1

Employment of Industrial Ethernet in Automated Plants

An automated control system for an industrial plant should allow the plant to work autonomously according to scheduled tasks, stop the plant or put it in a safe state in case of dangerous situations, and provide information for the supervision and monitoring of the plant. A distributed architecture of the control system allows for the division of an automated plant into modular computational units and provides the ability to assign to each unit, or group of units, a defined set of tasks. However, the distributed allocation of resources generates problems with response times due to the time needed to transfer relevant information among the remote computational units.

After a certain time, the information received by a functional unit (a Programmable Logic Controller [PLC] or an actuator) is no longer current, and is therefore outside the acceptable parameters for the unit's required reaction times. These reaction times are usually very strictly defined since the Distributed Control System (DCS) must coordinate operations in "real-time" to ensure production efficiency and meet safety requirements. Therefore, it is desirable to abstract completely from the geographical location of the computational units and to guarantee bounded times for the transmission of information.

The characterization of real-time communication will be dealt with in Chapter 2; here it is sufficient to realize that real-time implies a strict time correlation between the need to transmit information and the actual execution of the transmission.

Before proceeding to the description of real-time capability and safety compliance for Ethernet-based communication systems, it is necessary to briefly introduce some basics related to communication

network architecture. To do so, the OSI reference model for a generic communication system will be presented. Then, the specifics of standard Ethernet, as the basis for the development of modern industrial communication system architectures, will be described including a presentation of the frame structure and the *Data Link Layer*. Next, the hierarchical levels of the enterprise network and the need for a vertical integration among them will be presented.

1.1 Specification of a communication system through the OSI model

The transfer of information from one network station to another is realized through a series of steps. The sender inserts the relevant information in the message with additional data necessary for the management of packets by the other communication contributors and/or recipients. Then the sender codifies the data in the proper format, establishes a connection, and transmits the frame through the channel. The recipient decodes and interprets the frame, then executes the commands.

To achieve the interoperability of connected systems, a standardized procedure is necessary for the design of the communication system architecture. The approach of the Open System Interconnection (OSI) model, developed by the International Organization for Standardization (ISO), is to divide the set of functionalities into simpler services implemented through different functional levels, or layers (Cisco 2007). Each level executes functions and transfers services to adjacent levels, detailing the operations that are performed on the data. Therefore, the network architecture is realized, from a logical viewpoint, by a stack of communication services and protocol layers, as shown in Figure 1–1. The name OSI is derived from the objective to make systems open for communications with other systems.

Information that has to be transmitted crosses the OSI stack from the *Application Layer* to the *Physical Layer*, where it is coded as a sequence of bits and is inserted into the transmission channel. When those sequences of bits are received, information is decoded and crosses the OSI layers in the opposite direction to reach the application processes.

The Physical Layer is responsible for the management of the physical configuration (e.g., signal coding, wave form, voltage levels, layout of pins). The physical configuration depends on the characteristics of the medium through which the communication system makes available one or more logical and physical channels to the different communication

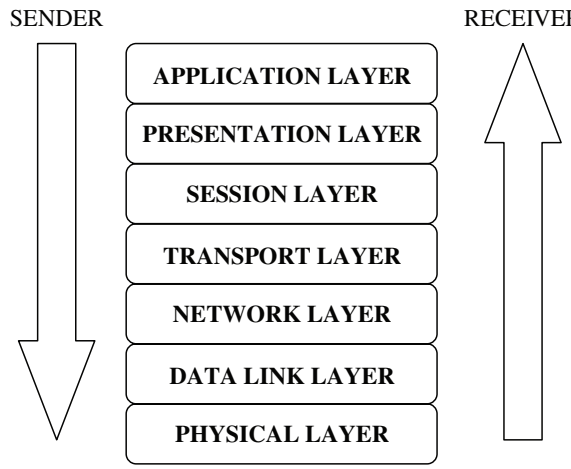

Figure 1–1 *OSI model structure and data transfer process*

participants. The specification of the Physical Layer is not the subject of this book.

The Data Link Layer oversees the management of the data exchange via the physical medium. In some communication system architectures, this layer can be divided into two sub-layers: the Medium Access Control (MAC) sub-layer and the Logical Link Control (LLC) sub-layer. The former is responsible for network access and the functions that depend on the network hardware. The latter is responsible for the integrity checking of data delivery (it checks for bit errors originating in the Physical Layer), the control of the frame structure, and packet retransmissions and acknowledgments.

The *Network Layer* is responsible for the routing of the packets through the network and specifies how the intermediate nodes handle the incoming packets. At this layer, services and protocols have some knowledge about the network structure and translate the physical addresses into logical ones (each node has its own network layer global address). The Internet Protocol (IP) is an example of a layer three protocol.

The *Transport Layer* is the link between the lower levels, dedicated to the physical communication, and the upper levels, dedicated to the application. This layer checks to ensure that the lower levels properly execute their tasks and provides a transparent data transfer service that is independent from the data organization of the different processing platforms connected to the network. Transmission Control Protocol (TCP)

and User Datagram Protocol (UDP) are the most representative layer four protocols.

The *Session Layer* of the OSI stack deals with the opening, maintaining, closing, and restarting of the connections between network stations, allowing different network stations to establish sessions between themselves and ensuring that the end-to-end transmission is successful. The Session Layer also provides a token management function that allows only the station holding the token to perform critical data transfer.

Since each network station may internally format data differently, agreements and conversions are needed to ensure that different stations can coexist on the same network. The *Presentation Layer* is concerned with the format of the transmitted information, encoding the structured data from the internal format of each station into a bitstream suitable for transmission. American Standard Code for Information Interchange (ASCII) and Extended Binary Coded Decimal Interchange Code (EBCDIC) are two such bitstream formats. At the destination end, the Presentation Layer decodes the data so that it is coherent with the required representation.

The Application Layer provides the application-specific services that use the services of lower layers to allow the connected stations to communicate with each other. These services are employed by the user's applicative network programs, such as file transfer protocol (FTP), to access a computer center or a web page (HTTP).

It is necessary to stress that the OSI model does not rigorously specify the services and protocols to be used in each layer; it simply states what every level has to do. Hence, many types of communication system architectures exist that can interact among layers, fulfilling the general specifications of the reference model.

1.2 Standard Ethernet

Ethernet is the major Local Area Network (LAN) technology in use today (Cisco 2007). The first version of standard Ethernet was the 2.94 Mbit/s version developed by Xerox's Palo Alto Research Center in the early 1970s. It was only in the 1980s that Intel and Xerox introduced 10 Mbit/s versions, and, at the same time, the Institute of Electrical and Electronic Engineers (IEEE) published open network standards through the 802 committee. The standards specify communication protocols with respect to signal coding, cabling, network access control, routing of packets, presentation of information, and interfacing with the application.

At first, Ethernet was standardized by the IEEE as a thick coaxial bus system, the so-called 10Base5 version. The bit rate was 10 Mbit/s, the maximum length of the cable segments was 500 m, and the maximum number of network stations connected to each segment was 100. The standardized version that followed was the thin coaxial cable Ethernet (10Base2). The maximum length of the cable segments was 185 m and the maximum number of network stations connected to each segment was 30. In the 1990s, the 10BaseT version introduced the twisted-pair cabling concept. The newest version, 10BaseF, is a series of fiber optical standards allowing longer coverage distance (up to 2 Km), noise immunity, and electrical isolation.

For each defined version, if further extension of the network is needed, a MAC layer device, for example a bridge or switch, can be used. Bridges and switches store and forward frames but do not transmit all received frames to all ports. Unlike repeaters, they use MAC layer addresses to forward information to selected recipients. If the network must be longer, repeaters, which are physical layer devices that retransmit signals in both directions, may be used (Anttalainem 2003). Collision detection requires that the cycle time does not exceed 51.2 μs to ensure that a station can confirm that transmission has been successful or that collision has occurred during transmission. Therefore, the maximum segment length controls the number of cable segments that can be connected with repeaters; the standard definition states that the maximum number of repeaters in a 10 Mbit/s network is four (Anttalainem 2003). Nowadays, higher bit rate (100 Mbit/s and 1 Gbit/s) Ethernet networks are available, which provide additional performance and network intelligence. Those variants use the same frame structure and the same collision detection mechanism, so the following description will refer to the original 10 Mbit/s standard.

This section describes in detail the frame structure and the Data Link Layer for standard Ethernet, including those relevant characteristics discussed in the following chapters.

1.2.1 Frame structure

Each frame starts with the preamble of 7 bytes, each containing the bit pattern 10101010. The start-of-frame delimiter (1 byte) contains the bit sequence of 10101011 and indicates the start of useful data in the frame. Both the destination and source addresses contain 6 bytes, with the first bit indicating whether the address is that of an individual network station or a group, and the second bit indicating whether the address is defined

locally or is a unique global address. Group addresses may be used for multicast when all stations belonging to the same group need to receive the same frame. If all the destination address bits are set to 1, all stations in a LAN receive the message. The length of the data field is variable from 0 to a maximum of 1500. For collision detection, the minimum length of the frame is 64 bytes from the destination address to the checksum. If the data field is very short, the Padding (PAD) field contains random data to extend the frame length to the minimum of 64 bytes. The Frame Check Sequence (FCS) is added to the end of the frame and, with the help of the 32-bit Cyclic Redundancy Check (CRC-32) code, the receiver is able to determine if bit corruptions have occurred in the frame during transmission. If errors are detected, the MAC layer discards the frame, and its recovery or non-recovery is left up to the upper protocol layers (Anttalainem 2003). Figure 1–2 shows the Ethernet frame structure.

1.2.2 CSMA/CD mechanism

The MAC layer is defined by the IEEE 802.3 standard. At this layer, the CSMA/CD (Carrier Sense Multiple Access/Collision Detection) protocol is implemented to allow many stations to share the same channel. According to the CSMA/CD, any network station that has to send a frame senses the channel and does not transmit until it is free. If the channel is free, the station may transmit at any time. If more than one station decides to transmit simultaneously, a collision will occur. Each station that detects the collision aborts its transmission, ensures that all other stations detect the collision as well, waits for a random period of time, and then tries again if no other station has started to transmit in the meantime. Contention and transmission periods alternate, with idle periods occurring when all stations are quiet. After a collision is detected, time is divided into discrete contention cycle times with a length equal to the worst-case round-trip propagation latency on the network.

The minimum time needed to detect a collision is based on the worst-case scenario. A station can be sure that it has transmitted successfully after two times the worst-case signal propagation latency. In fact, consider two stations: A and the farther away B. Station A may transmit just before receiving the signal from station B, and then takes another end-to-end propagation latency before this transmission is detected at the transmitting station B. Considering the signal transmission speed on the cable, to accommodate the longest path allowed (that is, 2.5 km with four repeaters), the cycle time is set to 512 bit times, corresponding, in the case

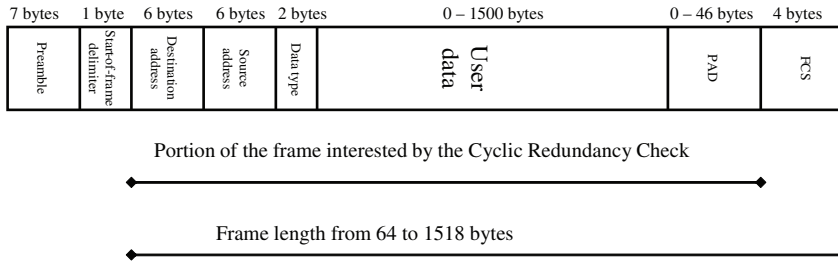

Figure 1–2 *Ethernet frame structure*

of a 10 Mbit/s network, to 51.2 μs. After the first collision, each station waits randomly either 0 or 1 contention cycle times before trying again.

If two stations collide and each waits for the same random time, they will collide again. After the second collision, each station waits again 0, 1, 2, or 3 cycle times, randomly. In general, after "i" number of collisions, a random number between 0 and 2i–1 is chosen, and that number of cycle times is skipped; therefore, the probability of the next collision decreases with the number of previous collisions. After 10 collisions have occurred, the randomization interval is frozen at the maximum of 1023 cycle times. After 16 collisions, the controller reports a network failure. The probability that this situation will occur in normal operation is very small, but it may happen. This algorithm, called binary exponential backoff, was chosen to allow the network to adapt dynamically to the number of stations trying to send. If the number of stations trying to send is high, significant delays will arise. No simple mathematical solution is available to accurately estimate CDMA/CD delays. Practical experience has proven that to have reasonable performance under the 802.3 standard, the loading has to be kept to approximately 40% or less of the maximum physical data rate (Anttalainem 2003).

1.3 CIM pyramidal structure

The CIM (Computer Integrated Manufacturing) model separates the services and functions of a completely automated plant and assigns them to a set of functional units allocated in different hierarchical levels:

- Management level
- Supervisory control level
- Field level