Module
8
Industrial Embedded and Communication Systems

Lesson
38
Networking of Field Devices via Fieldbus

Lesson Objectives

- To motivate a field level networked digital communication architecture for implementation of distributed plant wide control
- To describe the Fieldbus network protocol
- To describe the basic computation and communication architecture for Fieldbus devices
- To explain issues related to time synchronization, interoperabilty, communication efficiency etc. in the Fieldbus network.

1. Introduction

Embedded electronics technology has given rise to significant rise in the number of automatic devices for industrial data acquisition, transmission, monitoring, diagnostics, control and supervision. Each of these devices is configurable and capable of two way communication with other devices. Effective use of their capabilities can only be enabled by reliable and high speed communication architecture for extensive and rapid information exchange among automation devices for coordination and control. Below we introduce some of the major motivations that led to major users and suppliers from the U.S., Japan and Europe coming together to establish the Fieldbus Foundation in 1994. Their objective has been to develop a worldwide, unified specification of "Fieldbus", a network communication architecture for field devices for process control and manufacturing automation.

Motivations for the Fieldbus

Among the major motivations for the Fieldbus are the following.

- Replacement of analog and digital (serial) point-to-point communication technology with much superior digital communication network for high speed ubiquitous and reliable communication within a harsh industrial environment.
- Enhanced data availability from smart field bus devices needed for advanced automation functions such as control, monitoring, supervision etc.
- Easy configurability and interoperability of system components leading to an easily installable, maintainable and upgradeable open system that leverages the computing and networking hardware and software solutions

In industrial automation systems, the field signals have been traditionally transmitted to the control room using point-to-point communication methods that employ analog technologies such as the 4-20 mA current loop or, more recently, digital ones such as the RS-422 or RS-485. The main disadvantages of this are the highly increased cost of cabling due to the need for a separate pair of wires for each device connected to the mainframe. Apart from this, with 4-20 mA analog current loop, signals can be transmitted only in one direction. With the need for more complex monitoring and control of a process plant, installation and maintenance of these point to point communication media and their signal integrity become more and more difficult. As an alternative the network communication architecture presents an attractive option. Firstly the cabling requirements are marginally increased as more and more devices are added to the network. Secondly, a vast array of high speed networking technologies is available at attractive costs from the computer market. Thirdly, with the addition of intelligent devices, such a system enables advanced monitoring supervision and control, leading to improvements in productivity, quality and reliability of industrial operations.

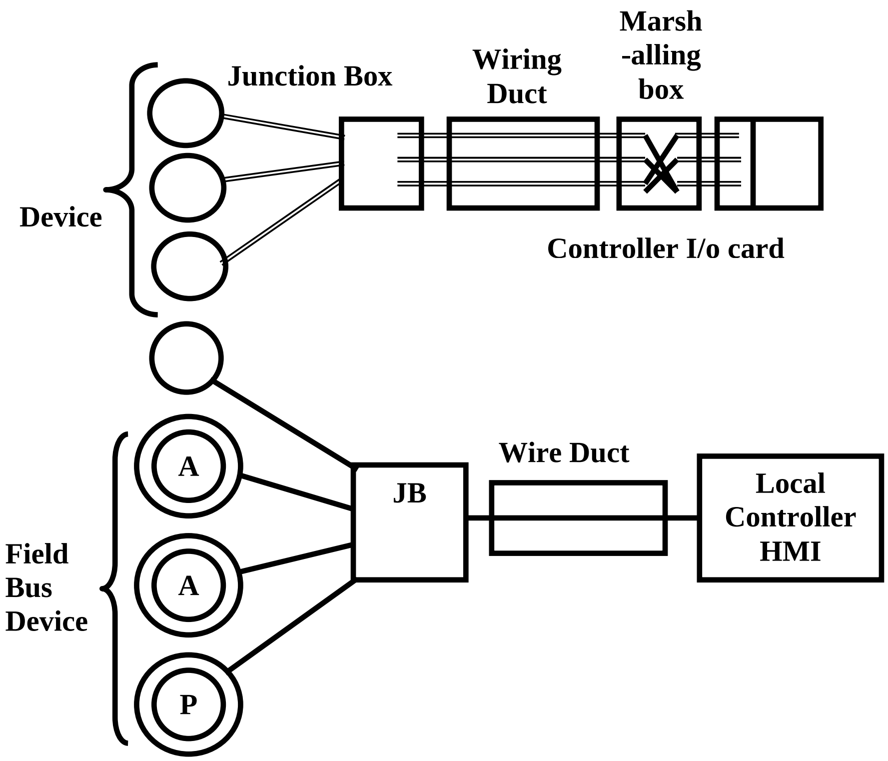

Fig. 38.1 Wiring system for conventional point-to-point communication systems and the Fieldbus

Fieldbus is a standard for Local Area Network (LAN) of industrial automation field devices that enables them to intercommunicate. Typical Fieldbus devices are sensors, actuators, controllers of various types, such as PLCs, and DCS, and other computer systems such as human-machine interfaces, process management servers etc. It includes standards for the network protocol as well as standards for the devices on the network.

Fieldbus allows many input and output variables to be transmitted on the same medium such as, a pair of metallic wires, optical fibre or even radio, using standard digital communication technologies such as baseband time-division multiplexing or frequency division multiplexing. Thus sensors transmit the measured signal values as well as other diagnostic information; the controllers compute the control signals based on these and transmit them to actuators. Further, advanced features such as process monitoring can be carried out leading to increased fault tolerance. Online process auto-tuning can be performed leading to optimized performance of control loops.

Table 1 compares some of the key features of 4-20mA and Fieldbus technology. It should be mentioned that Fieldbus becomes cost-effective only beyond a certain scale of operations.

Table 38.1 Comparison of Fieldbus with 4-20mA current loop

Item No.	Specification	4-20mA	Fieldbus
1.	No. of devices per wire	1	32
2.	Qty. of data/variables per device	1	Up to thousands
3.	Control functions in field	No	Yes
4.	Device Failure Notification	Minimal (O/C, S/C)	Yes, detailed
5.	Signal degradation over wire	Possible	None
6.	Power distribution over wire	Yes	Yes
7.	Interchangeability of field devices	Yes	Yes (with some restrictions)
8.	Maximum run-length	2Km	1.9Km (5.7 Km with repeaters)
9.	Failure diagnosis	Technician reqd.	Operator informed at console
10.	Intrinsic safety	With barriers	With barriers
11.	Sampling delay	Vendor defined	User defined (within limits)

Fieldbus technology was designed for geographically distributed harsh environment of process control applications. Also, it was conceived that there would be frequent changes in the installations. To meet these requirements the protocol includes the following aspects which are not necessarily found in other Protocols:

- Control algorithms may be in field-mounted Devices, central controlled or a combination of both.
- The End User does not have to be concerned with Device numerical address allocation. The Protocol handles this task, so 'plug and play' services are available for commissioning, modification and replacement.
- Devices do not have to be 'configured' before they are attached to the network.
- Device Definition and Function Blocks create a standard vendor-independent device interface for each device type which, in turn, facilitate installation, commissioning and upgradation of multi-vendor applications.
- The Physical Layer of the Protocol was designed from the outset to cope with installed cables and flammable atmospheres (hazardous areas).
- Both precise cyclic updates as well as acyclic and sporadic communications are catered for within the Protocol.
- Each variable transmitted on the Fieldbus carries with it tags indicating the current health of the source. Using this information, recipient Devices can take appropriate action immediately (for example switch to Manual, Off-line, etc.).

Point to Ponder: 1

A. *How exactly does a network lead to reduced cabling over a point-to-point communication topology?*

B. *Can you cite three reasons why the reduction in cabling is considered a significant advantage of the Fieldbus?*

C. *Can you state two advantages of interoperability?*

D. *For what kind of factories is a Fieldbus implementation for plant automation justified?*

2. Fieldbus Topology

As shown in Figure 38.1, Fieldbus generally uses one of the two topologies - Bus and Tree. With the Bus Topology, devices are connected to the network 'back-bone'. Either through a 'Drop Cable' Device, or are directly connected to the Bus by a 'Splice' connection. The Tree arrangement is used where a number of Devices share a similar location remote from the equipment room. A junction box, installed at the geographic center of gravity of the Devices, communicates with each Device is connected to it via a cable. In general Fieldbuses can use a combination of both topologies. Thus, trees can be hung from network buses.

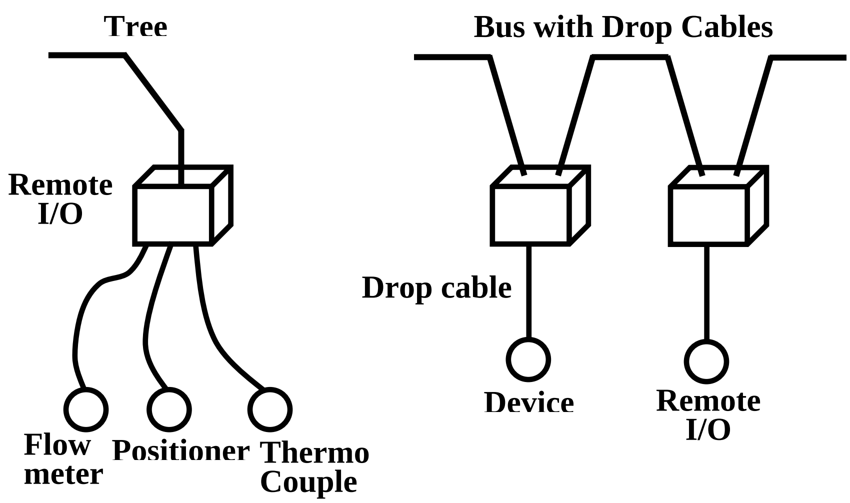

Fig. 38.2 Tree and Bus Structures for Fieldbus

3. Architecture of the Fieldbus

The Open Systems Interconnect (OSI) model published by the International Standards Organization is a well known definition of network communications based on seven generic layers. It defines seven generic 'Layers' required by a communication standard capable of supporting vast networks.

The first two layers, namely the Physical and the Data Link layers incorporate the technologies to realize a reliable, relatively error free and high speed communication channel among the communicating devices. It provides support for all standard and medium dependent functions for physical communication. DLL actually manages the basic communication protocol as well as error control set up by higher layers.

In Fieldbus, since the communication takes place over a fixed network routing and transport layers are made redundant. Moreover, in an industrial control environment, the network software entities or processes are also generally invariant. Under such a situation, requirements of the session and the presentation layers are also minimized. Therefore, the third, fourth, fifth and sixth layers of the ISO protocols have been omitted in the Field bus protocol. In fact the requirements of the omitted layers, although limited, have been included within the Fieldbus Application Layer (FAL) (7), which is sub-divided into two sub-layers, namely the Fieldbus Message Sub-layer (FMS) Fieldbus Message Sub-layer (FMS) that builds up a message data structure for communication as per requirements of user layer and includes the roles of the session and presentation layers of the ISO-OSI model , and the Field Access Sub-layer (FAS) that manages the functionality of the networking and transport layers to the extent needed and provides a virtual communication channel. Thus, the Foundation Fieldbus utilizes only three ISO model Layers (1, 2 and 7), plus an additional Layer referred to as the User Layer (8).

In the Fieldbus standard, the User Layer (8) is also included in the specification. In this it differs from other communication standards. A typical function of the User Layer is to define control tasks for a process plant. These are achieved through abstract software units called Function Blocks. Defining the User Layer functionality in terms of the open and published standards of Function Blocks enables interoperability of devices from different vendors. This is because any two devices that implement the standard abstract function block interface would interoperate, irrespective of their internal implementations.

Fieldbus Foundation has standardised a range of Function Block communications interfaces. The content of a Function Block is not standardised. For example: Company A and Company B may both supply PID control algorithms within their products. The Fieldbus Foundation specification dictates how each vendor's PID Function Block shall communicate Set-point, Controlled Variable, P, I &D constants etc., but not how the Function Block's internal algorithms would be realised.

The fieldbus protocol structure is shown alongside with that of the ISO-OSI model in Fig. 2. Below we discuss each of the above layers of the Fieldbus in more detail.

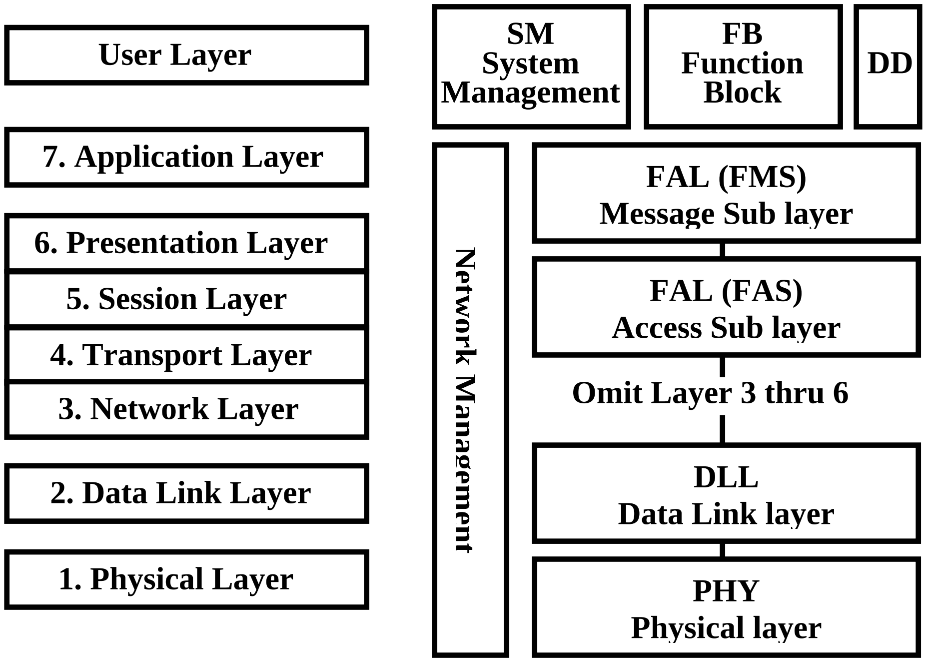

Fig. 38.3 Fieldbus Network Architecture vis-à-vis OSI

Point to Ponder: 2

A. Why are network and transport layers absent in a Fieldbus network?

B. Why are session and presentation layers absent in a Fieldbus network?

4. The Physical Layer

Fieldbus allows options for three types of communication media at this layer, namely, Wire, Fiber-optic and Radio. The Physical Layer is sub-divided into an upper section (the Media Independent Sub-Layer MIS) and a lower section which is media specific.

The MIS ensures that the selected Media interfaces in a consistent way with the Data Link Layer (2), regardless of the media used. The lower sections define the communications mechanism and media. For example, for wire medium they describe signal amplitudes, communication rate, waveform, wire types, etc.

An area-wide network can be implemented through the compartmentalization of the bus system in the bus segments that can be connected over repeaters. Standard-transmission rates can be in the range of from about 10KBaud 10 MBaud. The topology of the single bus segment is the line structure (up to 1200 m) with short drop cables (<0.3m). Transmission distances to 12 km are possible by electrical configuration and to 23.8 km with optical configuration. The distances are dependent on the transmission rate. With the help of repeaters, a tree structure can also be constructed as shown:

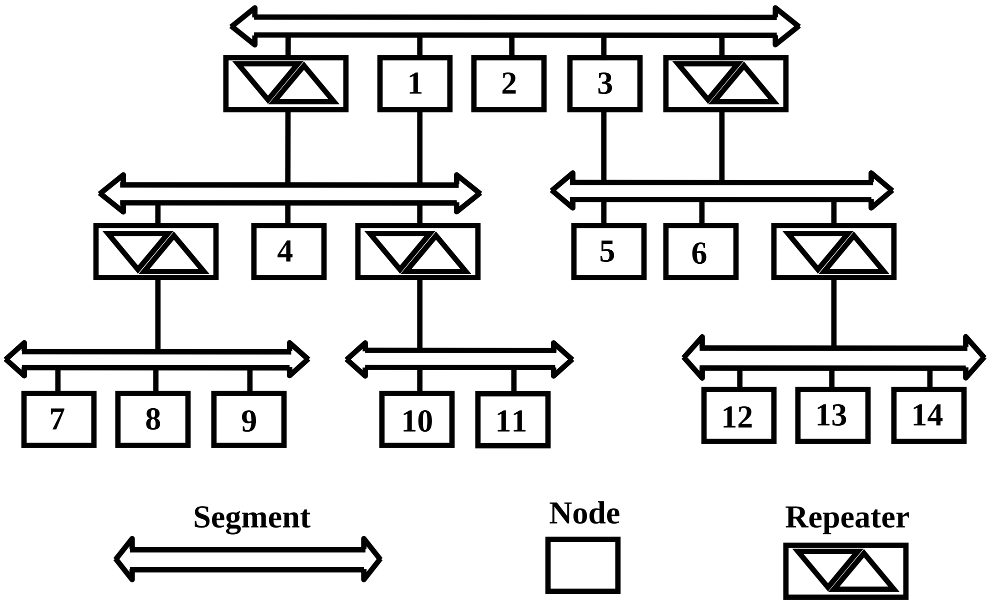

Fig. 38.4 Multi-bus segment Fieldbus network topology.

The maximum number of nodes per bus segment amounts to 32. More lines can connected under one another through performance enhancements (repeaters) where by it is noted that each repeater counts as a node. In total a maximum of 128 nodes are connectable (over all bus segments).

Point to Ponder: 3

A. *Why are repeaters required for extending the bus in a network?*

B. *Why does the maximum bus length depend on the transmission rate?*

6. The Data Link Layer

As the medium of transmission is a bus network, all device communications take place over the same physical medium. A mechanism is therefore necessary to ensure that it is shared effectively without collisions, i.e., when one device transmits none other does. The Fieldbus Data Link Layer protocol is a hybrid protocol that is capable of supporting both scheduled and asynchronous transfers. Its maximum packet size is 255 bytes.

It defines three types of data link layer entities, a Link Master(LM), a Basic Device(BD), and a bridge. Link master devices are capable of assuming the role of the bus master, called the link active scheduler (LAS). At any point of time only one of the LM devices act as the LAS. This is depicted in Figure 38.3.

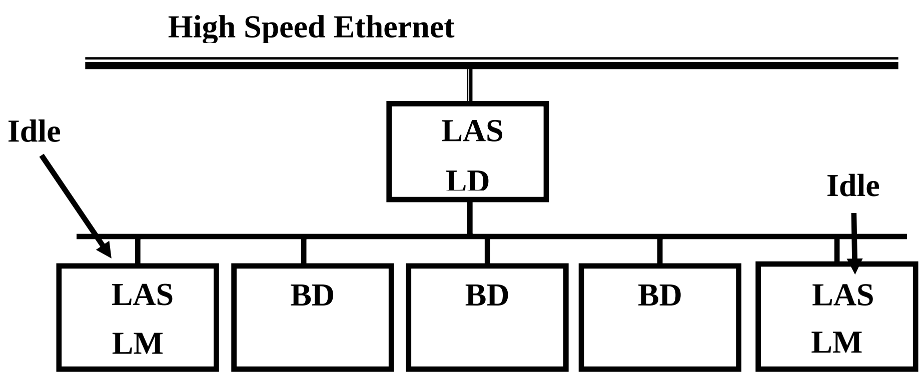

Fig. 38.5 Link Active Scheduler, Link Masters and Basic Devices for a Fieldbus implemented on a High Speed Ethernet

Basic devices are those devices not capable of becoming the LAS. They receive and send published data, and they receive and use tokens. When they hold the token, they are capable of initiating communications with all devices on the network.

Bridge devices connect link segments together. Bridged networks are configured into a spanning tree in which there is a single root link segment and a series of downstream link segments. Bridges interconnect the link segments. Each bridge may have a single upstream port (in the direction of the root) and multiple downstream ports (away from the root). The root port behaves as a basic device and the downstream ports are each the LAS for their downstream link.

Bridges are responsible for republishing scheduled transfers and forwarding all other traffic. Configured republishing and forwarding tables identify the packets that are to receive and republish or forward. Bridges are also responsible for synchronizing time messages received on their root port before regenerating them on their downstream ports.

6.1 The Link Active Scheduler (LAS)

One of the devices connected to the Fieldbus acts as the Link Active Scheduler (LAS). This decides which Device transmits next and for how long, thereby avoiding the collision of messages on the Bus. The LAS is responsible for the following list of tasks.

1. It detects the connection and disconnection of devices to the network, in order to maintain a "Live List" of functional devices and ensure they receive the "Right to transmit" when appropriate. Redundant LAS's maintain their own Live Lists in readiness to take over when the on-line LAS fails
2. It distributes time on the bus that can be used for scheduling and time stamping.
3. It polls device buffers for data according to a predefined schedule. This capability is used to support publisher/subscriber virtual communication relationships.
4. It distributes a token to devices in its live list that they can use for asynchronous transfers. This capability is used to support client/server and report distribution virtual communication relationships.

The LAS controls all cyclic data transmissions in this manner. In free time the LAS passes a message called the Pass Token (PT) to each Device in turn allowing them to use this idle period.

As mentioned before, the Link Active Scheduler (LAS) controls communications traffic on the Fieldbus. This is also called "Bus master function". The active LAS grants a "right to transmit" to each device on Fieldbus in a pre-defined manner. Devices other than LAS can communicate only when they have the "right to transmit". There are two ways of granting a "right to transmit". One is a polling method, which grants a right to transmit in sequence to each device. Another is a time slot method, which grants a right to transmit at a fixed time interval. The LAS uses these two methods combined to meet the requirements of precise cyclic updates and unscheduled traffic, for example, alarm reporting.

6.1 Cyclic Communication

Typically, cyclic communications in industrial operations involving input output operations related to process control loops or PLC scan cycles. Such communications must be performed at precise update rates. The LAS meets the requirement of precise cyclic updates of variables by issuing a "Compel Data" message (called the CD Token), to each source of data according to a fixed schedule. On receiving the CD, the addressed device transmits the current data on the bus. This message contains a reference to the source of the data. Any other device on the bus requiring the data takes a copy for its own use, for example an HMI or a control loop. Note that only one transmission is required to satisfy many destinations.

The device transmitting the data is referred to as the "Publisher" and those who take copies are called "Subscribers". The publisher may not know which devices are subscribers. The publisher's data is referred to as a Data Transfer Process Data Unit, or DT for short.

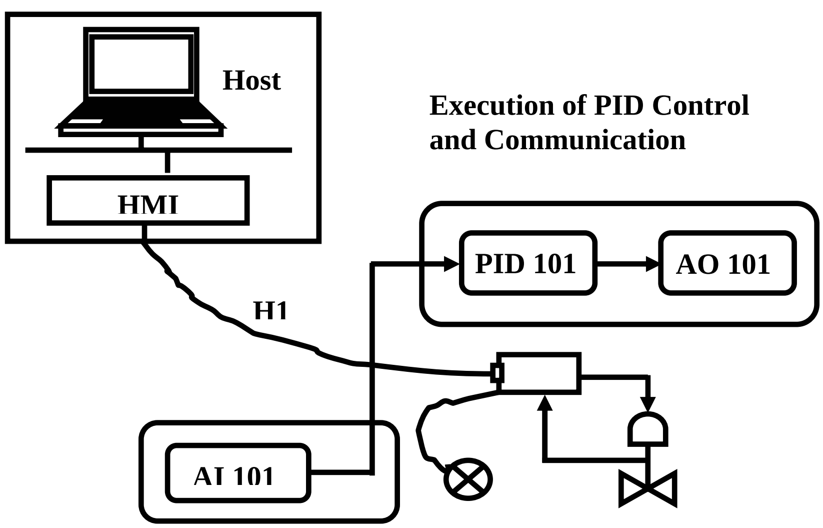

Fig. 38.6 Communication within a Control Loop

If a control loop requires a measured variable to be updated on a cyclic basis, the LAS instructs the source of the signal to transmit the variable by sending a special message called the Compel Data (CD) token. On receiving this message, the source transmits the variable on the bus. All devices on the bus receive the message, but only those with a use for the information take a copy.

In Figure 38.6, the Process Variable (PV) sensor transmits the measured variable when it receives the CD token. This is referred to as 'Publishing' the data. The control algorithm in the control valve copies it, as it is a Subscriber to this information. The HMI may also copy it for display and archiving purposes, but only one transmission of the PV is required.

6.2. Acyclic/Unscheduled Communication

Apart from cyclic communications, requirements for acyclic communications arise to handle sporadic process related events, such as,

- Alarm
- Operator Data Update
- Trend Data Update
- Set Point changes
- Controller Tuning

Once the requirements for cyclic data transmission have been met, the LAS will issue a Pass Token (PT) to each device in turn, thereby allowing them access to the bus to transmit data (a DT) or request data from another device, utilizing the bus up to an allocated time limit.

6.3 Macro Cycle and Elementary Cycle

A basic requirement of process control applications is that precise cyclic updates of process variables should be possible, to ensure good quality continuous control. Generally the number of such tasks in the system remains more or less fixed. Apart from these, communication tasks related to sporadic events, such as alarm reporting and operator changes of set points, must be scheduled. The LAS therefore organizes its overall schedule communication tasks in the system in “Macro Cycles”. The duration of each Macro Cycle is further subdivided into a number of “Elementary Cycles”. This is shown in Figure 38.7.

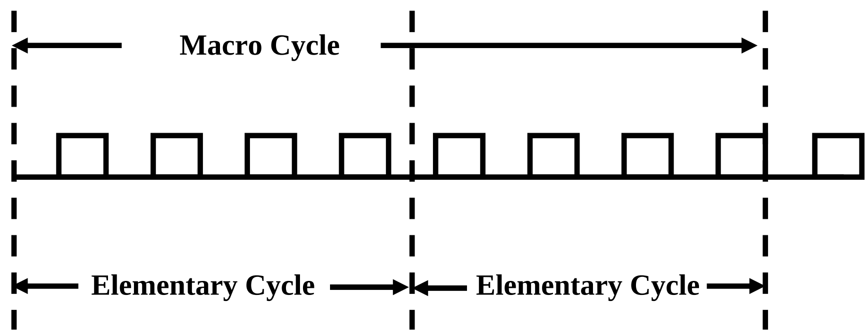

Fig. 38.7 Macro Cycles and Elementary Cycles

Each EC within an MC begins with the set of periodic tasks that is to be scheduled within that EC according to its update time period. The EC is chosen to be of such a duration that even after processing of the periodic tasks some time is left for servicing aperiodic tasks, should it be necessary, due to the occurrence of some event in the system.

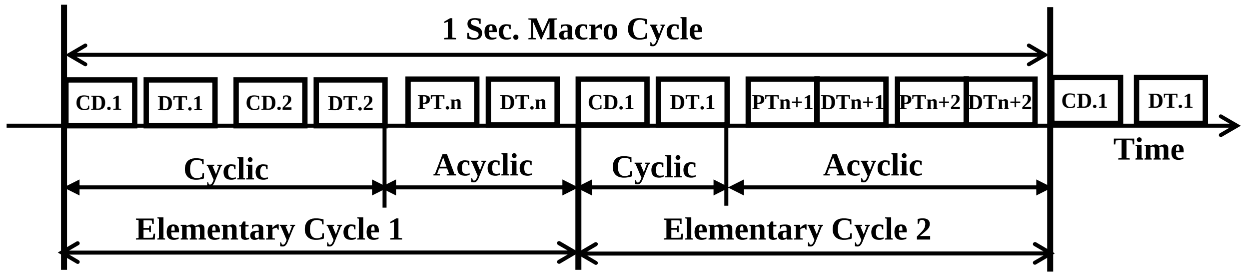

Fig. 38.8 An example task schedule

This is shown in Figure 38.8 in the case of a simple example of a system containing two devices requiring cyclic updates. The update requirements of the two devices are 1 sec. and 0.5 sec. respectively. The LAS sets the EC period as equal to the shortest update time requirement (0.5 Second. in this case). Similarly, the longest update time sets the MC period (1 Second. in this case).
The CD for Device 1 is generated at the beginning of each Elementary Cycle and the CD for Device 2 at the second time slot of alternate Elementary Cycles. In the 'free time' in each Elementary Cycle the LAS transmits PT's to devices on the Fieldbus segment in turn, allowing them to transmit unscheduled information. This is the unscheduled portion of the Elementary Cycle. There may be insufficient free time for all Devices to receive a PT before the end of the Elementary Cycle. In this case the LAS continues from where it left off in subsequent cycles.

Note that the time available for unscheduled traffic varies from one Elementary Cycle to another. For example in the first EC of an MC in the example, both periodic updates takes place, while in the second EC only one does, since the update requirement of device 2 is lower. Also, the CD's requiring the shortest update intervals are dealt with first in each Elementary Cycle. Thereby ensuring the interval between subsequent updates remains constant.

Point to Ponder: 4

A. *How does the LAS know how much time to give to which process and in what sequence?*

B. *Is the proposed medium access strategy adequate for industrial automation?*

C. *Can you think of an alternative strategy for network communication?*

7. The Application Layer

The objective of the Application Layer is to convert data and requests for services coming from the User Application (Layer 8), into demands on the communication system in the Layers below, and to provide the reverse service for received messages. Thus the application layer abstracts the technical details of the network from the user layer which can view the network devices to which communication is needed as if they are connected by virtual point to point communication channels. The Application Layer is subdivided into two sublayers namely the Fieldbus Access Sublayer (FAS) and the Fieldbus Message Sublayer (FMS). These are described below.

7.1 Fieldbus Access Sublayer

The FAS sits in-between the FMS and the DLL. The FAS provides three fundamental kinds of communications. The services offered by the higher layers such as the FMS are realized by the FAS using one of these modes of communication. They are described below.

7.1.1. One-to-one Bi-directional (QUB)

QUB is used for the communication between a device, which requests data on Fieldbus, and a device which provides the data. A typical example of such a communication is screen display updates and the change of setting of the function block parameter, etc. through an operator's station. QUB is initiated by a device (client) requesting read/write of parameters, and is terminated when another device (server) returns a response. Therefore, the communication is of the bi-directional, and confirmed, that is with acknowledgement from the server.

7.1.2. One-to-one Unidirectional 1 (BNU)

This type of communication is used for the distribution of data, which is generated in one device (publisher) and is transmitted to one or more devices (subscribers). The publishing application writes the data into a distributed network buffer. The network is responsible for copying the data to corresponding network buffers in subscriber devices. Subscribing applications subscribe to published data asynchronously by opening buffers for the receipt of the published data and identifying the associated publisher.

A typical example is a pressure transmitter sending measurement data as a Process Variable (PV), and a valve positioner receiving it and using it to modulate a valve.

Unlike QUB, BNU is initiated cyclically according to schedule, not by a request for data. Neither does it involve a response from the server. This is an unconfirmed communication service in single direction. BNU uses a connection-oriented service in the data link layer.

7.1.3. One-to-one Unidirectional 2 (QUU)

With QUU, one device on Fieldbus generates data, and interested recipients take a copy. The transmitter is referred to as the 'Source' and the recipients as 'Sinks'. This is typically for multicasting event reports and trend reports. Unlike published data, reports are sent to preconfigured group addresses when the bus is not scheduled for the transfer of published data. These virtual communication relationships used connectionless transfers. Note that QUU differs from BNU in that communication is unscheduled and new data does not overwrite older data in the recipient devices.

7.2 The Fieldbus Message Sublayer (FMS)

The FMS acts as the interface between the User Layer and FAS. There is a logical framework called Virtual Field Device (VFD), which manages various functions and parameters at the user layer.
A Fieldbus Device must have at least two VFD's, one for administering the network, the other for the control of the system or function blocks. The former has the parameters related to setting up the communication, the latter has the parameters related to Function Blocks defined by user layer and required by the control application.

The process control oriented VFD in a Fieldbus device is its Function Block Application Process (FBAP). Conceptually the Fieldbus specification allows for the development of other Application Processes in the future, for example a PLC Application Process might be defined.
In one field device, there are hundreds of parameters, such as the name of apparatus, an address, status variables and operating modes, function blocks, and those composed of data files. These parameters are defined as the objects in a VFD. They can be treated systematically, and are independent of the specification of the physical device.
Each VFD is an "object" and within it there are other objects. An index of these objects, referred to as the Object Dictionary (OD) is provided within the VFD. It details each object within the VFD, their data types and definition.
When another device, say a HMI host, wishes to access this data it can interrogate the VFD to determine what is available, its format etc.. This facility aids interoperability as well as automated configurability.

Point to Ponder: 5

A. *What is the advantage of implementing a virtual field device?*

B. *Why are most of the communications unidirectional in Fieldbus?*

8. Fieldbus Devices

Field devices are control devices connected to Fieldbus network. They execute analog and discrete I/O functions plus the algorithms necessary for closed loop distributed control.
From a communications perspective, field devices are composed of three components, namely, the function block application, the system management agent, and the communication stack, which includes the network management agent. This architecture for a field device, and its components, is described in detail by the Fieldbus Foundation Specifications. An overview is presented below.

8.1 Communications Stack

The communication stack of a field device is a three layer stack comprised of the Fieldbus physical, data link, and application layer protocols described above. The communication stack also contains a network management agent that provides for the configuration and management of the stack.

8.2 Transducer Block

Transducer Blocks may be output, input or a combination of the two. They interface between Fieldbus and the real world of sensors and actuators. An input Transducer Block converts signals coming from the plant into Fieldbus compatible variable and status messages. Output Transducer Blocks do the reverse. The content of Transducer blocks implementations are specific to the hardware technology they represent and consequently varies from vendors to vendors. They insulate function blocks from these specifics, making it possible to define and implement technology independent function blocks. Thus, while standardization is achieved through function blocks described below, technological innovations in terms of electronics or signal processing is not stifled.

8.3 Resource Block

Resource Block contains the resource information for hardware and software within the Fieldbus device. For example, the device type, the manufacturer's name and data such as, serial number and available memory capacity - are stored as parameters. Only one resource block exists in each Fieldbus device.

8.4 Function Block

The primary purpose of a field device is to perform low level I/O and control operations. The *Function Block Application Process* (FBAP), as defined by the Fieldbus Foundation, models these operations of a field device. The structure of an FBAP is shown in Figure 38.8. The FBAP is composed of a set of *function blocks* configured to communicate with each other. Outputs from one function block are linked to the inputs of another through configuration parameters called *link objects*. Function blocks may be linked within a device, or across the network. Function blocks are scheduled to execute their algorithms at predefined times that are coordinated with the transfer of their inputs and outputs. During the execution of the function block, the algorithm may detect events and trend parameter values (collect a series of values for subsequent reporting). Reporting of events and trends can be performed by multicasting them onto the bus to a group of devices. In addition some other types of objects such as *View Objects, Alert Objects* etc. may also be associated with an FBAP. These objects perform typical tasks related to the Function Blocks in the device. For example,

- The 'Alert Object' monitors the status of various kinds of blocks, and reports to the upper system with a time stamp, if a configured alarm or event is detected.
- The 'Trend Object' stores trend data within a device and sends it in one file upon request. This improves the communication efficiency.
- Similarly, the 'View Objects' construct dynamic files of variables, status indications etc., collected from various blocks, and required by external devices for monitoring and control purposes. By bundling together the required information it can be sent in a single transmission, thereby saving communication time.
- Finally, the 'Link Object'. Interfaces the function blocks and the objects to the FMS for implementing the configured Virtual Communication Channels between FBAPs residing within network devices

A function block is essentially a program that contains a set of network visible input parameters, output parameters, internally *contained* parameters, and an algorithm to process them. Parameters are identified by an index or a name (not recommended) that locates them in the *object dictionary* associated with the function block application. The object dictionary contains information used to encode and decode parameters, such as type and length, and also is used to map the parameter index to a local memory address. To promote interoperability, interface devices can access the object dictionary.
Function blocks are connected to the physical hardware they represent through transducer blocks. Devices can be configured across the network through the use of *contained* block

parameters. Contained block parameters are those that can be written to the device by interface devices. Interface devices are not able to write values to input and output parameters.
Figure 38.10 shows how a communication relating to a physical variable takes place over the network. The value of the local physical variable is acquired by the function block through the transducer block firmware. This is processed by the function block and the output is communicated to another field device with the help of the Link Object. The Link object locates, from the object directory, the network address of the destination device as well as the mode of communication service to be used for the communication task. These are then realized by the FMS and the FAS sub-layers, in turn using the lower layers.

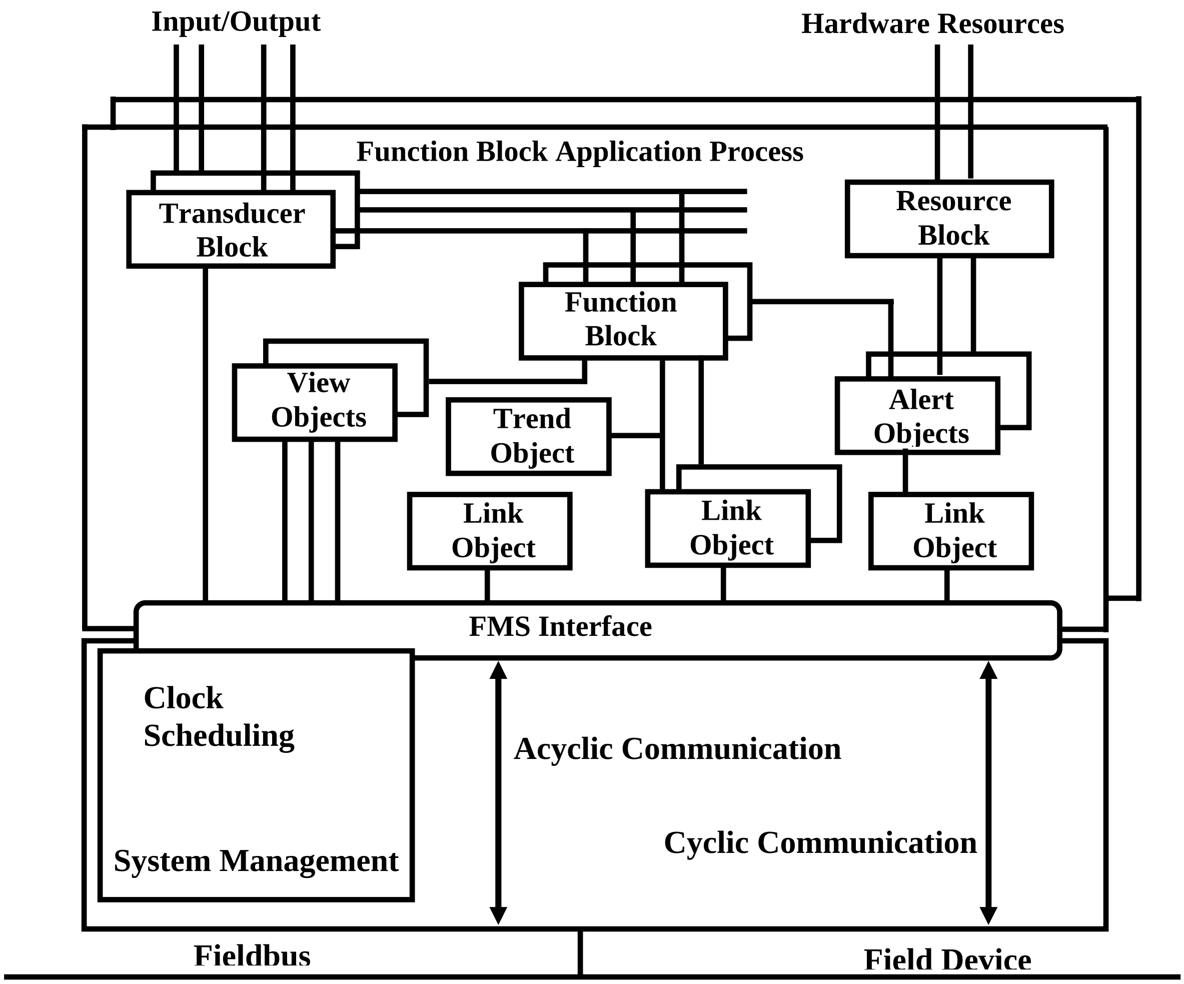

Fig. 38.9 Architecture of a Function Block Application Process

Point to Ponder: 6

A. *What is the difference between trend data and control data?*

B. *Explain the i/o connections for alert, link, trend and view objects.*

8.4.1 Realisation of Distributed Control Functions using Function Blocks in Fieldbus

Control functionality is realized over the network by a configured sequence execution of function blocks and communication tasks among them. For example, consider the control loop in Figure 38.6. The execution sequence is shown in Fig. 11. Note that there are three function blocks involved, namely, AI 101, PID 101 and AO 101. The first FB that executes is AI 101. This is followed by a cyclic communication of the process variable value to the PID 101 function block. The computation of the PID law in the FB PID 101 is followed by the computation of the valve stem position command to the positioner in the FB AO 101. Note there is no communication involved between these two FB executions, since both PID 101 and AO 101 are shown to be residing on the same Fieldbus device. Finally there is a communication between AO 101 and the host for the HMI station. This basic execution cycle repeats.

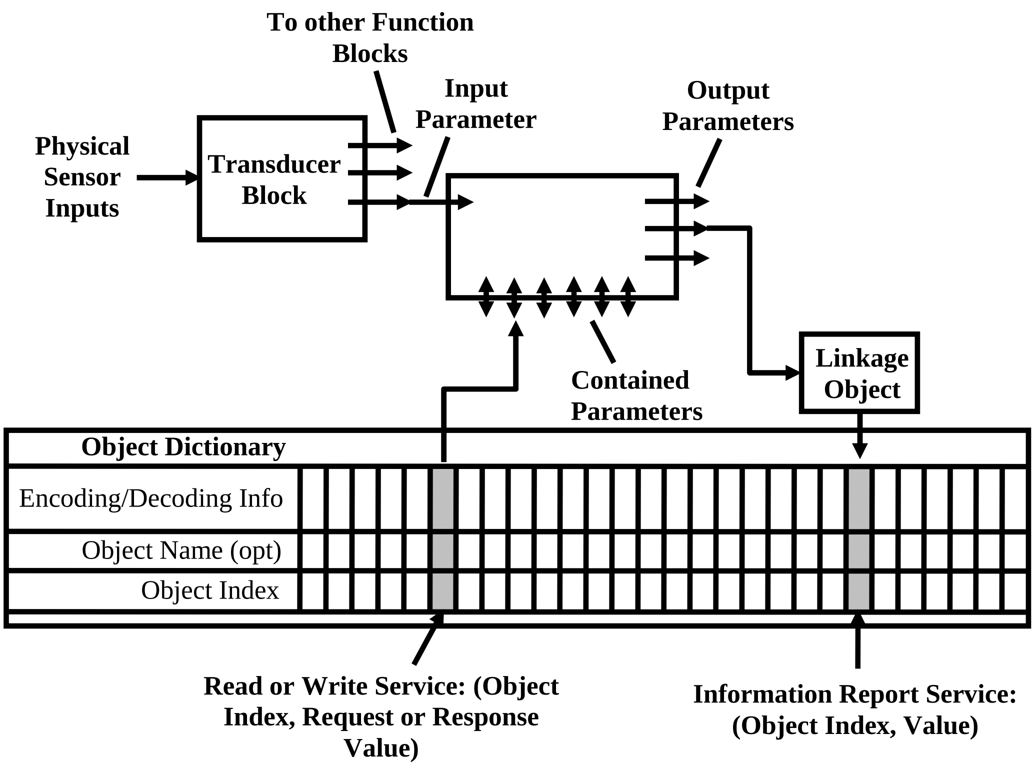

Fig. 38.10 Access to function block parameters through the object dictionary

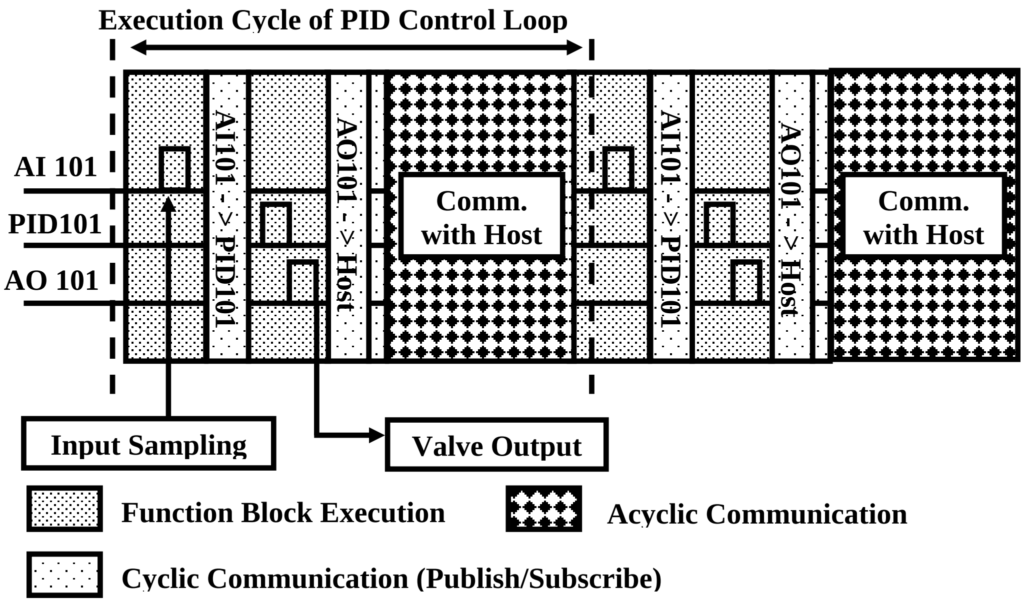

Fig. 38.11 Function Block execution and communication sequence for a control loop.

Point to Ponder: 7

A. *Is there any impact of the delay between input sampling and valve output, as shown in Fig. 11?*

B. *What is the relationship between the object dictionary and a function block?*

9. Network Management Structure

Network management is the function of managing various parameters to carry out Fieldbus communication. Generally, execution of the communication function is performed by communications software that resides in a communications ASIC.

The parameters for determining actual operation are called the Network Management Information Base (NMIB), and are grouped as one object. These parameters are accessed through Link Management Entity by the execution software at each layer. This function is transparent to the End User of the system.

10. System Management Functions

System Management (SM) performs the management of the parameters needed for the construction of a functional control system, rather than communication. The System Management Kernel is also modeled as an FBAP.

The System Management Kernel performs two primary functions. The first is to assign End User defined names, called *tags*, and data link layer addresses to devices as they are added to the fieldbus. It contains an object dictionary and can be configured and interrogated using FMS operating over client/server virtual communication relationships.

The second is to maintain distributed application time so that function block execution can be synchronized among devices. Fieldbus has a common clock for called Link Schedule Time (LS-Time). The LAS uses this to synchronize all devices on the bus frequently. Using LS-Time as a reference system management FBAP triggers each function block and synchronizes operation among Function Blocks in differing devices on the same bus. Furthermore, system management provides a real time reference, called Application time (AP-time). This time is used as the source of alarm or event time-stamps.

To support these functions, the System Management Kernel communicates directly with the data link layer.

11. Device Description

Electronic Device Descriptions (EDDs) created by Device Description Language (EDDL) for a field device support the management of intelligent field devices. Typical tasks such as operation, parameterizing and diagnostics can thus be solved efficiently. EDD describes product features which serve as a basis for the entire electronic product data management from planning to engineering, set-up, maintenance and diagnostics and the disintegration of a plant. EDDs are ASCII files. They primarily contain the description of all device parameters and their attributes (e.g. lower/upper value range, default value, write rights) and device functions, e.g. for the plausibility check, scaling, mode changes or tank characteristics. EDDs also include a grouping of device parameters and functions for visualization and a description of transferable data records.

Electronic Device Description Language (EDDL) is the mechanism that allows vendors to describe their products in a way that may be interpreted by any compliant host system. Thereby enabling compatibility and interoperability of devices. Also, the language allows vendors to include their specific product features while remaining compatible. Furthermore, the use of EDDL allows the development of new devices while still maintaining compatibility.
The Device Description (DD) may be supplied with the device on a disk, or down loaded from the Fieldbus Foundation web site, and loaded into the host system.

Point to Ponder: 8

A. *Device description languages enhance configurability and interoperability - justify or contradict.*

B. *What is the difference between Link Schedule Time and Application Time?*

Lesson Summary

In this lesson we have dealt with the following topics:

A. *Basic motivations for a networked communication architecture for distributed process automation:* It is seen that several advantages can be realised, such as, reduced cabling, reliable and high speed communication, configurability, interoperability, enhanced diagnostics etc.

B. *The communication architecture and its various layers:* The three layers of the Fieldbus protocol have been discussed. The possible network topologies and media are described. It is explained how the requirements of real-time communication are met using a shared medium in the protocol.

C. *Architecture of Fieldbus devices:* The software architecture of field bus devices is described. It is explained how the protocol is implemented within the device. The concept of Function and Transducer blocks are explained. The communication sequence involved in the distributed execution sequence of function blocks for realization of control functionality is explained. The mechanism of configuration using device description languages is introduced.

Answers, Remarks and Hints to Points to Ponder

Point to Ponder: 1

A. *How exactly does a network lead to reduced cabling over a point-to-point communication topology?*

Ans: The answer would be clear if you consider the sum of the lengths of line segments joining points on the circumference of a circle. This length grows proportional to the number of points. For a number of points more than 6, it crosses the length of the circumference of the circle. This is why the cabling requirement for point-to-point systems grows beyond that of a network.

B. *Can you cite three reasons why the reduction in cabling is considered a significant advantage of the Fieldbus?*

Ans:

1. Cost of data quality cables
2. Probability of interference and noise increasing with length of cable.
3. Complexity of maintaining a complicated hardware network of cables

C. *Can you state two advantages of interoperability?*

Ans:

1. Lower cost due to increased market competition
2. Better product functionality and quality of products

D. *For what kind of factories is a Fieldbus implementation for plant automation justified?*

Ans: For large factories employing sophisticated manufacturing processes, where the return on investment can be justified.

Point to Ponder: 2

A. *Why are network and transport layers absent in a Fieldbus network?*

Ans: Because the routing of messages in the network is fixed. Therefore the network layer is redundant. Also since the traffic is nearly fixed, transport layer functionality, such as flow control, are redundant.

B. *Why are session and presentation layers absent in a Fieldbus network?*

Ans: Because the task composition in the system and their communication requirements are fixed at configuration time. Also because the limited requirements of the layer are clubbed into the Application Layer.

Point to Ponder: 3

A. Why are repeaters required for extending the bus in a network ?

Ans: Repeaters clean up the distortion of digital data which increases with the length of the transmission channel.

B. Why does the maximum bus length depend on the transmission rate ?

Ans: Firstly because of increasing series mode interference along the channel. Secondly because of the increasing load capacitance seen by the channel drivers which needs to be driven at the higher data rates.

Point to Ponder: 4

A. How does the LAS know how much time to give to which process and in what sequence?

Ans: This is prefixed at configuration time by the configuration of the function blocks in the system.

B. Is the proposed medium access strategy adequate for industrial automation?

Ans: Yes it is. Since the sampling and control update rates are sufficiently low.

C. Can you think of an alternative strategy for network communication?

Ans: There are several other real-time networking protocols employed by networks such as Devicenet, Canbus etc.

Point to Ponder: 5

A. What is the advantage of implementing a virtual field device?

Ans: The advantage is that the network device tags and other communication parameters such as addresses, communication modes etc. can be made transparent to the high level function blocks.

B. Why are most of the communications unidirectional in Fieldbus?

Ans: Firstly because the nature of the typical application tasks are unidirectional. For example a transmitter only transmits data out to controllers. Secondly, communication takes place over short network channels and therefore in most cases data is received properly and the requirements for acknowledgements necessitating bidirectional communication does not exist.

Point to Ponder: 6

A. *What is the difference between trend data and control data?*

Ans: Trend data is generally averaged over several control data samples. Also requires a slower update rate.

B. *Explain the i/o connections for alert, link, trend and view objects.*

Ans: Alert objects report malfunctions in the device itself or abnormalities in the process. Thus it requires input from the hardware resources and function blocks. Link objects take input from all other blocks and outputs these in appropriate forms to the FMS. Trend and objects compute trend and view data based on the control data. Thus they take input from the function blocks.

Point to Ponder: 7

A. *Is there any impact of the delay between input sampling and valve output, as shown in Fig. 11?*

Ans: Introduces a time delay into the loop. However, the value of this delay is generally negligible in comparison with the dominant time constants of the control loop.

B. *What is the relationship between the object dictionary and a function block?*

Ans: The detailed communication and other parameters corresponding to function block parameters is stored in the object dictionary.

Point to Ponder: 8

A. *Device description languages enhance configurability, interoperability and maintainability - justify or contradict.*

Ans: Yes it does. The Device description is read by a host to re-configure the system to include the device and support its communications. Interoperability is enhanced, since the DDL provides a uniform syntax for describing products from all vendors.

B. *What is the difference between Link Schedule Time and Application Time?*

Ans: Link schedule time is relative time for scheduling tasks over elementary and macrocycles. Application time is running time until it is reset by specific events defined for the device.

Source:http://www.nptel.ac.in/courses/Webcourse-contents/IIT%20Kharagpur/Industrial%20Automation%20control/pdf/L-38(SM)%20(IA&C)%20((EE)NPTEL).pdf