ETHERNET

Ethernet is a widely used local area network (LAN) technology that allows multiple end stations (such as desktop computers, servers, printers, gateways to other networks, etc.) to exchange data among themselves within a single building or campus environment. The sending station segments the data into a sequence of frames, each of which is sent independently through the network to the destination(s). Every frame carries a globally unique 48-bit source and destination address and other information, laid out according to a standard format. However, the length of a frame can vary between a minimum of 64 bytes and a maximum of 1518 bytes. By design, Ethernet provides only a "best effort" delivery service: The network will not reorder or duplicate frames, but there is no guarantee that a particular frame will reach the destination. Applications must run a reliable transport protocol, such as TCP/IP, on top of the Ethernet service to guarantee delivery.

ETHERNET COMPONENTS

A typical Ethernet system is shown in Fig. 1. Each end station contains a *network interface,* which contains some temporary storage for frames being sent or received from the network, along with logic for executing the medium access control (MAC) algorithm, calculating the cyclic redundancy code (CRC) for error detection, and performing related functions. In many cases, the network interface is on a small card or printed circuit board that can be added to an end station if network connectivity is required. The network interface uses a *transceiver* to perform the actual data transmission and reception over the physical *link.* Initially, Ethernet used external transceivers, attached to the network interface by an *attachment unit interface* (AUI) cable. Today, however, the transceiver is often integrated with the network interface card. In some cases, changeable transceiver types may be plugged into the card through a *medium independent interface* (MII).

A variety of link types have been defined, including coaxial cable, unshielded twisted pair (UTP) cabling, and both multimode and single-mode optical fiber. Coaxial cable is restricted to 10 Mbit/s operation. If UTP cabling meets Category 5 requirements, then operation at 100 Mbit/s can be supported via 100BASE-TX and it is expected that operation at 1 Gbit/s will be supported in the future via the 1000BASE-T standard now being developed. Optical fiber can support speeds up to 1 Gbit/s. Multiple transceivers can be connected to a single coaxial cable segment (up to 100 stations per segment of "thick" cable in 10BASE5, and up to 30 stations per segment of 50 Ω "thin" RG58 cable in 10BASE2). Coaxial cable segments are inherently half duplex because the electrical signals travel in both directions away from the transmitters along a single metallic conductor, passing the transceivers belonging to all other stations before being absorbed by terminating resistors at the ends of the segment. On the other hand, UTP and fiber-optic segments can support full-duplex transmission because each segment uses a separate signaling path to carry data in each direction between two transceivers at its endpoints.

Larger networks are constructed by joining multiple segments together using active electronic devices that relay data from one of the attached segments to the other(s). A *repeater* (or "hub") immediately copies all bits arriving on each segment to all other segments, whether or not they are part of a valid frame. Segments joined together by repeaters form a single *collision domain.* If more than one end station within the given collision domain transmit frames at the same time, the data will get garbled together to form a *collision,* which cannot be understood by any of the receivers. A *bridge* (or "switch") copies frames that arrive on each segment to those segments that might contain its destination(s). Multicast, broadcast, and frames addressed to an unrecognized destination are copied to all other segments, while the rest are only copied to the segment that contains the destination station. Segments joined together by bridges form a single *broadcast domain.*

ETHERNET OPERATION

High-Level Service Interface

Ethernet provides a service interface to each end station that consists of independent, asynchronously operating frame transmitter and receiver functions. These functions are invoked by the higher-layer protocols (such as TCP/IP) in the end station. To send some data, the station creates a higher-layer datagram and passes it to the Ethernet transmit function. After the transmit function returns the outcome (success or failure) of this request, the station can call the transmit function again with another frame. The transmit function begins by converting the higher-layer datagram to an Ethernet frame by adding a 64-bit preamble and start-frame delimiter,

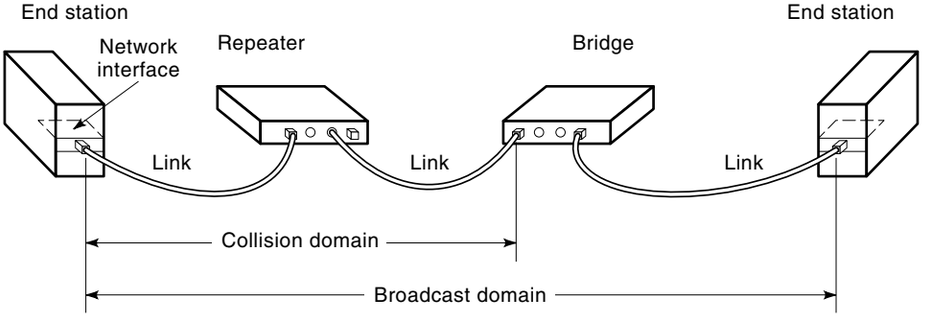

Figure 1. Typical Ethernet system.

a 48-bit source and destination addresses, and a 16-bit length/type value to the beginning and then adding a 32-bit frame check sequence computed with the CRC-32 polynomial to the end. Then the function attempts to transmit the frame over the outgoing link according to the rules of the medium access control (MAC) protocol. The transmit function reports successful delivery if it is able to transmit the entire frame without ever detecting a collision, and it reports failure if every one of the 16 allowable attempts to transmit the frame resulted in collisions. To receive some data, the station calls the Ethernet frame receiver function, and then it waits until the function returns with the next incoming datagram. Once activated, the receive function scans the incoming bit stream until it finds a valid preamble and start-frame delimiter, and then it gathers the rest of the incoming bits until the end of the transmission to form a candidate frame. If the candidate is shorter than the minimum frame length, then it is deemed to be a collision fragment and is discarded. If the candidate does not have a valid CRC, then it is discarded because of bit errors. Once a candidate frame has passed all the validation tests, its destination address is compared with the address of this end station and a list of recognized multicast and broadcast addresses. If there is an address match, then the receive function strips off the Ethernet encapsulation and returns the enclosed datagram to the station. Otherwise, the frame is discarded by the address filter and the receive function resumes looking for another frame in the incoming bit stream.

Medium Access Control

Ethernet uses a MAC algorithm called *Carrier Sense Multiple Access with Collision Detection* (CSMA/CD) to control the transmission of frames. CSMA/CD is a distributed algorithm for serializing the transmissions by multiple end stations over a shared channel. When the end station requests the transmission of a frame, the MAC layer frame transmitter starts executing a sequence of trial-and-error steps, as determined by network activity that is reported by its transceiver. In particular, the transceiver sets the *carrierSense* control signal whenever there are any data present on link, and it sets the *collisionDetect* control signal if it determines that the data originated from more than one transmitter. Originally, for the case of coaxial cables, both carrierSense and collisionDetect were obtained by *analog* logic. For example, in 10BASE5 networks, the signal levels used by the transceiver are offset from zero, so that each transmitter acts as a constant 41 mA current source acting on the two 50 Ω termination resistors connected in parallel. In this case, an analog voltmeter will read approximately 1 V when there are data present on the link, and a voltage threshold of approximately 1.5 V can be used to identify a collision. Thus, coaxial cable segments can support *receive mode collision detection,* which means that a transceiver can report collisions among third-party end stations. However, the Ethernet MAC algorithm does not use this feature. For other network types, such as UTP cabling and optical fiber, data are carried unidirectionally over a pair of physical links and both carrierSense and collisionDetect can be obtained using *digital* logic. A transceiver sets carrierSense if there are data present on the transmit or receive links, and it sets collisionDetect if data are present on both links.

Each transmitter is required to leave a 96-bit *interframe gap* between the last previous data on the link and the start of its own transmission. This provides some time for the receivers to handle one frame before the arrival of the next. The interframe gap is controlled by the *deferring* control signal, which becomes true when start-of-carrier is detected but becomes false 96 bit-times after end-of-carrier is detected. If carrierSense returns during the interframe gap part 1, then it is assumed to be caused by analog effects inside the transceiver or the arrival of additional fragments within the same collision event, so the 96-bit interframe gap timer is restarted at the next end-of-carrier. However, if carrierSense returns during the interframe gap part 2, then it is ignored and the interframe gap timer continues to run. The switch from part 1 to part 2 can occur at any time during the first 64 bit-times of the interframe gap. The reason for having part 2 is to prevent a station whose interframe gap timer runs too fast from enjoying an unfair advantage over the other stations. Suppose station A's clock ran 1% faster than station B's clock. If station A were to transmit a burst of consecutive frames, then station B's interframe gap timer would never expire and station B would be blocked from accessing the channel until station A had transmitted its entire burst. However, if station B were to transmit the same burst, then station A's interframe gap time would expire after every frame, allowing it to compete with B for access to the channel.

Under *half-duplex mode,* the MAC layer transmitter schedules the first attempt to transmit a frame immediately, if deferring is false, or as soon as the deferring control signal becomes false, otherwise. Obviously, this *persistent* strategy minimizes the delay between the request by the higher-layer protocol and the first attempt. However, it also means that the transmission of large frames are often followed by collisions on a busy network, as multiple stations wait for the same end-of-carrier event to trigger their respective attempts. If the entire frame is transmitted without triggering the collisionDetect control signal, then it is assumed that the frame reached its destination and its successful delivery is reported to the higher protocols at the station. Otherwise, the unsuccessful attempt must be aborted, and possibly rescheduled at a later time. When an attempt is aborted, the transmitter first finishes the 64-bit preamble and start-frame delimiter if necessary, and then it substitutes a 32-bit *jam sequence* in place of the remaining bits in the frame. In general, the jam can be any bit sequence as long as it is not intentionally chosen to be a valid CRC. Thus, in general a *collision fragment* starts with a normal preamble and start-frame delimiter (part of which may be garbled), followed by a string of bits that is at least as large as a 32-bit jam sequence and shorter than the minimum frame transmission time, so it can be easily discarded by the receivers using a length threshold.

The *slot time,* which is defined as 512 bit times for networks operating at speeds below 1 Gbps and 4096 bit times (or 512 *bytes*) for Gigabit Ethernet, is a key parameter for half-duplex operation. In order for each transmitter to reliably detect collisions, the minimum transmission time for a complete frame must be at least a slot time, whereas the round-trip propagation delay (including both logic delays in all electronic components and the propagation delay in all links) must be less than a slot time. Thus, none of the affected stations could have finished its transmission before detecting the collision. Similarly, the receivers must be able to identify

(and discard) incoming collision fragments using the fact that their length, after removing the preamble and start-frame delimiter, is less than a slot time. As a result of all these requirements, it can be shown that the round-trip delay in a half-duplex Ethernet collision domain must be less than a slot time minus the jam length. To see this, suppose the round-trip delay between stations A and B is τ bit times, and station A starts transmitting at time 0, while station B waits until time $\tau/2$ (when the data from A are about to arrive) before starting its own transmission. In this case, station B detects the collision immediately, but must still transmit a 96-bit minimum size collision fragment from $\tau/2$ to $\tau/2 + 96$. Meanwhile, station A detects the collision at τ, sends its jam signal, and stops transmitting at $\tau + 32$. In this case, a receiver adjacent to station A would have received a total of $\tau + 96$ bits between the start of A's preamble and the end of B's jam. Thus, after removing the 64-bit preamble and start-frame delimiter the receiver would be left with a $\tau + 32$-bit collision fragment.

After each collision, the *binary exponential backoff* (BEB) algorithm is used to schedule the next retransmission (if any) of the affected frame. Let *attempts* be the number of times this frame has already been transmitted. If attempts = 16, then the frame is dropped with an *excessive collision error.* Otherwise, the BEB algorithm generates a random integer r in the range $(0, 2^{\min\{\text{attempts}, 10\}} - 1)$ and instructs the transmitter to sleep for r slot times before making its next attempt. By selecting backoff delays that are multiples of the slot time, colliding stations that pick different delays will not collide with each other again. BEB adjusts the range after each attempt, in an effort to provide an average of one distinct backoff slot per transmitter. Initially, the BEB algorithm only knows that its own station has a frame, so it (greedily) selects a zero backoff before the first attempt. Thereafter, if a collision occurs after randomly selecting one of N slots, BEB raises its estimate to $2N$ active stations, based on the number of transmitters that selected its own slot. The doubling stops when $N = 1024$, since that is the maximum number of stations allowed in a single collision domain according to the Ethernet standard. It is important to note that each station's transmit function runs an *independent* copy of the BEB algorithm, which it restarts from the beginning for each new frame. Consequently, the backoff delays selected by different stations following the same collision can become very lopsided, resulting in an unfairness problem known as the *capture effect.* For example, suppose stations A and B collide at time 0, and it is the first attempt by A to transmit its packet and is the Kth attempt by B to transmit the other packet. In this case, A's range of backoff delays is 2^{K-1} times smaller than B's range, so A is very likely to retransmit in an earlier slot. Moreover, if A decides to transmit more frames, A's *first attempt* to transmit a *new* frame will collide with B's $(K + 1)$st attempt to transmit the *same* frame, so it is even more likely that A will retransmit its packet in an earlier slot. Because of the capture effect, a single station on a busy network can transmit large numbers of consecutive frames while many other stations are unable to transmit any packets.

Half-duplex operation in Gigabit Ethernet is more complex to allow the slot time to be increased to 4096 bit times (because higher-speed operation increases the bandwidth–delay product on a fixed diameter network), without changing the existing 512-bit minimum frame size or one-frame-at-a-time service interface. This was accomplished by introducing two new features. First, the minimum transmission time for a short frame was increased to 4096 bit times by appending *extended carrier symbols* to the end of the frame, if necessary. Extended carrier is a new code word that is neither a data bit nor an idle symbol. If a collision occurs before the end of the extended carrier has been sent, then the transmitter treats the attempt like a normal collision and retransmits the frame after a random backoff delay. Second, a technique called *frame bursting* was introduced to improve efficiency with short frames. In this case, once a station has successfully transmitted one frame, it is permitted to maintain control of the channel while it sends some additional frames by filling the interframe gap with extended carrier symbols instead of idle symbols. The station can keep adding more frames to the burst until it either runs out of frames or exceeds a total burst length of 65,536 bit times.

Full-duplex mode can be used on point-to-point segments that use separate signaling paths for each direction, such as UTP cabling and optical fiber. Under full-duplex operation, the MAC layer transmitter and receiver functions operate independently of each other. Thus, the collisionDetect control signal is never true, and the carrierSense control signal gets split into two signals: receiveDataValid indicates that data are present on the incoming link, while carrierSense indicates that data are present on the outgoing link and are only used to calculate the interframe gap. Full-duplex operation also includes an optional flow control method using *pause frames.* One end station can temporarily stop all traffic from the other end station (except control frames) by sending a pause frame. The duration of the pause (in multiples of a 512-bit time delay quantum) is controlled by a 16-bit parameter. Traffic resumes when the specified number of bit times has elapsed. If an additional pause frame arrives before the current pause time has expired, its parameter *replaces* the current pause time, so a pause frame with parameter zero allows traffic to resume immediately.

Repeater Operation

Repeaters may be attached to the same link types as end stations, but do not contain a MAC layer entity. Instead, a simple finite-state machine is used to control the forwarding of bits among its ports. If one port has incoming data, the data are sent to all other ports. If more than one port has incoming data, a jam signal is sent to all ports. Although some of the preamble bits may be lost while a transceiver is synchronizing with an incoming frame, the repeater is required to transmit the full 64-bit preamble and start-frame delimiter on each output port. Thus, the interframe gap between two consecutive frames can change every time they pass through a repeater. To limit the amount by which the interframe gap can shrink, a maximum of four repeaters is permitted in the path between any pair of end stations in the same collision domain. Repeaters also play a role in improving the robustness of large networks by automatically partitioning misbehaving ports from the rest of the network. For example, port partitioning will be triggered if an incoming data bit from a segment continues well beyond the maximum frame length (a condition known as "jabber"), or if many consecutive transmissions to that segment result in collisions (an indication

that a single frame might be colliding with itself after traveling around a loop).

Physical Layer Data Encoding

Ethernet uses a variety of physical layer data encoding schemes, depending on the link speed and type. In particular, 10 Mbit/s Ethernet uses *Manchester encoding,* which is a two-level encoding scheme using a baud rate of twice the bit rate, to distribute both data bits and the clock from the sender to the receiver(s). Each data bit is represented by a pair of channel symbols: either "HI" followed by "LO" (i.e., nominally 0 V followed by −2.05 V on a coaxial cable link) to send a logical "0" data bit, or "LO" followed by "HI" to send a logical "1" data bit. In this way, the receiver(s) can easily synchronize with the data stream and recover the incoming data based on the direction of the transition at the midpoint of each bit. A transceiver with no data to send is required to generate an idle pattern consisting of a constant string of "HI" symbols (i.e., 0 V), which allows multiple transceivers to be connected to a single link without interfering with each other (except through collisions). However, this approach also means that the receivers cannot distinguish between an *idle* link and a *broken* link, which limits the fault detection capabilities of the system. Thus, the 10 Mbit/s fiber-optic inter-repeater link (FOIRL) introduced a *Link Integrity Test* to provide some fault detection on each of its dedicated signaling paths. The same Link Integrity Test was also used in 10BASE-T. An idle transmitter must send a short burst of energy called a *link test pulse* once every 16 ms. If the corresponding receiver has not seen either data or a link test pulse for at least 50 ms, then it declares the link to be broken.

Manchester encoding is very simple and robust, but it is unsuitable for higher-speed operation because its high baud rate means that the link must carry frequencies much higher than the bit rate. Thus Fast Ethernet has adopted the same 4B/5B encoding used by the Fiber Distributed Data Interface (FDDI) for transmission over Category 5 UTP (via the 100BASE-TX standard) and optical fiber (via the 100BASE-FX standard). Under 4B/5B, each 4-bit "nibble" of data sent over the MII is converted into a 5-bit code word for transmission over the physical link. Two signaling levels are used, so the baud rate is 20% higher than the bit rate. Since there are twice as many 5-bit code words available compared to the number of distinct 4-bit data nibbles, the code words can be chosen in such a way that the encoder never outputs more than three consecutive logical "0" bits. Thus, since the transceiver indicates logical "0" and "1" bits by generating no change or a reversal of the current signal level, respectively, the receiver(s) will see at least one transition every 3 bit times, allowing them synchronize with the incoming signal and recover both data bits and the clock. (100BASE-TX also randomizes the output of the encoder, so particular data sequences don't create repeating signaling patterns that might cause high levels of electromagnetic interference.) Fast Ethernet also differs from the earlier designs by taking advantage (at the physical layer) of the fact that only point-to-point transmission over dedicated links will be used. In particular, the Link Integrity Test from 10BASE-T is not needed because all transmitters are always on. As soon as a device is turned on, its transceiver establishes a low-level connection with its peer at the other end of the link. If it has no data to send, the transmitter sends one of the unused 4B/5B code words, which has been reserved to indicate that the link is in the idle state, in order to maintain clock synchronization. Additional code words are used as delimiters to mark the start and end of each MAC frame, so the preamble and start-frame delimiter are reduced to framing overhead that no longer serves any real purpose.

Gigabit Ethernet yet again introduces some different encoding schemes. For transmission over optical fiber (via the 1000BASE-SX and 1000BASE-LX standards), the same 8B/10B encoding used in Fiber Channel is used. In this case, every 8-bit data byte transferred across the GMII is mapped into a 10-bit code word for transmission over the physical channel. Since there are four code words available for each data byte, only those code words with sufficient transitions to permit clock recovery at the receiver are used. Moreover, the 8B/10B code also maintains direct current (*dc*) *balance* over the long term in the following way. The *running disparity* of the data stream is defined as the difference between the total number of logical "1" bits minus the number of logical "0" bits transmitted. If the running disparity is positive, then one set of code words will be used; otherwise, another set of code words will be used. At least half the bits in every code word belonging to the first set are logical "0" bits, while at least half the bits in every code word belonging to the second set are logical "1" bits. At the time of this writing, the details of the encoding scheme for 1000 Mbit/s operation over Category 5 UTP cabling (via the proposed 1000BASE-T standard) was still under development. It is expected that a full-duplex link will be created by transmitting *simultaneously and in both directions* over all four pairs in a UTP cable. Each combination of an 8-bit data byte and an alternating clock bit is mapped into a code word that assigns one of five possible voltage levels to each of the four pairs in the UTP cable, giving a total of $5^4 = 625$ possible code words to represent 512 different combinations. Some of the unused code words are used for "nondata" control symbols, such as idle, extended carrier, and start-frame or end-frame delimiters. Notice that each pair in the UTP cable need only carry data symbols at the same baud rate as Fast Ethernet (i.e., 125 Mbit/s). However, since a five-level encoding is more prone to errors than a two-level encoding, a trellis decoder is used to reduce the bit error rate.

Autonegotiation of Link Capabilities

Over time, the Ethernet standard has been updated many times to support advances in technology. Initially, these changes were made to allow new media types (e.g., "thin" coaxial cable, optical fiber, UTP cabling, etc.) to be used with the existing Ethernet standard for 10 Mbit/s operation. However, following the introduction of Fast Ethernet, the same Category 5 UTP cabling and RJ-45 connectors could be used for transmitting at 10 Mbit/s (according to the 10BASE-T standard) or at 100 Mbit/s (using any one of the 100BASE-TX, 100BASE-T4, or 100BASE-T2 standards). Eventually, transmission at 1 Gbps will also be possible, when the proposed 1000BASE-T standard is completed. In addition, UTP cabling is also compatible with full-duplex operation and its optional flow control scheme. Therefore, it is now possible to design a standards-compliant Ethernet device that plugs into Category 5 UTP cabling and operates according to one of more

than a dozen different "modes," and many vendors have designed products that support several of these modes (e.g., 10/100 Mbit/s network interface cards). Unfortunately, this variety of operating modes also means that two standards-compliant Ethernet devices that are designed to use the same UTP cabling can't communicate unless they also use the same mode. Since manual configuration is tedious and error-prone, the Ethernet standard includes a method that allows the attached devices to automatically select their *highest common operating mode;* this method is called *Autonegotiation*.

When a device that supports autonegotiation is initialized, it transmits a *fast link pulse* (FLP) over the attached link every 16 ms. Each FLP looks like a normal link test pulse to existing 10BASE-T devices that do not support autonegotiation. However, the FLP is actually a burst of much shorter pulses that encodes a 16-bit "page" of information about the capabilities of the sending device. The encoding for a page consists of 33 pulse positions, each 125 μs apart. The odd pulse positions, which are always present, are used for clock recovery. The data are carried by the even pulse positions, where the presence of a pulse indicates a logical "1" data bit while absence of a pulse indicates a logical "0" data bit. The data in the page are used to advertise the set of capabilities supported by the sending device, and they also include an *acknowledgment bit* to indicate successful reception of the page being sent in the opposite direction (i.e., the data contained in at least three incoming FLPs were the same) and a *next page bit* to indicate there are more data to come after this page has been received.

Once both devices have finished exchanging their respective pages of data, the link is established using their highest common operating mode (if one exists) using a fixed priority list that gives preference to full-duplex operation over half-duplex operation and also gives preference to higher speeds over lower speeds. After the link has been established, no more FLPs are sent: Higher-speed operation uses an active idle pattern without any link test pulses, and if 10BASE-T is selected, then conventional link test pulses will be used.

ETHERNET SYSTEM DESIGN ISSUES

The design of Ethernet systems is limited by several factors. First, there is a maximum distance for each link type, which is determined by the given combination of transceiver type and medium. This distance limit is determined by the physical characteristics of the channel, as well as by how the signal quality changes as a function of distance due to such factors as attenuation (i.e., the signal level at the receiver is too low to be distinguished from background noise) and dispersion (i.e., successive code words blend together because of variability in the signal velocity and cannot be distinguished by the receiver). These factors determine the maximum length of the coaxial cable segments in 10BASE5 and 10BASE2, and they limit the maximum length of a UTP cable segment to 100 m no matter which data rate we use. For optical fiber, these limits are quite large for 10 Mbit/s and 100 Mbit/s operation, but become quite significant for 1 Gbit/s operation.

Second, when half-duplex operation is used, then the worst-case round-trip propagation delay must be restricted to less than one slot time in order for the CSMA/CD algorithm to function properly. This is why a 10 Mbit/s collision domain can span a maximum diameter of approximately 2.5 km whereas a 100 Mbit/s collision domain is limited to only 205 m. When full-duplex operation is used, then the propagation delay need not be related to the slot time in any way.

Third, when half-duplex operation is used, there can be at most four repeaters in the path between any pair of stations. This restriction comes about to prevent excessive shrinkage of the interframe gap, as explained above. And, finally, the network must be loop-free, whether it is constructed using repeaters and half-duplex links or using bridges (switches) and full-duplex links.

IEEE 802.3 ETHERNET STANDARD

The Institute of Electrical and Electronic Engineers (IEEE) Working Group 802.3 is responsible for defining an open vendor-independent standard for Ethernet. The Ethernet standard covers most of the functions of Layers 1 and 2 from the OSI reference model. However, these functions are divided into many sublayers by the *Ethernet Reference Model,* which is shown in Fig. 2 (1). The data link layer in the OSI reference model is divided into (1) logical link control (along with an optional MAC control sublayer to manage flow control in full-duplex links), which is outside the scope of the Ethernet standard, and (2) medium access control, which is the top sublayer within the Ethernet standard. All of the functions of the OSI physical layer are included in the Ethernet standard. The physical layer has been partitioned into sublayers in several ways, depending on the particular physical medium and/or data rate. Initially, the physical layer functions were separated into (1) physical layer signaling (PLS), which takes care of encoding and decoding the bit stream inside the end station, and (2) an external transceiver known as the medium attachment unit (MAU), which handles the actual transmission and reception of data over the link and generates the carrierSense and collisionDetect control signals. The communication between the PLS and MAU was defined by the attachment unit interface (AUI). However, as higher-speed operation was being developed, a different functional partitioning was adopted. First, Fast Ethernet introduced a new optional 4-bit-wide media independent interface (MII) to replace the bit-serial AUI, which defines a standard way to connect a removable transceiver. The MII can also be used for 10 Mbit/s operation, to simplify the design of equipment that can run at more than one speed. For Gigabit Ethernet, this interface was further changed into an 8-bit-wide Gigabit Media Independent Interface (GMII). The MII or GMII sits between (a) a small reconciliation sublayer, which manages the interface on behalf of the MAC sublayer, and (b) the physical coding sublayer (PCS), which handles encoding and decoding of the data stream and generation of the carrierSense and collisionDetect control signals. Below the PCS is the physical medium attachment (PMA) Sublayer, which contains Ethernet-specific functions for managing the physical transceiver, such as autonegotiation of speed and duplex settings and the operational status of the link. Finally, the lowest level functions were put into the physical-medium-dependent (PMD) sublayer so that the existing methods for transmission over optical fiber and UTP cable in the FDDI standard could be reused in Fast Ethernet. A similar partitioning was used

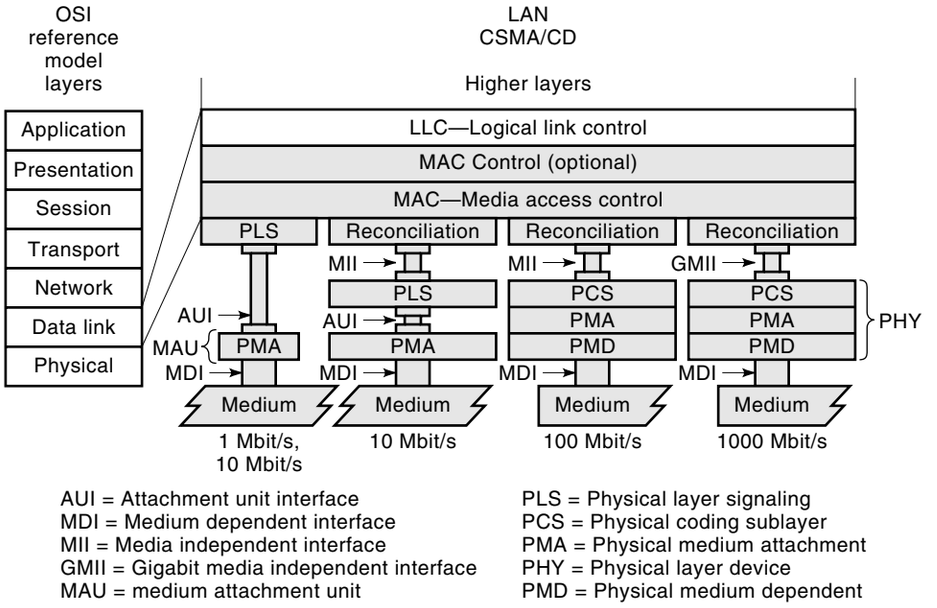

Figure 2. Ethernet reference model. (From Ref. 1, with permission.)

in Gigabit Ethernet, which uses a PMD derived from fiber channel.

In addition to the many sublayers that define the Ethernet functional specifications, the Ethernet standard also defines several groups of managed objects that provide a uniform set of attributes for getting information about the current status of the device (e.g., whether it is currently operating in half-duplex or full-duplex mode), retrieving data from cumulative activity counters (e.g., the number of octets of data sent or received), or setting operating parameters (e.g., whether or not it is set to promiscuously receive all incoming frames). These managed objects can be accessed through the Simple Network Management Protocol (SNMP), once the appropriate methods have been defined in the code contained in the Management Information Base (MIB) associated with the device.

HISTORY

Development of Ethernet began at the Xerox Palo Alto Research Center in 1973 with a prototype that operated at a data rate of 3.94 Mbit/s. An overview of this system was published by Metcalfe and Boggs (2) in 1976. By 1980, a commercial version of Ethernet, known as Ethernet version 2, had been jointly developed by Digital Equipment Corporation, Intel, and Xerox. Ethernet version 2 included a number of changes, including (1) larger values for the minimum frame size and slot time (2) and an increase in the data rate to 10 Mbit/s. The Ethernet version 2 "blue book" specification (3) formed the basis of the original IEEE 802.3 standard for Ethernet, published in 1983. However, the 802.3 standard introduced some technical changes, notably the replacement of the 16-bit "type field" by a 16-bit "length field" in the frame header. This distinction was eventually removed when a frame type was defined for the flow control pause frame in 1996.

Because support for new media types was added to the 802.3 standard, a new naming convention was adopted (4). The original version designed to operate over "thick" coaxial cable became 10BASE5, indicating that it operated at 10 Mbit/s, employed base band signaling, and had a maximum segment length of 500 m. In 1985, the standards for 10BASE2 (which defines Ethernet operation over "thin" RG-58 coaxial cable segments up to 185 m long) and 10BROAD36 (which defines Ethernet operation over broadband CATV systems in which the distance between the head end and the stations is at most 1.8 km) were approved. An even lower cost option called 1BASE5, based on the 1 Mbit/s AT&T Starlan design, was approved in 1987 but never became very popular. In 1987, 10 Mbit/s fiber-optic transceivers appeared in a limited way when a fiber-optic inter-repeater link (FOIRL) was approved, and they appeared more generally in 1993. 10BASE-T, which defines 10 Mbit/s operation over UTP cabling, was approved in 1990.

Fast Ethernet, which operates at a data rate of 100 Mbit/s, includes a variety of transceiver types. 100BASE-TX (for operation of two pairs of Category 5 UTP cabling), 100BASE-FX (for operation over optical fiber), and 100BASE-T4 (for operation over four pairs of Category 3 UTP cabling), was approved in 1995 and published as 802.3u (5). In 1996, 100BASE-T2 (for operation on two pairs of Category 3 UTP cabling) was published as 802.3y. However, neither 100BASE-T4 nor 100BASE-T2 has received widespread popularity. At the same time, full-duplex operation was defined in 802.3x, which was approved in 1996.

Gigabit Ethernet, which operates at a data rate of 1000 Mbit/s, also includes a number of subtypes. The 802.3z standard, which will be approved in 1998, includes 1000BASE-SX (a short-wavelength laser, suitable for limited distances over multimode fiber), 1000BASE-LX (a long-wavelength laser, suitable for moderate distances over multimode fiber and much longer distances over single mode fiber), and 1000BASE-CX (a short-haul copper jumper cable) (6). In addition, the development of 1000BASE-T (for transmission over distances of up to 100 m using four pairs of Category 5 UTP

cable) is well underway and should be published as 802.3ab sometime in 1999.

BIBLIOGRAPHY

1. IEEE 802.3 CSMA/CD (ETHERNET) Working Group Web Site [Online]. Available www: http://grouper.ieee.org/groups/802/3/
2. R. M. Metcalfe and D. R. Boggs, Ethernet: Distributed packet switching for local computer networks, *Commun. ACM,* **19** (7): 395–404, 1976.
3. Digital Equipment Corp., Intel Corp., and Xerox Corp., *The Ethernet: A local area network data link layer and physical layer specifications,* September 30, 1980.
4. ANSI/IEEE Std 802.3, *Carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications,* 5th ed., 1996.
5. IEEE Std 802.3u-1995, *Media access control (MAC) parameters, physical layer, medium attachment units, and repeater for 100 Mb/s operation, type 100BASE-T,* 1995.
6. IEEE Draft P802.3z/D4, *Media access control (MAC) parameters, physical layer, repeater and management parameters for 1000 Mb/s operation,* December 1997.

MART L. MOLLE
University of California, Riverside

ETHERNET. See LOCAL AREA NETWORKS.