CHAPTER 6

FIELD BUSES

6.1. CAN (The Controller Area Network).
6.2. DeviceNet.
6.3. Foundation Fieldbus
6.4. PROFIBUS
6.5. Hart Communication Protocol

PREVIEW

In this chapter, the most important and recent low level international instrumentation network standards are presented. Section 6.1 introduces the industrial network CAN (Controller Area Network) with its main features, CAN Network technology and HLP (Higher Layer Protocols). The essential features of DeviceNet are given in Section 6.2. Foundation Fieldbus is detailed in Section 6.3 where the 3 layers fieldbus model is presented. Section 6.4 covers PROFIBUS with its 3 compatible versions DP, FMS and PA. In Section 11.5 the Hart Communication Protocol is presented in details. The Appendix includes 3 informative tables on summary and comparison of Field Buses, Background Information, and Physical Characteristics.

INSTRUCTIONAL OBJECTIVES

After reading this chapter, you should be able to

- Describe the architecture of the most popular industrial FieldBuses.
- Discusses their various protocols.
- Give the main features of each Bus.
- Determine the domain of application of the different Buses.
- Determine technical specifications for each Bus.
- Give examples of how a given Bus is implemented in an industrial application.
- Contrast the transport mechanism of different Buses.
- Give the physical characteristics of each Bus.

INDUSTRIAL FIELD BUSES

There are many available industrial networks designed to meet differing application requirements. For example, a simple on/off switch on a conveyor belt has different communication requirements than a complex control valve in a petroleum refinery. It is important to understand the target application of a network in order to choose the right network for a given application.

The following sections are introductions to some of these industrial networks

6.1 CAN (The Controller Area Network)

CAN was developed in the late 80's by Robert Bosch[1] as a solution for networking in distributed real-time systems. CAN supplies a high level safety, even when working in a harsh environment, and can support transmission speeds up to 1 Mbit/s. One of the most important features of the CAN technology is the guaranteed maximal latency for bus access that makes it the choice for real-time systems. The CAN protocol becomes an ISO standard for serial data communication, ISO 11898 for high speed up to 1Mbps and true plug and play, and ISO 11519 for low speed up to 125 kbps. The CAN protocol was developed aiming at automotive applications. Today CAN has gained widespread use. It is currently used in industrial automation as well as in automotive and mobile machines, and becomes more and more popular in additional fields such as textile industry, medical equipment, and elevator controls. The CAN features also a low cost and ease of operation and maintenance. CAN controllers are available off the shelf at very low cost from many semiconductor

---

[1] Robert Bosch, *CAN Specifications Version 2.0*, BOSCH, Stuttgart, 1991.

manufacturers. It is estimated that at least 10 million CAN installations were available worldwide in 1996, and it is expected to grow over 120 million installations by year 2000[2].

The main Features of ISO 11898 can be summarized as follows:

- Topology: Bus terminated on both sides.
- Bus medium: twisted -pair cable.
- Transfer mode: serial asynchronous data transfer, multimaster capability, baseband transfer, NRZ coding with bit-stuffing.
- Bus Access Procedure: Non-destructive CSMA/CD, bit-wise arbitration.
- Transmitter output level : differential similar to RS-485.
- Max. number of nodes : up to 64 (Practical limit).
- A non-destructive bitwise arbitration is used to control access to the bus.
- Message length: 0-8 data bytes.
- Message addressing: There is no explicit address in the messages; instead, each message carries a numeric identification code value which controls its priority on the bus.
- Error handling: an elaborate error handling scheme that results in retransmitted messages when they are not properly received. There are effective means for isolating faults and removing faulty nodes from the bus.
- Transmission Rates: 1 Mbps up to 40 meters, 500 kbps up to 100 m, 250 kbps up to 200 meters, 125 kbps up to 500 meters, and 10 kbps up to 6 km.

CAN Network Technology

CAN handles the lower two layers of ISO reference model in a similar structure to IEEE802.3 format. It defines a Physical, MAC (media access control, and LLC (logical link layer) layers. The CAN standard includes a physical layer and a data-link layer which defines

---

[2] www.can-cia.de : CAN in Automation user group

a few different message types, arbitration rules for bus access and methods for fault detection and fault confinement.

The function of the *Physical layer* is, as in any other network technology, to transfer the bits from one destination to another. The *Medium Access Control* sublayer of the Data Link layer has the function to control frames, checking and signalling for errors, performing the needed operations for accessing the bus, discovering faults and signalling for them. *The Logical Link Control* sublayer of the Data Link layer has to filter incoming messages, to provide services for data transfer and data requests, to provide means for recovery management and overloading notification.

However, current ISO implementations are based on twisted pair cable. What is important, is that there are two signal lines, termed *CAN H and CAN L. A dominant (logical 0)* bit has CAN H higher than CAN L, and a *recessive (logical 1)* bit, in opposite, CAN H lower than CAN-L. as shown in Figure xx.xx That mechanism yields to a reliable data transfer, even in an extremely harsh electrical environment.

The MAC Sublayer uses CSMA/CD as Ethernet, but they differ in the way the collision is candled. In the Ethernet, if collision is detected, each transmitting station performs *Backoff*, and tries to transmit again after a random period of time. This mechanism is called destructive bus allocation, since all transmission are aborted. There is no guarantee for any computer that in the next time the bus will be free and there will be no collision. Theoretically, there is no an upper bound for the latency of the bus access, and also after each collision the bus remains idle for certain amount of time.

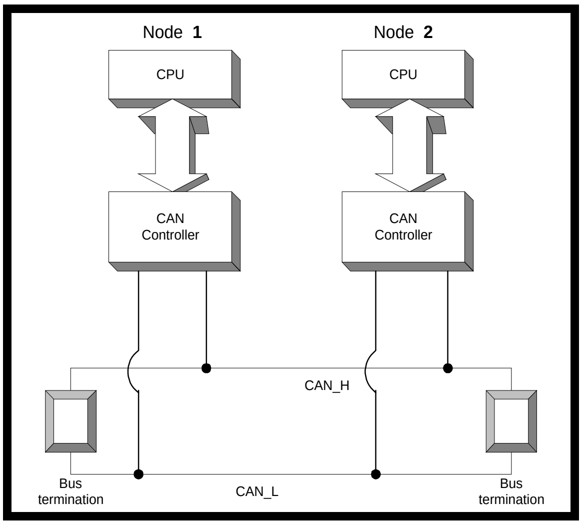
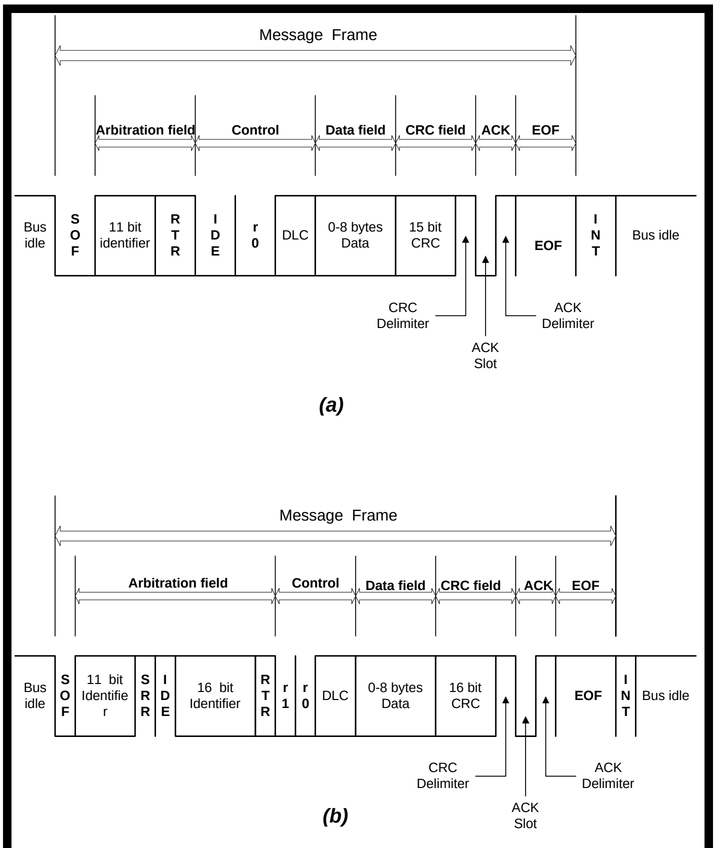


FIGURE 6-1

CAN uses *CSAM/CD with Non-Destructive Bitwise Arbitration* mechanism which ensures a low latency in a bus allocation for "important" messages, and also maximal bus utilization. First of all, in contrast to the Ethernet, the CAN communication is data-oriented (or content-oriented), and not destination-oriented. This means that frames are not sent to a specific destination, but they are identified according to the data they carry. The frame identifier defines also the data priority. When two nodes start transmission at the same time, they can't discover the problem until one of them tries to transmit 0 (dominant bit) and the second - 1 (recessive bit). In this case what is actually goes on the line is 0, and the node that tried to transmit 1 monitors 0 and recognizes a collision. It stops immediately the transmission .The node that tried to transmit 0, monitored 0, and thus didn't feel any problem and continued transmitting. It's important to mention that the identifiers of the data are unique, and it's impossible that two nodes try to transmit data with the same identifier.

The frame format is different from the Ethernet structure. There are two frame formats in the current CAN 2.0, standard frame format (compatible with CAN 1.0) CAN 2.0A, and extended frame format CAN 2.0B. The first one uses 11 bit identifier field, while the later one uses an additional 18 bit identifier field. Each frame starts by a single SOF bit, and ends with 7 recessive bits. The data field can be from 0-8 bytes. The frame contains as well a 15-bit error checking field using Cyclic-Redundancy Check (CRC), and other predefined bits and fields as shown in Figure 6.2.



Data Frame Formats: (a) Standard Frame Format ; (b) Extended Frame Format

Figure 6-2

One of the most powerful feature of CAN, is its overall error checking mechanism, which includes a number of fixed bits to validate the frame structure, the CRC, node to node protocol, and error monitoring, and confinement mechanisms.

Higher Layer Protocols (HLP)

The CAN protocol itself just specifies how small packets of data safely may be transported from point A to point B using a shared communications medium. It contains nothing on topics such as flow control, transportation of data larger than can fit in a 8-byte message, node addresses, establishment of communication, etc. These topics are covered by a HLP (higher layer protocol). The term HLP is derived from the OSI model and its seven layers. Higher layer protocols are used in order to standardize startup procedures including arbitrate setting distribute addresses among participating nodes or kinds of messages determine the layout of the messages provide routines for error handling on system level.

6.2 DeviceNet

Originally developed by Allen-Bradley, DeviceNet is managed by the Open DeviceNet Vendors Association (ODVA)[3], an independent supplier organization. DeviceNet is a low level network designed to connect industrial devices (sensors, actuators) to higher-level devices (controllers). DeviceNet focuses especially on the interchangeability of very low-cost, simple devices often used in manufacturing applications, such as limit switches, photoelectric sensors, motor starters, bar code readers, variable frequency drives, motor starters, etc.

DeviceNet builds upon the CAN (Controller Area Network) described in the previous section. While CAN specifies only portions of the Physical Layer and Data Link Layer (layers 1 and 2), DeviceNet adds the remainder of these two layers, plus the Media Layer and Application Layer. Layers 3-6 were not explicitly specified.

The main features of DeviceNet

- Basic Bus topology.

- Separate twisted-pair buses for both signal and power distribution, with signal and power carried in the same cable, (shielded cable).
- Hot insertion of devices without removing power from the network.
- Optional opto-isolated design so externally-powered devices can share bus cable with bus-powered devices.
- Data rates 125kbps (up to 500 m), 250kbps (up to 250 m), and 500kbps (up to 100 m).
- Maximum drop length is 6 meters.
- Up to 64 node addresses on a single network.
- Prioritized, Peer-to-Peer communication based on the non-destructive bitwise arbitration scheme of CAN protocol.
- Producer-Consumer Model for data transfer.

While CAN specifications define the form of data movement, the DeviceNet Application Layer (ISO layer 7) defines the semantics of the data on the network. DeviceNet specifications were based on the Object Oriented notations that will be introduced later in this chapter. In order to ensure compatibility between devices, each device usually comes with built in (in Electronic form) the device profile in a pre-specified format. The format includes the definition of the device object model, definition of the device I/O data format, and definition of the device's configurable parameters. The device object model specifies the device operation in terms of a limited set of special objects or functions, e.g., *Identity Object* (specifies vendor ID, device type, product code, serial number etc.), and *Device Net Object* (defines node address, baud rate, Bus-off action, Bus-off counter, etc.), *Connection Object, Message Route Object*, and *Parameter Object.*

---

[3] Www.odva.org

6.3 Foundation Fieldbus

FOUNDATION Fieldbus[4] is an industrial network designed specifically for distributed process control applications. This network was created by the Fieldbus Foundation, an organization of more than 100 companies that make up more than 80 percent of the world's supply of automation systems, devices and services. Specifications of the FFB were released in 1996 for the low speed field bus (H1) 31.25 kbit/s, and the high speed field bus (H2) 1.0 Mbit/s and 2.5 Mbit/s fieldbus, and details of the fieldbus' Data Link Layer, Application Layer, and System Management and Network Management functions.

The Foundation requires that vendors must register their field and host devices with the Foundation. The Fieldbus Foundation's product registration is granted through extensive testing procedures to ensure that a Foundation-registered host or field device will communicate and fully interoperate with any other registered device.

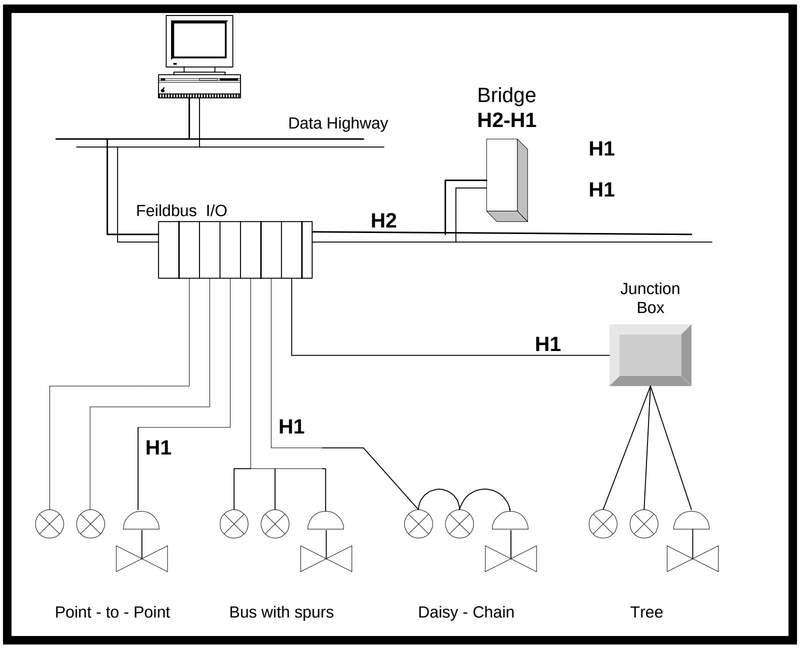


Figure 6-3

Foundation Fieldbus is based on existing technologies where possible, including work of the ISA (International Society for Measurement and Control) ISP50.02-1992, and the IEC

(International Electrotechnic Committee) standards IEC1158-2-1993. The Data Link Layer DLL and the Application Layers conform to the International IEC 61158 fieldbus standard. The Foundation has also recently standardized a 100 Mbps High Speed Ethernet (HSE), based on the open, commercial, high-speed Ethernet technology. Additional elements of the recommendations will provide redundant Ethernet physical media and multiple linking devices in order to meet the robust requirements of mission-critical applications.

FFB Network model

The field bus model consists of 3 layers:

(a) the Physical Layer corresponding to the ISO layer 1.

(b) the Communication "Stack" corresponding to layers 2 and 7 in the ISO model.

(c) the User Layer.

Figure 3 shows these components compared to the OSI 7-layer communication model.

FFB does not implement layers 3,4,5 and 6 of the OSI model because the services of these layers are not required in a process control application, however, a very important part of Foundation Fieldbus is the User Layer.

---

[4] www.fieldbus.org

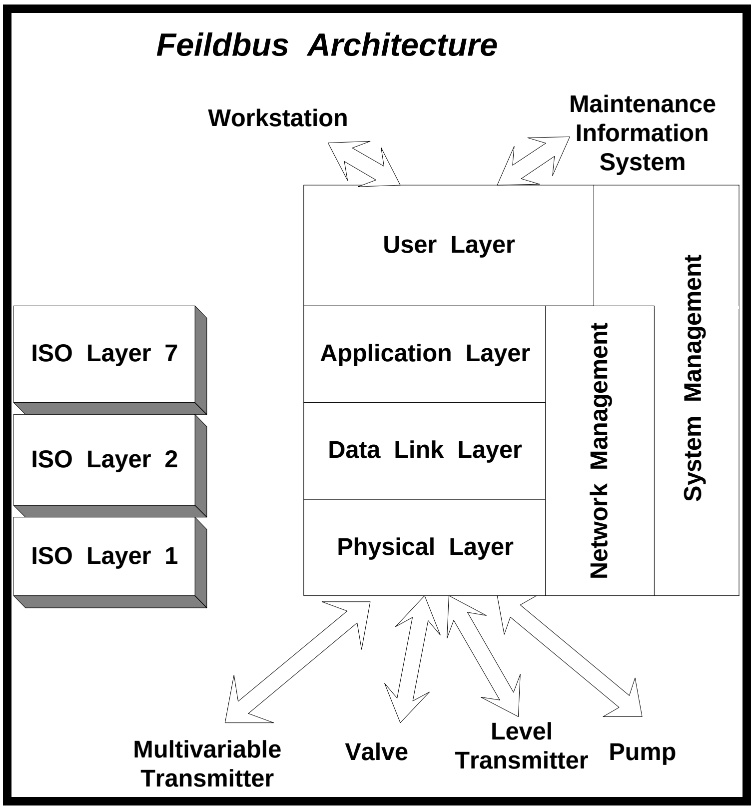


Figure 6-4

Physical Layer

Foundation Fieldbus uses the physical layer (electrical, wiring, and so on) standardized by ISA/IEC (ISA S50.02-1992, IEC 1158-2). Table 4 summarizes the main characteristics of this standardized physical layer.

	H1	H2 1Mbps	H2 2.5 Mbps
Topology	Bus/Tree	Bus	Bus
Devices	2-32	2-32	2-32
Bus-powered devices	2-13	none	none
Bus powered IS devices	2-6	none	none
Max disttance using shielded twisted-pair cable STP.	1900 m.	750 m.	500 m.
Spur length.	120 m.	non.	non

Table 4. Summary of IEC-1158 Physical Layer Characteristics

Several key characteristics to point out are:

- Several communication data rates, including a low speed that is compatible with existing twisted-pair 4-20 mA wiring, as shown in Table xxx..
- Devices draw their operating power from the same two wires as the signal, removing the need for external power supplies, as shown in table xxx.
- Supports bus and star topology.
- Capability for intrinsically safe operation, a requirement for hazardous environment.

Communication Stack

The Communication Stack performs the services required to interface the User Layer to the Physical Layer. Several characteristics and functions of the Data Link Layer, however, are key to the distributed, real-time capabilities of Foundation Fieldbus:

- The Data Link Layer is a token-passing protocol.
- The Link Active Scheduler (LAS) is a device that acts as the centralized arbitrator of the bus.
- The LAS executes a schedule that makes possible deterministic control and communication.
- The LAS distributes time to the bus to permit all devices to share the same sense of time.
- Control can be passed between multiple Link Masters (devices capable of being the LAS), providing redundancy on the fieldbus.

User Layer

FFB defines a unique communication layer called the User Layer. The User Layer does not exist in the ISO communication stack model. The User Layer defines an interface by which users of Foundation Fieldbus can communicate with devices through a set of blocks rather than as a collection of simple data points. Three types of blocks make up the Foundation Fieldbus User Layer:

Resource Block - describes characteristics of device such as name, manufacturer, and serial number.

Function Blocks - provide the control and I/O behavior of a device.

Function Block Name	Symbol
Analog Input	AI
Analog Output	AO
Bias	B
Control Selector	CS
Discrete Input	DI
Discrete Output	DO
Manual Loader	ML
Proportional/Derivative	PD
Proportional/Integral/Derivative	PID
Ratio	RA

Transducer Blocks - decouple Function Blocks from the functions required to read / write local inputs / outputs.

Foundation Fieldbus defines standard sets of Function Blocks, of which there is a set of 10 for the most basic of control and I/O functions (see Table 5). Other function blocks are being defined both by the Foundation and by individual manufacturers.

Users create applications on the fieldbus by connecting together the inputs and outputs of function blocks. In addition to specifying how these blocks “talk” to one another over the bus, Foundation Fieldbus also specifies how you can precisely schedule the time at which these blocks execute. The Function Blocks themselves reside in individual devices but the overall scheduling of execution is specified and executed across the network. Figure 5 shows a PID control loop for flow control and comparators for high alarm and low alarm. The figure uses four Function Blocks ; AI, PID, and AO, and CMP.

Because of the ability to interconnect different functions, even control algorithms, that reside with the field devices themselves, Foundation Fieldbus actually provides an architecture for distributing control into the field rather than concentrating the control in centralized controllers.

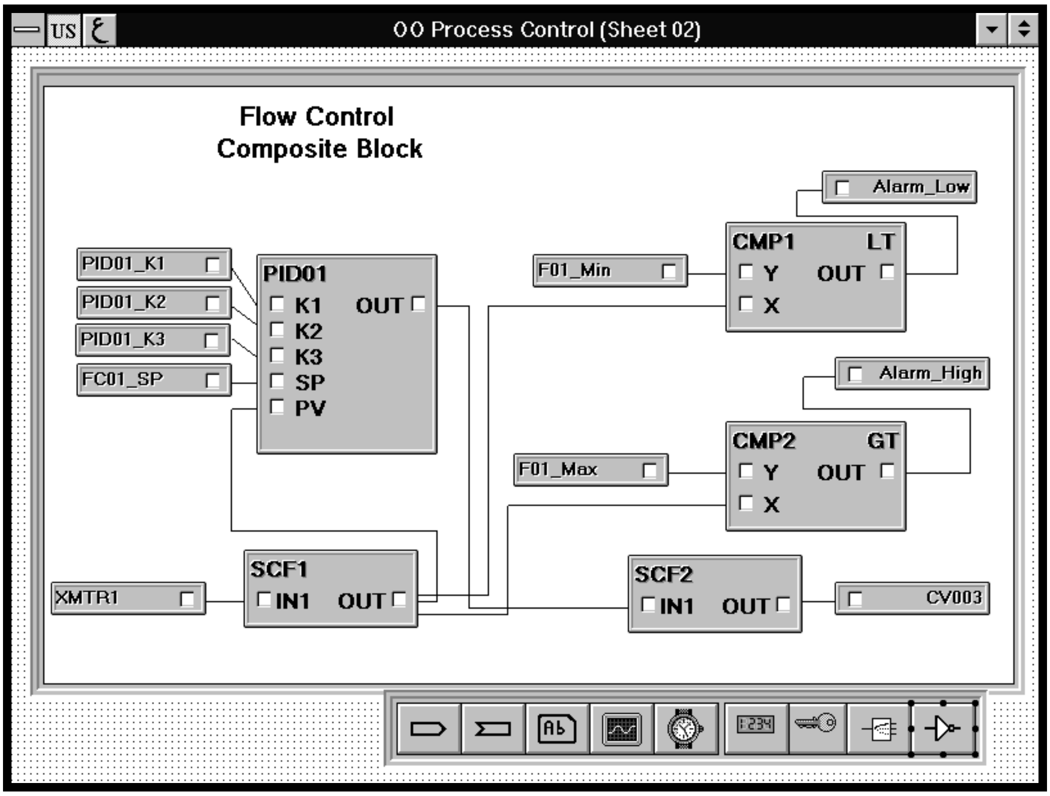


Figure 6-5

Device Description

A second important feature of the Foundation Fieldbus User Layer is Device Descriptions. A Device Description (DD) is a standardized description of the functions available in a device. Because it actually describes a device’s functions, the DD is a standard way that any host system can learn about the capabilities of the device. Even if the device contains a brand new capability never before seen in such a device, as long as the capability is included in the

DD, the host can access it. Thus, systems and devices, even those containing completely unique functionality, can interoperate by means of the DD.

6.4 PROFIBUS[5]

PROFIBUS is a leading open fieldbus system in Europe, is used worldwide in manufacturing, process, and building automation. PROFIBUS is standardized in the German standard DIN 19245 and European fieldbus standard EN 50170. PROFIBBUS technology is developed and administered by the PROFIBUS User Organization, with a membership of more than 600 manufacturers, users and research institutions.

Designed to meet a variety of application requirements, PROFIBUS can be used for both high-speed time-critical data transmission between controllers and I/O, and complex communications between programmable controllers. The PROFIBUS family consists of three compatible versions - DP, FMS, and PA. These three protocols are described briefly in the following sections. As by end of 1Q/99 it is estimated that over 2,500.000 Profibus devices are used worldwide.

---

[5] www.profibus.com

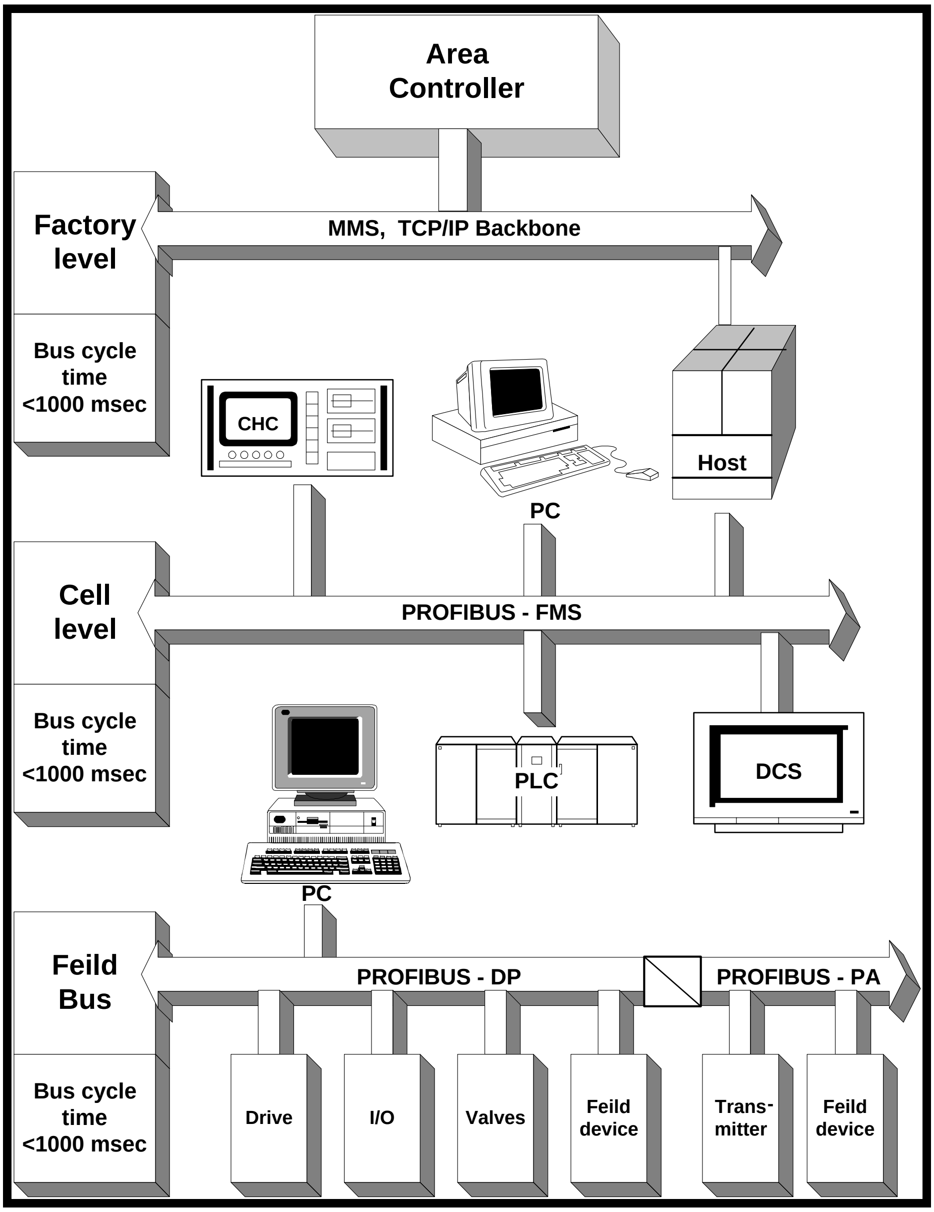


Figure 6-6

Protocol Architecture

Figure 6.7 shows the architecture of PROFIBUS protocols using the OSI 7-Layer model.

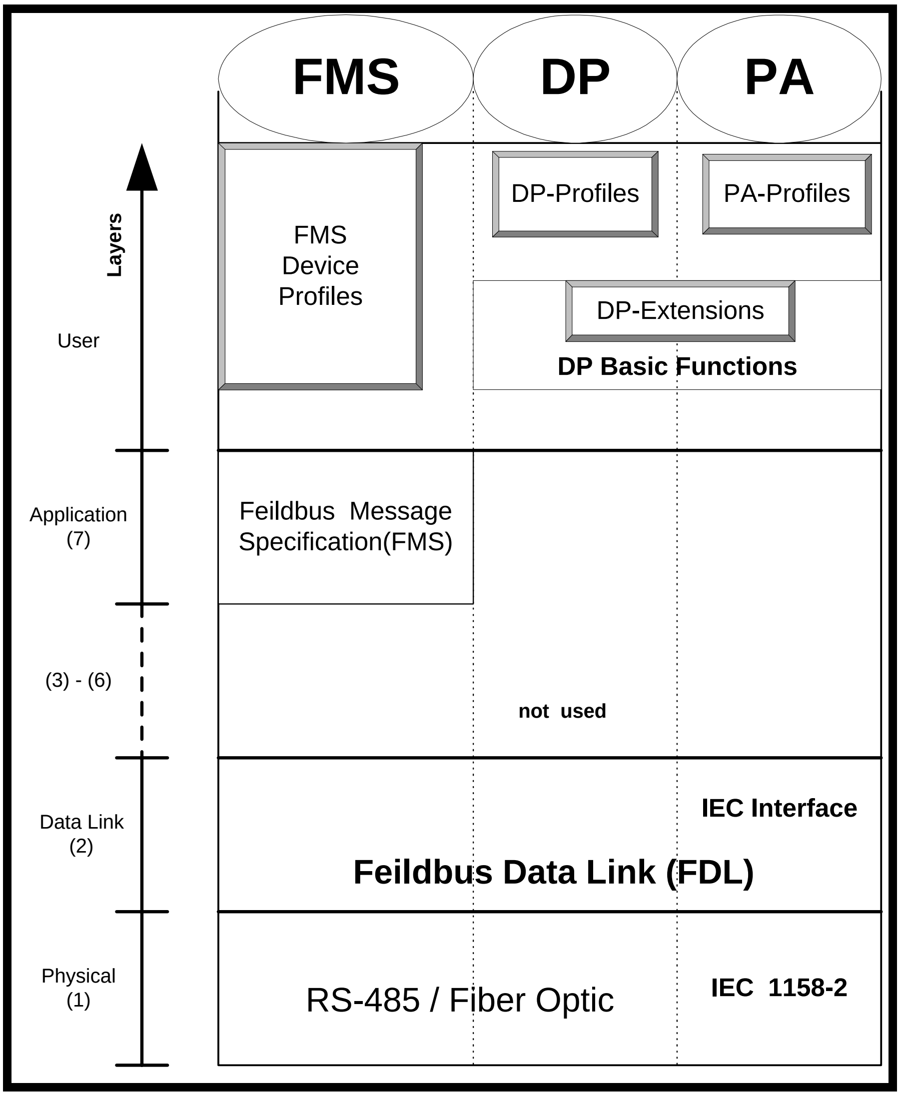


Figure 6-7 Protocol architecture of PROFIBUS

PROFIBUS-DP

PROFIBUS-DP is designed for high speed, cost-effective communication between industrial controllers and distributed I/O; it can replace parallel signal transmission with 24 V or 0 to 20 mA. On a PROFIBUS-DP network, central controllers, such as PLCs, or PCs, communicate with distributed field devices (such as I/O devices, drives, and valves) via a high-speed serial link. Most of the data communication with these distributed devices is done in a cyclic manner.

PROFIBUS-DP uses Layers 1 and 2, and the user interface (Figure 12.7). Layers 3 to 7 of the OSI model are not defined. The Direct Data Link Mapper (DDLM) gives access between the user interface and Layer 2. The user interface specifies both application functions that are available to the user and system and device behaviour of the PROFIBUS - DP device types. RS-485 and fiber-optic are available physical media for PROFIBUS-DP.

PROFIBUS-FMS

PROFIBUS-FMS is designed for general purpose communication primarily between programmable controllers, such as PLCs and PCs. FMS contains an application layer with communication service available to the user. These services make it possible to access variables, transmit programs, and control program execution as well as transmit events. PROFIBUS-FMS defines a communication model in which distributed application processes can be unified into a common process by using communication relationships.

In PROFIBUS-FMS, Layer 1, 2 and 7 are defined (Figure 6). The application layer consists of FMS (Fieldbus Message Specification) and LLI (Lower Layer Interface). FMS contains the application protocol and provides the user with a wide selection of communication service. LLI implements communication relationships and provides FMS with device-independent access to Layer 2. Layer 2 (FDL, Fieldbus Data Link) implements bus access control and data security. RS-485 and fiber-optic physical layers are available for PROFIBUS-FMS.

PROFIBUS-PA

PROFIBUS-PA is designed specifically for process automation, using the international fieldbus standard physical layer (IEC 1158-2) for bus-powered sensors and actuators to be operated in intrinsically safe areas. PROFIBUS-PA uses the extended PROFIBUS-DP protocol for data transmission. Using the IEC 1158-2 physical layer, field devices can be powered over the bus. PROFIBUS-PA devices can be integrated in PROFIBUS-DP networks by the use of segment couplers.

Frofibus distinguishes between master devices and slave devices.

Master devices determine the data communication on the bus. A master when it holds the “token” can send messages without an external request. Masters are also called active stations.

Slave devices are peripheral devices. Typical slave devices include I/O devices, valves, derives, and measuring transmitters. Slaves do not have bus access rights and they can only acknowledge received messages or send messages to the master when requested to do so. Slaves are called passive stations.

RS-485 Physical Layer for PROFIBUS-DP/FM

RS-485 is the physical layer most frequently used in PROFIBUS applications. Baud rates of 9.6 kb/s to 12 Mb/s can be used; one transmission speed is selected for all devices on the bus when the system is commissioned. Up to 32 stations can be attached to each segment without repeaters, and up to 127 stations can be attached with repeaters. The operating distance range from 100 m at 12Mbps to 1200 m at 9.6 kbps.

11.5 Hart Communication Protocol

HART is an acronym for "Highway Addressable Remote Transducer". HART was developed by Rosemount Inc. in the mid-1980’s, but has been made completely open, and all rights now belong to the independent HART Communication Foundation[6]. It is widely accepted in the industry as a de facto standard for digitally enhanced 4-20mA communications with smart field instruments. The HART protocol was designed specifically for use with intelligent measurement and control instruments which traditionally communicate using 4-20mA analogue signals. HART preserves the 4-2OmA signals and enables two-way digital communications to occur without disturbing the integrity of the 4-20mA signals. The HART protocol permits the process variable to continue to be transmitted by the 4-2OmA analogue signal, and additional information pertaining to other variables, parameters, device configuration, calibration, and device diagnostics to be transmitted digitally at the same time

The HART Protocol uses the Open Systems Interconnection (OSI) reference model as a guide. However, it implements only layers 1,2 and 7, i.e. the Physical layer, the Data Link layer, and the Application layer.

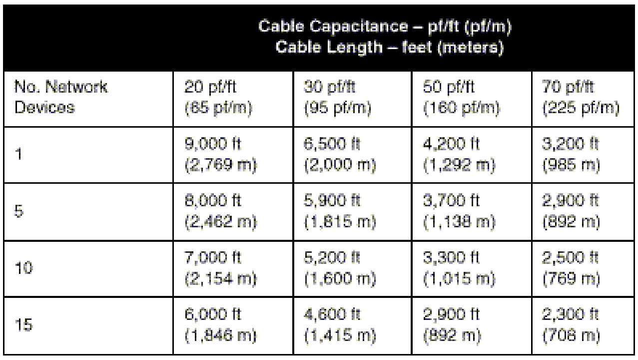
The Physical Layer makes use of the Bell 202 Frequency Shift Keying (FSK) standard to superimpose digital communication signals at a low level on top of the 4-2OmA as shown in Figures xxx . The HART protocol communicates at 1200 bps without interrupting the 4-2OmA signal and allows a host application (master) to get two or more digital updates per second from a field device. As the digital FSK signal is phase continuous, there is no interference with the 4-2OmA signal.

The bit stream is encoded into two sinusoidal tones. A “1” bit is represented by a 1200 Hz tone, and a “0” bit is transmitted by a 2200 Hz tone. Cable length can be up to 1500 m if 24AWG Multiple twisted pair with common shield. And up to about 3,000 meters using 20AWG single twisted pair with shield. Table xxx gives the allowable cable length for versus the number of devices. The HART protocol permits all digital communication with field devices in multidrop network configurations. In conventional point-to-point mode, the 4-20 mA signal continues to be used for analog transmission, while other data can be transferred digitally. In the mult-drop mode, up to 15 devices can be connected on a single pair of wires.

---

[6] www.hartcomm.org

In multidrop mode, device address can be from 1-15, and each device sets its current output at a fixed value of 4.0 mA.



	Cable Capacitance – pf/ft (pf/m) Cable Length – feet (meters)			
No. Network Devices	20 pf/ft (65 pf/m)	30 pf/ft (95 pf/m)	50 pf/ft (160 pf/m)	70 pf/ft (225 pf/m)
1	9,000 ft (2,769 m)	6,500 ft (2,000 m)	4,200 ft (1,292 m)	3,200 ft (985 m)
5	8,000 ft (2,462 m)	5,900 ft (1,815 m)	3,700 ft (1,138 m)	2,900 ft (892 m)
10	7,000 ft (2,154 m)	5,200 ft (1,600 m)	3,300 ft (1,015 m)	2,500 ft (769 m)
15	6,000 ft (1,846 m)	4,600 ft (1,415 m)	2,900 ft (892 m)	2,300 ft (708 m)

Table 3: Allowable cable lengths for 1.02 mm (#18 AWG) shield twisted pair

The Data Link Layer (DLL) is responsible for the reliable transmission of data packets across the network.

HART is a master/slave, character oriented, protocol which means that a field (slave) device only speaks when spoken to by a master. HART provides for up to two masters (primary and secondary) as shown in Figure xxx. This allows secondary masters such as handheld communicators to be used without interfering with the master operation.

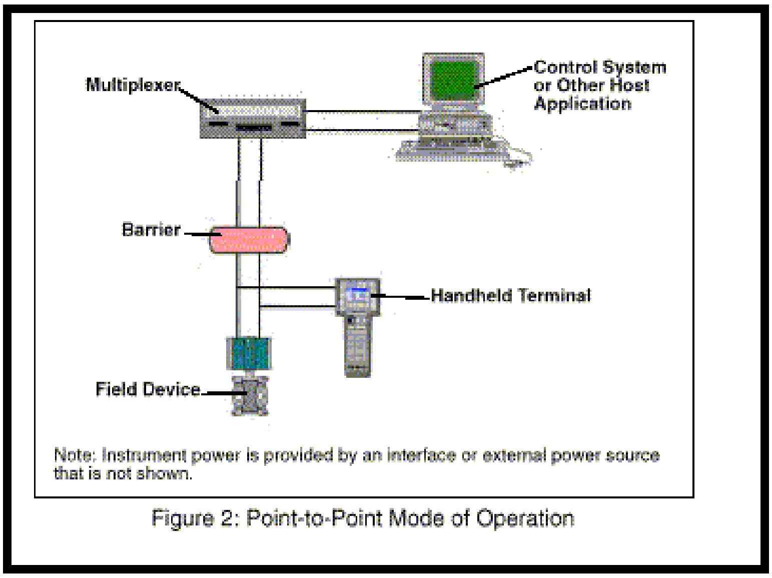
The DLL defines HART message structure. A HART message consists of synchronous 8-bit data bytes organized into frames. . The message structure is shown in Figure xxxx.

The *preamble*, 5-20 bytes of hex FF (all 1's), helps the receiver to synchronise to the character stream.

The *start* character may have one of several values, indicating the type of message: master to slave, slave to master, or burst message from slave; also the address format: short frame or long frame.

The *address* field includes both the master address (a single bit: I for a primary master, 0 for a secondary master) and the slave address. In the short frame format, the slave address is 4 bits containing the "polling address" (O to 1 5). In the long frame format, it is 3 8 bits containing a "unique identifier" for that particular device. (One bit is also used to indicate if a slave is in burst mode.)

The *command* byte contains the HART command for this message. *Universal Commands* are in the range 0 to 30; *Common Practice Commands* are in the range 32 to 126; *Device-Specific Commands* are in the range 128 to 253. We will talk about these commands shortly.



Note: Instrument power is provided by an interface or external power source that is not shown.

Figure 2: Point-to-Point Mode of Operation

Figure 6-8

The *byte count* byte contains the number of bytes to follow in the status and data bytes. The receiver uses this to know when the message is complete. (There is no special "end of message" character.)

The *status field* (also known as the "response code") is two bytes, only present in the response message from a slave. It contains information about communication errors in the outgoing message, the status of the received command, and the status of the device itself.

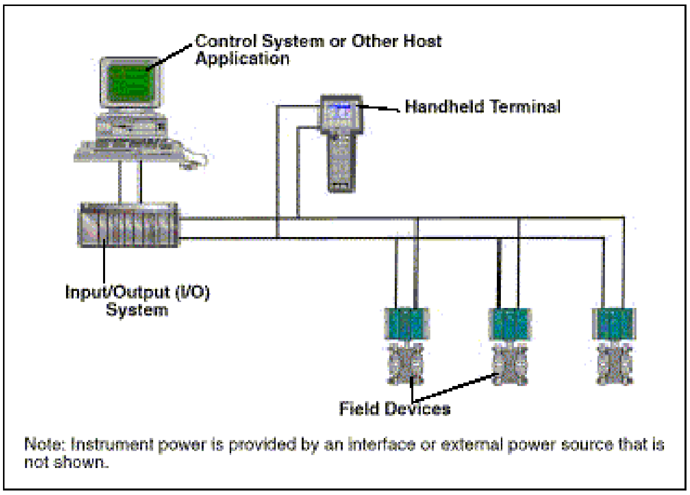
The *data field* may or may not be present, depending on the particular command. A maximum length of 25 bytes is recommended, to keep the overall message duration reasonable. (But some devices have device-specific commands using longer data fields.)

Finally, the *checksum* byte contains an "exclusive-or" or "longitudinal parity" of all previous bytes (from the start character onwards). Together with the parity bit attached to each byte, this is used to detect communication errors.

Transactions usually consist of a Master command ands slave response pair. The integrity of HART communication is very secure as status information is included with every reply message and extensive error checking occurs with each transaction. Up to four process variables can be communicated in one HART message and each device may have up to 256 variables.

The **Application Layer** defines the semantics of these messages and commands.

The HART Command Set is organized into three groups and provides read/write access to the wealth of additional information available in smart field instruments employing this technology. Universal Commands must be implemented by all HART devices and provides interoperability across the large and growing base of products and manufacturers. The HART Application Layer (OSI Layer 7) consists of three classes of commands or messages: Universal Commands, Common Practice Commands, Device Specific Commands.



Note: Instrument power is provided by an interface or external power source that is not shown.

Figure 3: Multidrop Mode of Operation

Figure 6-9

Universal commands must be implemented by all hart devices and provides interoperability across the large base of products and manufactures. Universal commands provide access to information that is useful in normal plant operation such as the instrument manufactures, model, tag, serial number, descriptor, range, limits, and process variable.

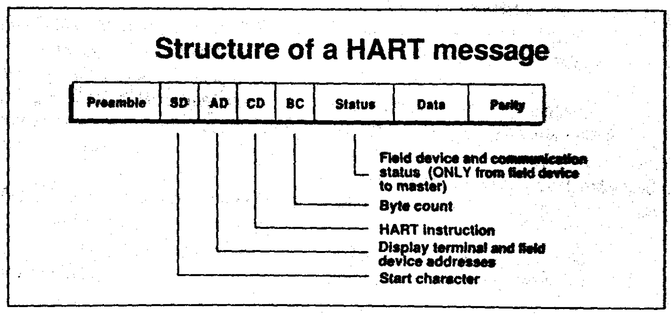


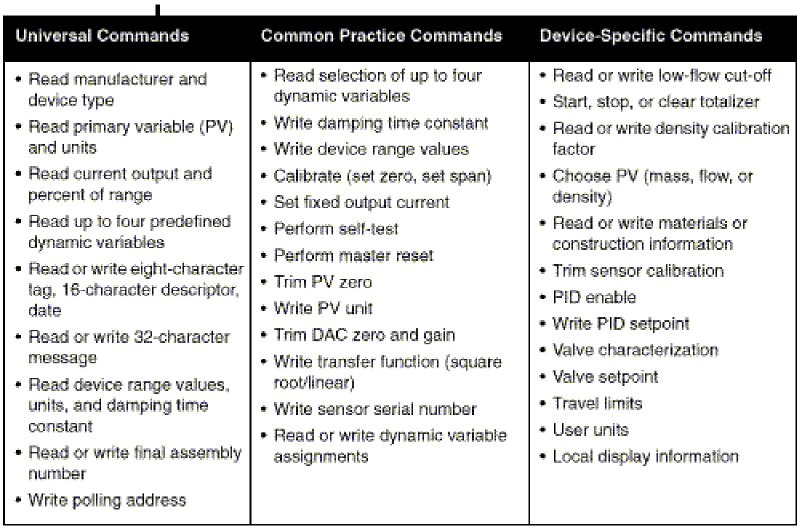
Figure 5
The HART message structure offers a high degree of data integrity.

Figure 6-10

Common practice commands access to functions which can be carried out by many devices though not all, and device specific commands provide access to functions which can be carried out by many devices.

Device Specific Commands provide access to functions which may be unique to a particular device. Table xxx gives examples of these types of commands.

Device Description Language (DDL), a recent enhancement to the HART technology extends interoperability to a higher level than provided through the Universal and Common Practice Commands. As reflected in Figure 8, DDL provides a field device (slave) product developer with the means to create a complete description of their instrument and all relevant characteristics, such that it can talk to any host device using the language. This **is** analogous to a printer driver in the personal computer world which enables an application to talk with a printer such that what gets printed on the page is what was expected by the application. Universal hand-held communicators capable of configuring any HART-based instrument through DDL are available today. Broader application in other types of host systems is expected. The HART Communication Foundation manages the centralized library of all registered Device Descriptions and DDL is being supported by all members of the Foundation.



Universal Commands	Common Practice Commands	Device-Specific Commands
• Read manufacturer and device type • Read primary variable (PV) and units • Read current output and percent of range • Read up to four predefined dynamic variables • Read or write eight-character tag, 16-character descriptor, date • Read or write 32-character message • Read device range values, units, and damping time constant • Read or write final assembly number • Write polling address	• Read selection of up to four dynamic variables • Write damping time constant • Write device range values • Calibrate (set zero, set span) • Set fixed output current • Perform self-test • Perform master reset • Trim PV zero • Write PV unit • Trim DAC zero and gain • Write transfer function (square root/linear) • Write sensor serial number • Read or write dynamic variable assignments	• Read or write low-flow cut-off • Start, stop, or clear totalizer • Read or write density calibration factor • Choose PV (mass, flow, or density) • Read or write materials or construction information • Trim sensor calibration • PID enable • Write PID setpoint • Valve characterization • Valve setpoint • Travel limits • User units • Local display information

Table 1: HART Commands

APPENDIX

SUMMARY AND COMPARISON OF FIELD BUSES

TRANSPORT MECHANISM

	Communication Methods	Transmission Properties	Data Transfer Size	Arbitration Method	Error Checking	Diagnostics
PROFIBUS DP/PA	Master/Slave peer to peer	DP up to 12 Mbps PA 31.25 kbps	244 bytes	Token passing	HD4 CRC	Station, module & channel diagnostics
INTERBUS-S	Master/slave with total frame transfer	500kBits/s, full duplex	512 bytes, h.s., unlimited block	None	16-bit CRC	Segment location of CRC error and cable break
DeviceNet	Master/slave, multi-master, others	500 kbps, 250 kbps, 125 kbps	8-byte variable message	Carrier-Sonac Multiple Access	CRC check	Bus monitoring
ARCNET	Peer to peer	31.25 K to 10 M	508 bytes	Token	16-Bit CRC	Built in
AS-I	Master/slave with cyclic polling	Data and power, EMI resistant	31 slaves with 4 in and 4 out	Master/slave with cyclic polling	Manchester Code, hamming-2, partly	Slave fault, device fault
Fieldbus Foundation	Client/server publisher / subscriber, Event notification	31.25 kbps 1 Mbps 2.5 Mbps	16.6 M objects / device	Deterministic centralized scheduler, multiple backup	16-bit CRC	Remote diagnostics, network monitors, parameter status
IEC/ISA SP50 Fieldbus	Client/server Publisher / subscriber	31.25 kbps IS+1, 2.6, 5Mbps	64 octets high & 256 low priority	Scheduler, tokens, or master	16-bit CRC	Configurable on network management
Seriplex	Master / slave peer to peer	200 Mbps	7680 / transfer	Sonal multiplexing	End of frame & echo check	Cabling problem
World FIP	Peer to peer	31.25kbps, 1 & 2.5 Mbps, 6 Mbps fiber	No limit, variables 128 bytes	Central arbitration	16-bit CRC, data “freshness” indicator	Device message time-out, redundant cabling
LonWorks	Master/slave peer to peer	1.25 Mbs full duplex	228 bytes	Carrier Sense, Multiple Access	16-bit CRC	Database of CRC errors and device errors
SDS	Master/slave, peer to peer, multi-case, multi-master	1 Mbps, 500 kbps, 250 kbps, 125 kbps	8-byte variable message	Carrier – Sonac Multiple Access	CRC check	Bus monitoring, Diagnostic slave

BACKGROUND INFORMATION

	Technology Developer	**Year Introduced**	**Governing Standard**	**Openness**
PROFIBUS DP/PA	PTO	DP-1994, PA-1995	DIN 19245 part ¾	Products from over 150 vendors
INTERBUS-S	Phoenix Contact	1984	DIN 19258	Products from over 4000 + manufacturers
DeviceNet	Allen-Bradley	March 1994	ISO 11898 & 11519	6 chip vendors, 100+ products
ARCNET	Datapoint/SMC	1975	ANSI 878	Chips, boards, ANSI docs
AS-I	AS-I Consortium	Fall 1993	Submitted to IEC	AS-II.C. Market item
Fieldbus Foundation	Fieldbus Foundation	1995	ISO SP50/IEC TC65	Chips/software from multiple vendors
IEC/ISA SP50 Fieldbus	ISA & Fieldbus F.	1992-1996	IEC 1158/ ANSI 850	Multiple chip vendors
Seriplex	APC, Inc.	1990	Seriplex spec	Chips available multiple interfaces
World FIP	WorldFIP	1988	IEC 1158-2	Multiple chip vendors
LonWorks	Echolon Corp.	March 1991	ASHRAE of BACnet	Public documentation on protocol
SDS	Honeywell	Jan., 1994	Honeywell Specification, Submitted to IEC, ISO11989	6 chip vendors, 200+ products

PHYSICAL CHARACTERISTICS

	Network Topology	**Physical Media**	**Max. Devices (nodes)**	**Max. Distance**
PROFIBUS DP/PA	Line, star & ring	Twisted-pair or fiber	127 nodes	24 Km (fiber)
INTERBUS-S	Segmented with "T" drops	Twisted-pair fiber, and slip-ring	256 nodes	400 m/segment, 12.8 Km total
DeviceNet	Trunkline/ dropline with branching	Twisted-pair for signal & power	64-nodes	500m
ARCNET	Bus, multidrop, staf	Twisted-pair coax fiber	255 nodes	5 miles
AS-I	Bus, ring, tree star, of al	Two wire cable	31 slaves	100 meters, 300 with repeater
Fieldbus Foundation	Multidrop with bus powered devices	Twisted-pair	240/segment, 65,000 segments	1900m @ 31.25 K 500 m @ 2.5M
IEC/ISA SP50 Fieldbus	Star or bus	Twisted-pair fiber, and radio	IS 3-7 Non IS 128	1700m @ 31.25K 500M @ 5Mbps
Seriplex	Tree, loop, ring, multi-drop, star	4-wire shrolded cable	500+ devices	500+ ft
World FIP	Bus	Twisted-pair, fiber	256 nodes	Up to 40 Km
LonWorks	Bus, ring, loop, star	Twisted-pair, fiber, power line	32,000/ domain	2000m @ 78 kbps
SDS	Trunkline/ Dropline	Twisted-pair for signal & power	64 nodes, 126 addresses	500 m

SUMMARY

1. The main functions of an instrumentation system are: value and quality assessment, safety and protection, control, and data collection.
2. Block diagrams help to view the subfunctions of each part of a process and determine its input and its output, and how it is linked with the other parts of the process.
3. The main parts of a control loop are the process, the measurement, error detector, controller, and control element.
4. P&I diagrams consist of graphical symbols and lines which illustrate the flow of a process and identify the location and functions of its instruments, e.g. sensors, valves, recorders, indicators, and instrument interconnections
5. An instrumentation system consists of four basic function parts; sensors, signal conditioning, signal processing, and indicators.

References

Standards publications from Instrument Society of America

[1] S5.1-Instrumentation Symbols and IdentificationANSI/ISA-1984 (Reaffirmed 1992) **2ISBN: 0-87664-844-8**

[2] S5.2-Binary Logic Diagrams For Process Operations-ANSI/ISA-1976 (Reaffirmed 1992) **ISBN: 0-87664-331-4**

[3] S5.3-Graphic Symbols for Distributed Control/Shared Display instrumentation, Logic, and Computer Systems-1983 **ISBN: 0-87664-707-7**

[4] S5.4-Instirument Loop DiagramsANSI/ISA-1991, **ISBN: 1-55617-227-3**

[5] S5.5-Graphic Symbols for Process Displays-ANSI/ISA-1985, **ISBN:0-87664-935-5**

Other references

[6] M.R. Skrokov, Mini-and Microcomputer Control in Industrial Processes, Van Nostrand Reinhold Company, 1980.

[7] Curtis D. Johnson, Process Control Instrumentation Technology, 5th ed., Printice Hall, 1997.

[8] R. S. Figliola and D.E. Beasley, Theory and Design For Mechanical Measurements, 2nd ed. John Wiley & Sonss, Inc, 1995.

[9] C. A. Smith and A.B. Corripio, Principles and Practice of Automatic Process Control, John Wiley & Sons, Inc, 1985.

[10]John R. Bentley, Principles of Measurement Systems, 3rd ed. , Longman Group Limited, 1995.

[11]*American National Standard,* American National Standards Institute, New York.

[12]*ASTM Standards,* American Society for Testing and Materials, Philadelphia, PA.1-

[13]W.G.Andrew, Applied Instrumentation in the Process Industries, Vol.II, Gulf Publishing Company, 1974.

[14]John A. Hill, “10th Annual Control Market Outlook”, ISA InTech magazine, pp.36-42, January, 1997.

[15]W.G. Wilbauk,”50 Years of Progress in Measuring and Controlling Industrial Processes”, IEEE Control Systems Magazine, pp. 62-65, 1996.

Further reading

IEEE 8023: Carrier Sense Access with Collision Detection.

IEEE-802-5: Token Ring Access Method. (www.ieee.org; IEEE, 445 Hoes Lane, Piscataway, NJ 08855-1331)

Lawrence M. Thompson, *Industrial Data Communications: Fundamentals and applications*, ISA

Paul J. Fortier, *Handbook of LAN Technology,* 2nd edition, McGraw-Hill Publishing, ISBN 0-07-021625-8

Frank J. Derfler, *Guide to Connectivity,* 2nd edition, Ziff-Davis Press, ISBN 1-56276-047-5

Uyless Black, *TCP/IP and Related protocols,* McGraw-Hill Publishing, ISBN 0-07-005553-X

John E. McNamara , *Technical Aspects of data Communication,* Digital Press,
ISBN 1-55558-007-6

Daniel Capana, Fiber optics: the medium, *Intech, ISA*, April 1997.

Exercises

1. Define the industrial network CAN and discuss its essential characteristics.
2. Discuss ISO 11898 and ISO 11519 in conjuncture with CAN.
3. Situate CAN in the layers of the ISO reference model.
4. Explain the functions of the two layers of CAN.
5. Compare CAN and DeviceNet.
6. Elaborate on the features of DeviceNet.
7. What layers does DeviceNet specify?
8. Discuss in detail all the components of Foundation Fieldbus.
9. Discuss the three versions of the PROFIBUS family. Summarize the various differences.
10. What layers does HART implement in the ISO Reference model?
11. Define the following terms and indicate in what network are they used:
    - Medium Access Control.
    - Logical Link Control.
    - Non-Destructive Bitwise Arbitration.
    - Status field.
    - Data field.
    - Checksum.
    - Universal Commands.
    - Common Practice Commands.
    - Device Specific Commands.