Program : M.E.(Process control and Instrumentation)

Semester : II semester

Course code :EIPCPC21

Course title : - INDUSTRIAL DATA COMMUNICATION AND CONTROL

Course Teacher: Dr.G.Sakthivel
Professor
Department of Electronics and Instrumentation Engg.
Annamalai university

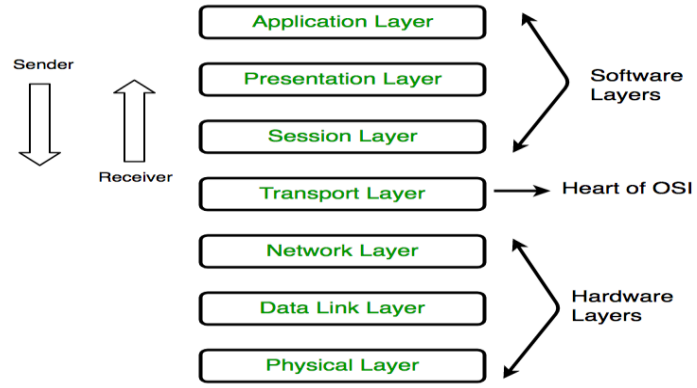
UNIT IV

Network Models and Protocols: OSI model - Data link Control protocol. Media access protocol: Command/response - Token passing - CSMA/CD, TCP/IP. Bridges - Routers - Gateways. Standard ETHERNET and Industrial ETHERNET Configuration - Special requirement for networks used for Control, Wireless LAN.Introduction to MODBUS, CANBUS, LON WORKS, FIP.

OSI Model

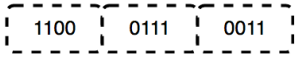
Layers of OSI Model

OSI stands for **Open Systems Interconnection**. It has been developed by ISO – '**International Organization of Standardization**', in the year 1984. It is a 7 layer architecture with each layer having specific functionality to perform. All these 7 layers work collaboratively to transmit the data from one person to another across the globe.



1. Physical Layer (Layer 1) :

The lowest layer of the OSI reference model is the physical layer. It is responsible for the actual physical connection between the devices. The physical layer contains information in the form of **bits.** It is responsible for transmitting individual bits from one node to the next. When receiving data, this layer will get the signal received and convert it into 0s and 1s and send them to the Data Link layer, which will put the frame back together.



The functions of the physical layer are :

1. **Bit synchronization:** The physical layer provides the synchronization of the bits by providing a clock. This clock controls both sender and receiver thus providing synchronization at bit level.
2. **Bit rate control:** The Physical layer also defines the transmission rate i.e. the number of bits sent per second.
3. **Physical topologies:** Physical layer specifies the way in which the different, devices/nodes are arranged in a network i.e. bus, star or mesh topolgy.
4. **Transmission mode:** Physical layer also defines the way in which the data flows between the two connected devices. The various transmission modes possible are: Simplex, half-duplex and full-duplex.

* Hub, Repeater, Modem, Cables are Physical Layer devices.

** Network Layer, Data Link Layer and Physical Layer are also known as **Lower Layers** or **Hardware Layers**.

2. Data Link Layer (DLL) (Layer 2) :

The data link layer is responsible for the node to node delivery of the message. The main function of this layer is to make sure data transfer is error-free from one node to another, over the physical layer. When a packet arrives in a network, it is the responsibility of DLL to transmit it to the Host using its MAC address.Data Link Layer is divided into two sub layers :

1. Logical Link Control (LLC)
2. Media Access Control (MAC)

The packet received from Network layer is further divided into frames depending on the frame size of NIC(Network Interface Card). DLL also encapsulates Sender and Receiver's MAC address in the header.

The Receiver's MAC address is obtained by placing an ARP(Address Resolution Protocol) request onto the wire asking "Who has that IP address?" and the destination host will reply with its MAC address.

The functions of the data Link layer are :

1. **Framing:** Framing is a function of the data link layer. It provides a way for a sender to transmit a set of bits that are meaningful to the receiver. This can be accomplished by attaching special bit patterns to the beginning and end of the frame.
2. **Physical addressing:** After creating frames, Data link layer adds physical addresses (MAC address) of sender and/or receiver in the header of each frame.
3. **Error control:** Data link layer provides the mechanism of error control in which it detects and retransmits damaged or lost frames.
4. **Flow Control:** The data rate must be constant on both sides else the data may get corrupted thus , flow control coordinates that amount of data that can be sent before receiving acknowledgement.
5. **Access control:** When a single communication channel is shared by multiple devices, MAC sub-layer of data link layer helps to determine which device has control over the channel at a given time.

** Packet in Data Link layer is referred as **Frame**.*
*** Data Link layer is handled by the NIC (Network Interface Card) and device drivers of host machines.*
**** Switch & Bridge are Data Link Layer devices.*

3. Network Layer (Layer 3) :

Network layer works for the transmission of data from one host to the other located in different networks. It also takes care of packet routing i.e. selection of the shortest path to transmit the packet, from the number of routes available. The sender & receiver's IP address are placed in the header by the network layer.
The functions of the Network layer are :

1. **Routing:** The network layer protocols determine which route is suitable from source to destination. This function of network layer is known as routing.
2. **Logical Addressing:** In order to identify each device on internetwork uniquely, network layer defines an addressing scheme. The sender & receiver's IP address are placed in the header by network layer. Such an address distinguishes each device uniquely and universally.

* *Segment* in Network layer is referred as **Packet**.

** Network layer is implemented by networking devices such as routers.

4. Transport Layer (Layer 4) :

Transport layer provides services to application layer and takes services from network layer. The data in the transport layer is referred to as *Segments*. It is responsible for the End to End Delivery of the complete message. The transport layer also provides the acknowledgement of the successful data transmission and re-transmits the data if an error is found.

• At sender's side:

Transport layer receives the formatted data from the upper layers,
performs **Segmentation** and also implements **Flow & Error control** to ensure proper data transmission. It also adds Source and Destination port number in its header and forwards the segmented data to the Network Layer.

• At receiver's side:

Transport Layer reads the port number from its header and forwards the Data which it has received to the respective application. It also performs sequencing and reassembling of the segmented data.

The functions of the transport layer are :

1. **Segmentation and Reassembly:** This layer accepts the message from the (session) layer , breaks the message into smaller units . Each of the segment produced has a header associated with it. The transport layer at the destination station reassembles the message.
2. **Service Point Addressing:** In order to deliver the message to correct process, transport layer header includes a type of address called service point address or port address. Thus by specifying this address, transport layer makes sure that the message is delivered to the correct process.

The services provided by the transport layer :

1. **Connection Oriented Service:** It is a three-phase process which include
   – Connection Establishment
   – Data Transfer
   – Termination / disconnection
   In this type of transmission, the receiving device sends an acknowledgement, back to the source after a packet or group of packet is received. This type of transmission is reliable and secure.
2. **Connection less service:** It is a one-phase process and includes Data Transfer. In this type of transmission, the receiver does not acknowledge receipt of a packet. This approach allows for much faster communication between devices. Connection-oriented service is more reliable than connectionless Service.

** Data in the Transport Layer is called as **Segments**.*
*** Transport layer is operated by the Operating System. It is a part of the OS and communicates with the Application Layer by making system calls.*
*Transport Layer is called as **Heart of OSI** model.*

5. Session Layer (Layer 5) :

This layer is responsible for establishment of connection, maintenance of sessions, authentication and also ensures security.

The functions of the session layer are :

1. **Session establishment, maintenance and termination:** The layer allows the two processes to establish, use and terminate a connection.
2. **Synchronization :** This layer allows a process to add checkpoints which are considered as synchronization points into the data. These synchronization point help to identify the error so that the data is re-synchronized properly, and ends of the messages are not cut prematurely and data loss is avoided.

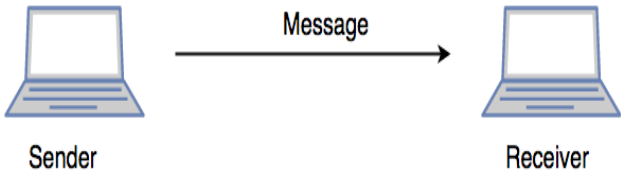
3. **Dialog Controller :** The session layer allows two systems to start communication with each other in half-duplex or full-duplex.

***All the below 3 layers(including Session Layer) are integrated as a single layer in the TCP/IP model as "Application Layer".*

***Implementation of these 3 layers is done by the network application itself. These are also known as **Upper Layers** or **Software Layers**.*

SCENARIO:

Let's consider a scenario where a user wants to send a message through some Messenger application running in his browser. The "Messenger" here acts as the application layer which provides the user with an interface to create the data. This message or so-called Data is compressed, encrypted (if any secure data) and converted into bits (0's and 1's) so that it can be transmitted.



6. Presentation Layer (Layer 6) :

Presentation layer is also called the **Translation layer**.The data from the application layer is extracted here and manipulated as per the required format to transmit over the network.

The functions of the presentation layer are :

1. **Translation :** For example, ASCII to EBCDIC.
2. **Encryption/ Decryption :** Data encryption translates the data into another form or code. The encrypted data is known as the cipher text and the decrypted data is known as plain text. A key value is used for encrypting as well as decrypting data.
3. **Compression:** Reduces the number of bits that need to be transmitted on the network.

7. Application Layer (Layer 7) :

At the very top of the OSI Reference Model stack of layers, we find Application layer which is implemented by the network applications. These applications produce the data, which has to be transferred over the network. This layer also serves as a window for the application services to access the network and for displaying the received information to the user.

Ex: Application – Browsers, Skype Messenger etc.

***Application Layer is also called as Desktop Layer.*

The functions of the Application layer are :

1. Network Virtual Terminal
2. FTAM-File transfer access and management
3. Mail Services
4. Directory Services

OSI model acts as a reference model and is not implemented in the Internet because of its late invention. Current model being used is the TCP/IP model.

Data link control protocol

A data link control is a service that ensures reliable network data communication by managing frame error detection and flow control. DLC is based on the Data Link layer of the OSI model.

DLC handles the following tasks:

- Reliable link packet transmission
- Recovery and error detection during high-layer packet retransmission
- Error framing, which determines start and end packetization via three approaches: length counts, bit-oriented framing and character-oriented framing

DLC character codes are based on standard character codes, such as the American Standard Code for Information Interchange (ASCII). Extended Binary Coded Decimal Interchange Code (EBCDIC) is comprised of hidden characters.

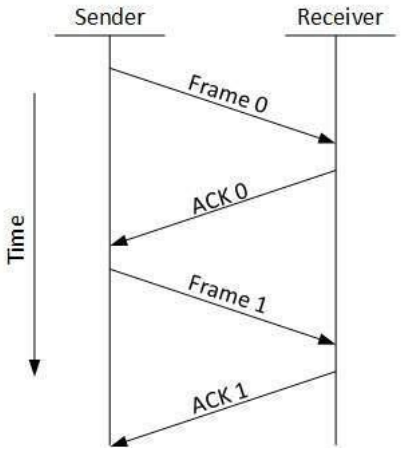
Flow Control

When a data frame (Layer-2 data) is sent from one host to another over a single medium, it is required that the sender and receiver should work at the same speed. That is, sender sends at a speed on which the receiver can process and accept the data. What if the speed (hardware/software) of the sender or receiver differs? If sender is sending too fast the receiver may be overloaded, (swamped) and data may be lost.

Two types of mechanisms can be deployed to control the flow:

- **Stop and Wait**

  This flow control mechanism forces the sender after transmitting a data frame to stop and wait until the acknowledgement of the data-frame sent is received.



- **Sliding Window**

  In this flow control mechanism, both sender and receiver agree on the number of data-frames after which the acknowledgement should be sent. As we learnt, stop and wait flow control mechanism wastes resources, this protocol tries to make use of underlying resources as much as possible.

Error Control

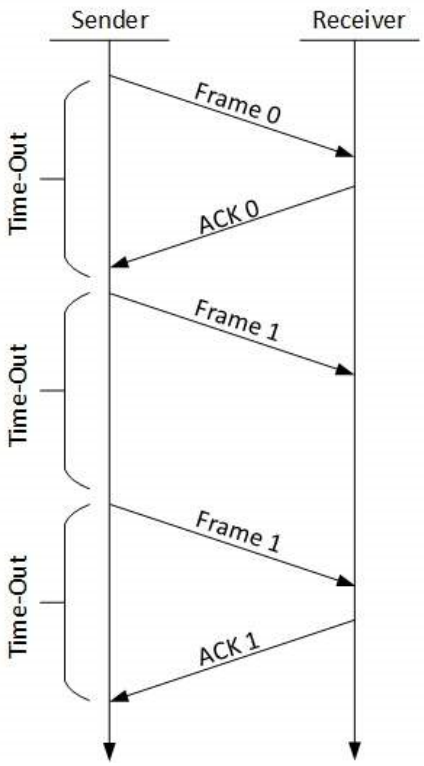
When data-frame is transmitted, there is a probability that data-frame may be lost in the transit or it is received corrupted. In both cases, the receiver does not receive the correct data-frame and sender does not know anything about any loss.In such case, both sender and receiver are equipped with some protocols which helps them to detect transit errors such as loss of data-frame. Hence, either the sender retransmits the data-frame or the receiver may request to resend the previous data-frame.

Requirements for error control mechanism:

- **Error detection** - The sender and receiver, either both or any, must ascertain that there is some error in the transit.
- **Positive ACK** - When the receiver receives a correct frame, it should acknowledge it.
- **Negative ACK** - When the receiver receives a damaged frame or a duplicate frame, it sends a NACK back to the sender and the sender must retransmit the correct frame.
- **Retransmission:** The sender maintains a clock and sets a timeout period. If an acknowledgement of a data-frame previously transmitted does not arrive before the timeout the sender retransmits the frame, thinking that the frame or it's acknowledgement is lost in transit.

There are three types of techniques available which Data-link layer may deploy to control the errors by Automatic Repeat Requests (ARQ):

- **Stop-and-wait ARQ**



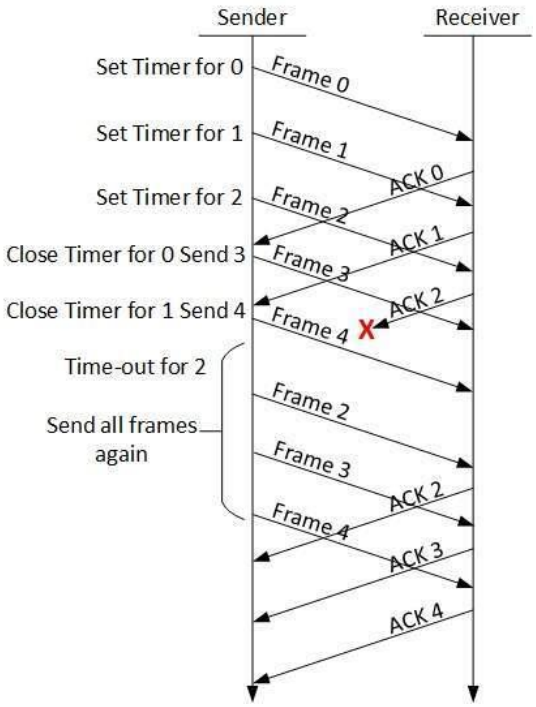
The following transition may occur in Stop-and-Wait ARQ:

- The sender maintains a timeout counter.
- When a frame is sent, the sender starts the timeout counter.

- If acknowledgement of frame comes in time, the sender transmits the next frame in queue.
- If acknowledgement does not come in time, the sender assumes that either the frame or its acknowledgement is lost in transit. Sender retransmits the frame and starts the timeout counter.
- If a negative acknowledgement is received, the sender retransmits the frame.

- **Go-Back-N ARQ**

Stop and wait ARQ mechanism does not utilize the resources at their best.When the acknowledgement is received, the sender sits idle and does nothing. In Go-Back-N ARQ method, both sender and receiver maintain a window.

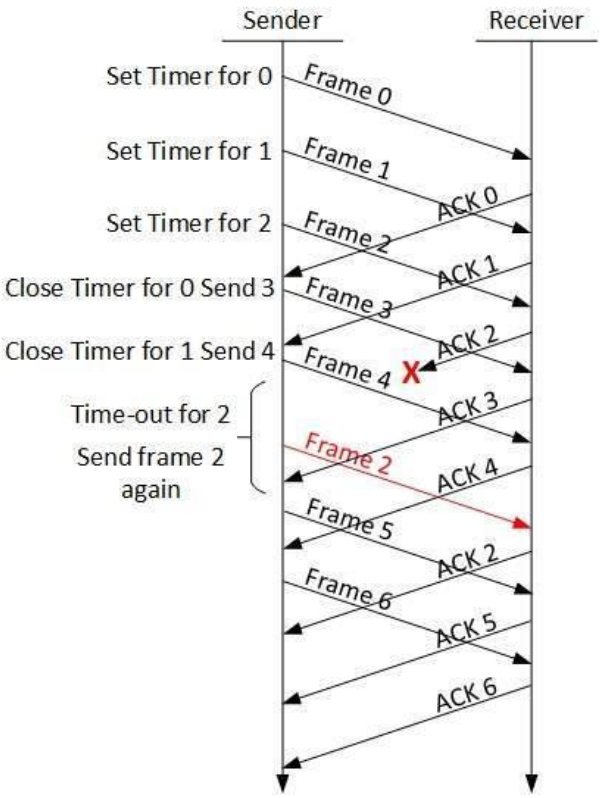


The sending-window size enables the sender to send multiple frames without receiving the acknowledgement of the previous ones. The receiving-window enables the receiver to receive multiple frames and acknowledge them. The receiver keeps track of incoming frame's sequence number.

When the sender sends all the frames in window, it checks up to what sequence number it has received positive acknowledgement. If all frames are positively acknowledged, the sender sends next set of frames. If sender finds that it has received NACK or has not receive any ACK for a particular frame, it retransmits all the frames after which it does not receive any positive ACK.

- **Selective Repeat ARQ**

In Go-back-N ARQ, it is assumed that the receiver does not have any buffer space for its window size and has to process each frame as it comes. This enforces the sender to retransmit all the frames which are not acknowledged.



In Selective-Repeat ARQ, the receiver while keeping track of sequence numbers, buffers the frames in memory and sends NACK for only frame which is missing or damaged.

The sender in this case, sends only packet for which NACK is received.

Medium access control (MAC)

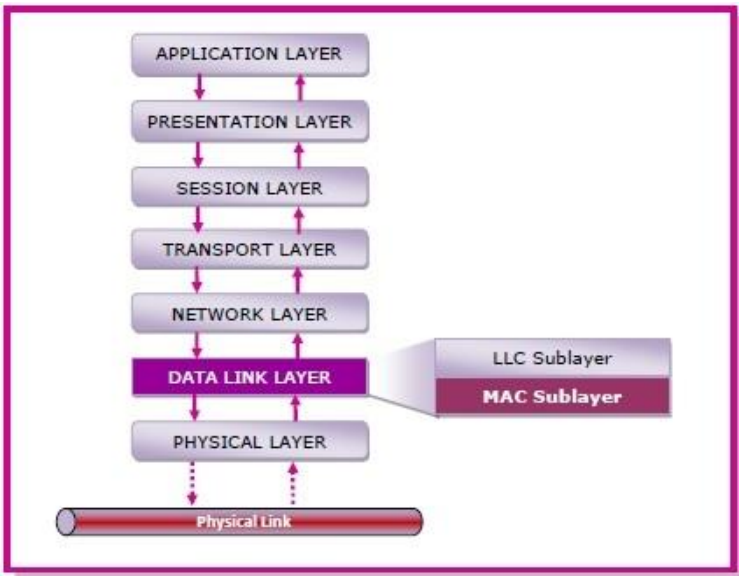
The medium access control (MAC) is a sublayer of the data link layer of the open system interconnections (OSI) reference model for data transmission. It is responsible for flow control and multiplexing for transmission medium. It controls the transmission of data packets via remotely shared channels. It sends data over the network interface card.

MAC Layer in the OSI Model

The Open System Interconnections (OSI) model is a layered networking framework that conceptualizes how communications should be done between heterogeneous systems. The data link layer is the second lowest layer. It is divided into two sublayers −

- The logical link control (LLC) sublayer
- The medium access control (MAC) sublayer

The following diagram depicts the position of the MAC layer −



Functions of MAC Layer

- It provides an abstraction of the physical layer to the LLC and upper layers of the OSI network.
- It is responsible for encapsulating frames so that they are suitable for transmission via the physical medium.
- It resolves the addressing of source station as well as the destination station, or groups of destination stations.
- It performs multiple access resolutions when more than one data frame is to be transmitted. It determines the channel access methods for transmission.
- It also performs collision resolution and initiating retransmission in case of collisions.
- It generates the frame check sequences and thus contributes to protection against transmission errors.

MAC Addresses

MAC address or media access control address is a unique identifier allotted to a network interface controller (NIC) of a device. It is used as a network address for data transmission within a network segment like Ethernet, Wi-Fi, and Bluetooth.

MAC address is assigned to a network adapter at the time of manufacturing. It is hardwired or hard-coded in the network interface card (NIC). A MAC address comprises of six groups of two hexadecimal digits, separated by hyphens, colons, or no separators. An example of a MAC address is 00:0A:89:5B:F0:11

Controlled Access Protocols in Computer Network

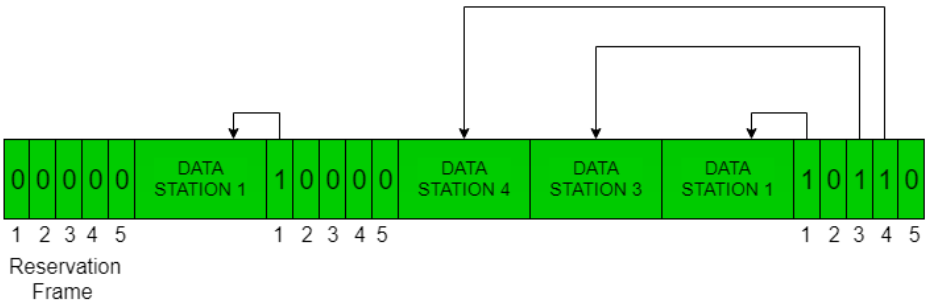
In controlled access, the stations seek information from one another to find which station has the right to send. It allows only one node to send at a time, to avoid collision of messages on shared medium.
The three controlled-access methods are:

1. Reservation
2. Polling
3. Token Passing

Reservation

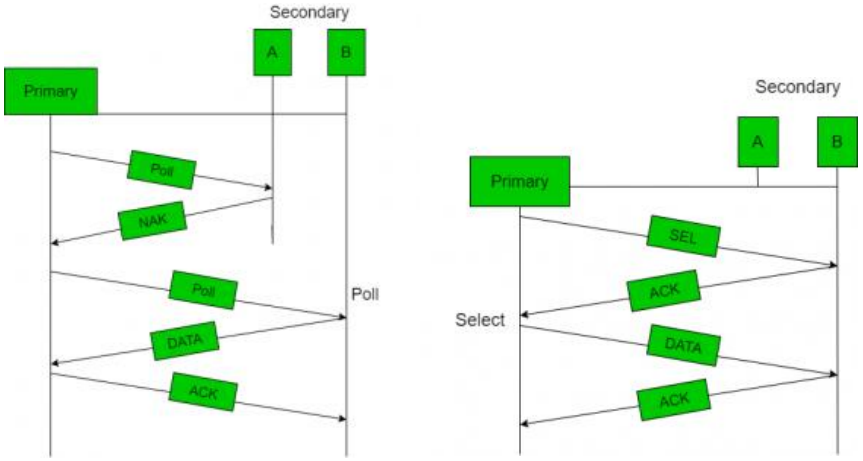
- In the reservation method, a station needs to make a reservation before sending data.
- The time line has two kinds of periods:
  1. Reservation interval of fixed time length
  2. Data transmission period of variable frames.
- If there are M stations, the reservation interval is divided into M slots, and each station has one slot.
- Suppose if station 1 has a frame to send, it transmits 1 bit during the slot 1. No other station is allowed to transmit during this slot.
- In general, i th station may announce that it has a frame to send by inserting a 1 bit into i th slot. After all N slots have been checked, each station knows which stations wish to transmit.
- The stations which have reserved their slots transfer their frames in that order.
- After data transmission period, next reservation interval begins.
- Since everyone agrees on who goes next, there will never be any collisions.

The following figure shows a situation with five stations and a five slot reservation frame. In the first interval, only stations 1, 3, and 4 have made reservations. In the second interval, only station 1 has made a reservation.



Polling

- Polling process is similar to the roll-call performed in class. Just like the teacher, a controller sends a message to each node in turn.
- In this, one acts as a primary station(controller) and the others are secondary stations. All data exchanges must be made through the controller.
- The message sent by the controller contains the address of the node being selected for granting access.
- Although all nodes receive the message but the addressed one responds to it and sends data, if any. If there is no data, usually a “poll reject”(NAK) message is sent back.
- Problems include high overhead of the polling messages and high dependence on the reliability of the controller.



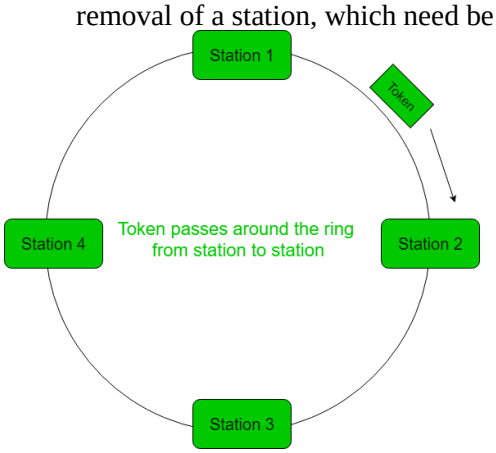
Efficiency

Let T_{poll} be the time for polling and T_t be the time required for transmission of data. Then,

Efficiency = $T_t/(T_t + T_{poll})$

Token Passing

- In token passing scheme, the stations are connected logically to each other in form of ring and access of stations is governed by tokens.
- A token is a special bit pattern or a small message, which circulate from one station to the next in the some predefined order.
- In Token ring, token is passed from one station to another adjacent station in the ring whereas incase of Token bus, each station
  uses the bus to send the token to the next station in some predefined order.
- In both cases, token represents permission to send. If a station has a frame queued for transmission when it receives the token, it can send that frame before it passes the token to the next station. If it has no queued frame, it passes the token simply.
- After sending a frame, each station must wait for all N stations (including itself) to send the token to their neighbors and the other N – 1 stations to send a frame, if they have one.
- There exists problems like duplication of token or token is lost or insertion of new station, removal of a station, which need be tackled for correct and reliable operation of this scheme.



Performance

Performance of token ring can be concluded by 2 parameters:-

1. **Delay**, which is a measure of time between when a packet is ready and when it is delivered.So, the average time (delay) required to send a token to the next station = a/N.
2. **Throughput**, which is a measure of the successful traffic.

Throughput, $S = 1/(1 + a/N)$ for $a<1$

and

$S = 1/\{a(1 + 1/N)\}$ for $a>1$.

where N = number of stations

$a = T_p/T_t$

(T_p = propagation delay and T_t = transmission delay)

CSMA with Collision Detection (CSMA/CD)

Carrier Sense Multiple Access with Collision Detection (CSMA/CD) is a network protocol for carrier transmission that operates in the Medium Access Control (MAC) layer. It senses or listens whether the shared channel for transmission is busy or not, and defers transmissions until the channel is free. The collision detection technology detects collisions by sensing transmissions from other stations. On detection of a collision, the station stops transmitting, sends a jam signal, and then waits for a random time interval before retransmission.

Algorithms

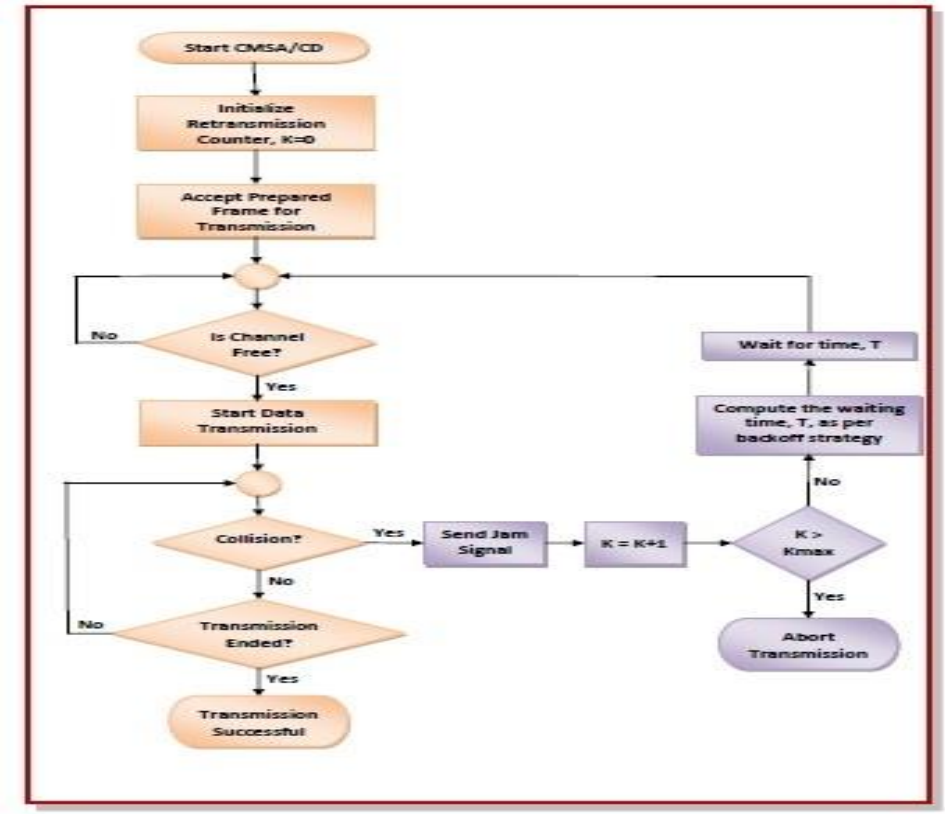
The algorithm of CSMA/CD is:

- When a frame is ready, the transmitting station checks whether the channel is idle or busy.
- If the channel is busy, the station waits until the channel becomes idle.
- If the channel is idle, the station starts transmitting and continually monitors the channel to detect collision.
- If a collision is detected, the station starts the collision resolution algorithm.
- The station resets the retransmission counters and completes frame transmission.

The algorithm of Collision Resolution is:

- The station continues transmission of the current frame for a specified time along with a jam signal, to ensure that all the other stations detect collision.
- The station increments the retransmission counter.
- If the maximum number of retransmission attempts is reached, then the station aborts transmission.
- Otherwise, the station waits for a backoff period which is generally a function of the number of collisions and restart main algorithm.

The following flowchart summarizes the algorithms:



TCP/IP Model

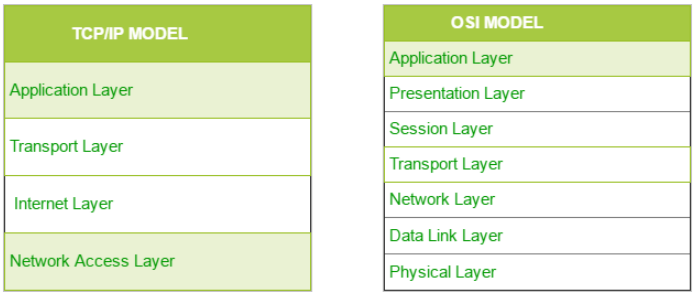
Prerequisite – Layers of OSI Model

The **OSI Model** we just looked at is just a reference/logical model. It was designed to describe the functions of the communication system by dividing the communication procedure into smaller and simpler components. But when we talk about the TCP/IP model, it was designed and developed by Department of Defense (DoD) in 1960s and is based on standard protocols. It stands for Transmission Control Protocol/Internet Protocol.

The **TCP/IP model** is a concise version of the OSI model. It contains four layers, unlike seven layers in the OSI model. The layers are:

1. Process/Application Layer
2. Host-to-Host/Transport Layer
3. Internet Layer
4. Network Access/Link Layer

The diagrammatic comparison of the TCP/IP and OSI model is as follows :



Difference between TCP/IP and OSI Model:

TCP/IP	OSI
TCP refers to Transmission Control Protocol.	OSI refers to Open Systems Interconnection.
TCP/IP has 4 layers.	OSI has 7 layers.
TCP/IP is more reliable	OSI is less reliable
TCP/IP does not have very strict boundaries.	OSI has strict boundaries
TCP/IP follow a horizontal approach.	OSI follows a vertical approach.
TCP/IP uses both session and presentation layer in the application layer itself.	OSI uses different session and presentation layers.
TCP/IP developed protocols then model.	OSI developed model then protocol.
Transport layer in TCP/IP does not provide assurance delivery of packets.	In OSI model, transport layer provides assurance delivery of packets.
TCP/IP model network layer only provides connection less services.	Connection less and connection oriented both services are provided by network layer in OSI model.
Protocols cannot be replaced easily in TCP/IP model.	While in OSI model, Protocols are better covered and is easy to replace with the change in technology.

The first layer is the Process layer on the behalf of the sender and Network Access layer on the behalf of the receiver. During this article, we will be talking on the behalf of the receiver.

1. Network Access Layer –

This layer corresponds to the combination of Data Link Layer and Physical Layer of the OSI model. It looks out for hardware addressing and the protocols present in this layer allows for the physical transmission of data.
We just talked about ARP being a protocol of Internet layer, but there is a conflict about declaring it as a protocol of Internet Layer or Network access layer. It is described as residing in layer 3, being encapsulated by layer 2 protocols.

2. Internet Layer –

This layer parallels the functions of OSI's Network layer. It defines the protocols which are responsible for logical transmission of data over the entire network. The main protocols residing at this layer are :

1. **IP –** stands for Internet Protocol and it is responsible for delivering packets from the source host to the destination host by looking at the IP addresses in the packet headers. IP has 2 versions: IPv4 and IPv6. IPv4 is the one that most of the websites are using currently. But IPv6 is growing as the number of IPv4 addresses are limited in number when compared to the number of users.
2. **ICMP –** stands for Internet Control Message Protocol. It is encapsulated within IP datagrams and is responsible for providing hosts with information about network problems.
3. **ARP –** stands for Address Resolution Protocol. Its job is to find the hardware address of a host from a known IP address. ARP has several types: Reverse ARP, Proxy ARP, Gratuitous ARP and Inverse ARP.

3. Host-to-Host Layer –

This layer is analogous to the transport layer of the OSI model. It is responsible for end-to-end communication and error-free delivery of data. It shields the upper-layer applications from the complexities of data. The two main protocols present in this layer are :

1. **Transmission Control Protocol (TCP) –** It is known to provide reliable and error-free communication between end systems. It performs sequencing and segmentation of data. It also has acknowledgment feature and controls the flow of the data through flow control mechanism. It is a very effective protocol but has a lot of overhead due to such features. Increased overhead leads to increased cost.
2. **User Datagram Protocol (UDP) –** On the other hand does not provide any such features. It is the go-to protocol if your application does not require reliable transport as it is very cost-effective. Unlike TCP, which is connection-oriented protocol, UDP is connectionless.

4. Application Layer –

This layer performs the functions of top three layers of the OSI model: Application, Presentation and Session Layer. It is responsible for node-to-node communication and controls user-interface specifications. Some of the protocols present in this layer are: HTTP, HTTPS, FTP, TFTP, Telnet, SSH, SMTP, SNMP, NTP, DNS, DHCP, NFS, X Window,

LPD. Have a look at Protocols in Application Layer for some information about these protocols. Protocols other than those present in the linked article are :

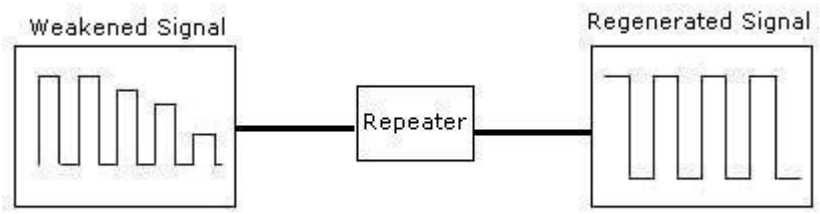
1. **HTTP and HTTPS –** HTTP stands for Hypertext transfer protocol. It is used by the World Wide Web to manage communications between web browsers and servers. HTTPS stands for HTTP-Secure. It is a combination of HTTP with SSL(Secure Socket Layer). It is efficient in cases where the browser need to fill out forms, sign in, authenticate and carry out bank transactions.
2. **SSH –** SSH stands for Secure Shell. It is a terminal emulations software similar to Telnet. The reason SSH is more preferred is because of its ability to maintain the encrypted connection. It sets up a secure session over a TCP/IP connection.
3. **NTP –** NTP stands for Network Time Protocol. It is used to synchronize the clocks on our computer to one standard time source. It is very useful in situations like bank transactions. Assume the following situation without the presence of NTP.

Repeaters, Bridges, Routers, and Gateways: A comparative study:

Repeaters

As signals travel along a network cable (or any other medium of transmission), they degrade and become distorted in a process that is called attenuation. If a cable is long enough, the attenuation will finally make a signal unrecognizable by the receiver.

A Repeater enables signals to travel longer distances over a network. Repeaters work at the OSI's Physical layer. A repeater regenerates the received signals and then retransmits the regenerated (or conditioned) signals on other segments.

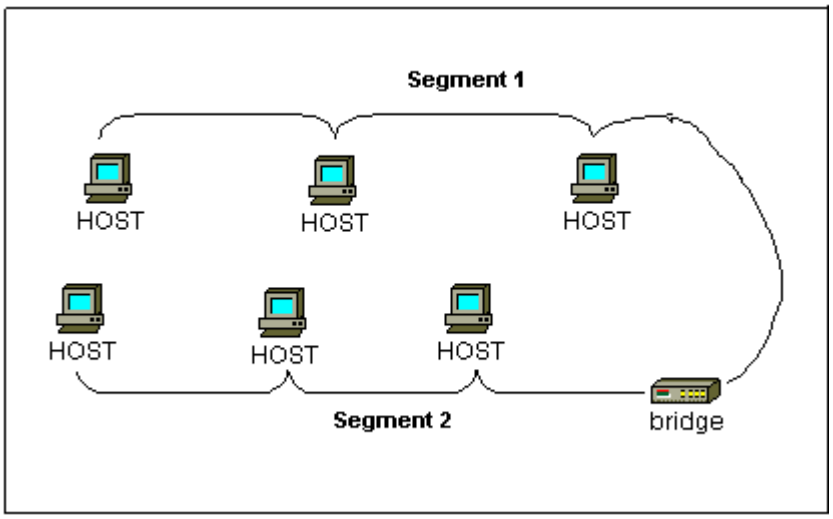


To pass data through the repeater in a usable fashion from one segment to the next, the packets and the Logical Link Control (LLC) protocols must be the same on the each segment. This means that a repeater will not enable communication, for example, between an 802.3 segment (Ethernet) and an 802.5 segment (Token Ring). That is, they cannot translate an Ethernet packet into a Token Ring packet. In other words, repeaters do not translate anything.

Bridges

Like a repeater, a bridge can join segments or workgroup LANs. However, a bridge can also divide a network to isolate traffic or problems. For example, if the volume of traffic from one or two computers or a single department is flooding the network with data and slowing down entire operation, a bridge can isolate those computers or that department.

In the following figure, a bridge is used to connect two segment segment 1 and segment 2.



Bridges can be used to:

i. Expand the distance of a segment.
ii. Provide for an increased number of computers on the network.
iii. Reduce traffic bottlenecks resulting from an excessive number of attached computers.

Bridges work at the Data Link Layer of the OSI model. Because they work at this layer, all information contained in the higher levels of the OSI model is unavailable to them. Therefore, they do not distinguish between one protocol and another.

Bridges simply pass all protocols along the network. Because all protocols pass across the bridges, it is up to the individual computers to determine which protocols they can recognize.

A bridge works on the principle that each network node has its own address. A bridge forwards the packets based on the address of the particular destination node.

As traffic passes through the bridge, information about the computer addresses is then stored in the bridge's RAM. The bridge will then use this RAM to build a routing table based on source addresses.

Routers

In an environment consisting of several network segments with different protocols and architecture, a bridge may not be adequate for ensuring fast communication among all of the segments. A complex network needs a device, which not only knows the address of each segment, but also can determine the best path for sending data and filtering broadcast traffic to the local segment. Such device is called a Router.

Routers work at the Network layer of the OSI model meaning that the Routers can switc h and route packets across multiple networks. They do this by exchanging protocol-specific information between separate networks.Routers have access to more information in packets than bridges, and use this information to improve packet deliveries. Routers are usually used in a complex network situation because they provide better traffic management than bridges and do not pass broadcast traffic.

Routers can share status and routing information with one another and use this information to bypass slow or malfunctioning connections.

Routers do not look at the destination node address; they only look at the network address. Routers will only pass the information if the network address is known. This ability to control the data passing through the router reduces the amount of traffic between networks and allows routers to use these links more efficiently than bridge

Gateways

Gateways make communication possible between different architectures and environments. They repackage and convert data going from one environment to another so that each environment can understand the other's environment data.

A gateway repackages information to match the requirements of the destination system. Gateways can change the format of a message so that it will conform to the application program at the receiving end of the transfer.

A gateway links two systems that do not use the same:

i. Communication protocols
ii. Data formatting structures
iii. Languages
iv. Architecture

For example, electronic mail gateways, such as X.400 gateway, receive messages in one format, and then translate it, and forward in X.400 format used by the receiver, and vice versa.

To process the data, the gateway:

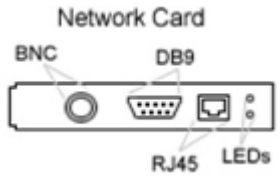
Decapsulates incoming data through the networks complete protocol stack. Encapsulates the outgoing data in the complete protocol stack of the other network to allow transmission.

NIC

A NIC or Network Interface Card is a circuit board or chip, which allows the computer to communicate to other computers on a Network. This board when connected to a cable or other method of transferring data such as infrared can share resources, information and computer hardware. Local or Wide area networks are generally used for large businesses as well as are beginning to be found in homes as home users begin to have more then one computer. Utilizing network cards to connect to a network allow users to share data such as companies being able to have the capability of having a database that can be accessed all at the same time send and receive e-mail internally within the company or share hardware devices such as printers.

Connectors

Network cards have three main types of connectors. Below is an example of what a network card may look like.



BNC connector: As illustrated in the above picture the BNC connector is a round connector, which is used for thin net or 10Base-2 Local Area Network.

DB9 (RJ45 JACK): The DB9 connector not to be confused with the Serial Port or sometimes referred to as the RJ45 JACK not to be confused with the RJ45 connection is used with Token Ring networks.

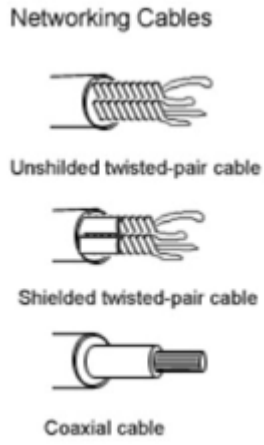
DB15 Connector: The DB15 connector is used for a Thick net or 10Base-5 Local area network.

RJ45 connector: Today one of the most popular types of connections used with computer networks. RJ45 looks similar to a phone connector or RJ11 connector however is slightly larger.

LED - The LED's as shown in the above illustration indicates if it detects a network generally by a green light which may flash as it communicates and then a red light which indicates collisions which will generally flash or not flash at all.

Cables

The following is a few examples of some of the more commonly used types of cables found with networks.



As illustrated in the above picture you can see three of the main types of cable used today. The first two pictures illustrate Unshielded and Shielded twisted pair cables. Unshielded twisted-pair cable is generally found in phone cables today and are used more often then shielded twisted pair today as it has been found that simply twisting the cable provides more efficient means of protection against interference. In addition shielded twisted-pair cable required the one end of the cable to be grounded. If both ends were to be grounded however this would cause a grounding loop causing low voltage and infinite amperage and various other hazards to the network.

The third picture in the above illustration shows a coaxial cable, which are the most commonly used and known types of cables. This cable can be found for cable TV and when used with networks utilize the BNC connector.

As illustrated in the above picture you can see three of the main types of cable used today. The first two pictures illustrate Unshielded and Shielded twisted pair cables. Unshielded twisted-pair cable is generally found in phone cables today and are used more often then shielded twisted pair today as it has been found that simply twisting the cable provides more efficient means of protection against interference. In addition shielded twisted-pair cable required the one end of the cable to be grounded. If both ends were to be grounded however this would

cause a grounding loop causing low voltage and infinite amperage and various other hazards to the network.

The third picture in the above illustration shows a coaxial cable, which are the most commonly used and known types of cables. This cable can be found for cable TV and when used with networks utilize the BNC connector.

Hub – A hub is basically a multiport repeater. A hub connects multiple wires coming from different branches, for example, the connector in star topology which connects different stations. Hubs cannot filter data, so data packets are sent to all connected devices. In other words, collision domain of all hosts connected through Hub remains one. Also, they do not have intelligence to find out best path for data packets which leads to inefficiencies and wastage.

Types of Hub

- **Active Hub:-** These are the hubs which have their own power supply and can clean, boost and relay the signal along with the network. It serves both as a repeater as well as wiring centre. These are used to extend the maximum distance between nodes.
- **Passive Hub :-** These are the hubs which collect wiring from nodes and power supply from active hub. These hubs relay signals onto the network without cleaning and boosting them and can't be used to extend the distance between nodes.

Ethernet

What is Ethernet?

Ethernet is the major local-area network (LAN) in which we connect our computers, routers, and printers. It has taken an important role in the industrial engineering world, having become the established standard connection for the Internet of Things.According to Cisco in 2003, Ethernet makes up 85% of the world's LAN connections. Industrial Ethernet differs from commercial Ethernet in that it applies the Ethernet standards toward the development of data communication to control and operate manufacturing networks.

Beginnings of Ethernet

ALOHAnet was a wireless data network that connected several computer systems separated throughout the Hawaiian island college campuses. They attempted to have independent data radio nodes to communicate with each other on a peer-to-peer basis without interference. ALOHAnet's solution was a multiple access with collision detection (CSMA/CD) concept. This idea inspired Bob Metcalfe from Xerox to base his PhD studies on finding improvements.

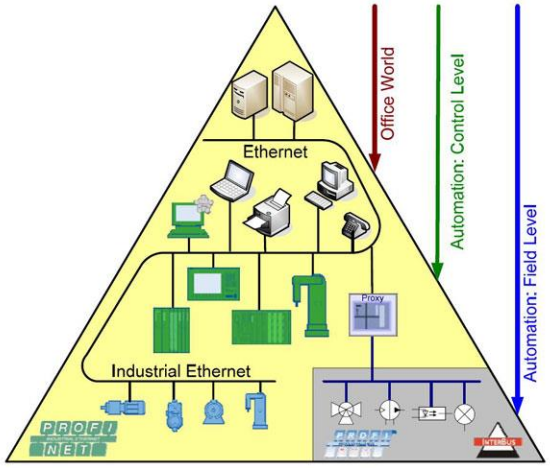
In the initial days of the Ethernet, there were two common options for configurations: 10Base2 and 10Base5. The operating speed for both configurations was 10 Mbps and used coaxial cable with nodes via tee connectors or through attachment unit interfaces in a multi-drop bus configuration.

The unit interfaces or nodes are typically computers with network interface cards (NIC). The max allowed segment lengths for 10Base2 was 185 ft when using RG 58 coaxial cable, also known as "Thin Ethernet." 10Base 5 offered greater distances between segment nodes; however, it required thick coaxial cable which was difficult and bulky to manage.

Other configurations started to develop, such as 10Base-FL, which allowed networks to use fiber optic media and increase distances to greater than 2,000 ft. 10Base-T became a popular physical layer option due to its ease of install and use of inexpensive unshielded twisted pair (UTP) Category 3 (CAT3) cable. Each computer is required to be less 100 feet from each other with standard RJ-45 connectors. By the 1990s, 100 Mbps Ethernet equipment became available. These NICs would adjust automatically between 10 Mbps or 100 Mbps operating speed.

Today's standard for computer setups is to have NICs implement 100Base-TX. Category 5e UTP cables (CAT 5) are the standard, with the lengths the same as used with 10Base-T networks—100 ft or less. What was once a coaxial-based network is being replaced by fiber optics specifically for point-to-point links. 100Base-FX uses two optical fibers for full duplex point-to-point communications which a reach 2,000 ft. Gigabit Ethernet or 1000 Mbps connections are available using twisted pair and fiber optic media.

Ethernet Data Link Layer and Frame



The hierarchy depicted above shows how Ethernet connections are used at different levels, depending on the devices being connected. (Image courtesy of Profibus and Profinet North America)Ethernet specifies the physical layer and data link layers of a network's function. It became the basis for the IEEE 802.3 network standard. The physical layer specifies electrical signals, signaling speeds, media, connector types, and network topologies. The technology can be used over optical fiber and twisted pair-cables. Four different type of data rates are:

Name	IEEE standard	Data rate	Media type	Maximum distance
Ethernet	802.3	10 Mbps	10Base-T	100 meters
Fast Ethernet/100Base-T	802.3u	100 Mbps	100Base-TX	100 meters
			100Base-FX	2,000 meters
Gigabit Ethernet/GigE	802.3z	1000 Mbps	1000Base-T	100 meters
			1000Base-SX	275/550 meters
			1000Base-LX	550/5000 meters
10 Gigabit Ethernet	IEEE 802.3ae	10 Gbps	10GBase-SR	300 meters
			10GBase-LX4	300m MMF/ 10km SMF
			10GBase-LR/ER	10km/40km
			10GBase-SW/LW/EW	300m/10km/40km

The data link layer defines the media access method. Half-duplex link are those connected in a bus or star topologies: 10/100Base-T, 10Base2, 10Base 5, etc. They use carrier sense, multiple access with collision detection (CSMA/CD). This allows for multiple nodes to have equal network access. All nodes on an Ethernet network continuously monitor for transmissions on the given media.

Nodes wait for a network to be idle prior to starting a transmission. When nodes transmit at the same time, the signals overlap and corrupt the originals. When nodes see a different signal from the one they are trying to send, they detect the collision and stop transmitting. They re-attempt to transmit after a preset delay. This method of media access allows for nodes to be simply added or removed from a network.

A node once connected begins to listen and transmit on the network. This, however, will eventually lead to decreased access and an increase in collisions. This makes Ethernet networks probabilistic networks. In full-duplex point-to-point Ethernet links like 10Base-FL or 100Base-FX collisions are not an issue. This is due to the fact that only two nodes are present and there are separate send and receive channels available. Data can also be sent in both directions simultaneously, hence doubling the data transfer rate.

The Ethernet Frame defines the format of the data message sent on the network. The message format contains several fields of information, including the data to be transferred. The data unit is defined as the actual data to be sent and can contain between 46 and 1500 eight-bit bytes of binary information. The length of the data unit is determined and included in the message as a field for the receiver to determine which part of the message is data.

The MAC address is the six-byte binary number set which includes the source and destination information for the nodes. A MAC address is included in each message sent over the network, and each Ethernet node has a unique MAC address.
The data link layer specifies the frame structure of messages transmitted or received and how communication occurs over the media. Nodes, switches, and hubs can be added and removed simply, and the technology can be easily troubleshot. These factors have made Ethernet connections the new standard for industrial network applications. The functions of the OSI layers are to designate how data and application demands are transmitted.

There are seven layers in the OSI reference model. The lower layers (1-4) focus on data-transport, while the upper layers (5-7) focus on applications. The lowest layer (1) is the physical layer closest to the physical network medium. The physical layer and the lower data link layers are implemented in hardware and software such as cabling, or Ethernet (which exists on Layer 2).

Layer 3 is used for logical addressing and routing. Its most common application is the use of the Internet Protocol (IP). Layer 4 is the transport layer that ensures the data is delivered error-free and in the correct sequence. It uses Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) to transmit data. Industrial Ethernet differs from commercial Ethernet in that it encompass all the lower layers for its solution ,and not just Layer 2.

The upper layers for the OSI reference model are used for application tasks, and are usually implemented for software only. Layer 5 deals with the session. It is responsible for dialing control and synchronization of the session connection (i.e., establishing and managing sessions) between networks and applications.

Layer 6 is for data presentation use. This layer presents the data and coding type, and defines the used characters. It ensures that data may be exchanged between hosts and across the network, and that it is compressed and encrypted. Layer 7 is for application use; it is used by software to prepare and interpret the data. As the top layer, it is closest to the end user.
Ethernet Connection Types and Industrial Systems

Above is an example of an industrial layoutâ□”from the programmable logic controllers, to the machine-operated hardware, back to the human machine interface operated by the engineer. (Source: Siemens USA)

TCP/IP over Ethernet networks offers the possibility of a level of standardization. Historically, network applications based on time-critical processes have used deterministic networks. When using Industrial Ethernet, it is important to note the speed and determinism of the communication. Speed is how fast the network can send information over the Ethernet.

Determinism is the network's ability to communicate in a predictable timespan. For motion control systems, it is essential for Ethernet connections to transmit data to and from devices on a regularly scheduled basis. These are networks based on master/slave or token passing schemes.

The use of Ethernet networks must be controlled at levels no greater than 10% or they would suffer from inadequate performances. Segmenting networks via switches and routers minimize unwanted network traffic and reduce use. Another method is to use newer and higher-level protocols to incorporate prioritization and synchronization of messages to ensure better timed delivery.

The result of these methods is shift of using Ethernet for industrial control on the plant floor and into the cell field levels. This increase of Ethernet implantation is due to their ease of installation and low cost of hardware. The use of bridges and fast switches help raise the determinism of a network. Eventually as gigabit, 10Gbit, and 100Gbit become more commonly used, determinism concerns will reduce.

The major Ethernet connection types are:

Modbus TCP/IP

- First industrial protocol on Ethernet, introduced in 1999.
- Based on Modbus protocol, which was developed by Modicon in 1979.
- Advantages:
  - Uses standard Ethernet layers: hardware and TCP/IP transport layer
  - Open and relatively simple Ethernet protocol
- Disadvantages:
  - Not a hard real-time protocol
- Largest supplier: Schneider Electric
- Multi-vendor consortium: Modbus IDA
- Factory automation technology: RTPS

EtherCAT

- Open-source, based on IEC 61158 and other similar standards.
- Advantages:
  - Hard real-time industrial protocol
  - Efficient and straightforward communication
- Disadvantages:
  - Total number of devices used is limited
  - Not designed for standard TCP/IP packets and EtherCAT packets
- Largest supplier: Beckhoff
- Multi-vendor consortium: EtherCAT Technology Group (ETG)
- Factory automation technology: Shared Frame

Ethernet/IP

- Extends DeviceNET concepts to Ethernet
- Advantages:
  - Uses Ethernet transport layers (i.e., TCP and UDP)
- Disadvantages:
  - Networks can be overloaded with UDP messages if not correctly configured
- Largest supplier: Rockwell Automation
- Multi-vendor consortium: Open Device Vendors Association (ODVA)
- Factory automation technology: CIP

Profinet

- Application protocol that extends Profibus to Ethernet
- Advantages:
  - Supports both standard and deterministic Ethernet traffic
  - Implements IEEE 1588 and Quality of Service (QoS) to add determinism
- Disadvantages:
  - For Real Time and Isochronous Real Time managed switches, QoS is recommended
- Largest supplier: Siemens
- Multi-vendor consortium: Profibus and Profinet International
- Factory automation technology: Profinet IO

Wireless LANs

Wireless LANs are those Local Area Networks that use high frequency radio waves instead of cables for connecting the devices in LAN. Users connected by WLANs can move around within the area of network coverage. Most WLANs are based upon the standard IEEE 802.11 or WiFi.

IEEE 802.11 Architecture

The components of an IEEE 802.11 architecture are as follows

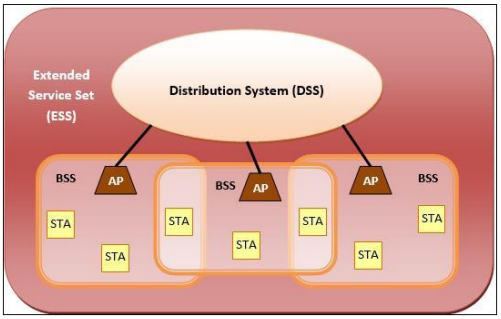
1) Stations (STA) − Stations comprise all devices and equipments that are connected to the wireless LAN. A station can be of two types:

- **Wireless Access Pointz (WAP)** − WAPs or simply access points (AP) are generally wireless routers that form the base stations or access.
- **Client.** − Clients are workstations, computers, laptops, printers, smartphones, etc.

Each station has a wireless network interface controller.
2) Basic Service Set (BSS) −A basic service set is a group of stations communicating at physical layer level. BSS can be of two categories depending upon mode of operation:

- **Infrastructure BSS** − Here, the devices communicate with other devices through access points.
- **Independent BSS** − Here, the devices communicate in peer-to-peer basis in an ad hoc manner.

3) Extended Service Set (ESS) − It is a set of all connected BSS.
4) Distribution System (DS) − It connects access points in ESS.



Advantages of WLANs

- They provide clutter free homes, offices and other networked places.
- The LANs are scalable in nature, i.e. devices may be added or removed from the network at a greater ease than wired LANs.
- The system is portable within the network coverage and access to the network is not bounded by the length of the cables.
- Installation and setup is much easier than wired counterparts.
- The equipment and setup costs are reduced.

Disadvantages of WLANs

- Since radio waves are used for communications, the signals are noisier with more interference from nearby systems.
- Greater care is needed for encrypting information. Also, they are more prone to errors. So, they require greater bandwidth than the wired LANs.
- WLANs are slower than wired LANs.

Introduction to Modbus

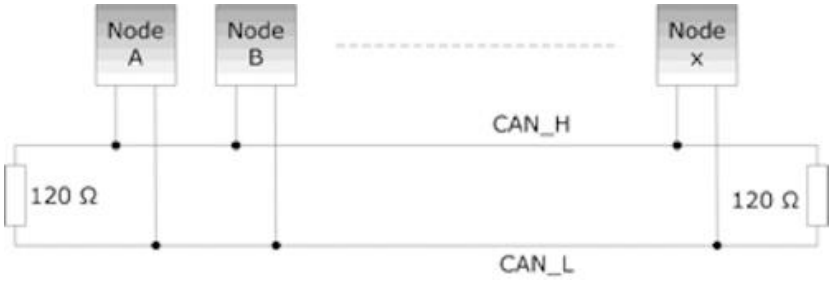
Modbus is a communication protocol developed by Modicon systems. In simple terms, it is a method used for transmitting information over serial lines between electronic devices. The device requesting the information is called the Modbus Master and the devices supplying information are Modbus Slaves.
Modbus is a communication protocol for transmitting information between electronic devices over serial lines (original version) or via the Ethernet, and is commonly used in process and factory automation. While it's an open protocol and anybody can use it, "Modbus" is a registered trademark of Schneider Electric USA, Inc. (current owner of the Modicon brand). The Modbus.org organization was created to further the use of Modbus and Schneider Electric has been a partner in it. This article is an introduction to Modbus and its basic functions—Modbus.org has extensive coverage on Modbus, the specifications for the various types of Modbus, software, testing, interface code and more. The Internet also has available tutorials and specific information on individual device Modbus implementations.

Modbus serial protocol (the original version) is a master/slave protocol, e.g. one master that controls the Modbus data transactions with multiple slaves that respond to the master's requests to read from or write data to the slaves.

Modbus is one of the most popular protocols used in the industrial world. Supporting traditional serial protocols of RS232/422/485 and Ethernet protocols allow industrial devices such a PLCs, HMIs and meters to use Modbus as their communication method. When communicating with Modbus via serial and Ethernet networks, a communication gateway is a necessity. Modbus serial servers from B+B SmartWorx enable smooth connectivity by translating Modbus/TCP to Modbus/ASCII/RTU protocols. This allows devices such as PLCs to communicate to devices such as sensors, meters and instruments.

Introduction to Controller Area Network

Controller Area Network (CAN) is a serial network technology that was originally designed for the automotive industry, especially for European cars, but has also become a popular bus in industrial automation as well as other applications. The CAN bus is primarily used in embedded systems, and as its name implies, is a network technology that provides fast communication among microcontrollers up to real-time requirements, eliminating the need for the much more expensive and complex technology of a Dual-Ported RAM.
CAN is a two-wire, half duplex, high-speed network system, that is far superior to conventional serial technologies such as RS232 in regards to functionality and reliability and yet CAN implementations are more cost effective.

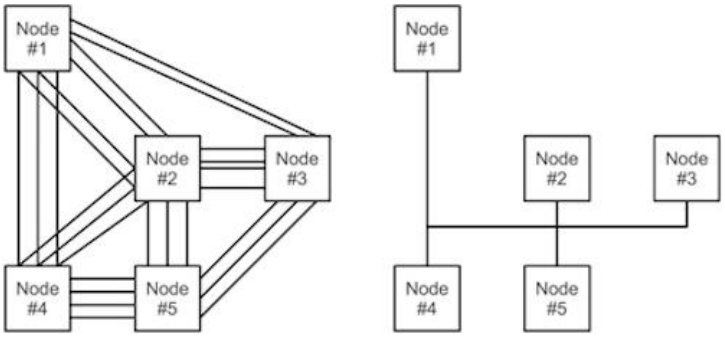


While, for instance, TCP/IP is designed for the transport of large data amounts, CAN is designed for real-time requirements and with its 1 MBit/sec baud rate can easily beat a 100 MBit/sec TCP/IP connection when it comes to short reaction times, timely error detection, quick error recovery and error repair.

CAN networks can be used as an embedded communication system for microcontrollers as well as an open communication system for intelligent devices. Some users, for example in the field of medical engineering, opted for CAN because they have to meet particularly stringent safety requirements.

Similar requirements had to be considered by manufacturers of other equipment with very high safety or reliability requirements (e.g. robots, lifts and transportation systems).

The greatest advantage of Controller Area Network lies in the reduced amount of wiring combined with an ingenious prevention of message collision (meaning no data will be lost during message transmission).



Without CAN - With CAN

The following shows a need-to-know overview of CAN's technical characteristics.

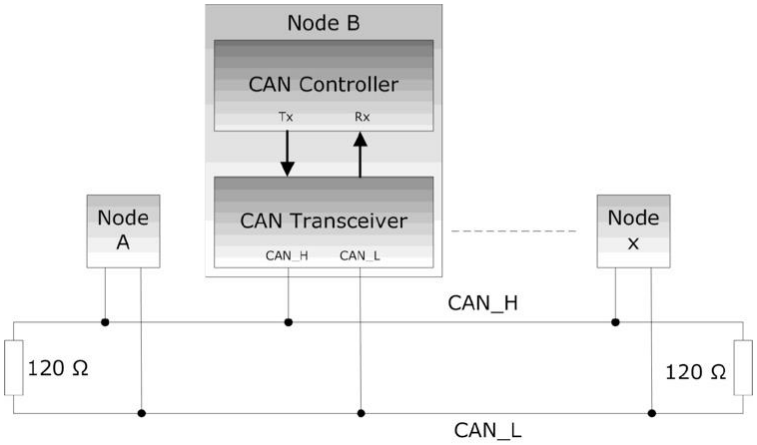
Controller Area Network

- Is a serial networking technology for embedded solutions.
- Needs only two wires named CAN_H and CAN_L.
- Operates at data rates of up to 1 Megabit per second.
- Supports a maximum of 8 bytes per message frame.
- Does not support node IDs, only message IDs. One application can support multiple message IDs.
- Supports message priority, i.e. the lower the message ID the higher its priority.
- Supports two message ID lengths, 11-bit (standard) and 29-bit (extended).
- Does not experience message collisions (as they can occur under other serial technologies).
- Is not demanding in terms of cable requirements. Twisted-pair wiring is sufficient.

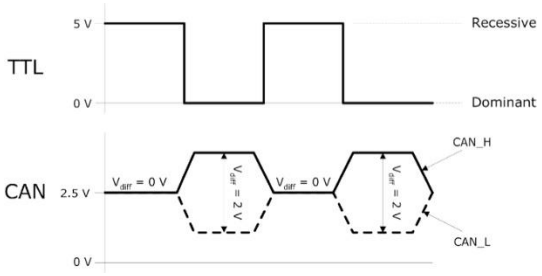
CAN Interface Hardware

A great variety of microprocessor chips, such as the ARM Cortex-M3 processor, provide interfaces such as Ethernet, digital I/O, analog I/O, USB, UARTS, and, last but not least, Controller Area Network. However, that does not mean that you can use the chip "as is" and connect it to a network, sensors, etc. All of these interfaces require some kind of a "hardware driver." In case of serial technologies such as RS232 or CAN, you will need the corresponding transceiver.

In the specific case of the CAN bus controller, we need a line driver (transceiver) to convert the controller's TTL signal to the actual CAN level, which is a differential voltage. The use of differential voltage contributes to the vast reliability of CAN.



The next image compares both signals, TTL and differential voltage:



The actual signal status, recessive or dominant, is based on the differential voltage between CAN_H and CAN_L (2V during dominant bit time; 0V during recessive bit time).
Controller Area Network (CAN) is a serial network technology that was originally designed for the automotive industry, especially for European cars, but has also become a popular bus in industrial automation as well as other applications. The CAN bus is primarily used in embedded systems, and as its name implies, is a network technology that provides fast communication among microcontrollers up to real-time requirements, eliminating the need for the much more expensive and complex technology of a Dual-Ported RAM.
CAN networks can be used as an embedded communication system for microcontrollers as well as an open communication system for intelligent devices. Some users, for example in the field of medical engineering, opted for CAN because they have to meet particularly stringent safety requirements.

Similar requirements had to be considered by manufacturers of other equipment with very high safety or reliability requirements (e.g. robots, lifts and transportation systems).

The greatest advantage of Controller Area Network lies in the reduced amount of wiring combined with an ingenious prevention of message collision (meaning no data will be lost during message transmission).

Controller Area Network

- Is a serial networking technology for embedded solutions.
- Needs only two wires named CAN_H and CAN_L.
- Operates at data rates of up to 1 Megabit per second.
- Supports a maximum of 8 bytes per message frame.
- Does not support node IDs, only message IDs. One application can support multiple message IDs.
- Supports message priority, i.e. the lower the message ID the higher its priority.
- Supports two message ID lengths, 11-bit (standard) and 29-bit (extended).
- Does not experience message collisions (as they can occur under other serial technologies).
- Is not demanding in terms of cable requirements. Twisted-pair wiring is sufficient.

Applications

- Passenger vehicles, trucks, buses (gasoline vehicles and electric vehicles)
- Agricultural equipment
- Electronic equipment for aviation and navigation
- Industrial automation and mechanical control
- Elevators, escalators
- Building automation
- Medical instruments and equipment
- Pedelecs

LonWorks

LonWorks (local operating network) is a networking platform specifically created to address the needs of control applications. The platform is built on a protocol created by Echelon Corporation for networking devices over media such as twisted pair, powerlines, fiber optics, and RF. LonWorks is a standard technology for many of the global standards organizations including ASHRAE, IEEE, ANSI, SEMI and many others. In fact, LonWorks capability is a prerequisite for participation in a growing number of automation projects.. LonWorks is becoming a major network standard in the commercial buildings market with a number of Building Automation Systems suppliers standardizing on LON including Siemens Building Systems and Honeywell. In order to profit in this market it is important to support LON. The technology has its origins with chip designs, power line and twisted pair, signaling technology, routers, network management software, and other products from Echelon Corporation. In 1999 the communications protocol (then known as LonTalk) was submitted to ANSI and accepted as a standard for control networking (**ANSI/CEA-709.1-B**). Echelon's power line and twisted pair signaling technology was also submitted to ANSI for standardization and accepted. Since then, ANSI/CEA-709.1 has been accepted as the basis for IEEE 1473-L (in-train controls), AAR electro-pneumatic braking systems for freight trains, IFSF (European petrol station control), SEMI (semiconductor equipment manufacturing), and in 2005 as EN 14908 (European building automation standard). The protocol is also one of several data link/physical layers of the BACnet ASHRAE/ANSI standard for building automation.

China ratified the technology as a national controls standard, GB/Z 20177.1-2006 and as a building and intelligent community standard, GB/T 20299.4-2006; and in 2007 CECED, the European Committee of Domestic Equipment Manufacturers, adopted the protocol as part of its Household Appliances Control and Monitoring – Application Interworking Specification (AIS) standards.

During 2008 ISO and IEC have granted the communications protocol, twisted pair signaling technology, power line signaling technology, and Internet Protocol (IP) compatibility standard numbers ISO/IEC 14908-1, -2, -3, and -4.

Modbus

Modbus is a network protocol best used for industrial automation systems specifically for connecting electronic equipment. Although Modbus is best for industrial applications, its simplicity allows it to be a useful tool for building automation as well.

	BACnet	Modbus	LonWorks
Developed By:	ASHRAE	Modicon Inc.	Echelon Corporation/ Motorola
Use	Communication across devices	Connection between devices	Networking devices through power lines, fiber optics, and other media
Markets	Industrial, Transportation, Energy Management, Building Automation, Regulatory and health and safety	HVAC, Lighting, Life Safety, Access Controls, transportation and maintenance	Home automation, industrial, transportation, and public utility control networks.
Examples	Boiler Control, Tank Level Measurements	Tasks such as request temperature reading, send status alarm, or fan schedule	Security, lighting systems, HVAC, machine control, manufacturing, metering
Proprietary	No	No	Yes
Transmission Modes	Ethernet, IP, MS/TP, Zigbee	ASCII, RTU, TCP/IP	MS/TP, network, SNVT
Standards	ANSI/ASHRAE Standard 185 ;ISO-16484-5; ISO-16484-6	IEC 61158	ANSI/EIA 709.1; ISO/IEC 14908-1, 14908-2, 14908-3, 14908-4
Costs	Low; No charge for usage or licensing fees	Low; No charge for usage or licensing fees	High (proprietary); Limited users (exclusive to actual members; mostly manufacturers)
Network Interfaces	Existing LANs and LANs infrastructure	Traditional serial and Ethernet protocols	U10/U20 USB Network Interface; i.LON SmartServer; i.LON 600
Testing	BACnet Testing Labs	Modbus TCP Conformance Testing Program	Products must conform to LonWorks protocol
Advantages	• Scalability between cost, performance and system size • Endorsement and adoption by nearly every major vendor in North America and many other countries • Robust internetworking including multiple LAN types and dial-up • Unrestricted growth and the ability to add new innovations and new features anytime	• Easy connection to Modicon • Suitable for small/medium volumes of data (≤255 bytes) • Data transfer designed for industrial applications • Openly published and royalty-free • Easy to deploy and maintain • Moves raw bits or words without placing restrictions on vendors	• Web based tool; saves time and cost • Numerous developers of Lonworks products in the market • Less Architecture at device level
Disadvantages	• Limited the number of field devices that can connect to a master station except Ethernet TCP/IP • MT/TP-Wire Length • Ethernet-Infrastructure • New standard has security standard but not implemented in all devices	• Limited the number of data types; Large binary objects are not supported. • No standard method for a node to find the description of a data object, i.e. finding a register value represents a temperature between 30° and 175°. • No security against unauthorized commands or interception of data • Transmissions must be contiguous which limits the types of remote communications devices to those that can buffer data to avoid gaps in the transmission. • Great amount of configuration and programming required	• Outdated • Controlled devices & variables are connected to a separate control device. (Not recommended due to network interruptions producing system failures) • Extensions are allowed only through the LonMark Consortium. • Hardware specific, and requires the Neuron chip for network movement of the protocol. • Close to “plug & play” ability, yet still far from achieving interconnectivity using Microsoft Windows.

LonWorks™ was Created by Echelon Corp (www.echelon.com) in 1988. LonWorks is a leading networking solution for Building Automation. Estimates for the number of nodes installed worldwide range into the millions. The LonMark Interoperability Association with over 300 member companies reflects the strength LON now has in the automation market.

When used in an industrial environment a LonWorks solution is very different from the open device networks like DeviceNet, Profibus and Modbus typically found on the factory floor. First unlike these popular device busses LonWorks is a completely peer-peer network. Instead of moving data through a "Master" device, any device can exchange data with any other LonWorks device on the network. Second, LonWorks is not tied a single physical communication layer. Where DeviceNet is limited to CAN and Profibus and Modbus are limited to RS485, LonWorks can use twisted pair, Ethernet or even a power line as its communication channel. Finally, network data exchanged on LonWorks is configured by a network configuration tool. This operation called "binding" ties an input of one device to an output of another device independent of the operation or application software in either device.

LonWorks is a standard technology for many of the global standards organizations including ASHRAE, IEEE, ANSI, SEMI and many others. In fact, LonWorks capability is a prerequisite for participation in a growing number of automation projects.. LonWorks is becoming a major network standard in the commercial buildings market with a number of Building Automation Systems suppliers standardizing on LON including Siemens Building Systems and Honeywell. In order to profit in this market it is important to support LON.

UNIT V

Common Industrial Protocols:HART: Introduction - Evolution of Signal standard - HART Communication protocol - Communication modes - HART Commands – HART and the OSI model.Field Bus: Introduction - General Field bus architecture - basic requirements of field bus standard - field bus topology - Interoperability - Interchangeability.

HART

HART ("Highway Addressable Remote Transducer") is a communication protocol designed for industrial process measurement and control applications. HART is an open standard and vendor independent. Because of this, it is the world's most broadly supported protocol for the process industry, with thousands of HART based products available from different vendors. It's called a hybrid protocol because it combines analogue and digital communication. It can communicate a single variable using a 4-20mA analogue signal while also communicating added information on a digital signal that is superimposed on the standard 4-20mA current loop. Using an analogue signal, information is sent only one way, either from the device to the host (inputs) or from the host to the device (outputs). Digital information can travel in both directions using HART. Traditional analogue devices communicate only a single process variable, and you typically have no easy way to tell if the information they're sending is valid. With HART, you will get the process variable – but other types of information too. Information items are standard in every HART device such as:

- Device status and diagnostic alerts
- Process variables and units
- Loop current and percentage range
- Basic configuration parameters
- Manufacturer and device tag information

How does HART work?

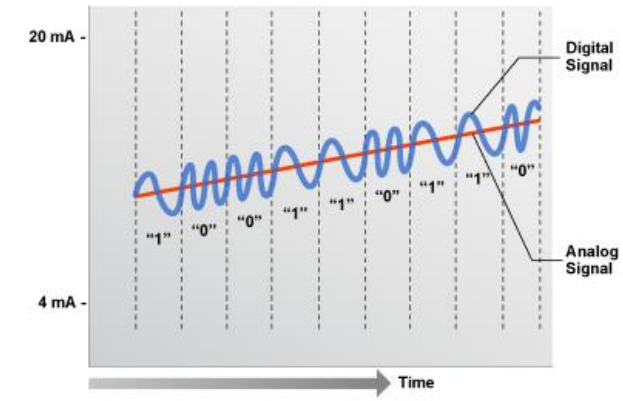
HART digital communications uses a request/reply communications model. This means that in general, HART devices won't transmit any information unless a request is sent from the host to the device. The exception to this is the burst mode where the HART device can send a single piece of information continuously without repeated host requests. A common use for burst mode is to send the process variable as a digital value to verify the analogue signal. Many control systems aren't designed to accept HART information in digital form so it is common to see external multiplexers reading the digital signal. In this approach, the HART device is attached to both the control host and to the multiplexer. Although this increases the cost of the installation, reductions in maintenance cost generally pay back the investment in a very short time. Some hosts, such as Emerson's DeltaV are able to capture and pass HART digital information to other applications (e.g., AMS Asset Management Suite) using a mechanism commonly called "HART pass through". Using a system that supports HART pass through reduces the cost of acquiring and using the HART information by eliminating the need to install separate multiplexer systems.

"HART" is an acronym for Highway Addressable Remote Transducer. The HART Protocol makes uses Frequency Shift Keying (FSK) standard to superimpose digital communication signals at a low level on top of the 4-20mA. This enables two-way field communication to take place and makes it possible for additional information beyond just the normal process variable to be communicated to/from a smart field instrument.

The HART Protocol communicates at 1200 bps without interrupting the 4-20mA signal and allows a host application (master) to get two or more digital updates per second

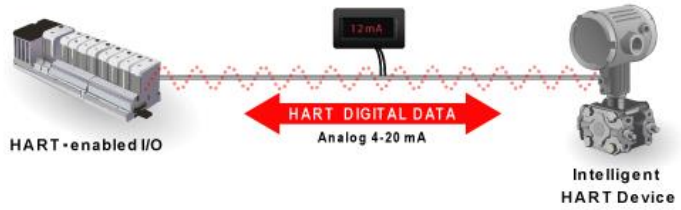
from a smart field device. As the digital FSK signal is phase continuous, there is no interference with the 4-20mA signal. The HART Protocol provides two simultaneous communication channels: the 4-20mA analog signal and a digital signal. The 4-20mA signal communicates the primary measured value (in the case of a field instrument) using the 4-20mA current loop - the fastest and most reliable industry standard. Additional device information is communicated using a digital signal that is superimposed on the analog signal.

The digital signal contains information from the device including device status, diagnostics, additional measured or calculated values, etc. Together, the two communication channels provide a low-cost and very robust complete field communication solution that is easy to use and configure.



Digital over Analog

HART Communication occurs between two HART-enabled devices, typically a smart field device and a control or monitoring system. Communication occurs using standard instrumentation grade wire and using standard wiring and termination practices.



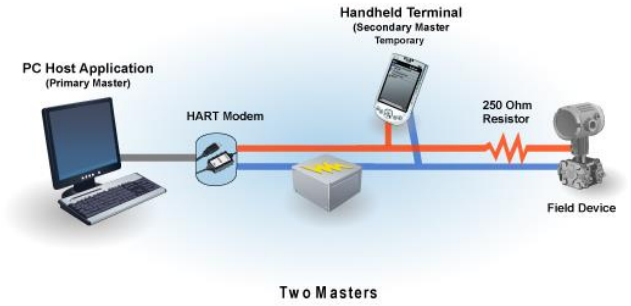
I/O to Device

Two Communication Channels

HART technology is a master/slave protocol, which means that a smart field (slave) device only speaks when spoken to by a master. The HART Protocol can be used in various modes such as point-to-point or multidrop for communicating information to/from smart field instruments and central control or monitoring systems.

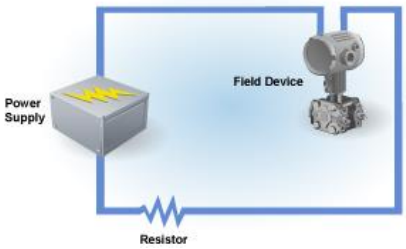
The HART Protocol provides for up to two masters (primary and secondary). This allows secondary masters such as handheld communicators to be used without

interfering with communications to/from the primary master, i.e. control/monitoring system.

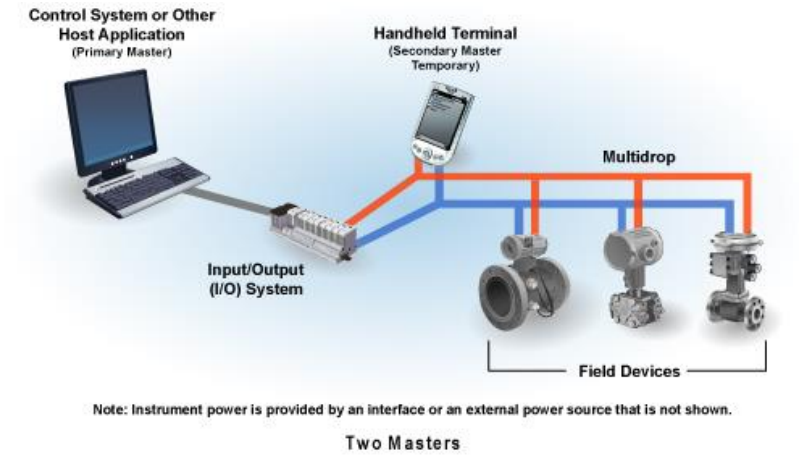


Primary and Secondary Masters

The HART Protocol permits all digital communication with field devices in either point-to-point or multidrop network configurations:



Point-to-Point Configuration



Multidrop Configuration

There is also an optional "burst" communication mode where a single slave device can continuously broadcast a standard HART reply message. Higher update rates are possible with this optional burst communication mode and use is normally restricted to point-to-point configuration.

HART COMMANDS

The HART Protocol is a master-slave communication protocol which means that during normal operation, each slave (a field device) communication is initiated by a request (or command) from the master (host) communication device. The master or host is generally a distributed control, PLC, or PC-based asset management system for example. The slave device is typically a field measurement device such as pressure, level, temperature, flow or other transmitters.

In order to make certain any HART-enabled device from any supplier can communicate properly and respond to a command with the correct information, the set and types of commands are defined in the HART Specifications and implemented in all HART registered devices.

Users need not worry about these commands because they are included in the functions of the host. The specific capabilities of a device (device specific commands) are available to the host when the host is given the instructions included in the Device Description (DD) of a specific device.

An important point is that defined device status indications are included with each communication response to the host. The host then interprets these status indicators and may provide basic device diagnostic information.

The **HART Command Set** provides uniform and consistent communication for all field devices. Host applications may implement any of the necessary commands for a particular application. The command set includes three classes:

Universal
All devices using the HART Protocol must recognize and support the universal commands. Universal commands provide access to information useful in normal operations (e.g., read primary variable and units).

Common Practice
Common Practice commands provide functions implemented by many, but not necessarily all, HART Communication devices.

Device Specific
Device Specific commands represent functions that are unique to each field device. These commands access setup and calibration information, as well as information about the construction of the device. Information on Device Specific commands is available from device manufacturers.

HART Commands:

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 pre-defined 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 set point • Valve characterization • Valve set point • Travel limits • User units • Local display information

HART Communication modes

The HART digital communication signal has a response time of approximately 2-3 data updates per second without interrupting the analog signal. A minimum loop impedance of 230 Ω is required for communication though 250 Ω is typically used in practice. HART communication occurs in two modes:

(a) Request - Response Mode

(b) Burst Mode

Request-Response Mode

During normal operation (2-3 data updates per second), each field device (slave) communication is initiated by a host (master) communication device. Two hosts can connect to each HART loop. The primary host is generally a distributed control system (DCS), programmable logic controller (PLC), or a personal computer (PC). The secondary host can be a handheld terminal or another PC. Field devices include transmitters, actuators and controllers that respond to commands from the primary or secondary host.

Burst Mode

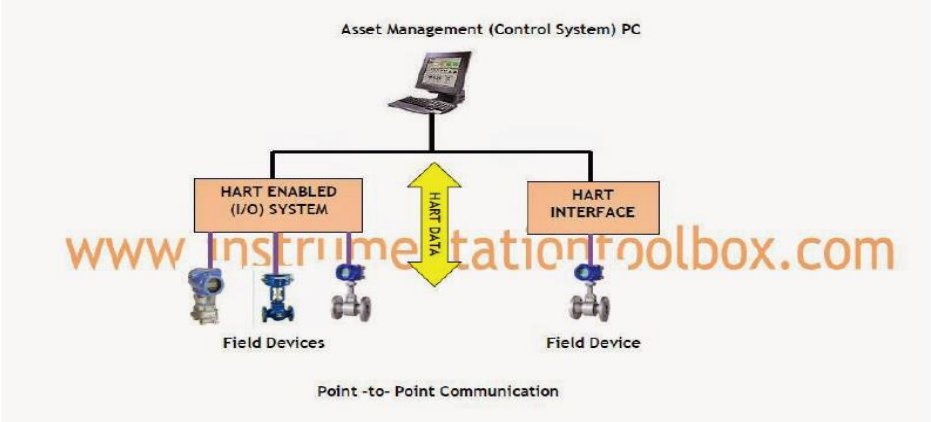
Some HART devices support an optional burst communication mode. Burst mode enables faster communication (3-4 data updates per second). In burst mode, the host instructs the field device to continuously broadcast a standard HART reply message (e.g., the value of the process variable). The host receives the message at the higher rate until it instructs the device to stop bursting. Burst mode is used when more than one HART device is required to listen to communication from the HART loop.

HART Communication Networks

HART devices can operate in one of two network configurations—point-to-point or multi-drop.

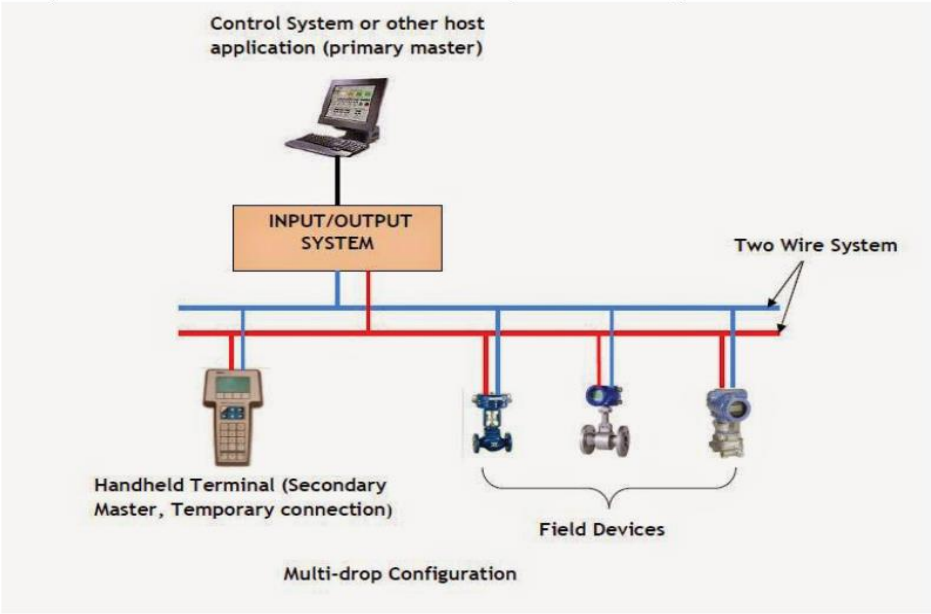
Point-to-Point Network Mode

In point-to-point mode, the 4-20mA signal is used to communicate one process variable,
while additional process variables, configuration parameters, and other device data are transferred digitally using the HART Protocol. The 4-20mA analog signal is not affected by the HART signal and can be used for control. The HART communication digital signal gives access to secondary variables and other data that can be used for operations, commissioning, maintenance and diagnostic purposes. A typical point-to-point communication network is shown below:



Multi-Drop Mode

The HART Communication Protocol enables several instruments to be connected on the same pair of wires multi-drop network configuration. The current through each field device is fixed at a minimum value (typi 4mA) sufficient for device operation. The analog loop current does not change in relation to the process thus does not reflect the primary variable. Communications in multi-drop mode are entirely digital. Multi- connection is mostly used for supervisory control installations that are widely spaced such as pipelines, cus transfer stations, and tank farms. Below is shown a multi-drop network configuration:



The HART Protocol is the leading communication technology used with smart process instrumentation today. The HART Protocol continues to grow in popularity and recognition in the industry as a global standard for smart instrument communication. More than two-thirds of all smart instruments shipping today communicate using the HART Protocol.

EASY TO USE

HART is field-proven, easy to use and provides highly capable two-way digital communication simultaneously with the 4-20mA analog signaling used by traditional instrumentation equipment.

UNIQUE COMMUNICATION SOLUTION

Unlike other digital communication technologies, the HART Protocol provides a unique communication solution that is backward compatible with the installed base of instrumentation in use today. This backward compatibility ensures that investments in existing cabling and current control strategies will remain secure well into the future.

Designed to compliment traditional 4-20mA analog signaling, the HART Protocol supports two way digital communications for process measurement and control devices. Applications include remote process variable interrogation, cyclical access to process data, parameter setting and diagnostics.

STRUCTURE

Specification of the HART protocol is based largely on the OSI 7-Layer Communication Model (see Figure 1).

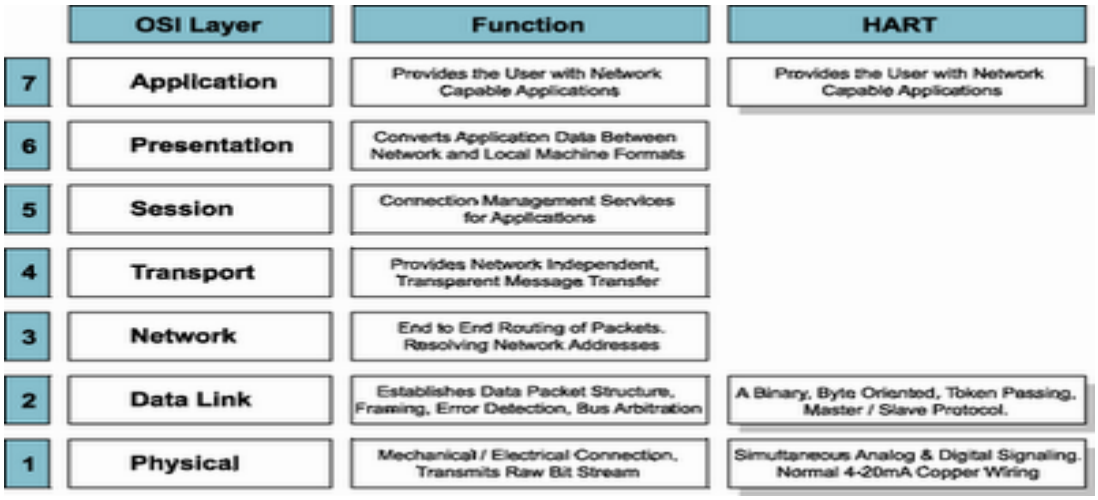


Figure 1. OSI 7-Layer Model

The HART protocol specifications directly address 3 layers in the OSI model: the Physical, Data Link and Application Layers. The Physical Layer connects devices together and communicates a bit-stream from one device to another. It is concerned with the mechanical and electrical properties of the connection and the medium (the copper wire cable) connecting the devices. Signal characteristics are defined to achieve a raw uncorrected reliability (see the FSK Physical Layer Specification).

While the Physical Layer transmits the bit stream, the Data Link Layer is responsible for reliably transferring that data across the channel. It organizes the raw bit stream into packets (framing), adds error detection codes to the data stream and performs Media Access Control (MAC) to insure orderly access to the communication channel by both master and slave devices.

The bit stream is organized into 8-bit bytes that are further grouped into messages. A HART transaction consists of a master command and a slave response. Media access consists of token passing between the devices connected to the channel. The passing of the token is implied by the actual message transmitted. Timers are used to bound the period between transactions. Once the timer expires, control of the channel is relinquished by the owner of the token. For more information see the Data Link Layer Specification.

The Application Layer defines the commands, responses, data types and status reporting supported by the Protocol. In addition, there are certain conventions in HART (for example how to trim the loop current) that are also considered part of the Application Layer. While the Command Summary, Common Tables and Command Response Code Specifications all establish mandatory Application Layer practices (e.g. data types, common definitions of data items, and procedures), the Universal Commands specify the minimum Application Layer content of all HART compatible devices.

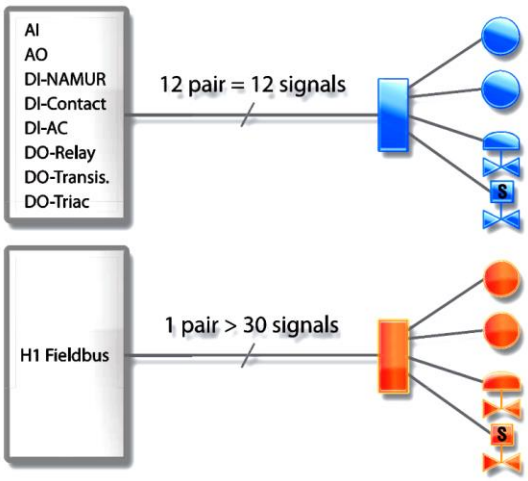
Fieldbus

Fieldbus is a digital two-way multidrop communication link between intelligent field devices. It is a local area network dedicated to industrial automation. It replaces centralized control networks with distributed control networks and links the isolated field devices such as smart sensors/ transducers/actuators/controllers.
Foundation Fieldbus H1 and PROFIBUS-PA are the two fieldbus technologies used in process control. In this two-way communication, it is possible to read data from the smart sensor and also write data into it. The multidrop communication facility in fieldbus results in enormous cable savings and resultant cost reduction.

A comparison between a 4–20 mA system and a fieldbus system is shown in Table.

S.NO	Description	4-20mA System	Fieldbus System
1	Number of field devices per wire	1	Max. 32
2	Signal/data	1	Up to thousands
3	Power supply over the wires	Yes	Yes
4	Signal degradation	Yes	Managed with terminators
5	Failure analysis	By human intervention	Reported at HMI
6	Max. run length	2000 m with proper cables and power supply	1900 m, extendable to 5700 m with repeaters
7	Field device interchangeability	Yes	Yes
8	Intrinsic safety	Yes, more barriers needed	Yes, less barriers needed
9	Control in the field	No	Yes
10	Device failure notification	Very limited	Extensive
11	Networking of field devices	No	Yes

A conventional 4–20 mA current transmission system has two wires each for each of the individual field devices employed.
HART Versus Fieldbus Devices
Compared with this, a fieldbus system has two wires running for many devices that belong to the same segment. A segment may consist of 32 devices.



The above Figure shows a conventional point-to-point communication system and its fieldbus counterpart. There are as many wire pairs as the number of field devices for a point-to-point communication system, while it can be only a single wire pair for a fieldbus system.

History

With digital communication making its way into process automation systems several decades back, different vendors started developing their own protocols—independent of each other. In the initial stages of fieldbus introduction, design engineers were confronted with several problems.First, a particular vendor could not provide all the parts/components needed for a plant and that a particular manufacturer cannot make all the devices better than others. This led to either choosing the less-than-the-best devices from a single manufacturer or else settling for choosing the best devices from different manufacturers.

The latter option would give rise to an integrability problem and lead to isolated islands of automation and consequent inter operability difficulty with devices from different manufacturers.

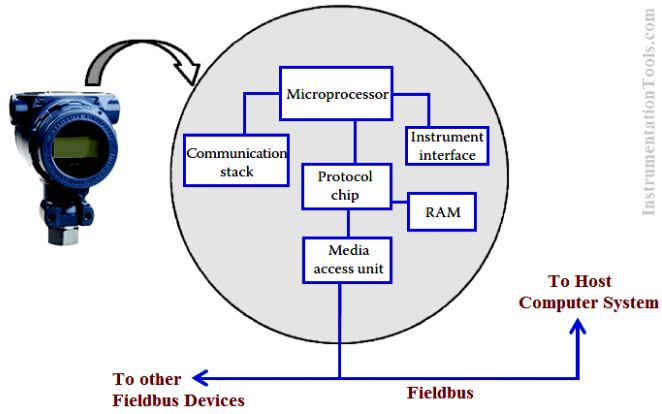


Fig : Fieldbus interface unit in a fieldbus transmitter

Initially when fieldbus was introduced, it suffered from numerous problems such as proprietary protocols, slow transmission speed, and different data formats. Improvements in field signal transmission technology resulted in increasing levels of decentralization.

In 1985, industry experts in the field sat together to work out a vendor independent fieldbus standard—i.e., it would be interoperable. The bus standard would provide bus power, intrinsic safety, and the ability to communicate long distances over existing wires—the basic requirements for a process plant automation system.

Partly due to the complexities of instrumentation automation systems and mostly due to the reluctance on the part of the manufacturers, a single standard protocol architecture is yet to be established. Foundation Fieldbus and PROFIBUS are now the two most dominant fieldbus technologies that are ruling the process automation field. Devices embracing these two technologies cannot communicate with each other because of protocol mismatch and thus seamless interoperability is yet to be achieved.

There are many types of fieldbuses in use today; the particular type to be used depends on the type of industry—discrete or manufacturing automation. Different types of fieldbuses include:

Foundation Fieldbus, PROFIBUS, DeviceNet, ControlNet, InterBus, HART, AS-i, MODBUS, CAN Bus, Ethernet, LonWorks, and WorldFIP.

Figure 1-1 – Digital control system architecture

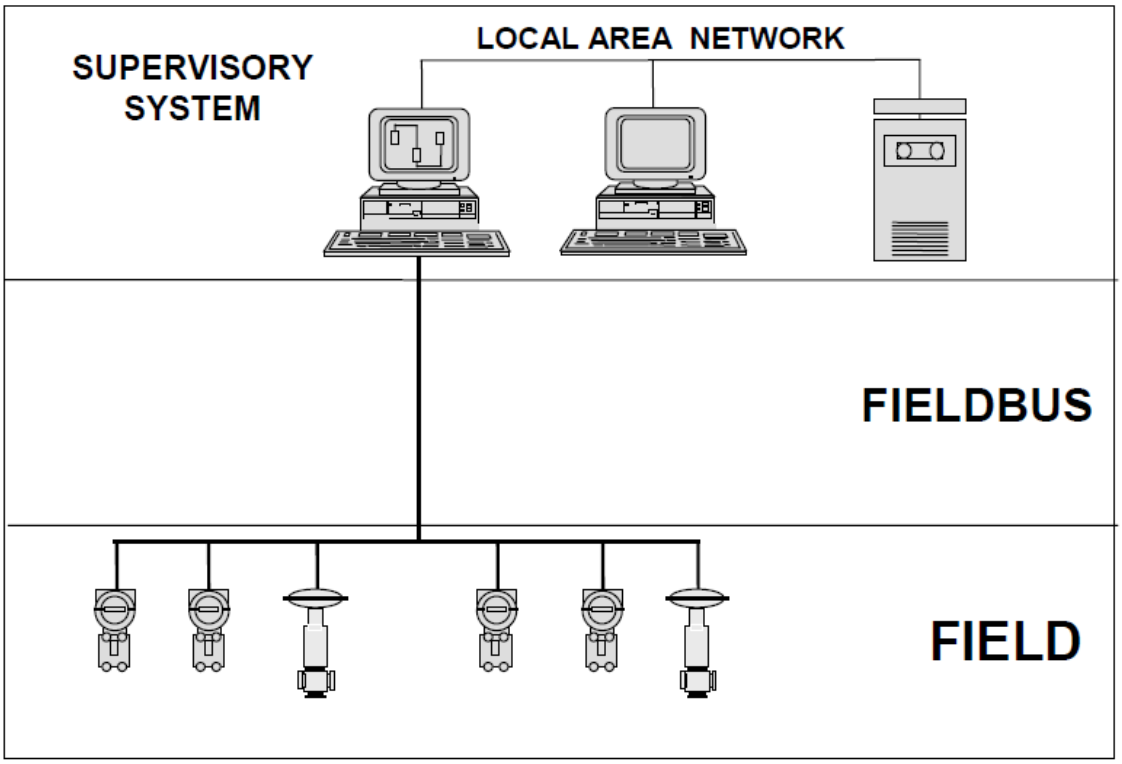


Figure 1-2a – OSI model compared with Fieldbus model

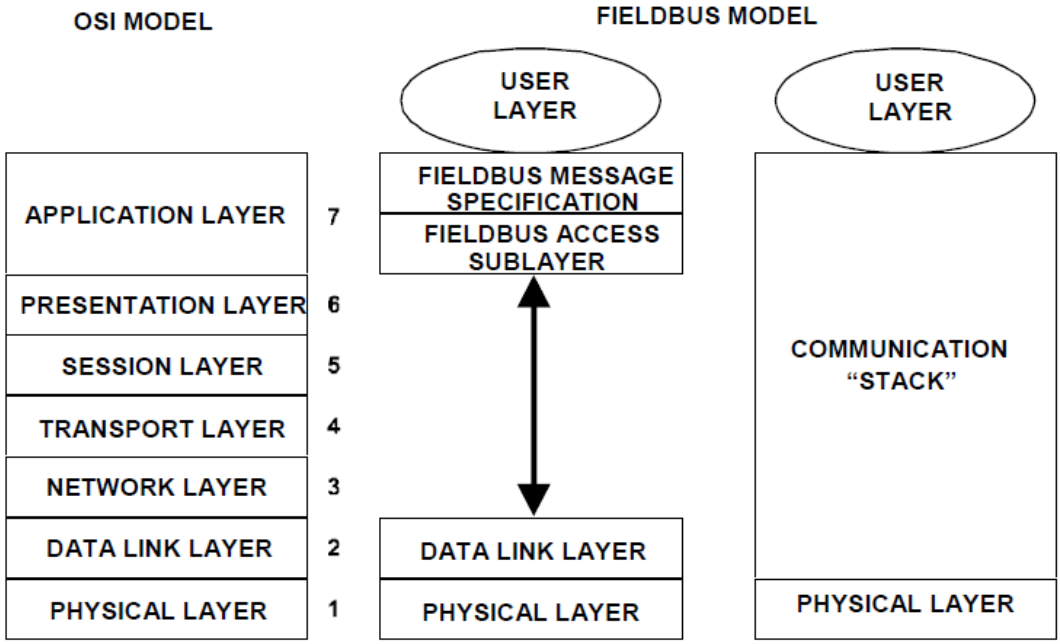
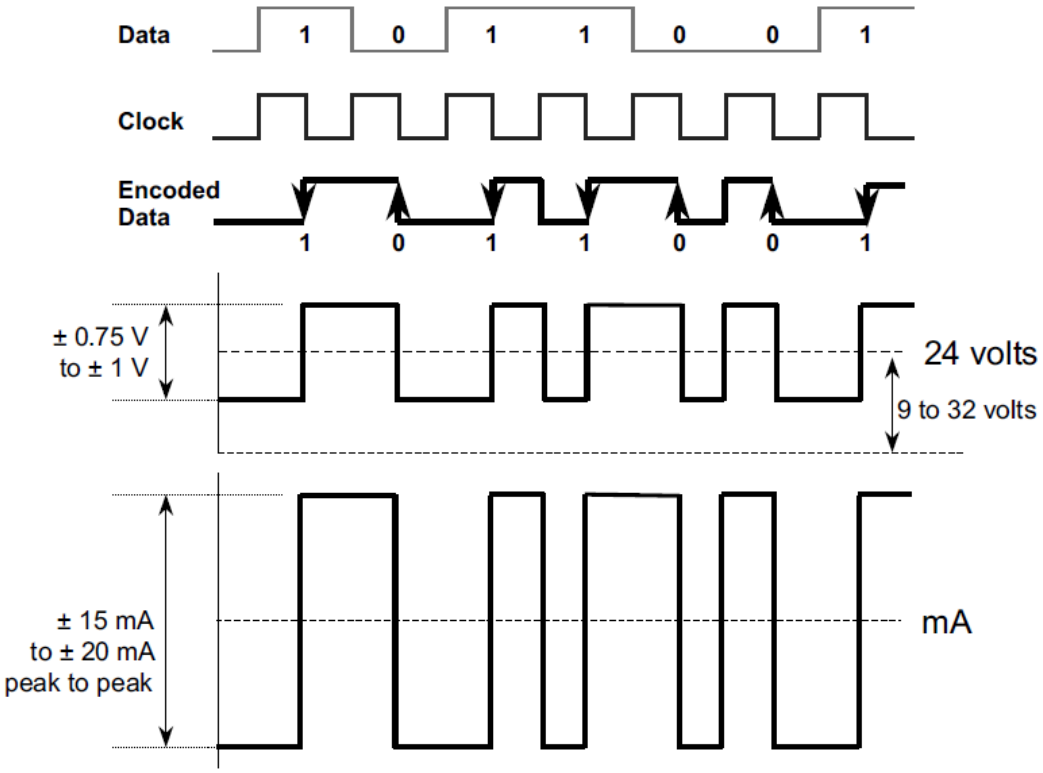


Figure 1-3 – Manchester encoding



1.1 Topology

1.1.1 Application Layer

The Application Layer provides an interface for the device's application software. This layer defines how to read, write, or start a task in a remote node. The main task of this layer is to define syntax for the messages.

Index 255 and below define standard data types such as Boolean, integer, float, bitstring, and data structures that are used to build all other object descriptions.

A Virtual Field Device (VFD) is used to remotely view local device data described in the object dictionary. A typical device will have at least two VFDs: a Network and System Management VFD and a User Application VFD.

Network Management is part of the Network and System Management Application. It provides for the configuration of the communication stack. The Virtual Field Device (VFD) used for Network Management is also used for System Management, and provides access to the Network Management Information Base (NMIB) and to the System Management Information Base (SMIB). NMIB data includes Virtual Communication Relationships (VCR), dynamic variables, statistics, and Link Active Scheduler (LAS) schedules (if the device is a Link Master). SMIB data includes device tag and address information and schedules for Function Block execution.

1.1.2 User Layer

The User Layer defines the way of accessing information within Fieldbus devices so that such information may be distributed to other devices or nodes in the Fieldbus network. This is a fundamental attribute for process control applications.

The architecture of a Fieldbus device is based on blocks, with the Function Block, which as the name implies is an object-based function designed to execute a range of control functions that are responsible for performing the tasks required for the current applications, such as data acquisition, feedback and cascade loop control, calculations, and actuation. Every Function Block contains an algorithm, a database (inputs and outputs), and a user-defined name, typically the loop or tag name since the Function Block tag number must be unique in the user's plant). Function Block parameters are addressed on the Fieldbus by means of their TAG.PARAMETER-NAME.

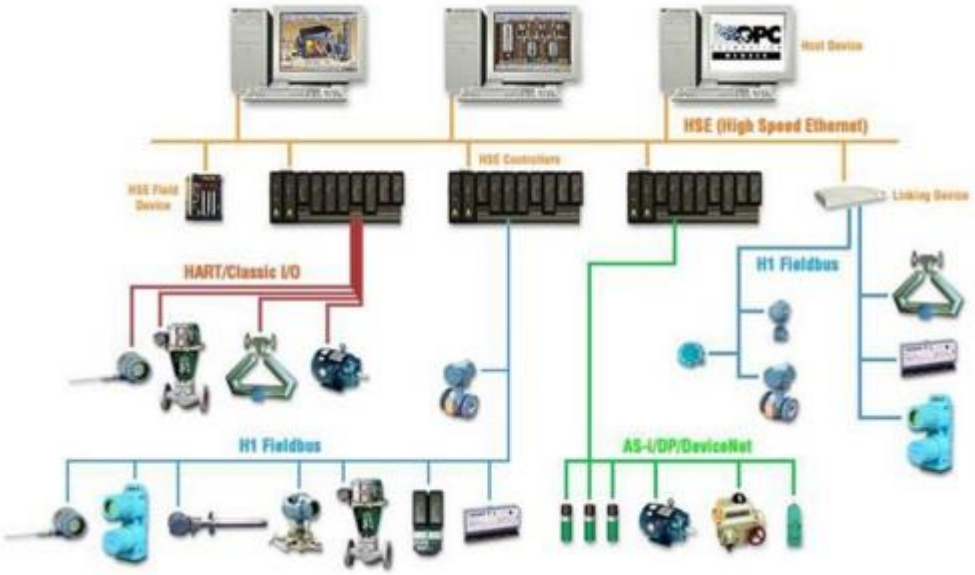
A Fieldbus device includes a defined quantity of Function Blocks of which at least one block must be instantiated or defined.

UNIT III

DCS :- Local Control Unit(LCU) and architecture - LCU languages - LCU - Process interfacing issues. Operator interface - requirements Engineering interface - requirements - displays - alarms and alarm management. Factors to be considered in selecting a DCS. Introduction to SCADA, OLE for Process control(OPC).

Introduction to DCS (Distributed Control System)

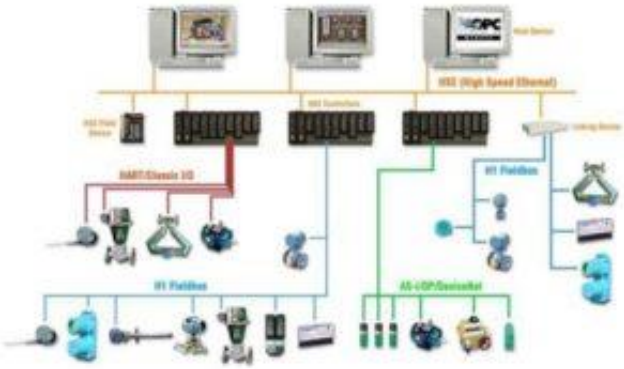
Posted on : April 18, 2016 **By** denizen robo



Automation

What is Distributed Control System ?

There are many definitions of distributed control, but the basic concept is always the same: divide a large application into multiple smaller subsystems, each of which carries out a portion of the application, and allow these subsystems to communicate with one another. Distributed control systems have evolved from two older technologies — direct digital control, and hybrid control made up of discrete devices.



Hybrid Control System :

Hybrid control systems include individual discrete-control hardware, typically programmable logic controllers (PLCs) or analog loop controllers, and a computer to collect process information and generate management reports. Panel-board instrumentation, located in a central control room, is typically used as the operator interface.

While hybrid control systems offer distributed hardware, they do have problems: Each system is custom-designed using controllers made by different vendors, making interfacing difficult. In order to expand a system, specialists in several different technologies are required. Actual control is usually done using an analog controller, which suffers the familiar problem of drift. Installations typically take a long time, due to complexities in interfacing various subsystems together. Because several subsystems are linked, system response is slow and unpredictable. And, lack of a central database makes decision-making more complex.

Digital Control System :

Direct digital control solves many of the interfacing problems of hybrid control. Using a computer to control an entire process allows information to be collected and reported quickly. A major disadvantage of this approach is that the computer represents a vulnerable point — the entire process could shut down if the computer fails. Redundant control systems that consist of either a second computer or analog instruments are usually installed to protect against a total system shutdown, but these backups add greatly to the cost and complexity of the control.

Distributed Control System :

With the advent of the microprocessor, true distributed-control systems became practical. This approach to problem-solving combines the best features of hybrid control systems and direct digital control. It allows the application to be broken into subsystems that use digital, rather than analog, control techniques and that can be interfaced together easily. These systems can easily be expanded to accommodate future plant requirements and to take advantage of the latest control technologies.

Many elements make up the subsystems of the distributed control system (DCS). Typical elements that make up a DCS today include input/output

(I/O) devices, individual controllers (such as PLCs and loop controllers), operator interfaces (such as color CRT cathode ray tube screens), computers for data manipulation, engineering workstations, and communication networks for remote and local information exchange.

An important concept in distributed control is that of breaking down the software program into logical pieces, independent of system hardware. Doing this makes system design much more efficient, since programs are smaller and, therefore, easier to write, debug and maintain than the larger programs in computer and PLC systems. This concept is known as multitasking. It allows multiple small programs to run on a single processor in a priority-structured manner, which ensures that system response will be repeatable.

Advantages of DCS

Because a system can be broken down into subsections that communicate freely with one another, DCS makes the design, implementation and maintenance of complex control strategies easier. Although simpler design and (to some extent) implementation appear to benefit primarily the DCS supplier, the user also benefits — from shorter leadtimes, simpler systems, improved reliability, reduced downtime, reduced installation costs due to wire savings, and greater flexibility to make future enhancements.

Ideally, the DCS will use common hardware throughout. This simplifies the design, since the programming methods and documentation become independent of the subsection that is being designed.

If the DCS is well designed, the engineer has a great deal of freedom to construct the system in a logical manner, taking into consideration the effects of single-point failures on the operation of the process. Unlike control systems based on other strategies, the DCS can continue to operate in a semiautomatic mode when one control section fails. The process can be kept operational while the failed section is repaired, although the process may operate at less than peak efficiency.

FACTORS TO BE CONSIDERED IN SELECTING A DCS

Depending upon the application and design, the controllers differ in size,I/O capability,range of functions & architectural parameters.

- Controller size

  This refers to the number of function blocks and or language statements that can be executed by the controller, as well as the number of process I/O channels provided by the controller.

- Controller functionality

  This refers to the mix of function blocks or language statements provided by the controller (continuous control,logic control,arithmetic functions,or combinations of the above) and also the mix of process input and output types provided by the controller.
  Controller scalability

- Controller performance

  It is the rate at which the controller scans inputs process function blocks or language statements,and generates outputs and also includes the accuracy with which the controller performs these operations.

- Communication channels

  In addition to the process inputs and output channels the controller must provide other communication channels to operator interface devices and to other controllers and devices in the system. The number type and speed of these channels are key controller design parameters.

- Controller output security

  In real-time process control system a mechanism must be provided usually manual backup or redundancy to ensure that the control output is maintained despite a controller failure so that a process shutdown can be avoided.

Other important features to examine in the selection of a DCS for continuous processes:
--Span of control (the number of points that can be handled)

--The number of operator stations supported and the incremental cost of adding operator stations
--The extent and accessibility of distributed databases
--The ability to reconfigure control software without shutting down the process

Advantages of DCS:

Flexibility – probably the most outstand advantage particularly changing the control strategies and expanding easily both size and capabilities of DCS

Lower installed cost

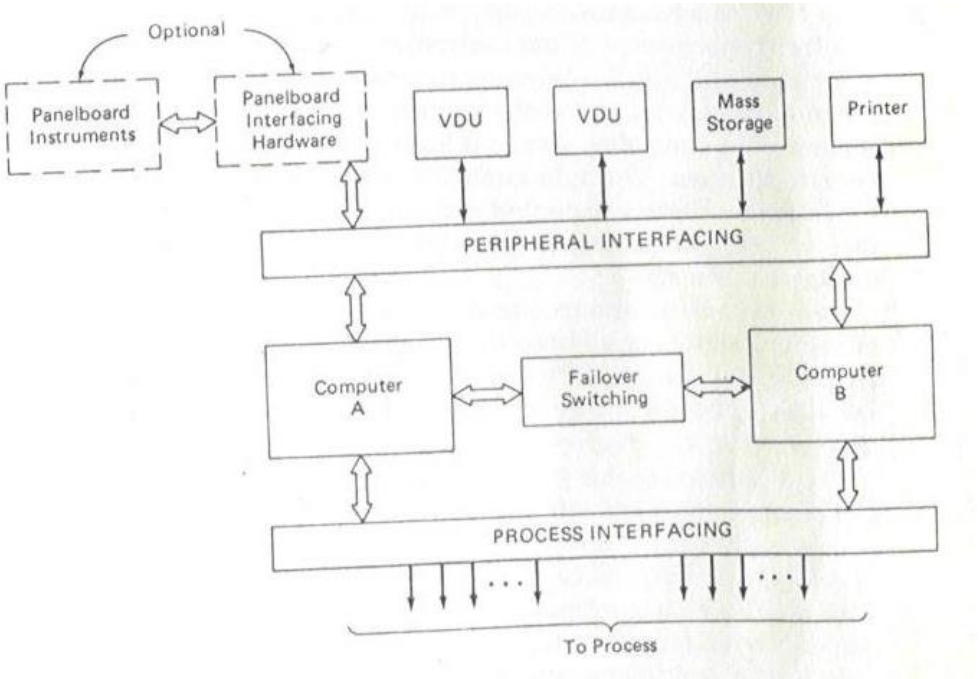
In the process can profit from extensive use of complex controls, then DCS is particularly favorable because conventional system require so much hardware and labor to accomplish similar function. One of the key saving is reduced cost of field wiring (or cabling)

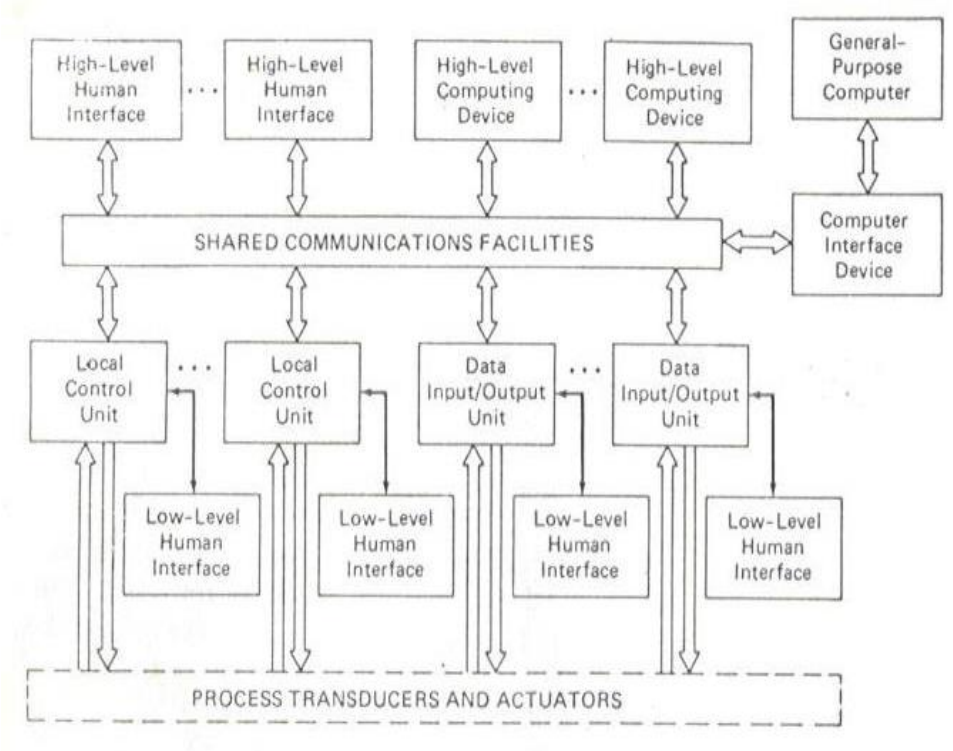
Higher reliability

In the DCS even if the man-machine interface has failed, the controller will continue to control the process and even if the controller fails it may affect either one or few loops. Reliability is further ensured by providing 100% redundancy for critical parts of the system.

Diagnostic facility

DCS offers automatic diagnostics to show system problems and helps in speeding up troubleshooting. If any hardware failure occurs it is displayed on the CRT and alerts the operator through in audible alarm by showing in which station which card/power supply failed. Software failure is also announced as in which file what type of error has occurred.





comparison of hardwired and distributed control system:

HARDWIRED	DISTRIBUTED
- CONTROL SYSTEM NOT FLEXIBLE	- CONTROL SYSTEM PROGRAMMABLE
- SIGNAL TRANSMISSION VIA NUMEROUS CABLES	- SIGNAL TRANSMISSION OVER COAXIAL CABLE
- COMPONENT DRIFT INTRODUCES ERROR IN PERFORMANCE	- DRIFT DOESN'T AFFECT PERFORMANCE
- DISTRIBUTION OF CONTROL NOT POSSIBLE	- GEOGRAPHICAL & FUNCTIONAL DISTRIBUTION OF CONTROL
- SYSTEM TUNING / OPTIMISATION NOT EASY	- SYSTEM TUNING / OPTIMISATION EASY WITH FRIENDLY MAN- MACHINE INTERFACE
- LIMITED DIAGNOSTIC FEATURES	- EXTENSIVE SELF DIAGNOSTIC FEATURES
- MANY VARIETIES OF MODULES	- FEWER VARIETIES OF MODULES
- DATA ACQUISITION IS A SEPARATE SYSTEM	- COMBINED DATA ACQUISITION & CONTROL SYSTEMS CAN BE OFFERED
- ALL SUBSYSTEMS ARE NOT AVAILABLE FROM SINGLE SOURCE	- SINGLE SOURCE OF SUPPLY

LOCAL CONTROL UNIT ARCHITECTURE

Introduction

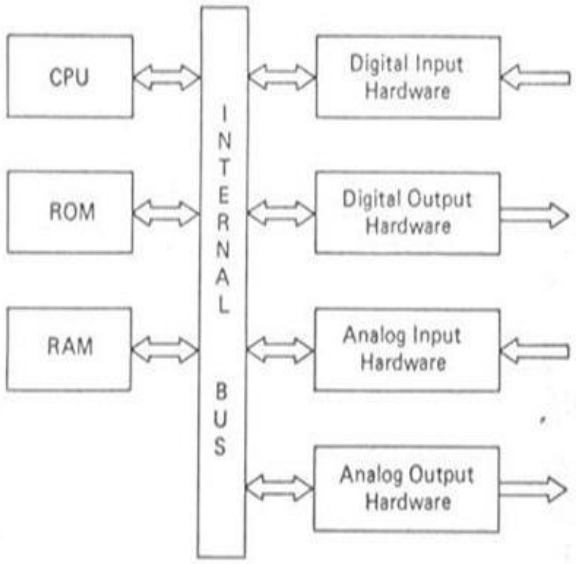
The local control unit (LCU) is the smallest collection of hardware in the distributed control system that performs closed-loop control. That is, it takes inputs from process-measuring devices and commands from the operator and computers the control outputs needed to make the process follow the

commands. It then sends the control outputs to make the process follow the commands. It then sends the control outputs to the actuators, drives, values, and other mechanical devices that regulate the flows, temperature, pressures, and other variables to be controlled in the plant. Since an LCU malfunction can cause a condition that is hazardous to both people and equipment, its proper design is critical to the safe and efficient operation of the plant.

Basic elements of a microprocessor-based controller

The basic elements of a generalized microprocessor-based LCU can be defined as shown in the Figure. The microprocessor along with its associated clock comprise the central processing unit (CPU) of the controller. Read only memory (ROM) is used for permanent storage controller programs, and random-access semiconductor memory (RAM) is used for temporary storage of information. Depending on the type of microprocessor used, RAM and ROM can be located on the microprocessor chip or on separate memory chips.

The LCU also must have input/output (I/O) circuitry so that it can communicate with the external world by reading in, or receiving, analog and digital data as well as sending similar signals out. Generally, the CPU communicates with the other elements in the LCU over an internal shared bus data that transmits addressing, data control, and status information in addition to the data.



The controller structure shown in the Figure is the minimum required to perform basic control function. In a non critical application in which the control function never changes, this structure might be adequate (e.g., in a home appliance whose failure would not cause a safety problem).

The control algorithms could be closed in assembly language and loaded into ROM. After the controller was tuned on, it would read inputs, execute the control algorithms, and generate control outputs in a fixed cycle indefinitely. However, because the situation is not this simple in industrial control applications, the controller structure shown in Figure 2.1 must be enhanced to include the following:

1. Flexibility of changing the control configuration – In industrial applications the same controller product usually is used to implement a great variety of different control strategies. Even for a particular strategy, the user usually wants the flexibility of changing the control system tuning parameters without changing the controller hardware. Therefore, the control configuration

cannot configuration cannot be burned into ROM but must be stored in a memory medium whose contents can be changed, such as RAM. Unfortunately, RAM is usually implemented using semiconductor technology that is volatile; that is it loss its contents if the power is turned off (whether due to power failure. routine maintenance, or removal of the controller from its cabinet). Therefore. Some provision must be made for restoring the control configuration, either from an external source or form a nonvolatile memory within the controller itself.

2. Ability to use the controller without being a computer expert-The typical user of an industrial control system is generally familiar with the process to be controlled. Knows the basics of control system design, and has worked with electric analog or pneumatic control systems before. However, the user is usually not capable of or interested in programming a microprocessor in assembly language. He or she simply wants to be able to implement the selected control algorithms. Therefore, a mechanism for allowing the user to "configure" the LCU's control algorithms in relatively simple way must be provided.
3. Ability to bypass the controller in case it fails so that the process still can be controlled manually-Shutting down the process is very expensive and undesirable for the control system user. Since all control equipment has the potential of failing no matter how carefully it has been designed, the system architecture must allow an operator to "take over" the control loop and run it by hand until the control hardware is repaired or replaced.
4. Ability of the LCU to communicate with there LCU's and other elements in the system has shown in the generalized system architecture in Figure 1.4 Controllers in an industrial control system do not operate in isolation but must work in conjunction with other controllers, data I/O devices and human interface devices. A mechanism for allowing the LCU to perform this interaction must be provided.

A COMPARISON OF ARCHITECTURES

While the above Figure describes basic element of all microprocessor based local control units, the current offerings of controllers in the marketplace exhibit endless variations on this structure. The controllers differ in size. I/O capability range of functions provided, and other architectural parameters depending on the application and the vendor who designed the equipment.

Architectural Parameters

When evaluating the controllers on the market when specifying a new one, the control system designer is faced with the problem of choosing a controller architecture that best meets the needs of the range of applications in which the controller is to be used. Some of the major architectural parameters that must be selected included the following:

1. Size of controller-This refers to the number of function blocks and or language statements that can be executed by the controller, as well as the number of process I/O channels provided by the controller.
2. Functionality controller-This refers to the mix of function blocks or language statements provided by the controller (e.g.. continuous control, logic control, arithmetic functions, or combinations of the above). This also refers to the mix of process input and output types (e.g.. analog or digital) provided by the controller.
3. Performance of controller-This refers to the rate at which the controller scans inputs processes function blocks or language statements, and generates outputs; it also includes the accuracy with which the controller performs these operations.

4. Communication channels out of controller-In addition to process inputs and output channels the controller must provide other communication channels to operator interface devices and to other controllers and devices in the system. The number type and speed of these channels are key controller design parameters.
5. Controller output security-In a real-time process control system a mechanism must be provided usually manual backup or redundancy to ensure that the control output is maintained despite a controller failure so that a process shutdown can be avoided .

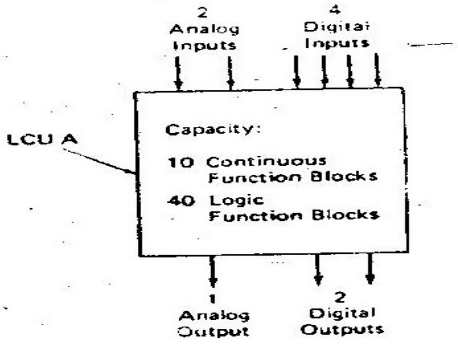


Figure 4.6 LCU Architecture- A

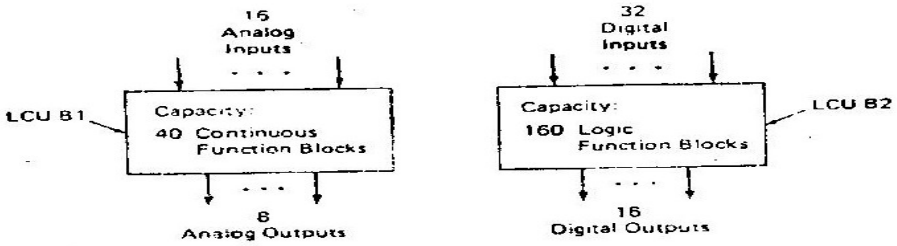


Figure 4.6 LCU Architecture- B

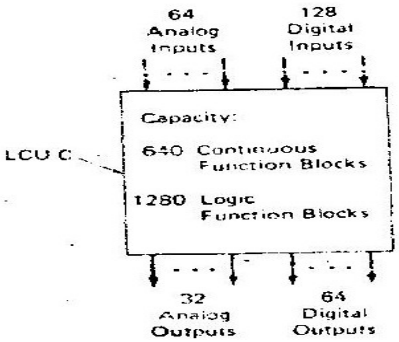


Figure 4.6 LCU Architecture- C

Unfortunately, it is not generally possible to select any one of these architectural parameters independently from all of the others. Since there is a great degree of interaction among them. Therefore, selecting the best combination for the range of applications to be considered is more a matter of engineering judgment than a science. Each vendor of microprocessor based systems has a different view of the range of applications intended for the controller and as a result designs an LCU architecture that quite often differs from that of its competitors.

To illustrate some of the difference in LCU architectures, three representative LCU configurations are shown in Figures 2.3 through 2.5 They are not intended to represent particular commercially available products but rather, different classes of controllers on the market today. Configuration A(Figure 2.3 represents a class of single-loop LCU that provides both analog and digital

inputs and outputs and executes both continuous and logic function blocks. Configuration B (Figure 2.4) represents an architecture in which two different types of LCU's implement the full range of required continuous and logic functions.

Configuration C (Figure 2.5) represents a multi loop controller architecture in which both continuous and logical functions are performed. For each example, the architecture is defined in terms of its analog and digital I/O capacity and its continuous and logic function block capacity. In Table 2.1; these architectures are compared on the basis of the parameters defined previously: size, functionality, performance, communications, and output security provisions.

Table 2.1. Comparison of Architectures

ARCHITECTURE PARAMETERS	CONFIGURATION A (SINGLE –LOOP)	CONFIGURATION B (2 LCU TYPES)	CONFIGURATION C (MULTI-LOOP)
Controller size	Number of functions needed for single PID loop or motor controller.	Includes functions and I/O needed for eight control loops and a small logic controller.	System size is equivalent to small DDC system.
Controller functionality	Uses both continuous and logic function blocks.	Continuous and logical function blocks split between controllers	Uses both continuous and logical function blocks; can support high-level languages.
Controller scalability	High degree of scalability from small to large systems	Requires both controller types even in smaller systems.	Not scalable to very small systems.
Controller performance	Requirements cam be met with inexpensive hardware	Because of functional split performance requirements are not expensive.	Hard ware must be high performance to execute a large number of functions.
Communication channels	Need intermodule communications for control: only minimum needed for human interface.	Functional separation requires closer interface between controller types.	Large communication requirement to human interface : minimum between controllers
Controller output security	Controller has single loop integrity: usually only manual backup is needed	Lack of single loop integrity requires redundancy in critical applications.	Size of controller requires redundancy in all applications.

LOCAL CONTROL UNIT LANGUAGE

Since the local control unit (LCU) is a microprocessor-based device, the only functions of its hardware elements are to bring signals in and out of the device and to provide an environment for the execution of its programs stored in read-only memory. The control and computing functions that the LCU performs are defined by these programs and by the parameters stored in nonvolatile memory. Of course, the programs and parameters are stored in the LCU memories in the form of bits (binary digits-0_s and 1s) and bytes (groups of eight bits) as in the case of any computer-based device. Unfortunately, the user of the distributed control system is not likely to be fluent in this machine-level-language of bits and bytes. Therefore, a higher-level language must be provided to allow him to interact with the digital hardware in defining the control and computational functions to be implemented in the LCU.

LANGUAGE REQUIREMENTS

The control-oriented language selected for use in a distributed control system has a critical impact on how well the user accepts the system. Since this language is one of the primary mechanisms by which the user interfaces with the system. Therefore, one of the first requirements on the language is that it must allow the user to specify the control and computing functions to be implemented in the LCU in a comfortable manner. The definition of comfortable depends to a great extent on the background of the user.

A user who has had experience primary in the area of conventional control devices probably is not interested in becoming a computer programmer simply to use the control system. The user is used to selecting control functions in the form of dedicated hardware modules (e.g.. summer, integrator, or PID control modules), then interconnecting these modules and tuning them to implement the desired control strategy. Such a user would like to interact with the distributed control system in a similar manner-configuring the control strategy instead of programming it. On the other hand, a user who comes from a computer control back ground is used to the power and flexibility of a general-purpose computer language. This kind of user would not be disturbed by the prospect of doing some programming to implement the precise control function in mind for the application. Selecting the control language (or languages) for the LCU must take into account the needs of both these types of user.

Another requirement is that the language must allow the user to implement at least the set of control functions that have been provided in the past by conventional analog, sequential and programmable control systems. Communication functions also are required to allow the LCU's to exchange information with other elements in the distributed control system. A representative list of these control and communication functions includes:

1. Data acquisition and Signal conditioning functions. Such as input scanning, filtering, linearization and conversion to engineering unit
2. Limit checking and alarming functions:
3. Modulating control function, including proportional-integral-durative (PID) control with all its variations:
4. Sequential control functions implementing Boolean logics such as AND. OR. and NOT and related functions. Such as time delays and latches:
5. Computational functions such as arithmetic addition subtraction multiplication and division trigonometric (e.g. sine and cosine) and dynamic signal processing (integral derivative and filter) fruitions:
6. Signal output functions both for control and for driving recorders and indicators:
7. Communication functions to allow signal transmissions between the LCU and other controllers and elements in the distributed control system:
8. Communication functions to human interface devices that allow operators and engineers to interact with the LCU.

Another desirable feature is for the language to allow the user to modify these standard functions or to include custom functions within the control language structure. As the LCU's become more powerful in processing capability it is becoming feasible and cost-effective to distribute functions previously performed by a central computer into the LCU's.

These functions include plant performance monitoring long-term data storage and retrieval and computing of optimal set points for plant operation. In the past these functions have been implemented

using a general purpose programming language. Such as FORTRAN or BASIC in the central computer. Therefore it is essential that the LCU's be able to support these types of high level language so that the user can make use of software already in his or her program library or available as standard commercial packages.

As much as possible the control language selected should be transportable: that is it should not be dependent on the microprocessor used in the LCU. There are two reasons for this requirement: (1) the user may want to implement the same control logic on systems provided by different vendors. and (2) the same control logic should be usable if the control system vendor decides to upgrade the LCU by going to a new more powerful microprocessor.

Obviously the control language must be compatible with the hardware environment in which it will run and with the performance requirements of the control system. The system must provide for sufficient memory capacity to store the control algorithms and data in the LCU. The microprocessor selected must be powerful enough to execute the control algorithms fast enough to meet the needs of the process to be controlled.

The major language alternatives currently available to the industrial control user are:

1. Function blocks, preprogrammed functions whose parameters can be set by the user and which can be linked together much like traditional discrete hardware modules:
2. Problem-oriented languages (POLs) that are customized for a specific type of control application:
3. High-level language that offer the user some degree of flexibility in designing his or her own control algorithms while maintaining compatibility with function blocks or POLs in the system.

FUNCTION BLOCKS

In current distributed control systems the most prevalent prepackaged control language at the LCU level is one of function block. In this language approach, the user receives a standard library of control and stored in ROM. The user then configures the control system by selecting the appropriate blocks from the library linking them together and setting the various input and tuning parameters associated with each block. Table 3.1 gives one example of a library of function blocks used an a commercially available controller. It is clear from the example that many of the functions provided are very similar to those available using hardware modules in discrete modulating and sequential control systems. However, the functional capabilities of the software function blocks goes beyond that of hardware modules in several areas:

Examples of function block library:

Sum-2 input 4	Integrator
Multiply	Lead/Lag
Divide	Moving average
Square root	Analog time delay
Y^x E^x	High/low limit
LogX in X	Rate time
Trigonometric	**Signal status functions**
Generalised polynomials	High/low alarm
Two-dimensional interpolation	High/low select
Matrix addition	Analog transfer
Matrix multiplication	Digital transfer
Control functions	**Logic functions**
PID control	AND

Pulse Positioner	OR
Adapt block	Qualified OR
Smith predicator	NOT
General predicator	Latch
Inter module communications	Digital Timer
Analog input	Up/Down counter
Analog input list	Remote control latch
Analog output	Pulse rate controller
Digital input	**Operator communications**
Digital input list	Control station
Digital output	Indicator station
	Ratio station

1. Functions are provided that would be very difficult or expensive to implement in hardware. Example include the analog time-delay block used in the control of processes that have transport delays the adapt block (which allows running parameters in one functions block to be adjusted from an externally computed or transmitted signal). and the more complex computational blocks (e.g.. exponential functions and matrix operations).
2. Since the functions blocks are implement in software instead of in hardware. One can easily alter the control configuration without changing any controller wiring or purchasing additional hardware assuming that the controller's function block capacity has not been exceeded).
3. The computational capability of the digital controller allows the conversion of all physical inputs to the system to engineering units (e.g.. from milliamps and volts to degrees Celsius or pressure in atmospheres). All of the control and computing operations can then be carried out using engineering units there by eliminating tedious scaling and conversion operations required in conventional analog control systems.

Example of Continuous Control

To introduce the function block concept, Figure 2.2 gave an example of how function blocks are used in configuring a control strategy. This example can be interpreted as the implementation of a simple flow control loop. Inputs 1 and 2 are the physical inputs to the LCU: input 1 is the flow set point from an external operator input device, and input 2 is a signal input from a differential pressure transducer. To convert input 2 is a signal, the user selects the square-root function block from the library and identifies its input as coming from physical input2.

The user then enters appropriate values of the gain and bias parameters to accomplish the desired square-root function and convert the input signal to engineering units (say, flow in gallons per minute). Next the user selects the difference block in order to generate an error signal between the actual flow and the flow set point. Again, the user identifies the inputs to the difference block and chooses the gain corresponding to input 1 so that the set point also is in units of gallons per minute. Using the output of the difference block as its input.

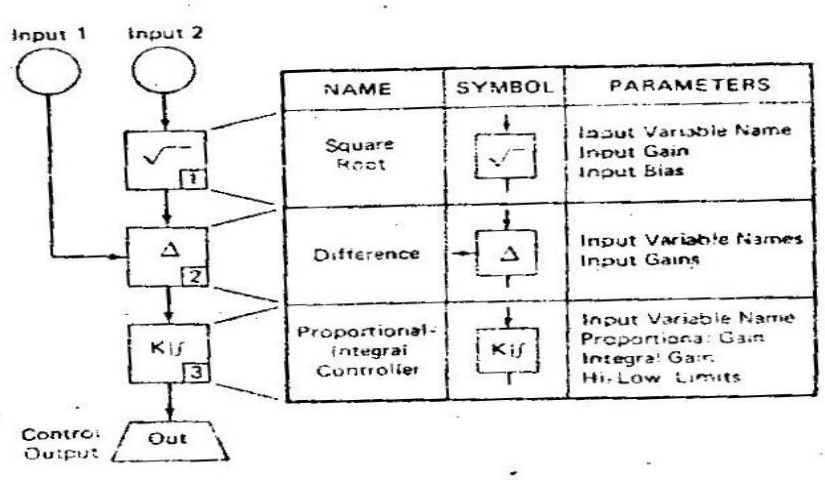


Fig. 4.9 Continuous control example

The user then selects a proportional-integral (PI) block to perform the control function. The tuning parameters (proportional gain and integral gain) that are appropriate for the valve and process dynamics then are selected, and the high low control limits (if any) are set. Once the PI block output is linked to the physical output from the LCU the control configuration is complete. Once the control configuration has been created the LCU can be put on-line so that it can start executing the algorithms in the configuration. In most systems, the function block parameters that do not affect the control configuration can be turned changes while the LCU is on-line however, the LCU generally mist be taken off-line if the control configuration itself must be changed. (This is equivalent to rewiring an analog control system.)

PROBLEM-ORIENTED LANGUAGES

As in the function block approach to defining control system logic, problem-oriented languages (POLs) do not require the user to have a knowledge of computer programming. In general, these are special-purpose languages: each one is tailored to a particular vendor's hardware, specific control application, or both.

There are two popular types for POL: fill-in-the-form and batch languages. In the first type, the user decides what the configuration of the control system will be and then literally fills in forms provided by the vendor. This defines the configuration using the vendor's hardware/software system. a technician then enters the control configuration into the LCU through a terminal or other device, using the form as the source of the input data. This approach was developed during the early years of computer control systems as a first attempt on the part of vendors to provide prepackaged software for nonprogrammers.

Figure gives an example of the fill-in-the-from approach to defining a single PID loop. Note that this is an expanded version of the PID function in the programmable controller example in Figure 3.5. since a different form must be filled out for every function, it is clear that a large amount of paperwork is involved in configuring a complex control scheme. On the other hand, once the forms have been filled out, they provide a convenient means of documenting the control system configuration for the user.

While this control language approach does not require user programming as such, it suffers form the following defects:

1. The user cannot easily determine the structure of the control system from the form entries themselves: separate documentation must be maintained to define the overall structure.

2. The definition procedure is tedious and has more of the flavor of programming than any other approach.
3. There is little or no standardization in the forms from one vendor to another.

The second popular type of POL is the batch language, used in control application of the same name. It consists of a set of commands that direct the LCU to perform certain operations, such as opening a valve, ramping a temperature (moving it from one value to another), or filling a tank. Associated with each of the commands in one or more parameters that define the valve to be opened, the beginning and end temperatures in a ramp, the time of the ramp, and so forth. Figure 3.7 lists a set of example.

```
Initial Conditions: L1 = 0
                    V1, V2, V3 are closed.
                    Agitator is stopped.
Batch Program:  START PROCESS
                    OPEN V1
                    WAIT UNTIL L1 = 10
                    CLOSE V1
                    START AGITATOR
                    START TIMER
                    ACTIVATE CONTROLLER C1
                    WAIT UNTIL TIMER = 10
                    DEACTIVATE CONTROLLER C1
                    WAIT UNTIL TIMER = 10
                    DEACTIVATE CONTROLLER C1
                    CLOSE V3
                    STOP AGITATOR
                    OPEN V2
                    WAIT UNTIL L1 = 0
                    CLOSE V2
                    STOP PROCESS
CONTROLLER C1 PARAMETERS:
    Loop Tag: TC1                       Proportional Gain: 1.5
    Set Point Source:   STA4            Rest Time: 30 min
    Engineering Units: Deg. F       Derivative Time: 30.0 min
    Process Variable:   T1              Track Set Source: R4
    PV S   PAN values: 0-300        Track Ref. Value: 0
    Loop Output:  v3                    Scan Period 5 sec.
```

Batch language statements

A batch language can consume a significant amount fo processor and memory resources in the OCU. Therefore, batch languages have not proved to be cost-effective for many general-purpose control applications. As in the case of fill-in-the-from languages, batch control languages have been used mainly in computer control applications. As the computing capabilities of microprocessor-based LCU's continue to increase, however, it is clear that LCU will be designed to support the use to batch control languages.

HIGH-LEVEL LANGUAGES

Until the early 1980s, high-level programming languages such as FORTRAN, BASIC, and PASCAL) were restricted primarily to larger minicomputer systems used in direct digital control, supervisory control, or data acquisition applications. In industrial control, they were not seriously considered for use in dedicated microprocessor-based systems for a number of reasons. First.

The performance and memory capabilities of the microcomputer hardware available were not inclined to learn high-level programming languages. They were more comfortable with the other language approaches., including function block and fill-in-the-form languages, discussed earlier.

However, this situation has changed significantly for a number fo reasons:

1. The increasingly widespread use of personal computer is creating a class of users who are comfortable with certain high-level languages, such as BAISC.
2. Control system engineers are becoming more sophisticated and are demanding control capabilities that are more specialized than can be provided by means of standard, "canned" functions.
3. The continuing explosion in the performance capability of microprocessor and memory systems is allowing the migration of control functions formerly performed by a computer down into the level of the LCU. Therefore, functions formerly implemented in high-level languages (e.g., supervisory control or plant performance calculations) can now be considered for implementation in the LCU. While many of these functions can be implemented using function blocks of POLs, the remainder require a high-level language implemented at the LCU level

OPERATOR INTERFACE

introduction:

This automated equipment to be the use in safe an defective manner, however, it is absolutely necessary to have a sell-engineered human interface system to permit error-free interactions between the humans and the automated system. two distinct groups of plant personnel interact with the control system on a regular basis:

1. Instrumentation and control system engineers – Theses people are responsible for setting up the control system initially and adjusting and maintaining it from time to time afterwards:
2. plant operators- These people are responsible for monitoring. Supervising, and running the process through the control system during startup, operation, and shutdown conditions

as the generalized distributed control system architecture in shows a human interface capability can be provided at one or both or two levels:

1. Through a low-level human interface (LLHI) connected directly to the local control unit or data input/output unit (DI/OU) via dedicated cabling:
2. through a high-level human interface (HLHI) connected to an LCU or DI/OU only through the shard communications facility.

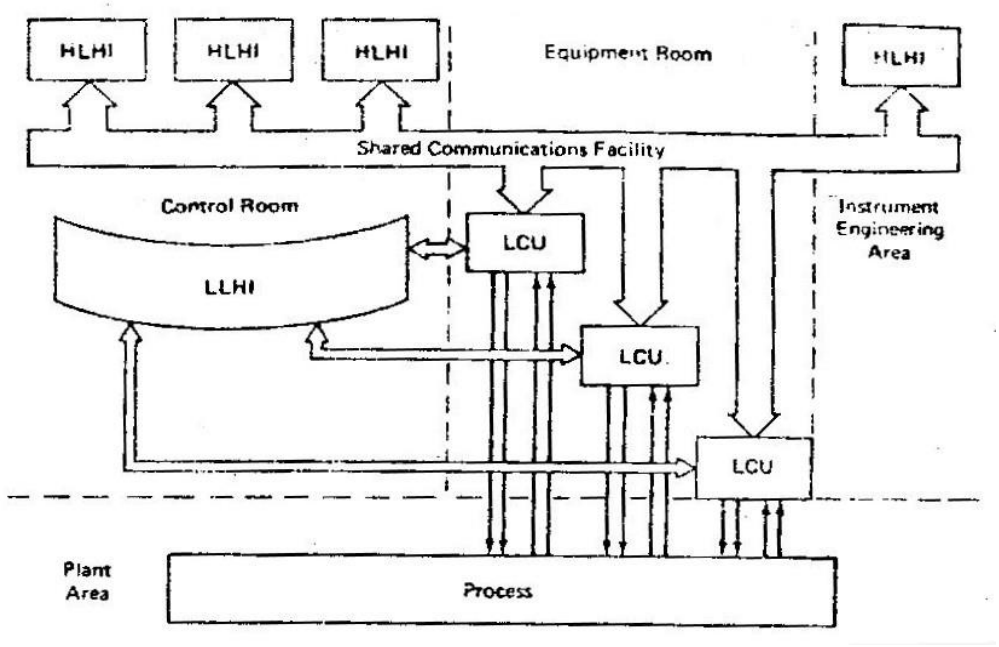


Figure. Geographically Centralized Control Configuration

The low-level human interface equipment used in distributed control systems usually resembles the panel board instrumentation (stations, indicators, and recorders) and thing devices used in conventional electronic the at least display technology (e.g., CRTs or flat-panel displays) and peripheral) devices (e.g., printers and magnetic storage) that are available on the market: it is configured in a console arrangement that allows operator and engineer to be seated during use.

When it is included in the system configuration, the LLHI generally is located geographically close to (within 100-200 feet of) the LCU or DI/OU to which it is connected. on the other hand, the HLHI can be located any where in the plant, including the central control room. The needs of the application will determine whether the particular installation has one or both levels of interface.

Figure illustrates a relatively small and simple installation. A single LCU located in the plant equipment room and the plant level human interface units locate in the plant equipment room (sometimes called the relay room) performs all of the required control functions. Low-level human interface units located in the equipment room and the plant control room provide the complete operator and instrument engineer interface for the control system. this type of equipment configuration is typical of a standalone control system for a small process or of a small digital control system installed in a plant controlled primarily with conventional electrical analog or pneumatic equipment.

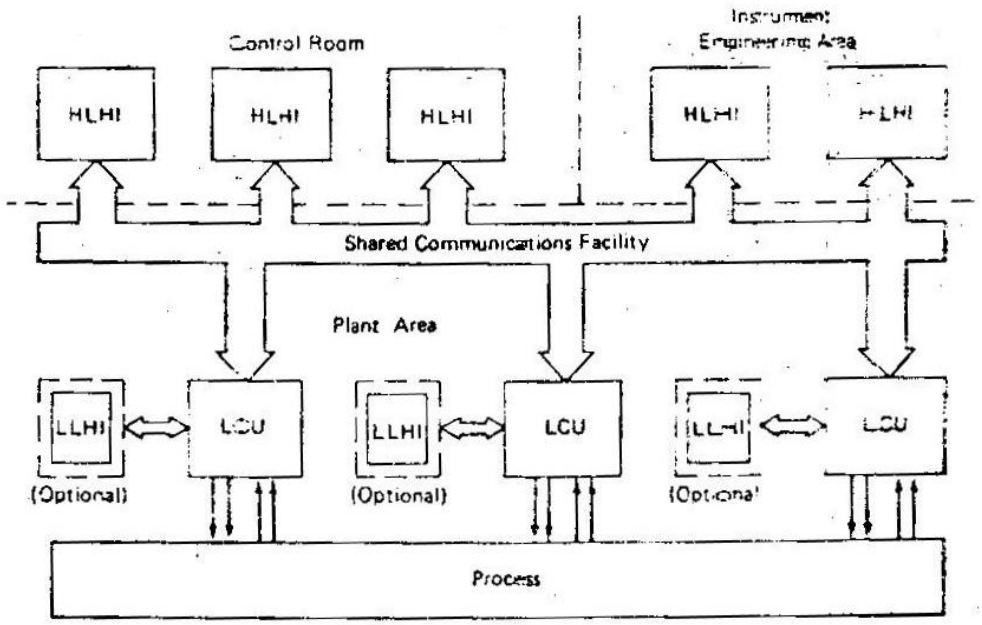


Figure . : Geographically distributed control configuration

Figure shows a typical structure of a complete plant wide control system. Several LCUs are used to implement the functions required in controlling the process; therefore, the control is functionally distributed. However, the LCUs are all located in a central equipment room area, and so it is not a geographically distributed control system. both high-level and low-level human interface devices are located in the control room area for operational purposes most of the operator control functions are performed using the high-level interface; the low-level interface is included in the configuration primarily to serve as a backup in case the high-level interface fails. A high-level human interface is located in the instrument engineer's area so that control system monitoring and analysis can be done without disturbing plant operations. This type of installation is typical of early distributed control system configurations in which equipment location and operator interface design followed conventional practices.

The Figure shows a fully distributed control system configuration. In this case, each LCU is located in the plant area closest to the portion of the process that it controls. Associated low-level human interface equipment (if provided) is also located in this area. The control room and instrument engineering areas contain high-level human interface units, which are used to perform all of the primary operational and engineering functions. The low-level units are used only as manual backup controls in case the high-level equipment fails or needs maintenance. This configuration takes advantage of tow areas of equipment savings that result from a totally distributed system architecture: (1) reduction in control room size (by eliminating panelboard equipment), and (2) reduction in field wiring costs (by placing LCUs near the process).

These examples of system configurations illustrate the point that human interface equipment in distributed control system must be designed to meet a wide range of applications:

1. Large as well as small systems;
2. centralized equipment configurations (often used in retrofit installations made long after original plant construction sas well as distributed ones likely in "grass roots" installations made during plant construction:

3. variety of human interface philosophies, ranging from accepting CRT-only operation to requiring panelboard instrumentation in at least a backup role.

OPERATOR INTERFACE REQUIRMETNS

Despite the continuing trend toward increased automation in process control and less reliance on the operator, the basis responsibilities of the operator have remained largely the same in the last fifty sears most of the changes have come in the relative emphasis on the various operator functions and the means provided to accomplish them. As a result, the operator interface in distributed control system must allow the operator to perform tasks in the following traditional areas of responsibility: process monitoring, process control, process diagnostics, and process record keeping. In addition, it is important to design the operator interface system using human factors design principles (also called ergonomics) to ensure that the operator can perform these tasks in an effective manner with that the operator can perform these tasks in an effective manner with minimum risk or confusion error.

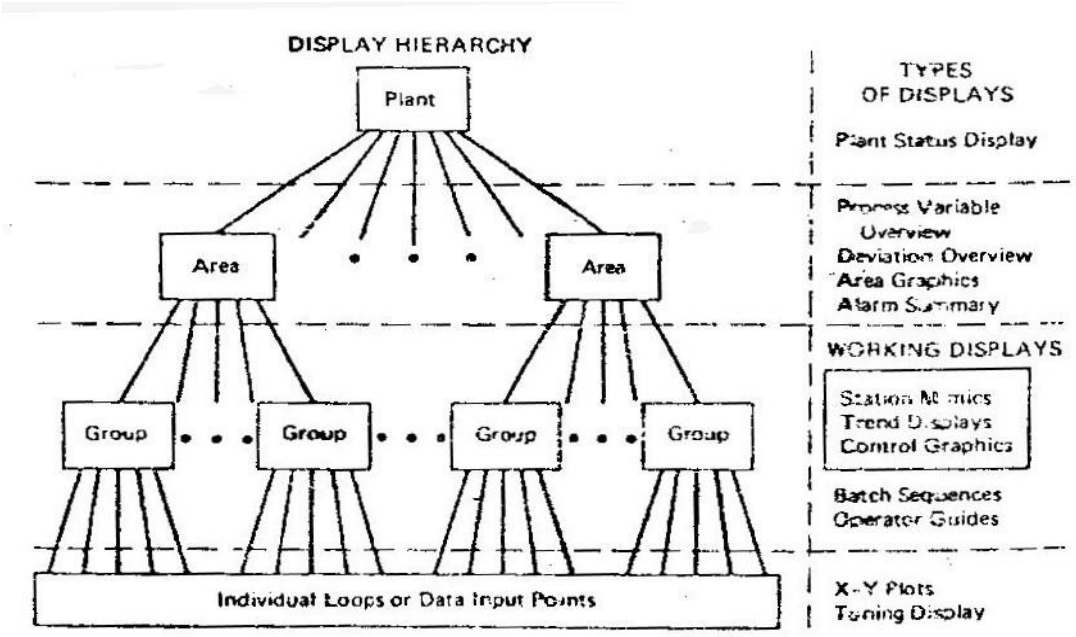
Operator Displays

The panelboard in a conventional control room uses many square feet of dedicated instruments to provide the operator with the information and mechanisms needed to control the plant. In theory, the operator has simultaneous access to all of these instrument s at one time, since they all are physically located in the same room. In practice, of course, the operator must move about the room to be able to see the indicators and manipulate the various stations needed to control the plant.

The video display unit in an system, in contrast provides a "window" to the process path allows the operator to see only a relatively small amount of information at any one time on one or more CRT displays. The operators is able to monitor and control the whole process only by calling up a number of these displays, which usually are arrang3ed in a fixed logical structure or hierarchy. If this display structure and the associated display access mechanisms are designed properly, the will provide the operator with much faster access to the needed information than is possible by moving around a panelboard (see Reference 6.8).

Typical Display Hierarchy. The flexibility of CRT display technology makes it possible to conceive a great number of different display structures that would be appropriate for industrial control systems. A typical version of this hierarchy, illustrated in Figure 6.13, is composed of displays at four levels :

1. Plant level-Displays this level provide information concerning the entire plant, which (if large enough) can be broken up into several areas of interest.
2. Area level-Displays at this level provide information concerning a portion of the plant equipment that is related in some way, e.g., a train of separation processes in a refinery or a boiler-turbine-generator set in a power plant.
3. Group level- Displays at this level deal with the control loops and data points relating to a single process unit within a plant area, such as a distillation column or a cooling tower.
4. Loop level- Displays at this level dean with individual control loops, control sequences, and data points.



This general display structure is attractive from several points of view first, it covers the full range of detail of information that is likely to be of interest to the operator, from overall plant conditions to the status of each lop in the plant. Also, it allows for the grouping of available information in a way that matches the structure of the process itself. Finally it provides a mechanism that allows the operator to from a mental model of the relationships between the various pieces of information in the plant. This is similar to the mental model of a panelboard that develops in the mind of an operator after gaining experience with its layout. After a period of weeks or months, the operator no longer has to refer to the labels on the panelboard to find a particular instrument, but moves to it instinctively. Similarly, a meaningful display structure such as the one shown in Figure 6.13 allows the operator to learn to move from one display to the next in a smooth and efficient manner.

Plant-Level Displays.

Typically, at the top level of this structure is a single type of plant of plant status display (perhaps consisting of several pages) as Figure 6.14 illustrates. This display summarizes the day information needed to provide the operator with the "big pocture" of current plant conditions. This example shows the overall production level at which the plant is operating compared to full capacity. It also indicates how well the plant is running (e.g., by plotting efficiency of energy usage) In addition, some of the key problem areas (e.g., equipment outages or resource shortages) are displayed. A summary of the names of the various areas in the plant serves as a main menu (index) to the next level of displays. At the top of the plant-status display is a status line of information provided in all operating displays. This line shows the current day of the week, the date, and the time of day for display labeling purposes. In addition, it provides a summary of process alarms and equipment diagnostic alarms by listing the numbers of the plant areas in which outstanding alarms exist. (The subject of alarming is discussed further in Section 6.4.5)

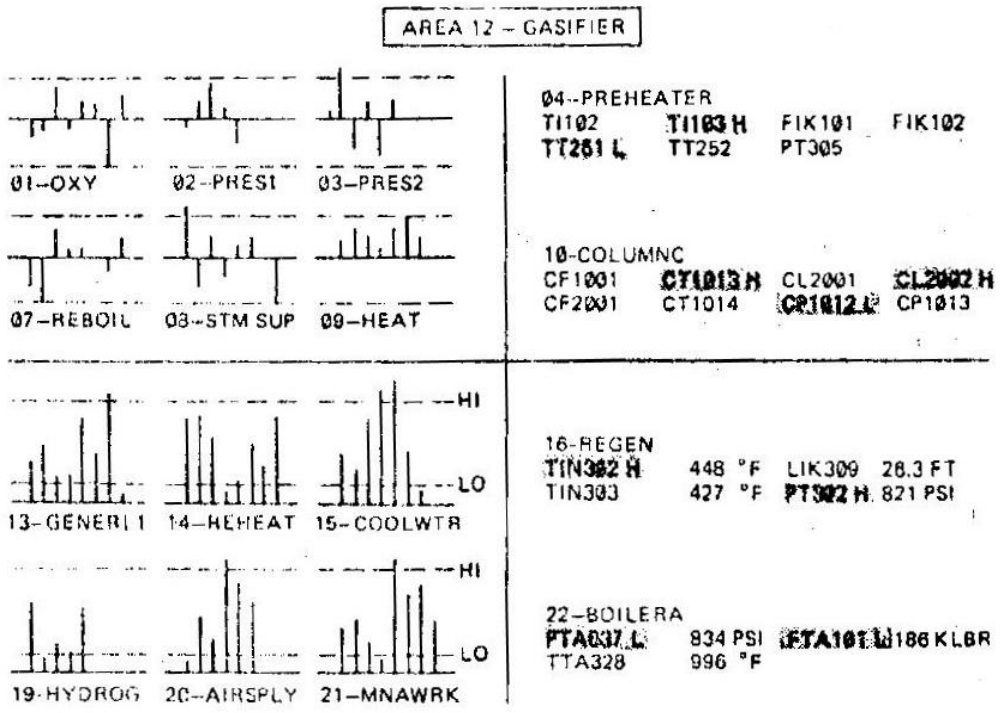
Area-Level Displays.

After obtaining a summary of the plant status display hierarchy, the operator can move down to the next level to look at the situation in a selected plant area. This can be done by means of several types of displays; Figure 6.15 shows a composite of four of these types. The top line of the display is

the system data and status line described previously. The upper left quadrant illustrates an area display type known as deviation overview, which display in bar graph form the deviation of key process variables from their corresponding set points. The deviation of key process variables from their corresponding set points.

The deviations are usually normalized to reflect a percentage of total span, and are clustered into a number of groups within each area. If the absolute value of deviation exceeds a predetermined level (e.g., 5 percent of span) the process variable enters a deviation-alarm status condition and the bar graph for that variable changes color. This approach to overview display derives from the green-band concept in conventional analog instrumentation, in which the manual-auto stations for continuous control loops are arranged side by side in a row on the panelboard. For each loop, if the process variable is within a small percentage of the set point, the analog pointer for that variable remains hidden behind a green band on the station face. The operator then can determine which loops are upset by simply scanning the row of stations and seeing which pointers are outside the green band. The deviation overview display provides the operator with the same information in a CRT display format.

The lower left quadrant of Figure 6.15 shows a variation of this approach in which a bar graph indicates the absolute value of the process variable instead of its deviation from set point. some versions of this display also show the set point and the high and low alarm limits on the process variable. When one of these limits is exceeded, the bar graph changes than the previous one in that it accommodates the alarming of process variables that are not used in control loops as well as those that are.



The two display types just described essentially mimic the analog portions of a conventional panelboard. The upper right-hand quadrant of Figure 6.15 shows another approach to the area overview display. Here the tag numbers of the various loops and process variables are arranged in clusters by group. If the point associated with a particular tag is not in alarm, its tag number is displayed in a low-key color. If it does go into alarm, it changes color and starts flashing to get the attention of the operator. Underlining also can be used under the tag number, so that a colorblind operator still can see the alarm

state clearly. This format of an overview display is similar to that of an alarm annuciator panel in a conventional panelboard.

The lower right-hand quadrant shows a variation of this display. In addition to the tag number it self, the current value of the process variable is displayed in engineering units to the right of the tag. This provides the operator with information on the values of the key variables in a group in addition to their alarm status.

In come implementations of area overview displays several of these approaches may be intermixed in a single display. Also tow other types of area-level display often are provided:

1. Area graphics display-This display is similar to a piping-and-in strumentation diagram (P&ID) or mimic panel used on conventional panelboards to illustrate the process equipment and its associated instrumentation. It usually is designed to provide the same type of information that other area displays give: alarm status and perhaps current values of key process variables. the capabilities and use of graphics displays are discussed in more detail later in this section.
2. Alarm summary display – This display is simply a listing of the most current alarms that are still outstanding in the area. Its format is similar to that of an alarm log produced by a computer, and would include the following information on the points in alarm: tag number and description of point in alarm, time of alarm, types of alarm (e.g., deviation, high, and low), current value of point, and current alarm status (e.g., in alarm or not in alarm, returned to normal, acknowledged or not acknowledged).

Group-Level Displays. The displays at the and area levels of the hierarchy in Figure 6.13 are designed to provide the operator with information on the alarm and operational status of the key process variables in the plant. To perform control operations, however, it is necessary of use the displays at the next lower level in the hierarchy – the group level. As in the case of the higher-level displays, many of the display formats at the group level are patterned after the layout of panelboard instrumentation designed to accomplish similar functions.

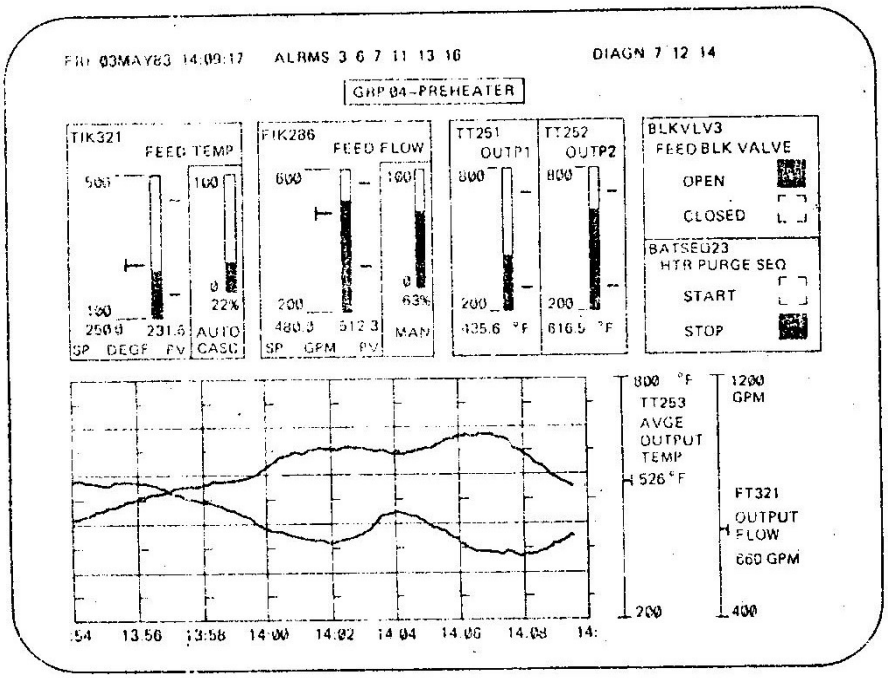


Figure shows one example of a typical group display. Mimics of manual and automatic stations for continuous control lops occupy the upper left-hand corner of the display. These mimics include all of the elements contained in a similar panelboard station: bar graphs showing values of set point, control output, and process variables: manual, automatic, and cascade mode indicators; high and low alarm levels; and other information as needed for the type of station implemented. (The next section will discuss the mechright section of Figure 6.16 shows indicator stations that let the operator view (but not control) selected process variables.

Also in this section is a logic operations such as opining and closing valves, as well as starting or stopping sequential control sequences (e.g., for batch processes). The bottom half of the display is devoted to plotting the trends of one or more process variables as a function of time, mimicking the operation of a trend recorder on a panelbord. In some operator interface system, each screen "page" of a group display can use only one type of station ro trend recorder: other provide much more flexibility by allowing the user to mix and match the types on each display.

The type of group display shown in Figure 6.16 can be viewed as the equivalent of a section of panelboard in conventional type of operator interface. Switching from one group display to another is the equivalent of having the operator move around a panelboard to accomplish the monitoring and control function. the CRT-based "panelboard" offers the user some significant benefits over the conventional panelboard, however. first stations and recorders can be added to or removed from the CRT "panelboard" by reconfiguring displays rather than cutting or patching real holes in a panel and procuring additional instrumentation hardware. This provides a significant flexibility advantage during initial plant startup. At which time the user often discovers that additional stations or recorders would be very helpful in plant operations. Another benefit is that one can duplicate stations and recorders in several displays without any additional hardware. This duplication capability can be significant aid to improving plant operations-one whose cost could not be justified in a conventional panelboard.

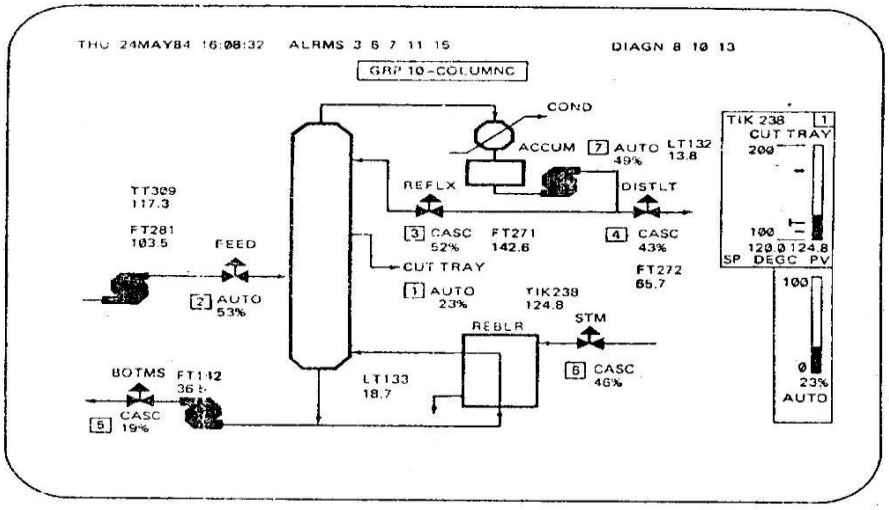


Figure 3. Example of a Control Graphics Display

Of course, the capabilities of a VDU permit the configuration of operator displays that go well beyond simple replacement of panelboard functions. One example of this is the graphic display for a piping-and-instrumentation diagram (P&ID), shown in Figure 6.17. This differs from the area level P&ID display described previously in two respects:

1. The scope of process covered is smaller- a group rather than an entire area.
2. Control capability is included in addition to the monitoring capability provided in area P&ID.

The controller station shown on the right side of the display allows the operator to perform control functions. The operator is able to select on e of the control loops shown on the graphic through one of several possible mechanisms described in the next section can be used to perform a sequence control or batch control operation using the graphic display.

It should be noted that in some systems, selected controller stations or logic stations on a particular console can be designated as monitor only the operator cannot perform any control actions but can only monitor the station variables on the display. This capability is useful in ensuring that operator control actions are coordinated when the consoles are physically distributed in several locations in the plant.

Two other types of displays also have proved useful at the group level:

1. Batch control displays-These are menu-oriented displays that allow an operator to observe the progression of a batch recipe (such as that shown in Figure In Chapter) and interact with the sequence; start it, stop it, provide permissive to allow it to continue, and so forth. This class of displays also allows the operator to diagnose problems in executing a sequence, such as identifying the part of the process that is preventing the sequence from continuing.
2. Operator guides-These are advisory displays that provide the following kinds of information to the operator: problems diagnosed by the automatic system, suggested alternative courses of action in an emergency, or step-by-step startup and shutdown procedures for a piece of plant equipment. These displays may combine alphanumeric and graphic information. One can think of them as CRT-based substitutes for a set of plant operating manuals. They differ from manuals in that they can take current plant conditions into account as well as simply provide standard operating procedures to the operator.

Loop-Level Displays. The displays at the group level are the operator's primary working displays. The operator uses a few types of displays dealing with single loops or data points for control and analysis purposes. Figure shows one example, the X-Y operating display. Here one process variable is plotted as a function of another to show the current operating point of this pair of variables.

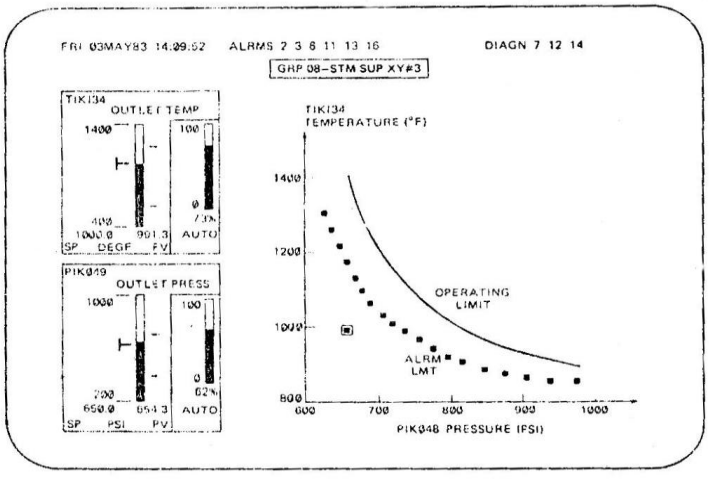


Figure : Example of an X-Y Operating Display

The operator then can compare this operating point against an alarm limit curve or an operating limit curve. In the example shown in the figure, the combination of temperature and pressure for a particular portion of the process may be critical to safety. Therefore, this pair of variables is made

available to the operator in the X-Y formal shown. If this pair can be controlled directly, manual/automatic stations also can be included in the display for direct operator manipulation. This approach to control and display is not possible using standard panelboard instrumentation. The CRT format makes it feasible and cost-effective. The above figure shows and example of a tuning display another single-loop display that is of use to both operating personnel and instrumentation engineers. This display includes several elements that make the tuning function possible:

1. A “feast” trend-plotting capability:
2. A manual/automatic station to allow the operator to control the loop:
3. A list of the tuning parameters (e.g., proportional band, reset rate, and derivative rate) with the loop.

The trend graph is used to plot set-point changes (in automatic mode), manual control output changes (in manual mode), and the resulting responses of the process variable being controlled, based on these responses, the operator or instrumentation engineer can make on-line adjustments to the tuning parameters to improve the performance of the control loop. This example of integrating control, trending, and tuning functions is one illustration of the ability of the CRT-based operator interface to provide the operator with a very usable and convenient tool for plant operation.

Function Block. The FOUNDATION Fieldbus Function Block, especially its models and parameters–through which you can configure, maintain, and customize your applications–is a key concept of Fieldbus technology.

What is a Function Block? A Function Block is a generalized concept of the functionality in field instruments and control systems, such as analog input and output as well as PID (Proportional-Integral-Derivative) control. The FOUNDATION Fieldbus specification, FF-890, "Function Block Application Process–Part 1," gives fundamental concepts, while Part 2 and later parts give various Function Block details.

The Function Block Virtual Field Device (VFD) contains three classes of blocks: Resource Block, Function Block, and Transducer Block.

Resource Block. A Resource Block shows what is in the VFD by providing the manufacturer's name, device name, Device Description (DD), and so on.

The Resource Block controls the overall device hardware and Function Blocks within the VFD, including hardware status.

Tip 1 – The mode of the Resource Block controls the mode of all other blocks in the device.

Transducer Block. A Function Block is a general idea while the Transducer Block is dependent on its hardware and principles of measurement. For example, a pressure transmitter and magnetic flow meter use different measurement principles but provide an analog measured value. The common part is modeled as an AI (Analog Input) Block. The difference is modeled as Transducer Blocks, which provide the information on the measurement principle. A Transducer Block is linked to a Function Block through the CHANNEL parameter of the Function Block.

In addition to converting the signal between a digital number and a physical signal (milliVolts, capacitance, frequency etc.) or output (pressure, current, etc.), Transducer Blocks are becoming ever more important because they are also the blocks used to capture and store all the diagnostic and maintenance-related data for a device. A number of Standard Transducer

Function Block. The FOUNDATION Fieldbus Function Block, especially its models and parameters–through which you can configure, maintain, and customize your applications–is a key concept of Fieldbus technology.

What is a Function Block? A Function Block is a generalized concept of the functionality in field instruments and control systems, such as analog input and output as well as PID (Proportional-Integral-Derivative) control. The FOUNDATION Fieldbus specification, FF-890, "Function Block Application Process–Part 1," gives fundamental concepts, while Part 2 and later parts give various Function Block details.

The Function Block Virtual Field Device (VFD) contains three classes of blocks: Resource Block, Function Block, and Transducer Block.

Resource Block. A Resource Block shows what is in the VFD by providing the manufacturer's name, device name, Device Description (DD), and so on.

The Resource Block controls the overall device hardware and Function Blocks within the VFD, including hardware status.

Tip 1 – The mode of the Resource Block controls the mode of all other blocks in the device.

Transducer Block. A Function Block is a general idea while the Transducer Block is dependent on its hardware and principles of measurement. For example, a pressure transmitter and magnetic flow meter use different measurement principles but provide an analog measured value. The common part is modeled as an AI (Analog Input) Block. The difference is modeled as Transducer Blocks, which provide the information on the measurement principle. A Transducer Block is linked to a Function Block through the CHANNEL parameter of the Function Block.

In addition to converting the signal between a digital number and a physical signal (milliVolts, capacitance, frequency etc.) or output (pressure, current, etc.), Transducer Blocks are becoming ever more important because they are also the blocks used to capture and store all the diagnostic and maintenance-related data for a device. A number of Standard Transducer

The main components of the Application Layer are the Fieldbus Access Sublayer (FAS) and the Fieldbus Message Specification (FMS).

The FAS uses the scheduled and unscheduled features of the Data Link Layer to provide a service for the Fieldbus Message Specification (FMS). The types of FAS services are described by Virtual Communication Relationships (VCR).

The VCR is like the speed dial feature on your memory telephone. There are many digits to dial for an international call–an international access code, country code, city code, exchange code, and the specific telephone number. This information only needs to be entered once and then a "speed dial number" is assigned. After setup, only the speed dial number needs to be entered for dialing to occur.

In a similar fashion, after configuration, only the VCR number is needed to communicate with another Fieldbus device.

Just as there are different types of telephone calls, such as person-to-person, collect, or conference calls, there are different types of VCRs. VCRs and their management are covered in more detail in Chapter 5.

Fieldbus Message Specification (FMS) services allow user applications to send messages to each other across the Fieldbus using a standard set of message formats.

FMS describes the communication services, message formats, and protocol behavior needed to build messages for the User Application.

Data that is communicated over the Fieldbus is described by an "object description." Object descriptions are collected together in a structure called an object dictionary (OD).

The object description is identified by its index in the OD. Index 0, called the object dictionary header, provides a description of the dictionary itself and defines the first index for the object descriptions of the User Application. The User Application object descriptions can start at any index above 255.