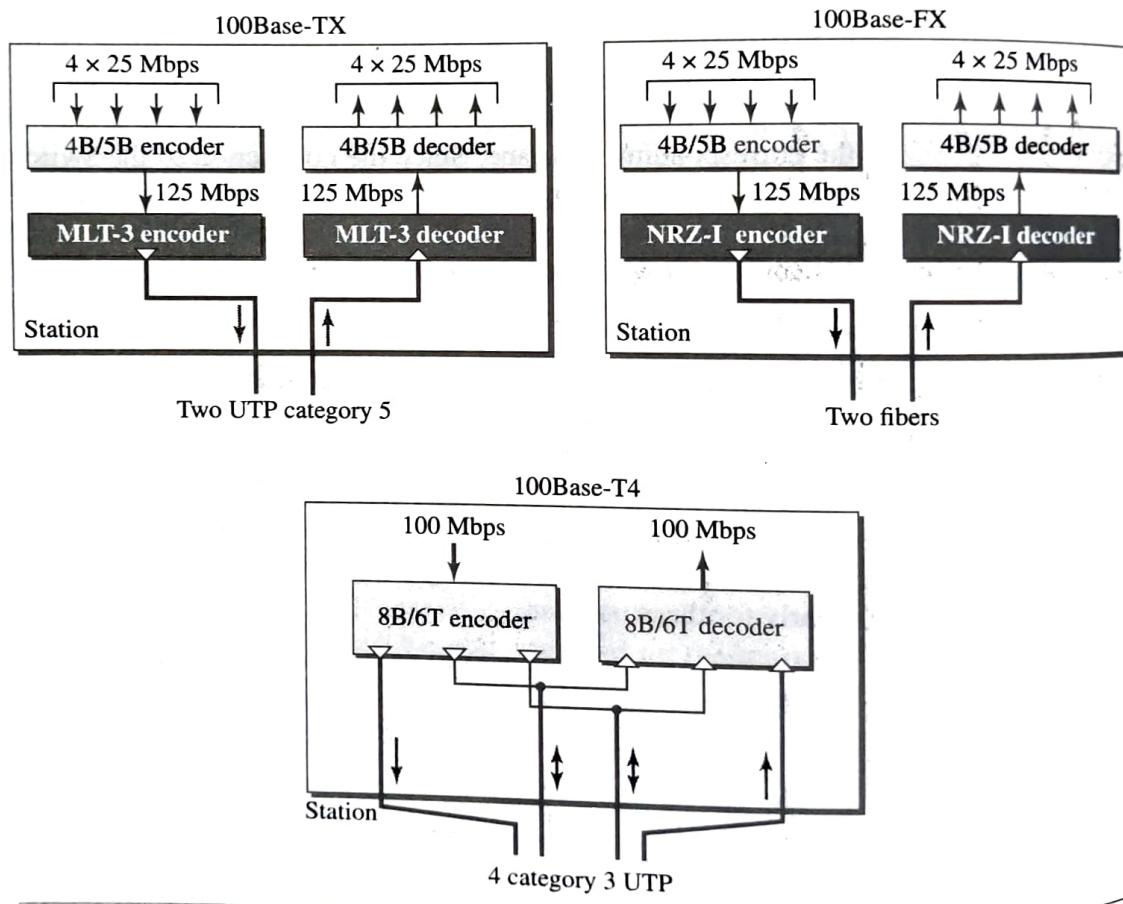
Topology

Fast Ethernet is designed to connect two or more stations. If there are only two stations, they can be connected point-to-point. Three or more stations need to be connected in a star topology with a hub or a switch at the center.

Encoding

Manchester encoding needs a 200-Mbaud bandwidth for a data rate of 100 Mbps, which makes it unsuitable for a medium such as twisted-pair cable. For this reason, the Fast Ethernet designers sought some alternative encoding/decoding scheme. However, it was found that one scheme would not perform equally well for all three implementations. Therefore, three different encoding schemes were chosen (see Figure 13.16).

Figure 13.16 Encoding for Fast Ethernet implementation



100Base-TX uses two pairs of twisted-pair cable (either category 5 UTP or STP). For this implementation, the MLT-3 scheme was selected since it has good bandwidth performance (see Chapter 4). However, since MLT-3 is not a self-synchronous line coding scheme, 4B/5B block coding is used to provide bit synchronization by preventing

the occurrence of a long sequence of 0s and 1s. This creates a data rate of 125 Mbps, which is fed into MLT-3 for encoding.

100Base-FX uses two pairs of fiber-optic cables. Optical fiber can easily handle high bandwidth requirements by using simple encoding schemes. The designers of 100Base-FX selected the NRZ-I encoding scheme (see Chapter 4) for this implementation. However, NRZ-I has a bit synchronization problem for long sequences of 0s (or 1s, based on the encoding), as we saw in Chapter 4. To overcome this problem, the designers used 4B/5B block encoding, as we described for 100Base-TX. The block encoding increases the bit rate from 100 to 125 Mbps, which can easily be handled by fiber-optic cable.

A 100Base-TX network can provide a data rate of 100 Mbps, but it requires the use of category 5 UTP or STP cable. This is not cost-efficient for buildings that have already been wired for voice-grade twisted-pair (category 3). A new standard, called **100Base-T4**, was designed to use category 3 or higher UTP. The implementation uses four pairs of UTP for transmitting 100 Mbps. Encoding/decoding in 100Base-T4 is more complicated. As this implementation uses category 3 UTP, each twisted-pair cannot easily handle more than 25 Mbaud. In this design, one pair switches between sending and receiving. Three pairs of UTP category 3, however, can handle only 75 Mbaud (25 Mbaud) each. We need to use an encoding scheme that converts 100 Mbps to a 75 Mbaud signal. As we saw in Chapter 4, 8B/6T satisfies this requirement. In 8B/6T, eight data elements are encoded as six signal elements. This means that 100 Mbps uses only $(6/8) \times 100$ Mbps, or 75 Mbaud.

Summary

Fast Ethernet implementation at the physical layer can be categorized as either two-wire or four-wire. The two-wire implementation can be either shielded twisted pair (STP), which is called *100Base-TX*, or fiber-optic cable, which is called *100Base-FX*. The four-wire implementation is designed only for unshielded twisted pair (UTP), which is called *100Base-T4*. Table 13.2 is a summary of the Fast Ethernet implementations. We discussed encoding in Chapter 4.

Table 13.2 Summary of Fast Ethernet implementations

Implementation	Medium	Medium Length	Wires	Encoding
100Base-TX	UTP or STP	100 m	2	4B5B + MLT-3
100Base-FX	Fiber	185 m	2	4B5B + NRZ-I
100Base-T4	UTP	100 m	4	Two 8B/6T

13.4 GIGABIT ETHERNET

The need for an even higher data rate resulted in the design of the Gigabit Ethernet Protocol (1000 Mbps). The IEEE committee calls it the Standard 802.3z. The goals of the Gigabit Ethernet were to upgrade the data rate to 1 Gbps, but keep the address length, the frame format, and the maximum and minimum frame length the same. The goals of the Gigabit Ethernet design can be summarized as follows:

1. Upgrade the data rate to 1 Gbps.
2. Make it compatible with Standard or Fast Ethernet.

3. Use the same 48-bit address.
4. Use the same frame format.
5. Keep the same minimum and maximum frame lengths.
6. Support autonegotiation as defined in Fast Ethernet.

13.4.1 MAC Sublayer

A main consideration in the evolution of Ethernet was to keep the MAC sublayer untouched. However, to achieve a data rate of 1 Gbps, this was no longer possible. Gigabit Ethernet has two distinctive approaches for medium access: half-duplex and full-duplex. Almost all implementations of Gigabit Ethernet follow the full-duplex approach, so we mostly ignore the half-duplex mode.

Full-Duplex Mode

In full-duplex mode, there is a central switch connected to all computers or other switches. In this mode, for each input port, each switch has buffers in which data are stored until they are transmitted. Since the switch uses the destination address of the frame and sends a frame out of the port connected to that particular destination, there is no collision. This means that CSMA/CD is not used. Lack of collision implies that the maximum length of the cable is determined by the signal attenuation in the cable, not by the collision detection process.

In the full-duplex mode of Gigabit Ethernet, there is no collision; the maximum length of the cable is determined by the signal attenuation in the cable.

Half-Duplex Mode

Gigabit Ethernet can also be used in half-duplex mode, although it is rare. In this case, a switch can be replaced by a hub, which acts as the common cable in which a collision might occur. The half-duplex approach uses CSMA/CD. However, as we saw before, the maximum length of the network in this approach is totally dependent on the minimum frame size. Three methods have been defined: traditional, carrier extension, and frame bursting.

Traditional

In the traditional approach, we keep the minimum length of the frame as in traditional Ethernet (512 bits). However, because the length of a bit is 1/100 shorter in Gigabit Ethernet than in 10-Mbps Ethernet, the slot time for Gigabit Ethernet is $512 \text{ bits} \times 1/1000 \mu\text{s}$, which is equal to $0.512 \mu\text{s}$. The reduced slot time means that collision is detected 100 times earlier. This means that the maximum length of the network is 25 m. This length may be suitable if all the stations are in one room, but it may not even be long enough to connect the computers in one single office.

Carrier Extension

To allow for a longer network, we increase the minimum frame length. The **carrier extension** approach defines the minimum length of a frame as 512 bytes (4096 bits). This means that the minimum length is 8 times longer. This method forces a station to add

extension bits (padding) to any frame that is less than 4096 bits. In this way, the maximum length of the network can be increased 8 times to a length of 200 m. This allows a length of 100 m from the hub to the station.

Frame Bursting

Carrier extension is very inefficient if we have a series of short frames to send; each frame carries redundant data. To improve efficiency, **frame bursting** was proposed. Instead of adding an extension to each frame, multiple frames are sent. However, to make these multiple frames look like one frame, padding is added between the frames (the same as that used for the carrier extension method) so that the channel is not idle. In other words, the method deceives other stations into thinking that a very large frame has been transmitted.

13.4.2 Physical Layer

The physical layer in Gigabit Ethernet is more complicated than that in Standard or Fast Ethernet. We briefly discuss some features of this layer.

Topology

Gigabit Ethernet is designed to connect two or more stations. If there are only two stations, they can be connected point-to-point. Three or more stations need to be connected in a star topology with a hub or a switch at the center. Another possible configuration is to connect several star topologies or let one star topology be part of another.

Implementation

Gigabit Ethernet can be categorized as either a two-wire or a four-wire implementation. The two-wire implementations use fiber-optic cable (**1000Base-SX**, short-wave, or **1000Base-LX**, long-wave), or STP (**1000Base-CX**). The four-wire version uses category 5 twisted-pair cable (**1000Base-T**). In other words, we have four implementations. 1000Base-T was designed in response to those users who had already installed this wiring for other purposes such as Fast Ethernet or telephone services.

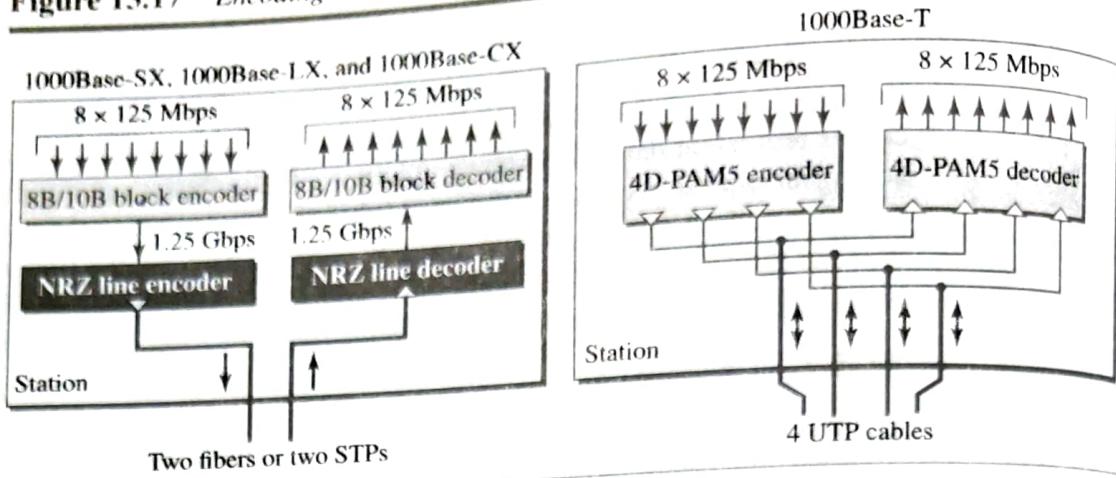
Encoding

Figure 13.17 shows the encoding/decoding schemes for the four implementations. Gigabit Ethernet cannot use the Manchester encoding scheme because it involves a very high bandwidth (2 Gbaud). The two-wire implementations use an NRZ scheme, but NRZ does not self-synchronize properly. To synchronize bits, particularly at this high data rate, 8B/10B block encoding, discussed in Chapter 4, is used.

This block encoding prevents long sequences of 0s or 1s in the stream, but the resulting stream is 1.25 Gbps. Note that in this implementation, one wire (fiber or STP) is used for sending and one for receiving.

In the four-wire implementation it is not possible to have 2 wires for input and 2 for output, because each wire would need to carry 500 Mbps, which exceeds the capacity for category 5 UTP. As a solution, 4D-PAM5 encoding, as discussed in Chapter 4, is used to reduce the bandwidth. Thus, all four wires are involved in both input and output; each wire carries 250 Mbps, which is in the range for category 5 UTP cable.

Figure 13.17 Encoding in Gigabit Ethernet implementations



Implementation Summary

Table 13.3 is a summary of the Gigabit Ethernet implementations. S-W and L-W mean short-wave and long-wave respectively.

Table 13.3 Summary of Gigabit Ethernet implementations

Implementation	Medium	Medium Length	Wires	Encoding
1000Base-SX	Fiber S-W	550 m	2	8B/10B + NRZ
1000Base-LX	Fiber L-W	5000 m	2	8B/10B + NRZ
1000Base-CX	STP	25 m	2	8B/10B + NRZ
1000Base-T4	UTP	100 m	4	4D-PAM5

13.5 10 GIGABIT ETHERNET

In recent years, there has been another look into the Ethernet for use in metropolitan areas. The idea is to extend the technology, the data rate, and the coverage distance so that the Ethernet can be used as LAN and MAN (metropolitan area network). The IEEE committee created 10 Gigabit Ethernet and called it Standard 802.3ae. The goals of the 10 Gigabit Ethernet design can be summarized as upgrading the data rate to 10 Gbps, keeping the same frame size and format, and allowing the interconnection of LANs, MANs, and WAN possible. This data rate is possible only with fiber-optic technology at this time. The standard defines two types of physical layers: LAN PHY and WAN PHY. The first is designed to support existing LANs; the second actually defines a WAN with links connected through SONET OC-192.

13.5.1 Implementation

10 Gigabit Ethernet operates only in full-duplex mode, which means there is no need for contention; CSMA/CD is not used in 10 Gigabit Ethernet. Four implementations are the most common: **10GBase-SR**, **10GBase-LR**, **10GBase-EW**, and **10GBase-X4**. Table 13.4 shows a summary of the 10 Gigabit Ethernet implementations. We discussed the encoding in Chapter 4.