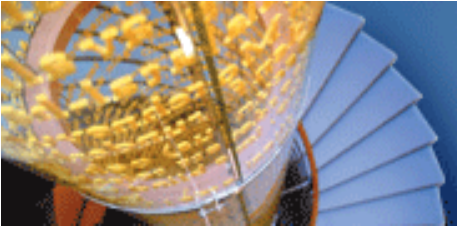


HALMSTAD UNIVERSITY

SCADA systems
Lecture 1

Urban Bilstrup

E 327

Urban.Bilstrup@hh.se

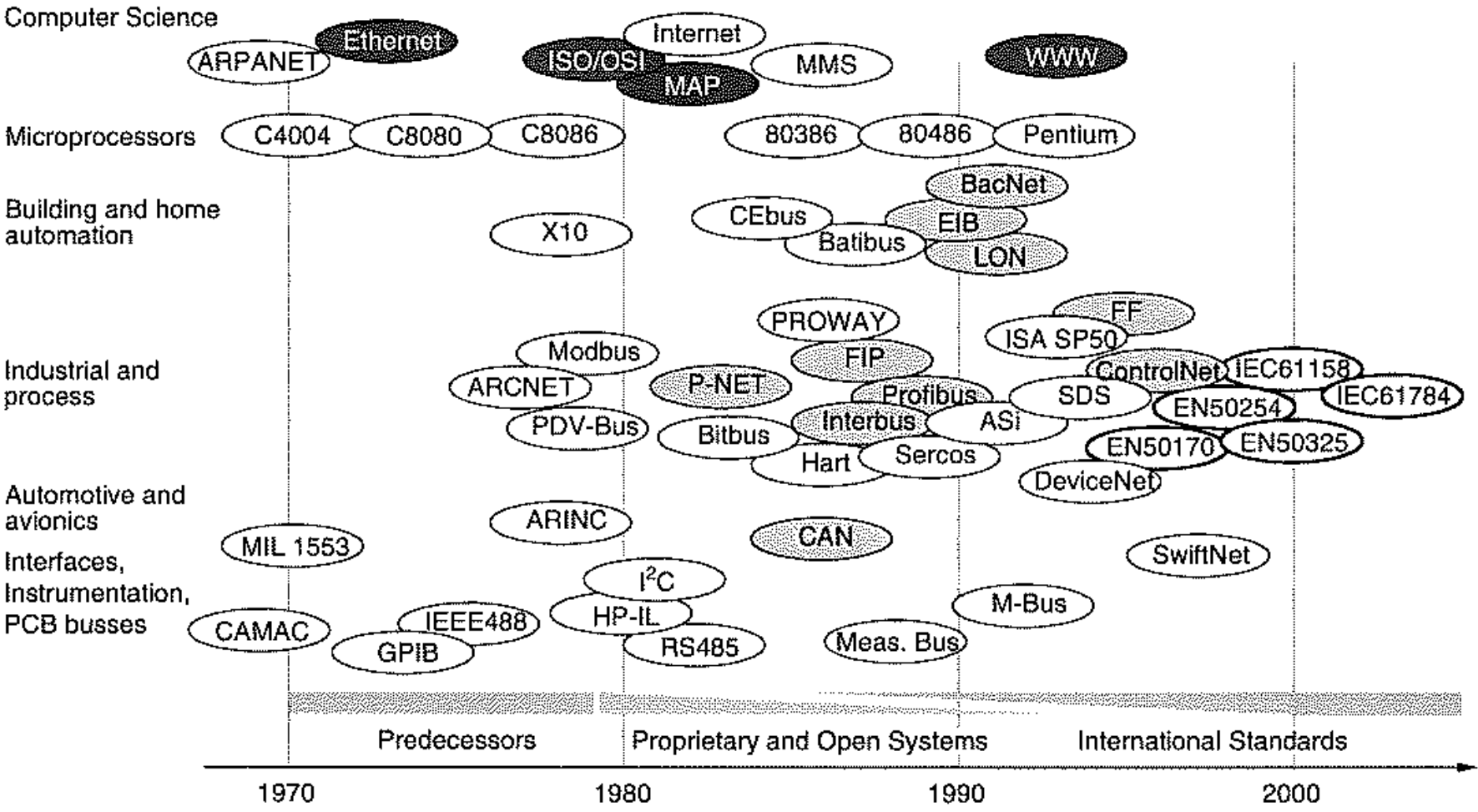
Introduction – Outline

- Motivation and historical perspective
- Interfaces that is replaced with SCADA (industrial networks)
- Applications
- Application requirements
- Industrial network architecture
- ISO/OSI model
- User layer
- Communication paradigm
- Sensor/Actuators network
- Fieldbus networks
- Control networks

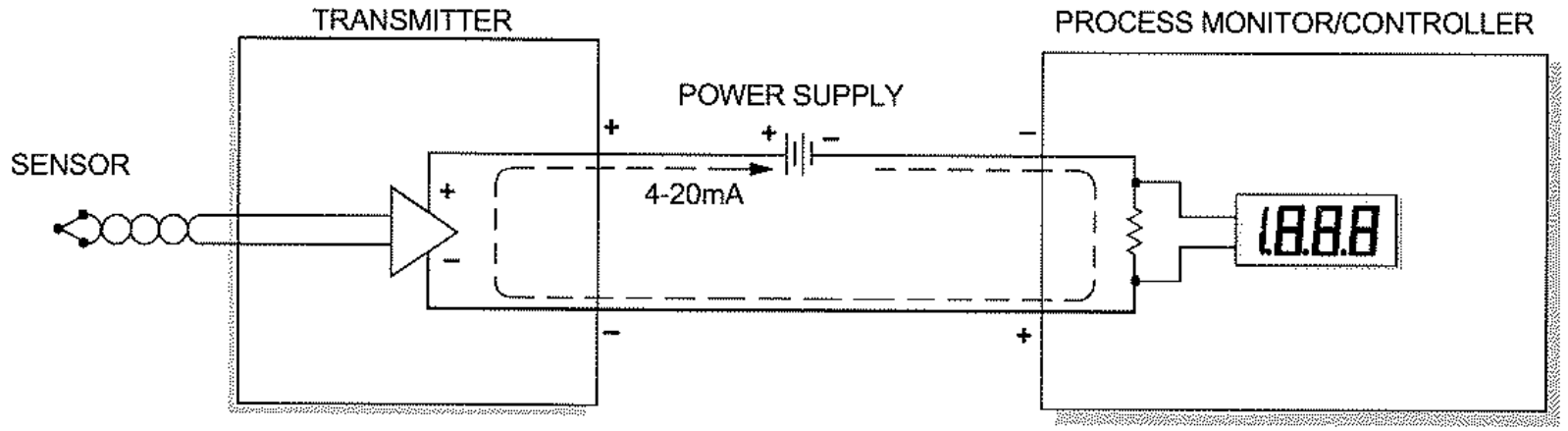
Motivation

- Save money
- Increase productivity
- Offering more services
- Test integration increase reliability
- Allows for more intelligent systems
- Flexibility
- Configurability
- Maintainability
- Distribution

Evolution history



History – Analog 4-20 mA current loop



The 4-20mA current loop is common in many industrial process-monitoring applications.

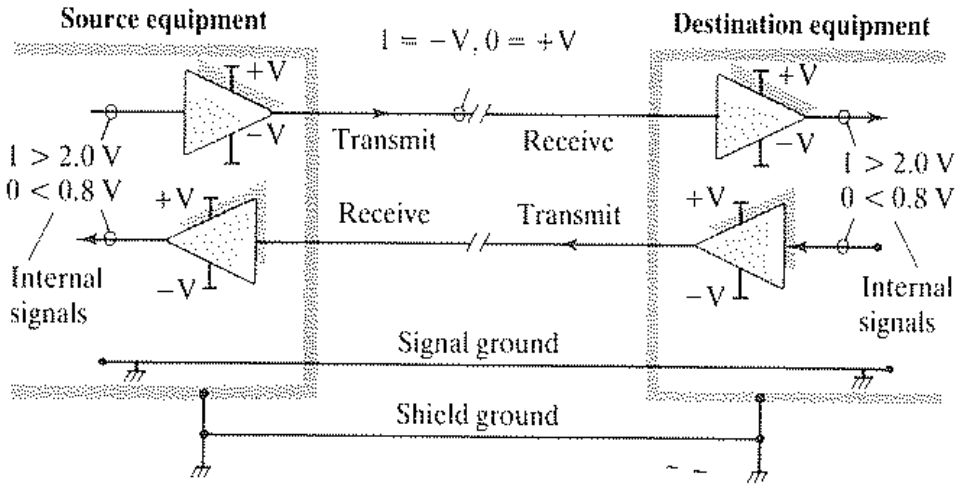
The loop's operation is straightforward: a sensor's output voltage is first converted to a proportional current, with 4mA normally representing the sensor's zero-level output, and 20mA representing the sensor's full-scale output.

The receiver at the remote end converts the 4-20mA current back into a voltage which in turn can be further processed by a computer or display module.

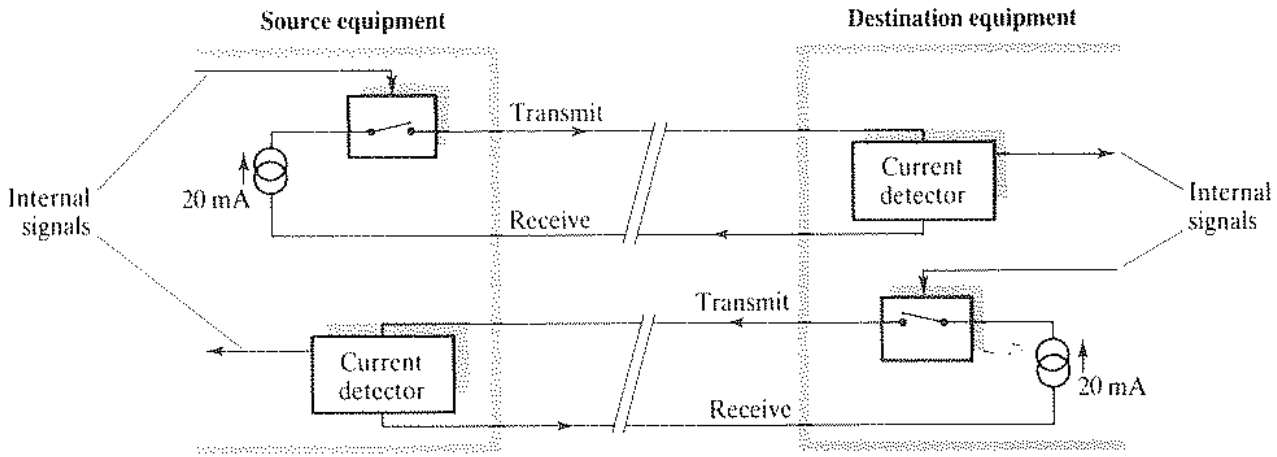
History – EIA 232

The EIA-232 standard defines the voltage levels that correspond to logical one and logical zero levels. Valid signals are plus or minus 3 to 15 volts. Point to point.

The standard specifies a maximum open-circuit voltage of 25 volts; signal levels of ±5V,±10V,±12V, and ±15V are all commonly seen depending on the power supplies available within a device.



History – Digital 20mA current loop

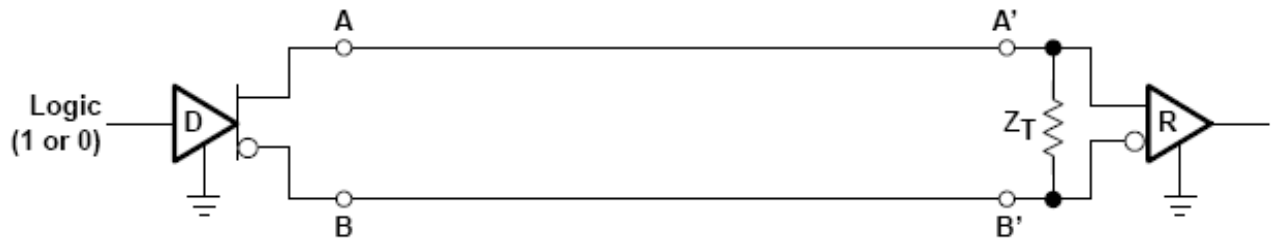


An alternative to the EIA 232 is the digital 20mA current loop. Essentially, the state of a switch is controlled by the bit stream to be transmitted; the switch is closed for a binary 1, thus passing a current pulse of 20mA and opened for a binary 0, thus stopping the current flow.

History – EIA 422/423

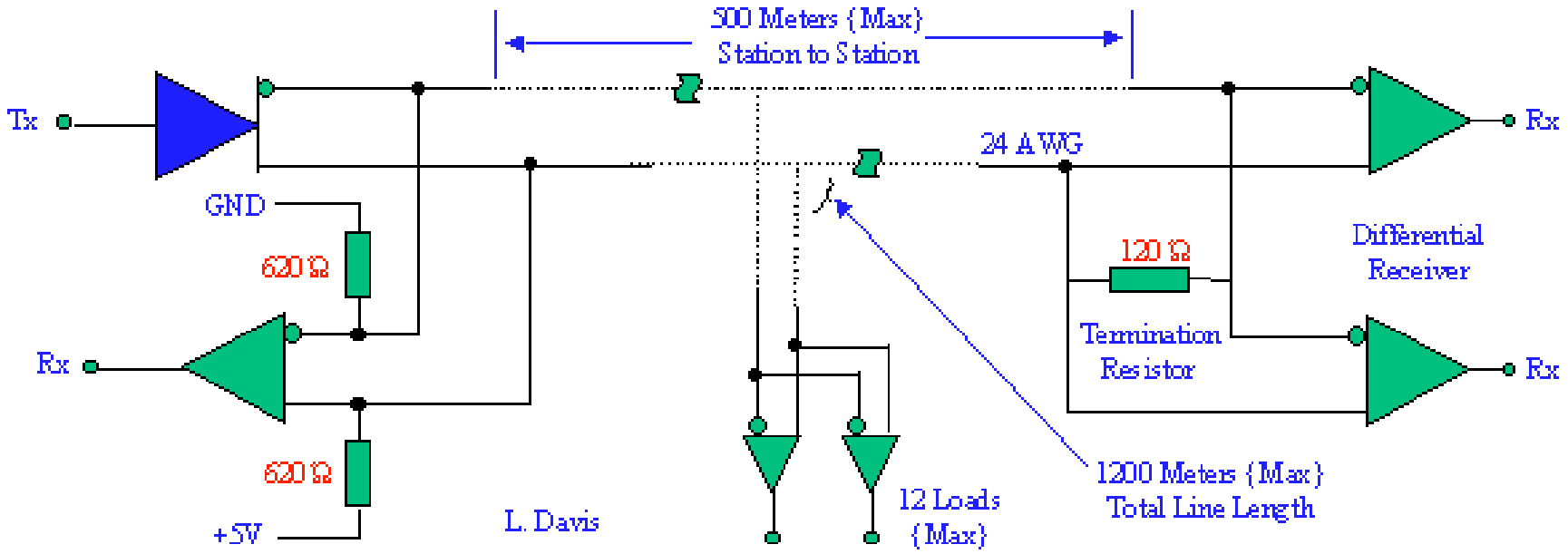
EIA-422 (formerly **RS-422**), now **TIA-422**, is a technical standard which specifies the "electrical characteristics of the balanced voltage digital interface circuit".

It provides for data transmission, using balanced or differential signaling, with unidirectional/non-reversible, terminated or non-terminated transmission lines, point to point, or multi-drop only multiple receivers.

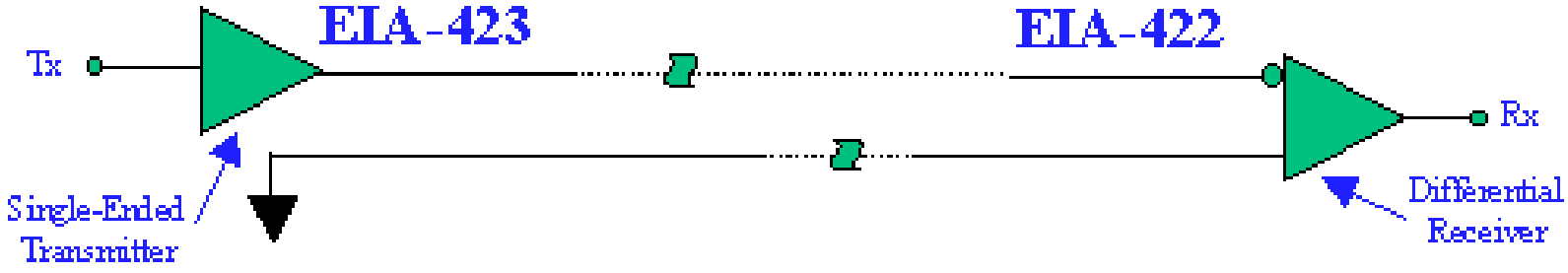


A. D = driver (or generator)
B. R = receiver
C. Z_T = termination impedance

History – EIA 422/423

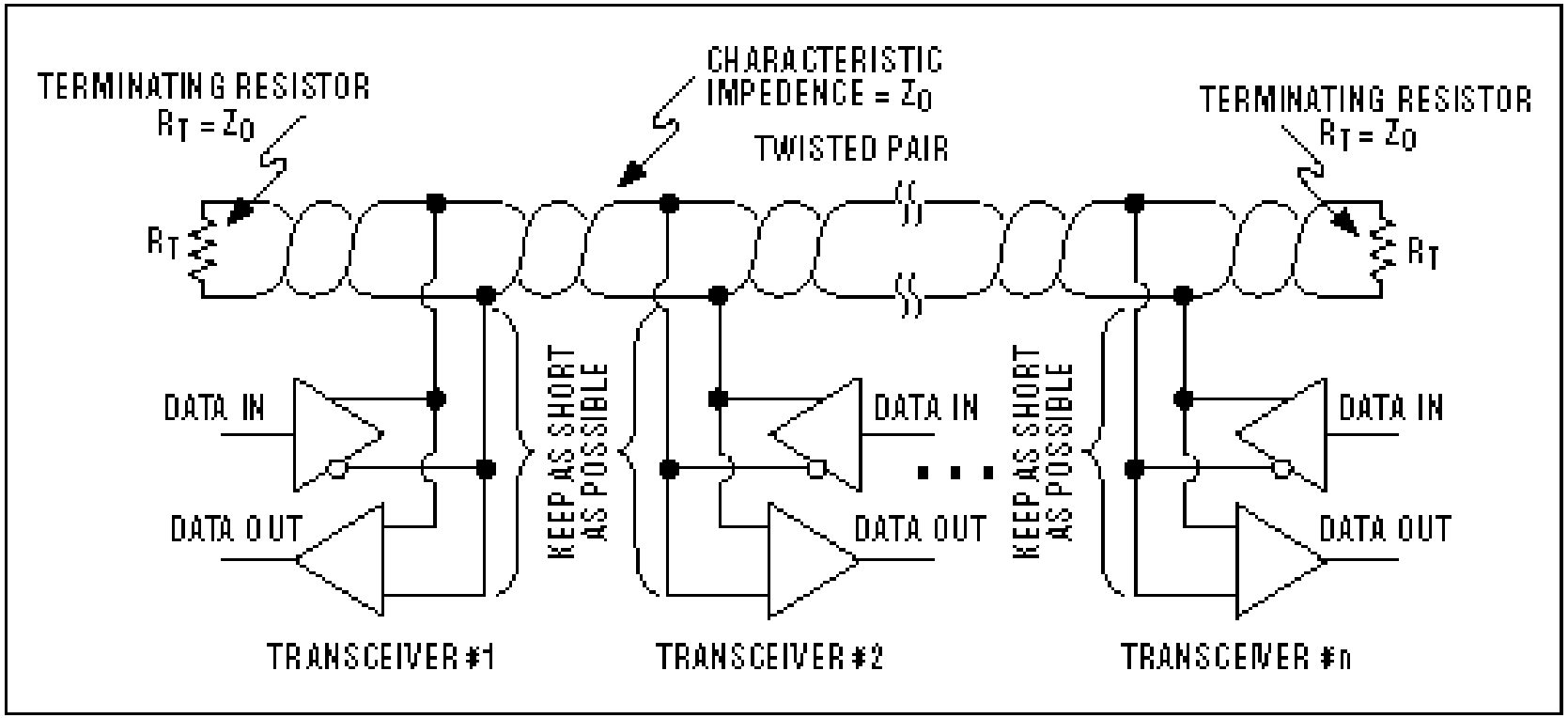


History – EIA 422/423



History – EIA 485

EIA-485 (formerly **RS-485** or **RS485**) is an OSI Model physical layer electrical specification of a two-wire, half-duplex, multipoint serial connection.



History – EIA 485

The standard specifies a differential form of signaling. The difference between the wires' voltages is what conveys the data.

One polarity of voltage indicates a logic 1 level, the reverse polarity indicates logic 0.

The difference of potential must be at least 0.2 volts for valid operation, but any applied voltages between +12 V and -7 volts will allow correct operation of the receiver.

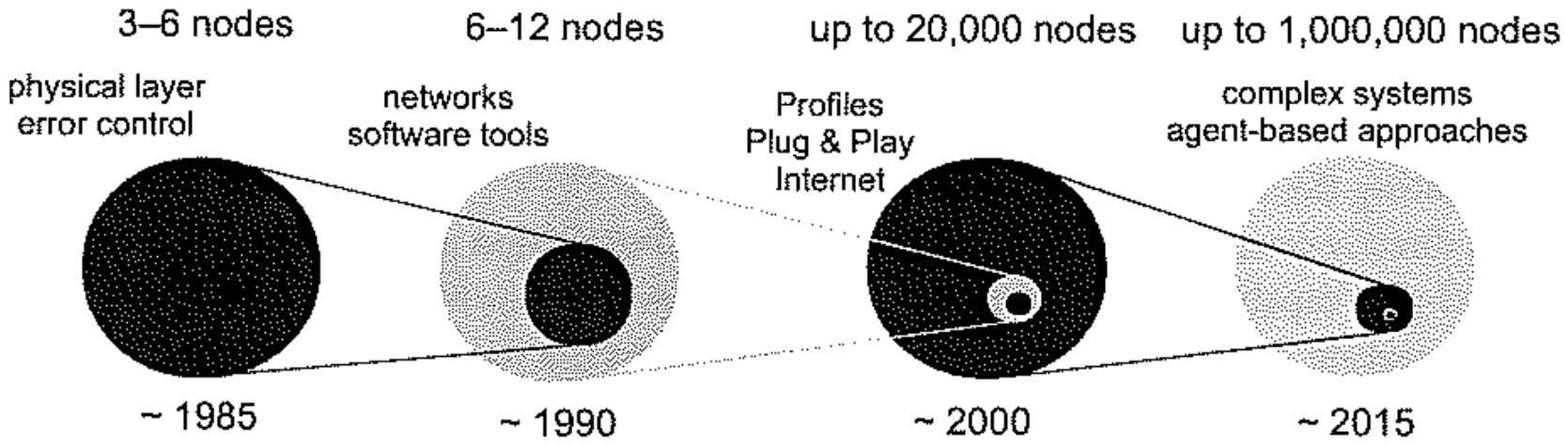
History – EIA 485

EIA-485 only specifies electrical characteristics of the driver and the receiver.

Since it uses a differential balanced line over twisted pair (like EIA-422), it can span relatively large distances (up to 4000 feet or just over 1200 meters).

In contrast to EIA-422, which has a single driver circuit which cannot be switched off, EIA-485 drives need to be put in transmit mode explicitly by asserting a signal to the driver. This allows EIA-485 to implement linear topologies using only two lines.

History – Complexity growth

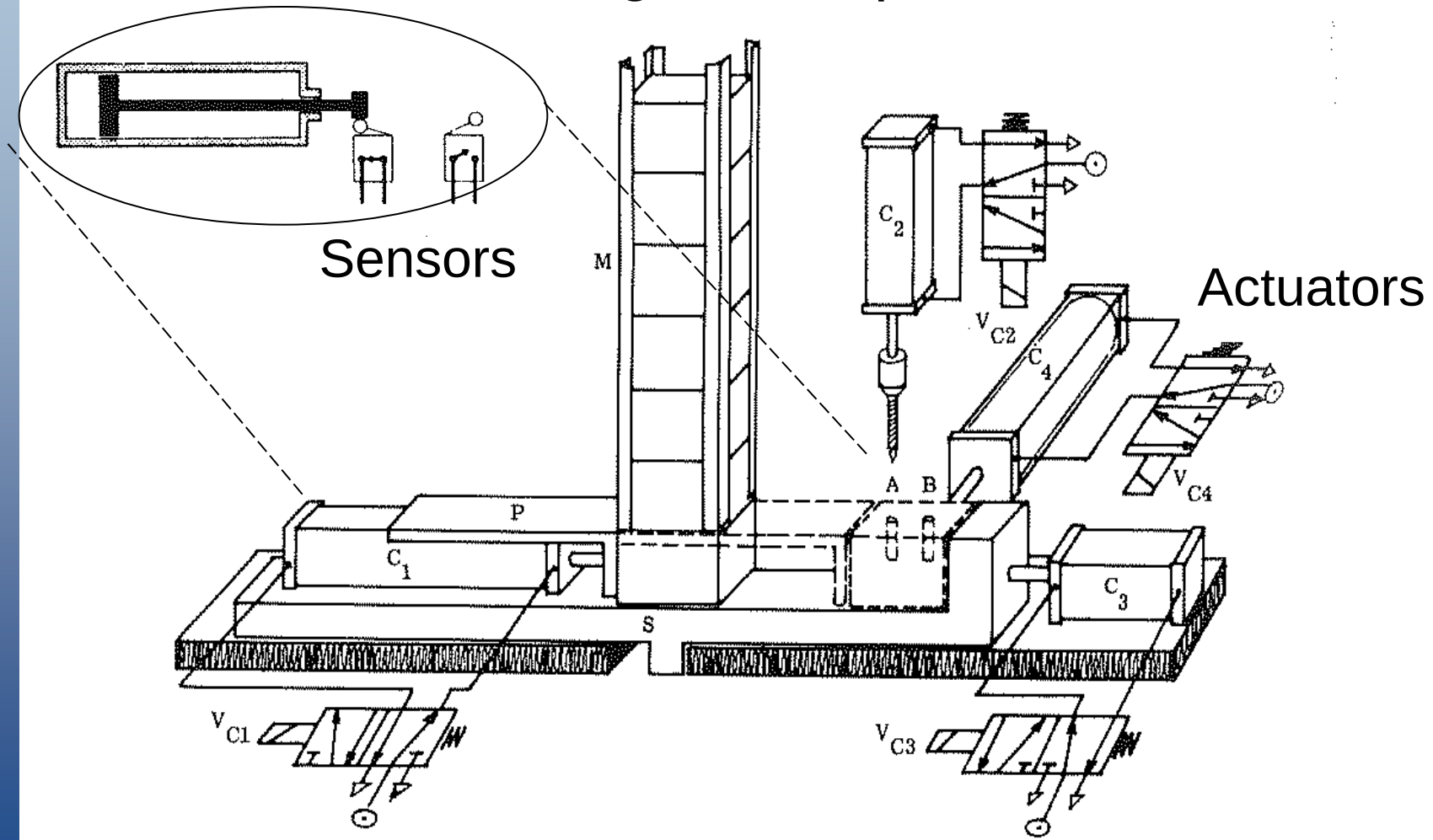


Applications

Logic Control

Applications

Controling a drill operation



Applications

Programmable logic controller (PLC)

A computer with special input/output arrangements, perform sequence of operations (state machines, ladder logic), traditionally intended for logic control.

A set of I/O ports connect the PLC to sensors and actuators.

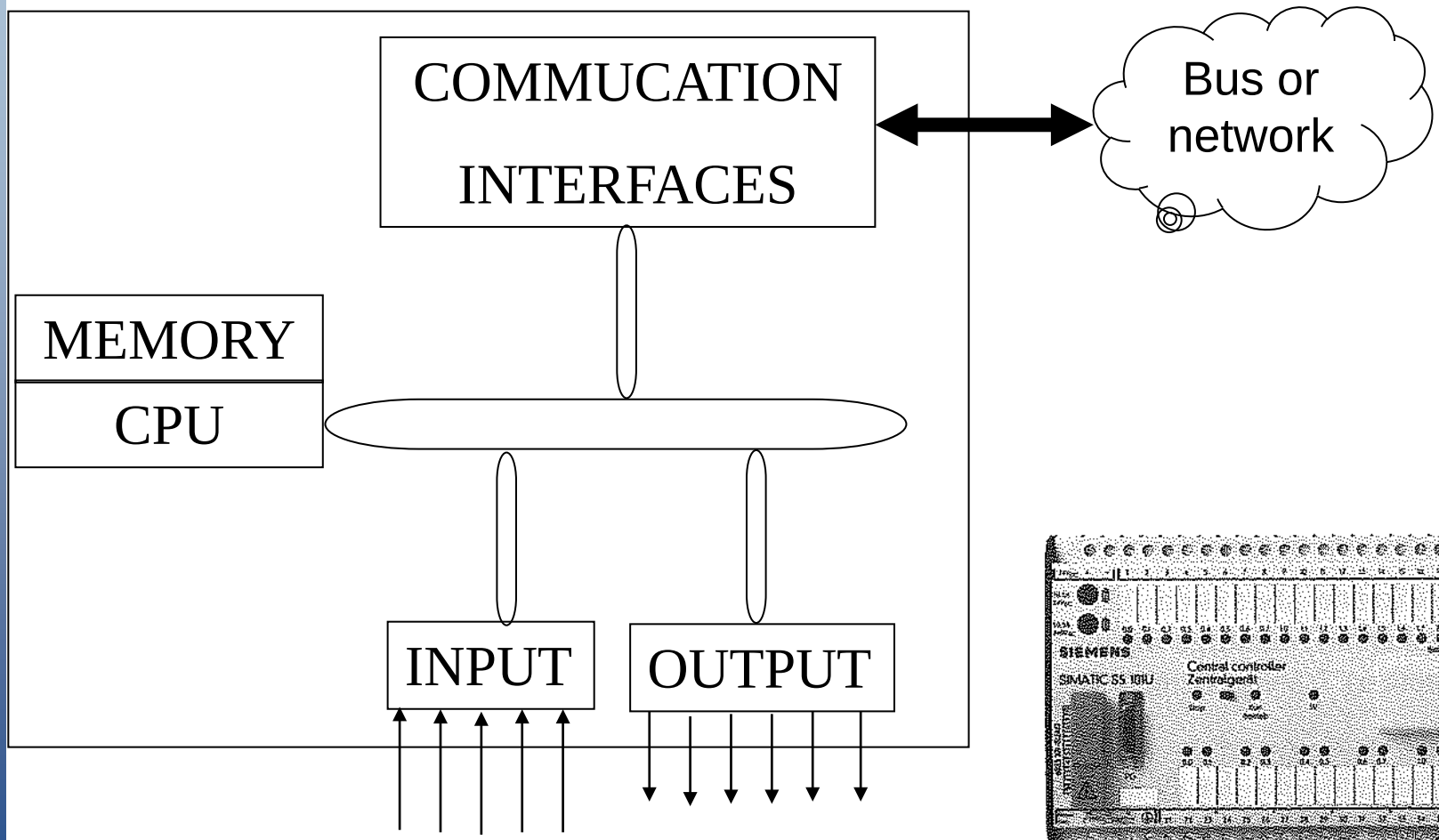
As inputs the PLC read limit switches, temperature indicators and the positions of complex positioning systems.

On the output side the PLC operates electric motors, pneumatic or hydraulic cylinders or diaphragms, magnetic relays or solenoids.

The PLC may have external I/O modules attached to a proprietary computer network that plugs into the PLC.

Applications

Typical PLC architecture

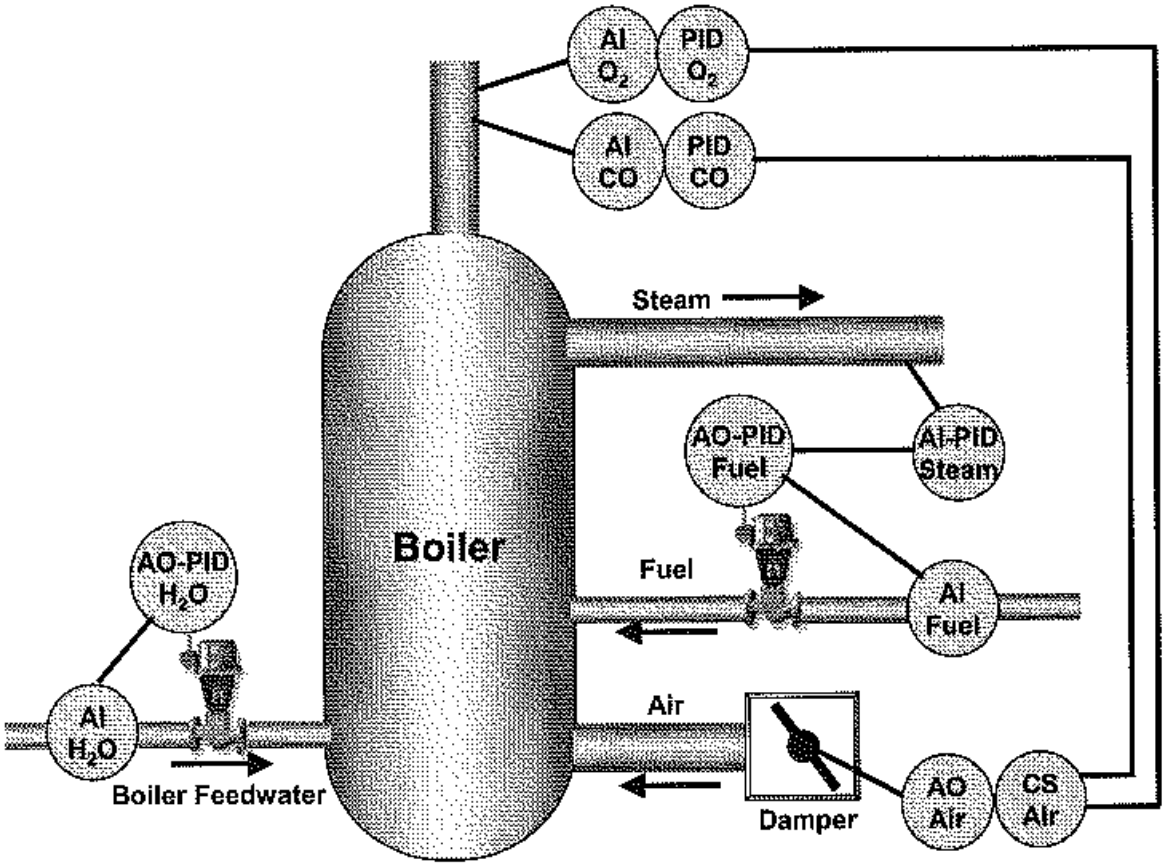


Applications

Process Control

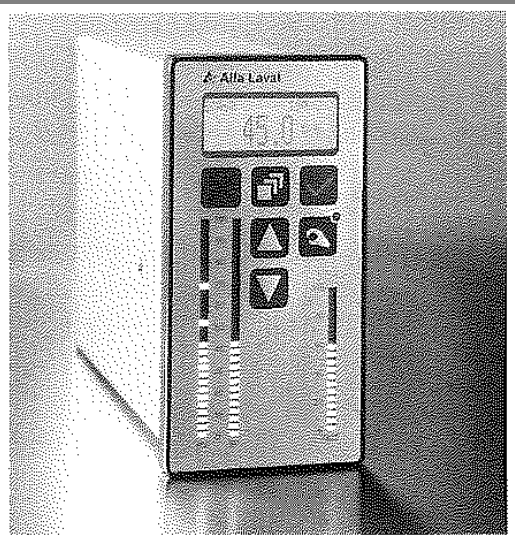
Applications

Feedback Process Control



Applications

Applications



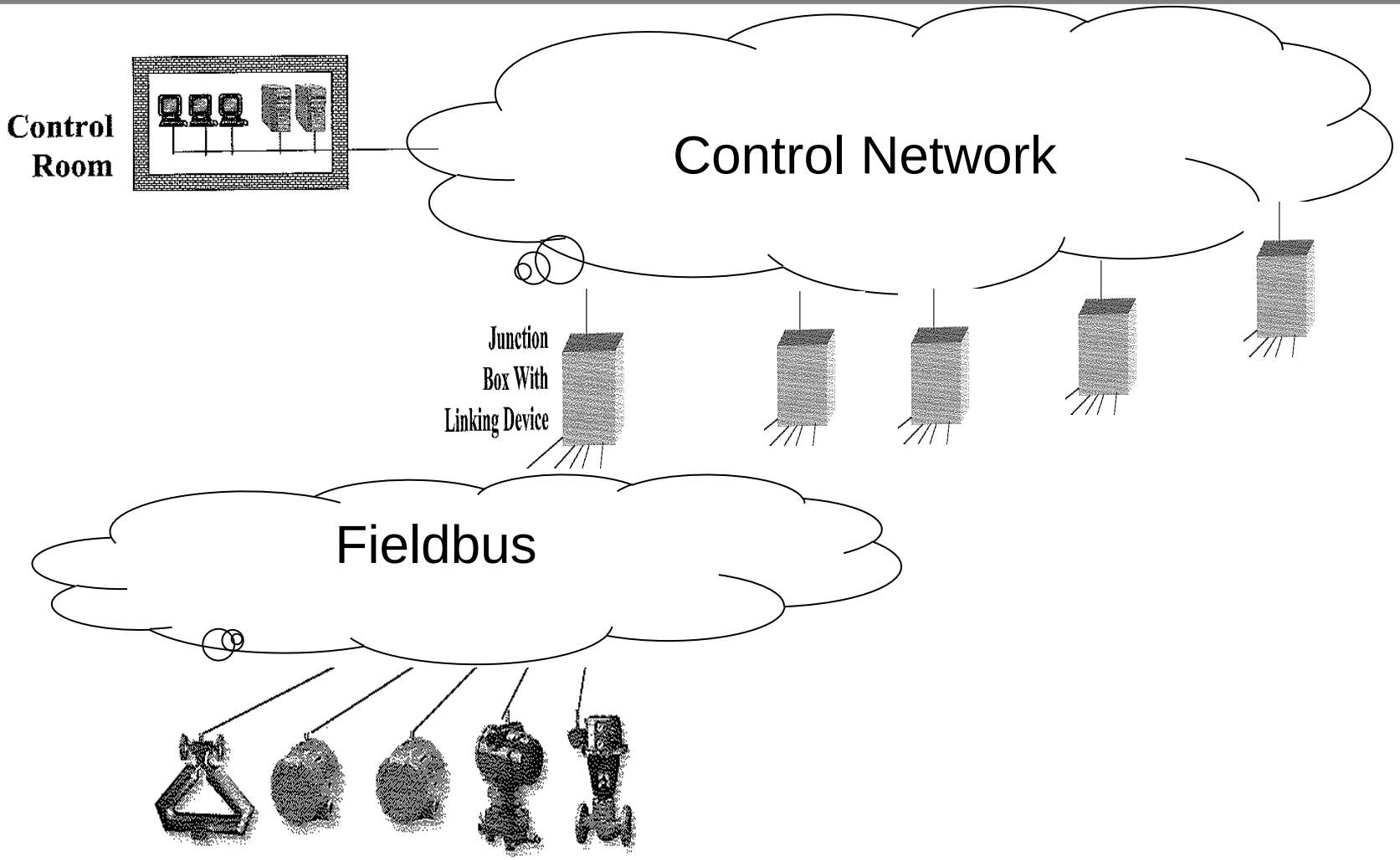
Single-loop
PID controller

A PID controller can described as a set of rules with which precise regulation of a closed-loop control system is obtained.

Closed-loop control means a method in which a real-time measurement of the process being controlled is constantly fed back to the controlling device to ensure that the value which is desired is, in fact, being realized.

The mission of the controlling device is to make the measured value, usually known as the PROCESS VARIABLE, equal to the desired value, usually known as the SETPOINT.

Applications



Applications

Numerical Control

Applications

A CNC milling machine

Computer Numerical Control (CNC), refers specifically to a computer "controller" that reads **G-code instructions** and drives the machine tool, a powered mechanical device typically used to fabricate metal components by the selective removal of metal.

CNC does numerically directed interpolation of a cutting tool in the work envelope of a machine.

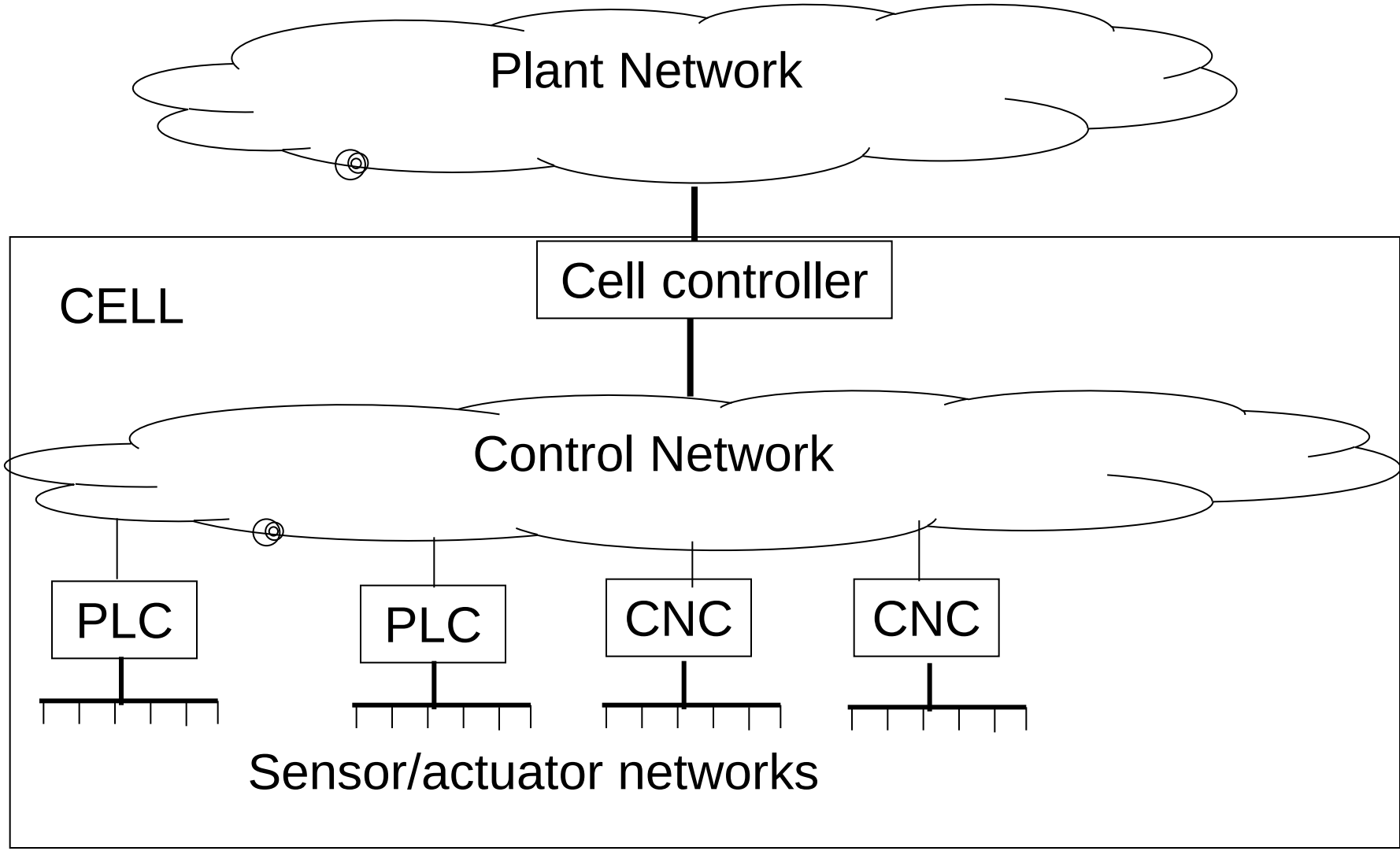
Applications

In a production environment, a series of CNC machines may be combined into one station, commonly called a "cell", to progressively machine a part requiring several operations.

CNC machines today are controlled directly from files created by CAD software packages. A part or assembly can go directly from design team to manufacturing plant.

CNC machines can run over night and over weekends without operator intervention. Error detection features have been developed, giving CNC machines the ability to call the operator's mobile phone if it detects that a tool has broken.

Applications



Applications

Industrial Robots

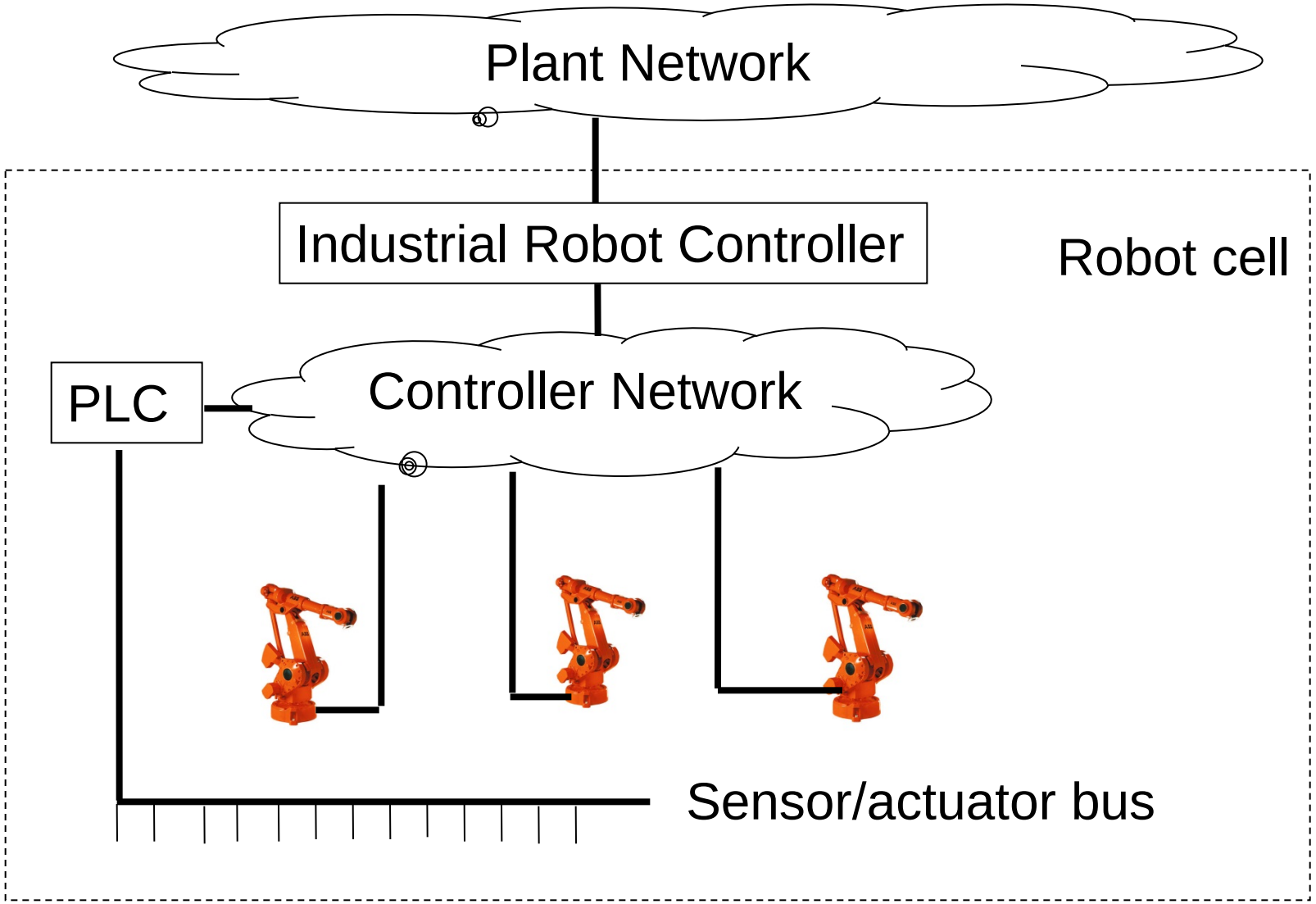
Applications

An **industrial robot** is officially defined as an *automatically controlled, reprogrammable, multi-purpose manipulator programmable in three or more axes*.

Typical applications of industrial robots include welding, painting, ironing, assembly, pick and place, palletizing, product inspection, and testing, all accomplished with high endurance, speed, and precision.

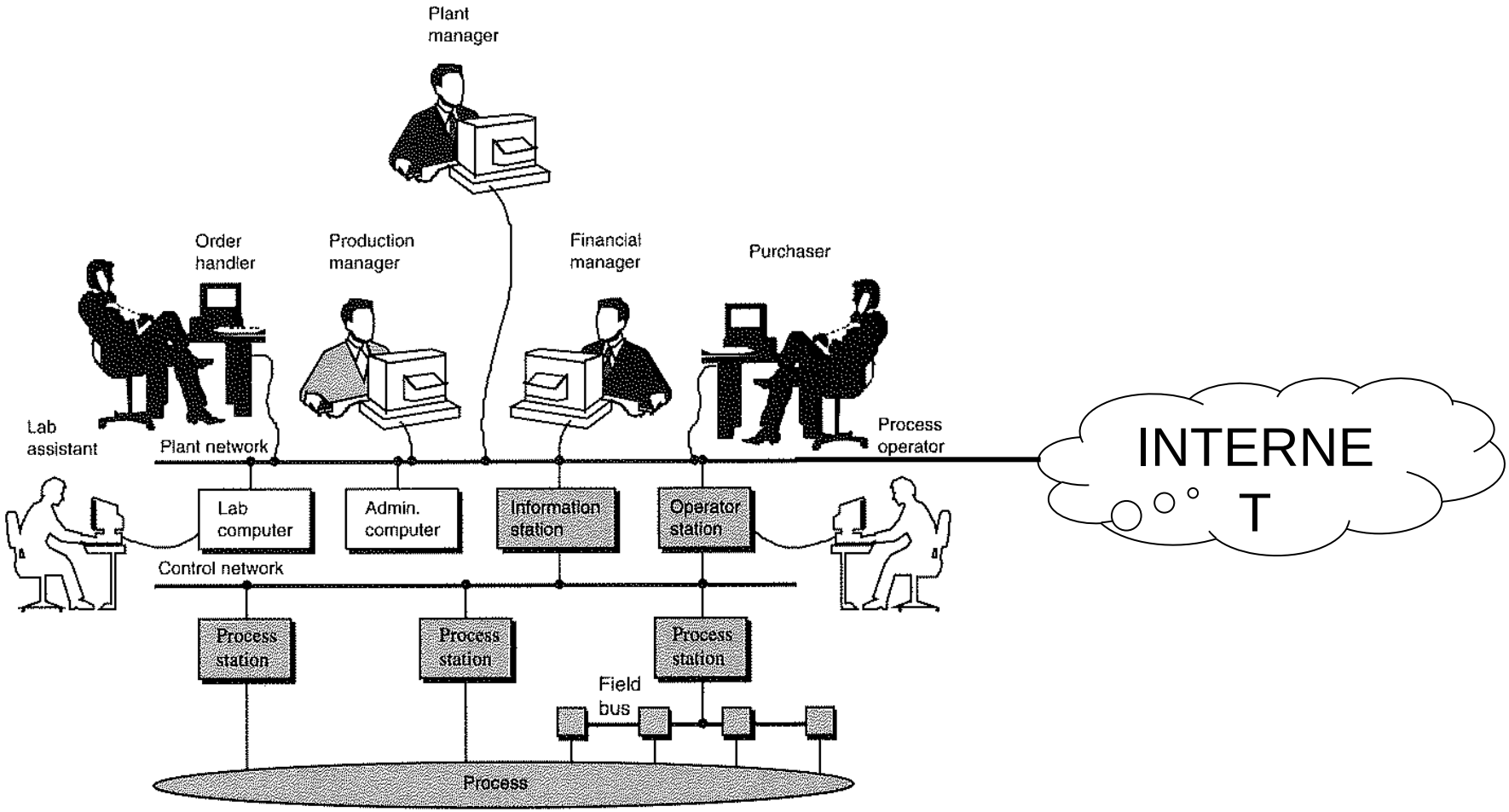
Machine vision systems, safety interlock systems, bar code printers and an almost infinite array of other industrial devices are accessed and controlled via the operator control panel.

Applications



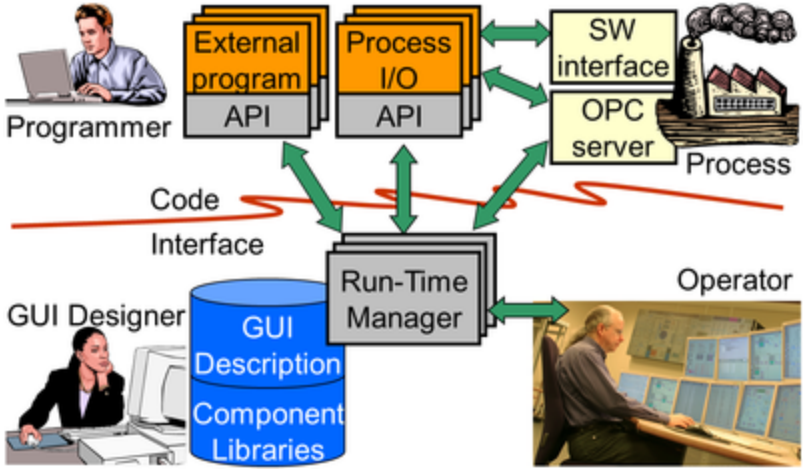
Applications

- Computer Integrated Manufacturing (CIM)



Applications

"ProcSee* has been used for simulators and process monitoring & control applications in various industries, including nuclear power plants, oil production platforms, electric power production and distribution, telecommunication networks, ship bridge systems, ship engine systems, paper mills, and environmental monitoring systems."



*http://www.ife.no/departments/visual_interface_technologies/products/procsee

Application requirements

- The requirements of an industrial communication system is directly deduced from the end user, the ***physical process*** that is controlled.
- This is actually what differentiates the requirements of an industrial network from a general computer network, a physical process obey under ***strict physical laws***.
- These introduce tight ***timing constraints***, i.e., the data has a temporal validity often referred to as ***critical*** data.
- There also exists data without timing constraints, these are often refered to as ***non critical***.

Application requirements

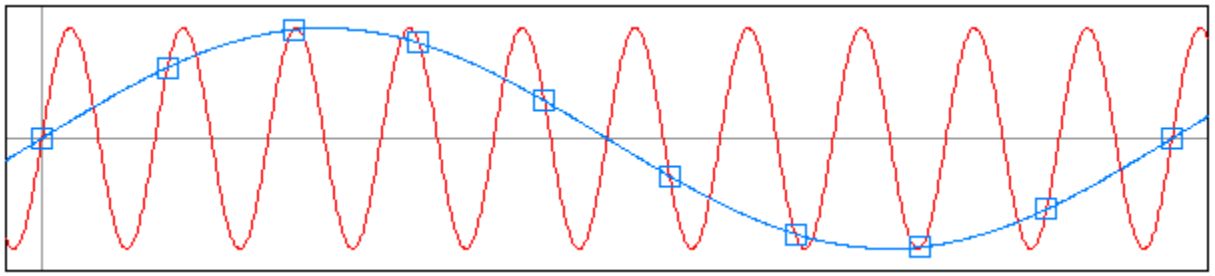
- Another time related aspect that concerns the data transferred over a communication network is the ***regularity and frequency***.
- Typically, there exists both ***periodic*** and ***random*** generation of data.
- When data transferred with a ***constant frequency*** it is called ***periodic*** otherwise the data is ***aperiodic***.
- In some cases it is possible to bound the ***minimum separation time*** between two consecutive aperiodic transmissions of the same source, this is known as ***sporadic***.

Application requirements

- Two approaches is present to design manufacturing applications ***event triggered*** and ***time triggered***.
- In event triggered systems the occurrence of an event initiate the application activity.
- A time triggered system is driven by periodic observations.
- The controlled physical processes often obey under the ***Nyquist sampling theorem***, the time triggered is well adapted for design of these applications.
- Aperiodic traffic is then handled by so called ***periodic servers***.

Application requirements

Nyquist sampling theorem states that to avoid **aliasing** make sure that the signal does not contain any sinusoidal component with a frequency equal to or greater than *fs* / 2.



Application requirements

- In a ***synchronous system*** operations are performed in rounds.
- At the beginning of each round, each processor sends messages and waits to receive messages.
- Upon receiving a these messages, the processor performs some operation and then decides what message to send in next round.
- In asynchronous system messages incur in an unbounded (but finite delay), and processors take steps at arbitrary rates.

Application requirements

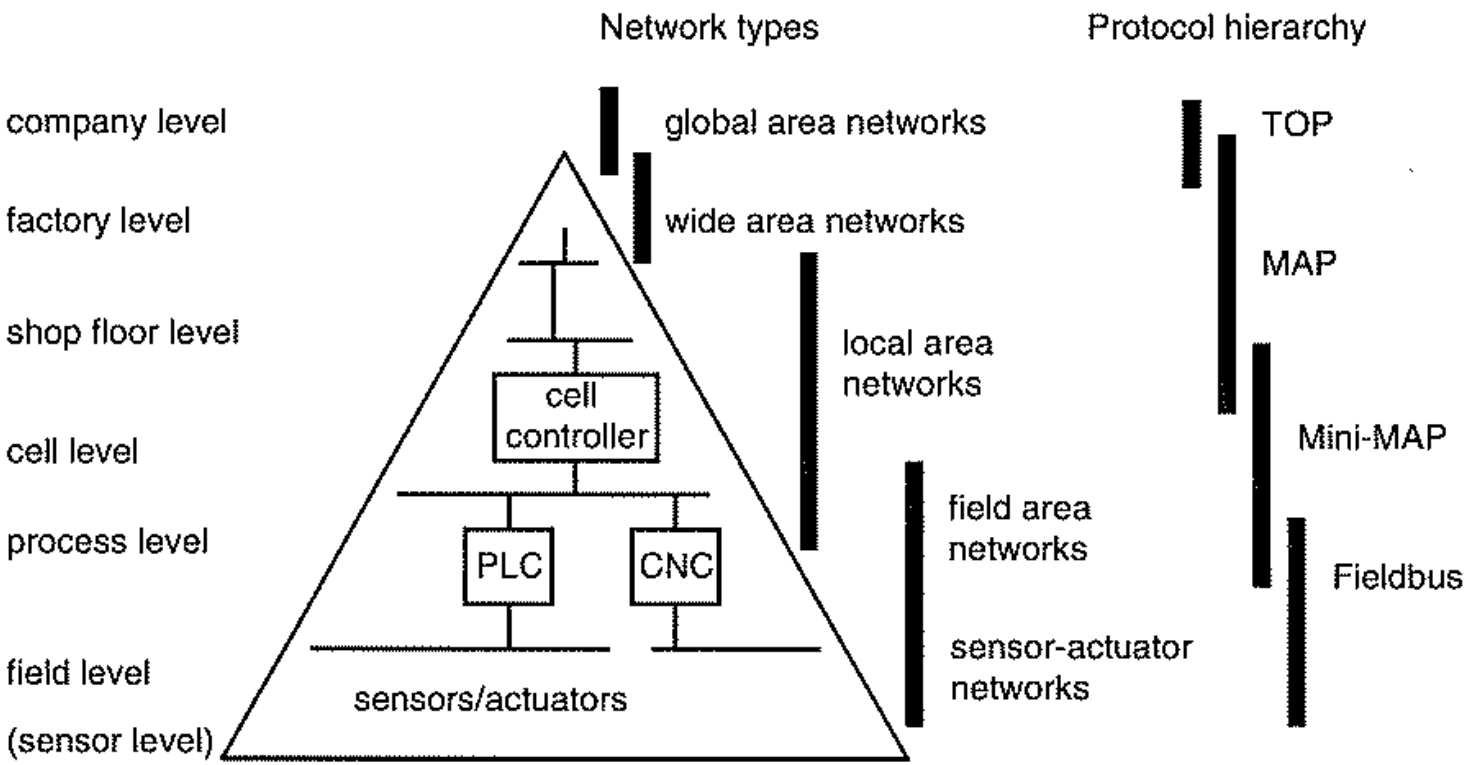
- In synchronous system there exist a ***fixed upper bound*** Δ, on the time needed to deliver a message from one processor to another.
- It also exist a ***fixed upper bound*** Θ on the relative speeds of different processors.
- In an asynchronous system no fixed bounds Δ and Θ exists.

Application requirements

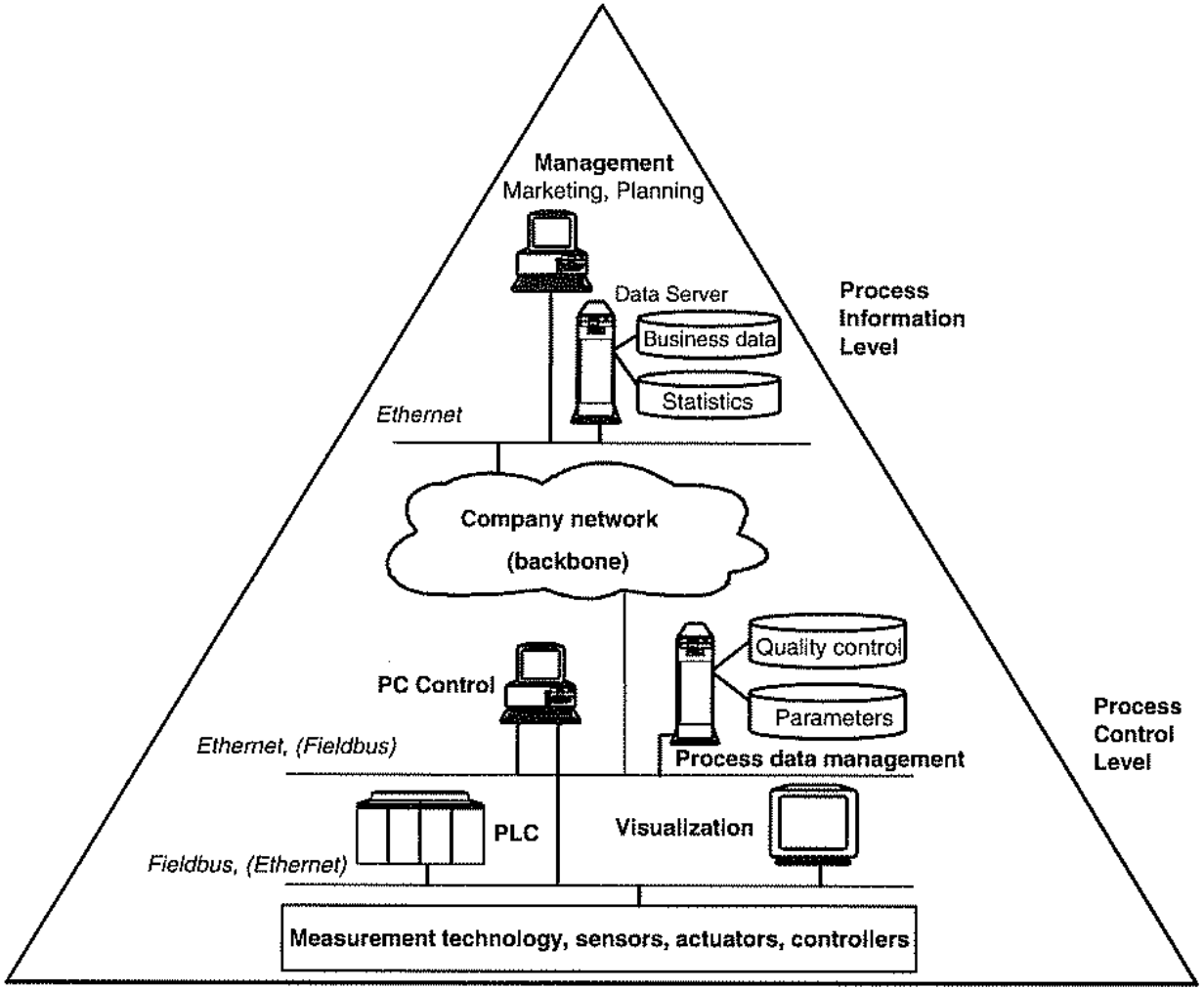
- Determinism is based on the capability to predict something in the future from the knowledge of the past and of the present.
- Determinism does not exist in general unless a hypotheses is explicitly stated, particularly in terms of reliability, bounded delays and absence of errors etc.

Industrial network architecture

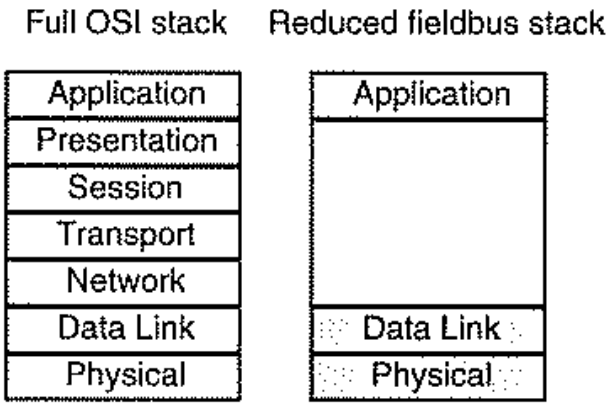
Automation pyramid (CIM pyramid)



Industrial network architecture



ISO/OSI model



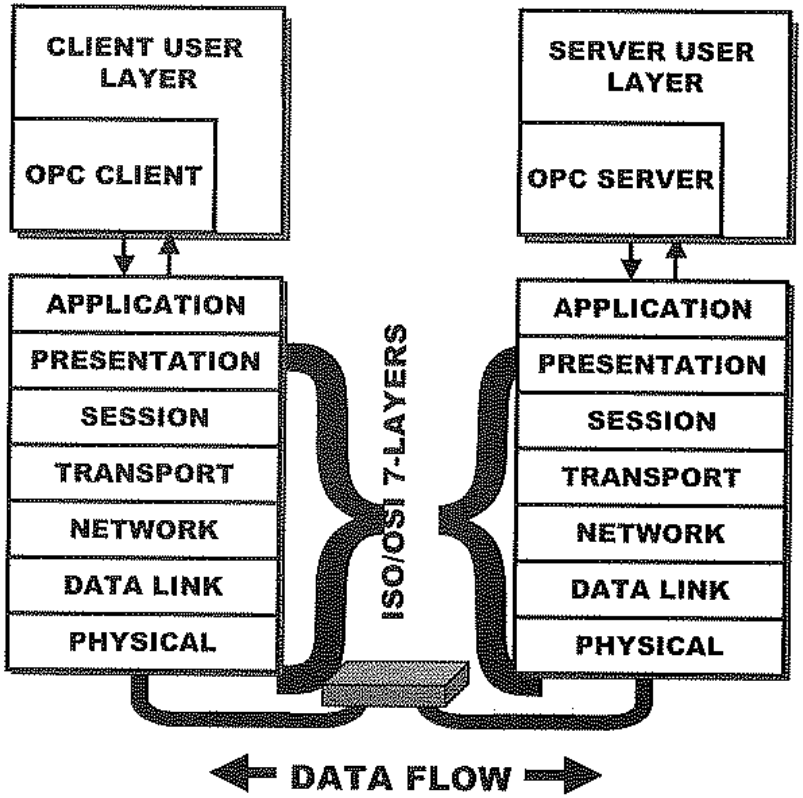
Like all modern data communication systems, fieldbus protocols are modeled according to the ISO/OSI model.

However, normally only layer 1,2 and 7 are actually used.

Fieldbuses are typically single segment networks, and extensions are realized via repeaters or bridges. Therefore network and transport layers are simply not necessary.

If the function of layer 3 to 6 is necessary they are typically implemented in layer 2 or 7.

Industrial network architecture



The end user only cares about the connection to the physical wires coming out at the bottom and features and functions made available on the top.

Notice that there are two layers above the application layer, often referred to as user layers.

User layer

- The creation of automation profiles originates from the fact that the definition of protocol layers alone is not sufficient to allow for implementation of interoperable product, because there are to many degrees of freedom.
- A profile limits the top-level functionality and defines specialized subsets for particular application areas.
- They specify communication objects, data types and their encoding.
- Often seen as an additional layer on the ISO/OSI model often referred to as ***layer 8*** or ***user layer***.

User layer – Profiles

- Profiles can be distinguished into communication, device, and branch profiles.
- Communication profiles define the mapping of communication objects onto services offered by the fieldbus (application layer).
- Branch profiles specify common definitions within an application area concerning terms, data types, their encoding and physical meaning.
- Device profiles build on communication and branch profiles and describe functionality, interfaces and in general the behavior of the entire class of devices, e.g., electrical drives, hydraulic valves and simple sensors and actuators.

User layer – OPC

- Many applications in industrial automation are highly generic and intended to operate with a variety of automation systems. As a result software developers, working with automation systems were required to write drivers for each of the automation systems to which they would connect.
- The ***object linking and embedding for process control*** (OPC) layer adapt the application layer to the host system.
- OPC solves that by enabling any application working with automation systems supplying an OPC server.

User layer – OPC

- "The OPC Specification is a non-proprietary technical specification that defines a set of standard interfaces based upon Microsoft's OLE/COM/DCOM platform and .NET technology. The application of the OPC standard interface makes possible interoperability between automation/control applications, field systems/devices and business/office applications."*

- The communication only take place between the OPC client and the OPC server, ***freeing software developers from the particularities of automation system***.

*http://en.wikipedia.org/wiki/Opc_server

User layer – OPC

- OPC is a high-level protocol for standardizing host-to-controller communication.
- OPC was first designed for operation on the Microsoft Windows operating system.
- It is founded on the Component Object Model (COM) and its network distributed equivalent (DCOM).
- Two applications can pass messages between each other using COM and DCOM.
- COM/DCOM is based on the general remote procedure call construction and its message passing interface between executing objects is Object Linking and Embedding (OLE).

User layer – OPC

- If the networked device is not running Windows it must support COM/DCOM, which is an open standard supported by the ActiveX Consortium.
- COM/DCOM is supported by many operating systems including several used for embedded systems.
- Object attribute definitions is provide by OPC/DX (data eXchange) built as an additional layer on top of OPC/XML (eXtension markup language).
- The purpose is to allow the definition of data independent of both the control system supplying the server, and the data management or presentation system supplying the OPC/DX client.

User layer – FDT

- Most fieldbuses define attributes of the devices used for sensing and control. These attribute definitions are contained in the user layer.
- Field Device Tools (FDT) was created to eliminate the need for the user to maintain the different attribute definitions for each fieldbus in use.
- FDT allows the field device suppliers to offer a single device type manager (DTM) independent of the fieldbus used for a project, where the host device uses an FDT framework server.
- FDT does not have the same support in factory automation since few PLC suppliers have created FDT framework servers or DTMs for binary field devices.

User layer – FDT

- FDT and OPC are used to build a layer of abstraction above the application layer in the protocol stack, to make the specific communication standard used, transparent to the end user and the system supplier.

Communication paradigms

- Client-Server
- Producer-Consumer
- Publisher-Subscriber

Communication paradigms

Client-Server

- Peer-to-peer
- Connection-oriented
- Monomaster, multimaster
- Confirmed, unconfirmed, acknowledge
- Parameter transfer, cyclic communication

Communication paradigms

Producer-Consumer

- Broadcast
- Connectionless
- Multimaster
- Unconfirmed, acknowledge
- Event notification, alarms, error, synchronization

Publisher-Consumer

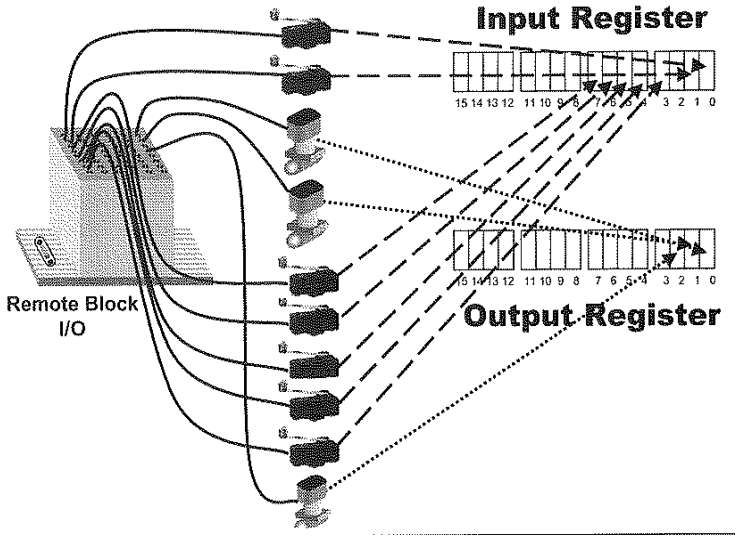
- Multicast
- Connectionless
- Multimaster
- Unconfirmed, acknowledge
- State change, event-oriented signal sources (e.g. switches)

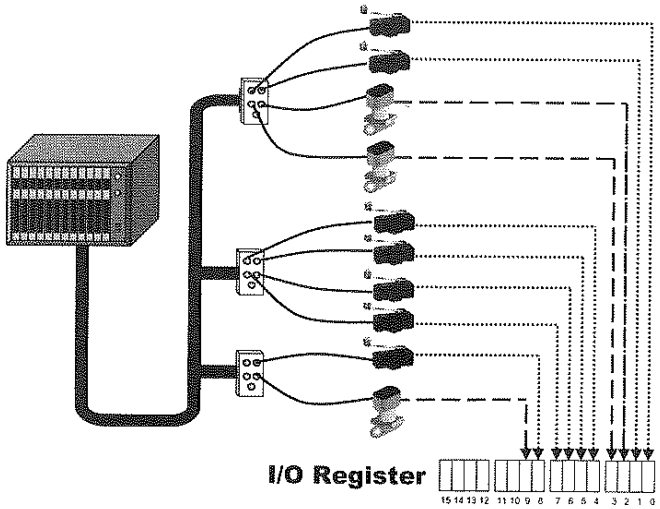
Sensor/actuator networks

- Sensor/actuator networks are designed to reduce the point-to-point wiring needed to connect sensors and actuators to a PLC.
- This is done in two ways:
  - Put a network driver inside the sensor or actuator.
  - Bring the I/O interface closer to the sensor or actuator so that the connection is very short. A typical I/O interface box terminates 4 to 16 I/O points.
- The distinguishing factor of a ***sensor/actuator network*** is that the sensor, actuator and the network nodes do nothing more than converts the sensor or actuator state to or from the network status word.

Sensor/actuator networks

- All sensor/actuator networks operate similarly by making the I/O appear in series of registers in the I/O unit in the same way as if they where directly point-to-point wired.
- This makes the sensor/actuator network invisible for a PLC.





Sensor/actuator networks

- Sensor/actuator networks work by actually detecting the status of the sensor or setting the status of an actuator. The status word is generated/terminated by a so called scanner, typically embedded in the PLC.
- In many of these applications the speed of detection is critical, meaning that a change of state must be detected or set within one scan cycle of the PLC (the time it takes for a PLC to scan all input and output registers).
- A scan cycle is typically 3 to 5 ms, sensor networks are designed to meet these requirements.

Fieldbus networks

- A ***fieldbus*** is defined as a network in which there are distributed and programmable intelligent nodes in the network.
- Devices that measure or set physical variables (scalars) are considerably more complex than devices (sensor/actuators) that are used for discrete values (on/off).
- These devices often require many configuring parameters to perform scaling and filtering of the raw measurement.
- These sensors are often referred to as ***smart sensors*** (or smart actuators).

Fieldbus networks

- Fieldbuses are also used to ***interconnect PLCs, PCs, and HMI***, into a network to share information.
- However there is not possible to bridge control logic over the network. This means that logic in one PLC cannot link directly with logic in another PLC on the same network
- One can obtain the status of a certain input over the network but the time determinism is lost, i.e., the timing of the fieldbus network is not synchronized with the timing of the PLC logic in this case.

Fieldbus networks

- Process control operation are typically much slower than trajectory applications, but often require large data transfer and tight time synchronization, e.g., between nodes forming an cascaded control loop.

Control networks

- Control level networks are intended to allow control systems to connect with each other and to serve as a bridge of control systems to business systems.
- Large amount of data is often passed on this level, messages tend to be longer and use higher transmission rate than fieldbus networks.
- Since they can be used to pass time-critical data between controllers, control networks must be deterministic.
- Control networks and fieldbuses can often be used for the same applications.