Intelligent distributed systems

Daniele Fontanelli

Department of Industrial Engineering
University of Trento
E-mail address: *daniele.fontanelli@unitn.it*

2022/2023

Outline

1. Automation protocols
2. Fieldbuses
   - FF: H1 Protocol
   - CAN bus
   - PROFIBUS
3. Take home message

Outline

Automation protocols

Wikipedia results...

Process automation protocols [edit]

- AS-i – Actuator-sensor interface, a low level 2-wire bus establishing power and communications to basic digital and analog devices
- BSAP – Bristol Standard Asynchronous Protocol, developed by Bristol Babcock Inc.
- CC-Link Industrial Networks – Supported by the CLPA
- CIP (Common Industrial Protocol) – can be treated as application layer common to DeviceNet, CompoNet, ControlNet and EtherNet/IP
- Controller Area Network utilised in many network implementations, including CANopen and DeviceNet
- ControlNet – an implementation of CIP, originally by Allen-Bradley
- DeviceNet – an implementation of CIP, originally by Allen-Bradley
- DF-1 - used by Allen-Bradley PLC-5, SLC-500, and MicroLogix class devices
- DirectNet – Koyo / Automation Direct[1] proprietary, yet documented PLC interface
- EtherCAT
- Ethernet Global Data (EGD) – GE Fanuc PLCs (see also SRTP)
- EtherNet/IP – IP stands for "Industrial Protocol". An implementation of CIP, originally created by Rockwell Automation
- Ethernet Powerlink – an open protocol managed by the Ethernet POWERLINK Standardization Group (EPSG).
- FINS, Omron's protocol for communication over several networks, including ethernet.
- FOUNDATION fieldbus – H1 & HSE
- HART Protocol
- HostLink Protocol, Omron's protocol for communication over serial links.
- Interbus, Phoenix Contact's protocol for communication over serial links, now part of PROFINET IO
- MACRO Fieldbus - "Motion and Control Ring Optical" developed by Delta Tau Data Systems.
- MECHATROLINK – open protocol originally developed by Yaskawa, supported by the MMA.
- MelsecNet, supported by Mitsubishi Electric.
- Modbus PEMEX
- Modbus Plus
- Modbus RTU or ASCII or TCP
- OSGP – The Open Smart Grid Protocol, a widely use protocol for smart grid devices built on ISO/IEC 14908.1
- Optomux – Serial (RS-422/485) network protocol originally developed by Opto 22 in 1982. The protocol was openly documented[2] and over time used for industrial automation applications.
- PieP – An Open Fieldbus Protocol
- Profibus – by PROFIBUS International.
- PROFINET IO
- RAPIEnet – Real-time Automation Protocols for Industrial Ethernet
- Honeywell SDS – Smart Distributed System – Originally developed by Honeywell. Currently supported by Holjeron.
- SERCOS III, Ethernet-based version of SERCOS real-time interface standard
- SERCOS interface, Open Protocol for hard real-time control of motion and I/O
- GE SRTP – GE Fanuc PLCs
- Sinec H1 – Siemens
- SynqNet – Danaher
- TTEthernet – TTTech

Figure: Available *process automation protocols* according to Wikipedia.

Automation protocols

What is covered in this course

In this course, we will see in some more details only a small subset of the proposed solutions.
The choice is dictated by the *popularity* gained for industrial and robotics automation solutions.
In particular, we will see:

- *Fieldbuses*: Foundation Fieldbus H1, CAN bus, PROFIBUS;
- *Real-Time Ethernet-based industrial communication systems*: Foundation Fieldbus HSE, EtherCAT, Ethernet Powerlink, PROFINET IO, PROFINET IRT (other slides);
- *Wireless communications*.

Outline

What is a fieldbus?

An industrial distributed automation solution is organised as a *hierarchy of controllers*.

At the top of the hierarchy there is usually the *human supervisor* that interact with a *Human Machine Interface* (HMI) to monitor data in a meaningful way or operate the system.

The second level of the hierarchy usually comprises a network of *computers* or *programmable logic controllers* (PLCs) through a non-time-critical communication system (e.g. Ethernet).

The lowest level of the hierarchical control chain there is the *fieldbus* connecting the PLCs with sensors, actuators, switches, etc.

What is a fieldbus?

Example

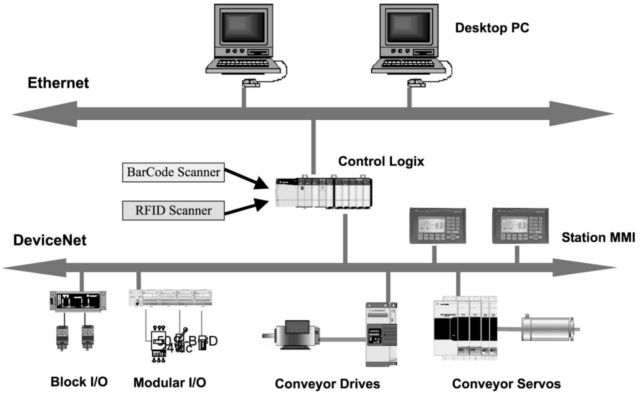


Figure: Example of hierarchical structure (A. Mousavi, M. Sarhadi, A. Lenk, S. Fawcett, "Tracking and traceability in the meat processing industry: a solution", British Food Journal).

Industrial networks

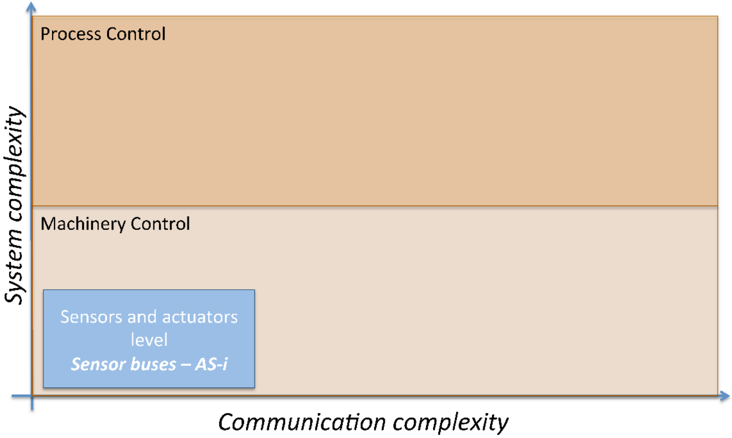


Figure: Main industrial networks and buses: sensor level.

Industrial networks

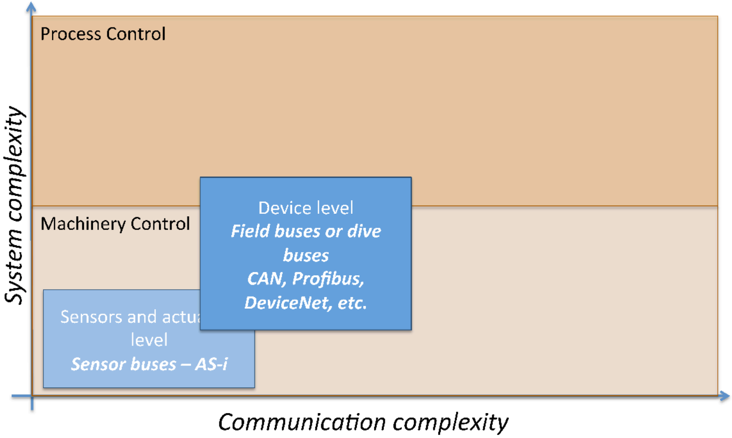


Figure: Main industrial networks and buses: device level.

Industrial networks

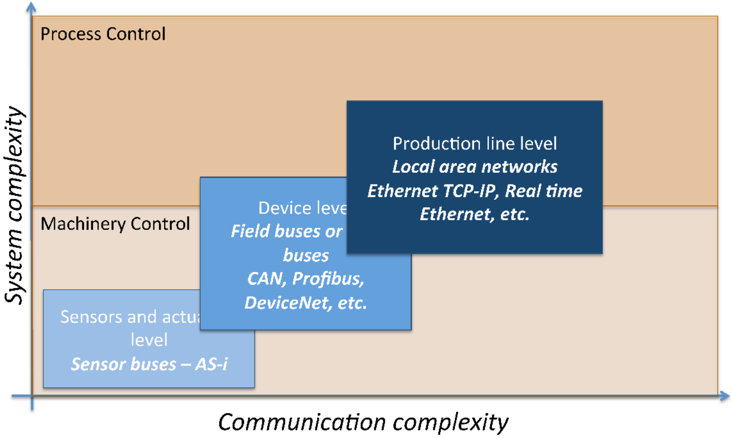


Figure: Main industrial networks and buses: production line level.

Industrial networks

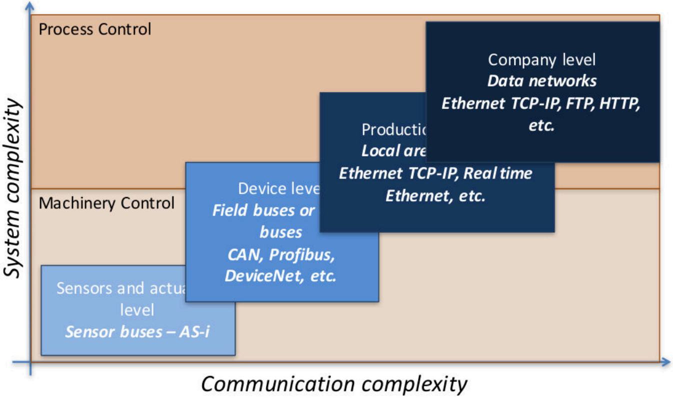


Figure: Main industrial networks and buses: company level.

What is a fieldbus?

Fieldbus is hence the name of a *family of industrial digital network protocols* used for *real-time distributed control*.

Definition

The generic term *fieldbus* refers to any bus that connects to field devices.

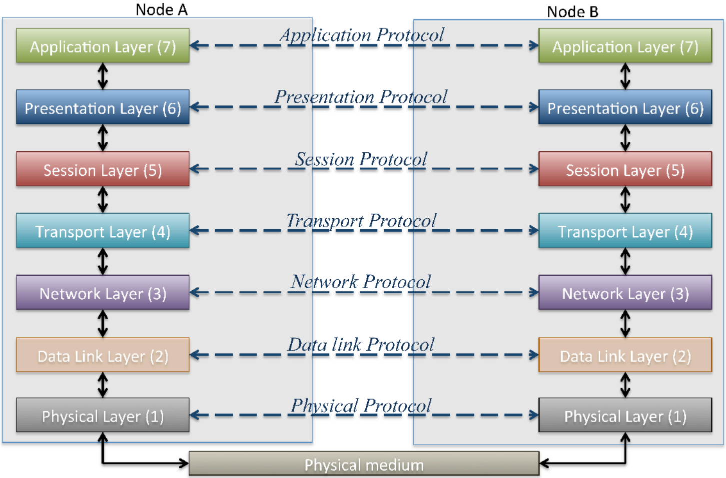
The *Fieldbus Foundation* defines as fieldbus:

Definition

A digital, two-way, multi-drop communication link among intelligent measurement and control devices.

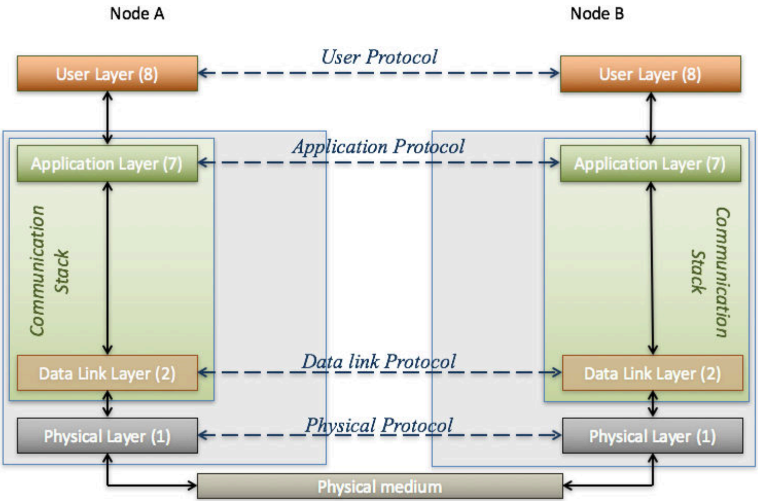
OSI Model

Graphical representation



Fieldbus OSI Model

Graphical representation



Outline

1. Automation protocols

2. Fieldbuses
   - FF: H1 Protocol
   - CAN bus
   - PROFIBUS

3. Take home message

Foundation fieldbus

OSI model: Physical layer

Physical layer

Foundation fieldbus

OSI model: Physical layer - Circuitry

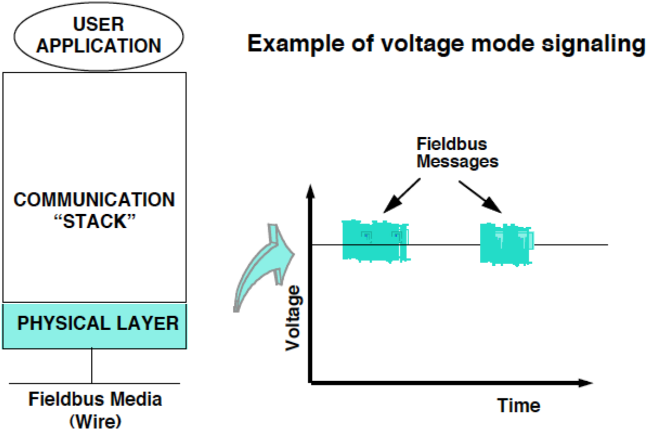


Figure: Example of voltage signalling (courtesy of FOUNDATION™ fieldbus).

Foundation fieldbus

OSI model: Physical layer - Circuitry

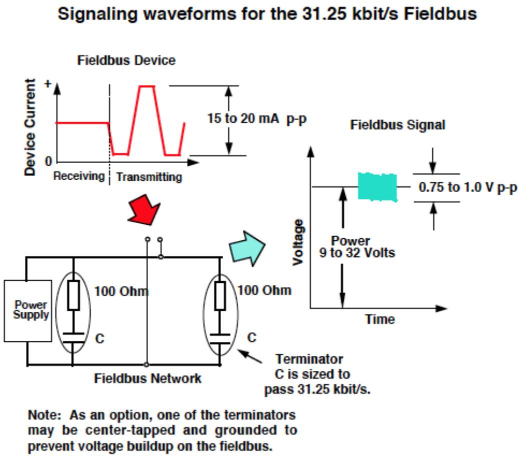


Figure: FF signalling waveforms (courtesy of FOUNDATION™ fieldbus).

Foundation fieldbus

OSI model: Physical layer - Circuitry

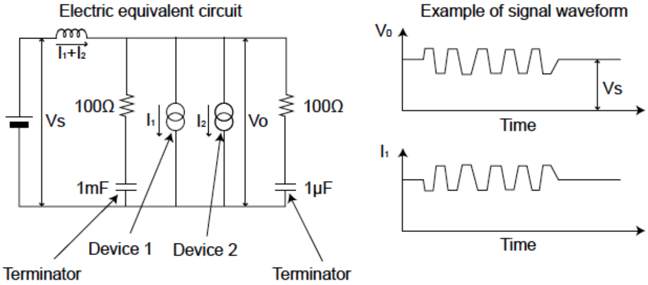


Figure: FF equivalent circuit of signal transmission (courtesy of Yokogawa Electric Corporation).

- Supply voltage is provided by a power supply through an *impedance conditioner* (between 9 to 32 V DC at the devices). DC current through the impedance conditioner feeds devices;
- A 100 Ω impedance terminator is installed at each cable terminal. It makes an instrumentation cable a *balanced transmission line* (relatively high frequency transmission);

Foundation fieldbus

OSI model: Physical layer - Circuitry

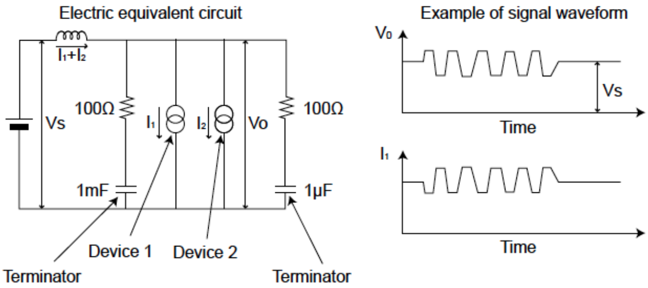


Figure: FF equivalent circuit of signal transmission (courtesy of Yokogawa Electric Corporation).

- For a *bipolar device*, there is commonly a 10 mA that is *wasted*, i.e. absorbed without using it.
- When the device wants to transmit a high signal, it *turns off* this 10 mA.
- For example, if device 1 drops the I_1 current of 10 mA, the voltage between the cables increases by 10 mA x 50 $\Omega =$ 0.5 V;

Foundation fieldbus

OSI model: Physical layer - Circuitry

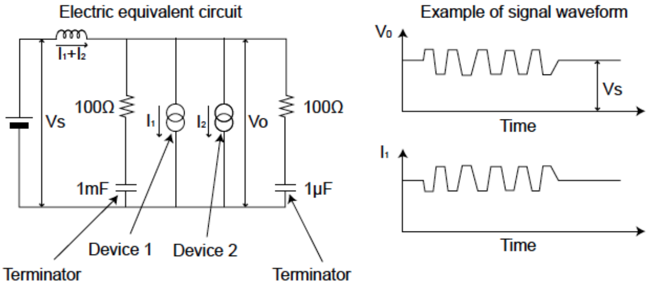


Figure: FF equivalent circuit of signal transmission (courtesy of Yokogawa Electric Corporation).

- When the device 1 absorbs, i.e. wastes, I_1 20 mA, the voltage between the cables decreases by 1 V DC and a *low level* is transmitted.
- Thus the average voltage (V_s) is maintained at the same level and generates a modulated signal of 1 V amplitude;
- For unipolar devices, the current is not absorbed when there is no transmission: power saving but more complicated transmission.

Foundation fieldbus

OSI model: Physical layer - Actual signal

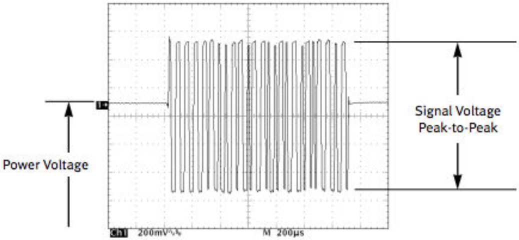


Figure: FF actual transmitted signal (courtesy of RELCOM, INC.).

Foundation fieldbus

OSI model: Physical layer - Waveform

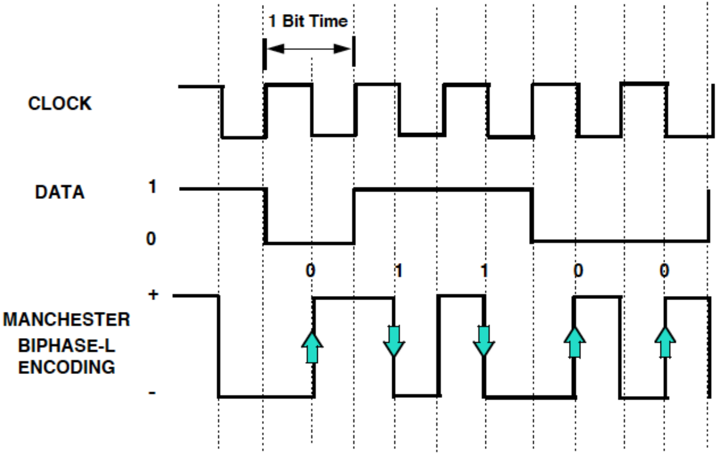


Figure: Example of signalling encoding (courtesy of FOUNDATION™ fieldbus). The bit time at 31.25 kbs is 32 μs.

Foundation fieldbus

OSI model: Physical layer - Waveform

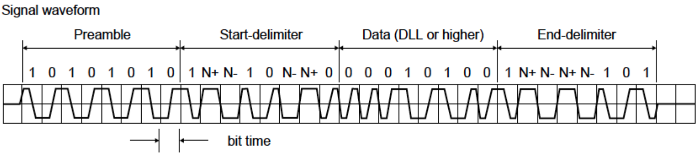


Figure: FF typical waveform of a *Physical layer* signal (courtesy of Yokogawa Electric Corporation).

- Data is encoded as a voltage change in the middle of the one-bit time (32 μs at 31.25 kbps): 1 (0) is encoded as a voltage drop (increase) in the middle of one bit time;
- $N+$ and $N-$ are encoded as the constant voltage between the bit times (e.g., start/stop signals used for dividing frames to generate specifically identified signals separated from ordinary data);

Foundation fieldbus

OSI model: Physical layer - Waveform

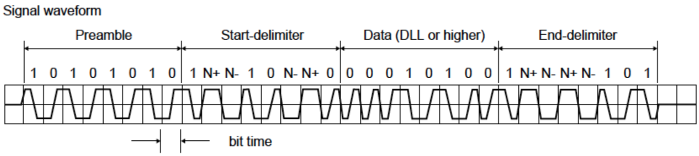


Figure: FF typical waveform of a *Physical layer* signal (courtesy of Yokogawa Electric Corporation).

- The receiving *Physical layer* retrieves the bit time using the preamble and the *octets* (bytes) of start delimiter signal;
- The end delimiter indicates the end of the *Physical layer* signal;
- The length of the preamble can be increased when the signal goes through the repeaters.

Foundation fieldbus

OSI model: Physical layer - Example of dimensionality

To understand how many devices can be connected, consider this example:

- The power supply and power conditioner output is 20 volts;
- The cable used has a resistance of $22\ \Omega/\text{km}$ for each conductor, hence $44\ \Omega$ for the two wires;
- The cable length is 1 km;
- Each device at the end of the cable draws 20 mA.

Foundation fieldbus

OSI model: Physical layer - Example of dimensionality

Since the minimum voltage needed by a device is 9 V, there are $20 - 9 = 11$ V that can be used by the cable.
The total current that can be supplied is $11/44 = 250$ mA.
Since each device draws 20 mA, the maximum number of devices is $250/20 = 12$ devices.
The power used by Fieldbus devices varies by device type and one important issue to take into account is the *initial inrush current and the lift-off voltage*. Hence, the power consumption should be considered for the *worst case*.
When the network becomes more complicated, e.g. tree topology with several sprung, the power consumption computation becomes more involved.

Foundation fieldbus

OSI model: Physical layer - Attenuation

The IEC standards define the *minimum amplitude* and the *worst waveform* of a received signal at a device at a place of the fieldbus network. The *Physical layer* receiver circuit must be able to receive this signal.
As signals travel on a cable, they become *attenuated*, that is, getting smaller. Attenuation is measured in units called *dB* or *deci-Bell*.
It is calculated as

$$dB = 20 \log_{10} \frac{\text{transmitted signal amplitude}}{\text{received signal amplitude}}.$$

Foundation fieldbus

OSI model: Physical layer - Attenuation

Cables have *attenuation ratings* for a given frequency.
Standard Fieldbus cables have 3 dB/km at 39 kHz.
In other words, the percentage of the transmitted signal at the receiver side after one kilometer is

$$\frac{1}{10^{\frac{3}{20}}}100 \approx 71\%.$$

For a shorter cable, say 500 m, we have

$$\frac{1}{10^{\frac{1.5}{20}}}100 \approx 84\%.$$

Foundation fieldbus

OSI model: Physical layer - Attenuation

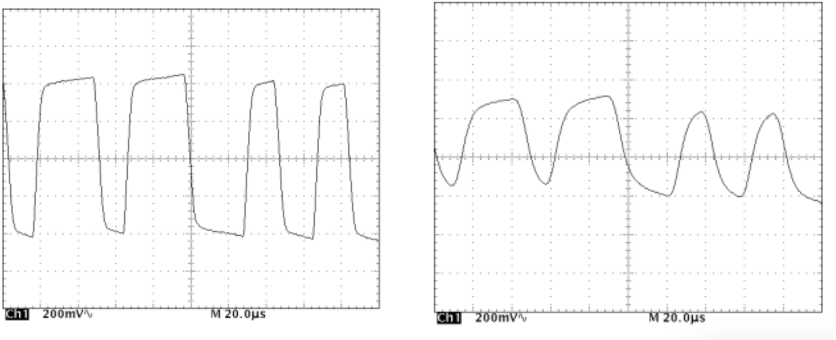
Assume that a *Fieldbus transmitter* can generate a signal as low as 0.6 V peak-to-peak in the worst case.
Moreover, assume that a *Fieldbus receiver* is able to detect a signal as little as 0.2 V peak-to-peak.
What should be the *maximum length of the cable* if it has an attenuation of 3 dB/km?
The cable can then attenuate the signal up to

$$20 \log_{10} \frac{0.6}{0.2} \approx 9.5 \text{ dB},$$

hence the Fieldbus can be as long as $9.5/3 \approx 3.2$ km.

Foundation fieldbus

OSI model: Physical layer - Attenuation



(a) (b)

Figure: FF actual transmitted (a) and received (b) signals due to attenuation (courtesy of RELCOM, INC.).

Foundation fieldbus

OSI model: Physical layer - Hazardous areas

Fieldbus solutions are suitable for hazardous areas, which require a higher degree of safety, i.e., *intrinsically safe* solutions.
Intrinsically safe solutions are hard to be designed and depends on the maximum power transmitted/received on the cable.

Foundation fieldbus

OSI model: Data-Link layer

Data-Link layer

Foundation fieldbus

OSI model: Data-Link layer

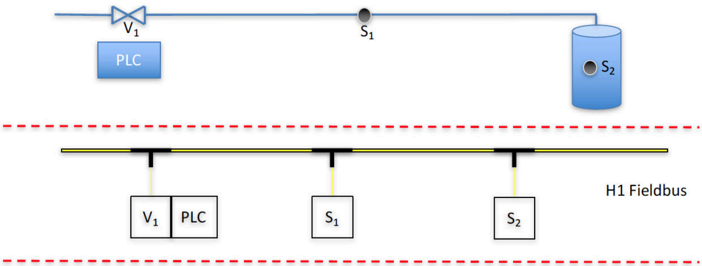
Communication is ruled by the *Link Active Scheduler* (LAS), which is one of the so-called *Link master*.

The communication services of the FF specification utilise *scheduled* (*time critical*) and *unscheduled* (*management*) data transmission:

- *Scheduled* transmission uses *publisher/subscriber* to notify all the nodes the time schedule to follow;
- *Unscheduled* transmissions take place only if *no scheduled communication is foreseen*, i.e., in free time slots, using the *Token Passing* (TP) scheme;
- A particular *unscheduled* message is the *probe*, used to automatically discover new devices or not responding devices to maintain a *live list*.

Foundation fieldbus

OSI model: Data-Link layer - Scheduled transmission example



Device	Type	Action	Offset [ms]
S1	Sensor	Ex. Sense Transmit	0 20
S2	Sensor	Ex. Sense Transmit	0 30
V1 + PLC	Controlled valve	Ex. PID Ex. Control	40 62

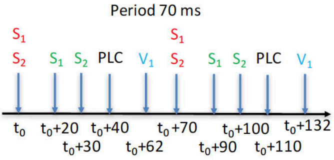


Figure: Scheduling example for a distributed plant with time-critical transmissions.

Foundation fieldbus

OSI model: Application layer

Application layer

Foundation fieldbus

OSI model: Application layer

The *Application layer* is divided into the *Fieldbus Message Specification* (FMS) and the *Fieldbus Access Sublayer* (FAS).
The FMS manage and schedule the communication using the device *profiles*.
The FAS uses the *scheduled* and *unscheduled* communications to provide services to the FMS using:

- *Publisher/Subscriber*: for scheduled control messages;
- *Client/Server*: for the human operator messages;
- *Report Distribution*: to acknowledge the other connected devices of warnings or problems.

Foundation fieldbus

OSI model: User layer

User layer

Foundation fieldbus

OSI model: User layer

The *User layer* provides the interface for user interaction with the system.
It uses the *device description*, i.e. a *configuration object*, to tell the host its inputs/outputs and the device capabilities, i.e. its *functionalities*.
Each device becomes a *function block* according to the IEC 61499.
To do so, the device vendor supplies *device description files*, which describe the parameters of the function and transducer blocks contained in a device.
The *linkages* connects such functionalities.
The *control loops* are sets of function blocks connected by linkages executed with a desired rate.
It is possible to have *multiple loops* with different rates hosted on the same link.

Foundation fieldbus

OSI model: User layer

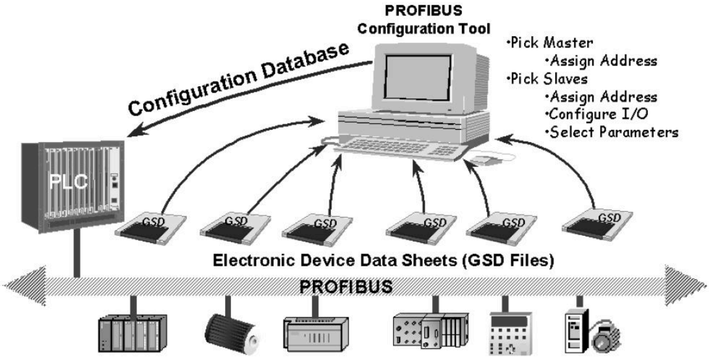


Figure: Example of configuration of devices in PROFIBUS.

Outline

1. Automation protocols

2. Fieldbuses
   - FF: H1 Protocol
   - CAN bus
   - PROFIBUS

3. Take home message

Random access with collision arbitration: CAN

Introduction

The *CAN* is a *serial communication protocol* developed mainly for applications in the automotive industry but also capable of offering good performance in other time-critical industrial applications.
The CAN protocol is optimised for *short messages* and uses a CSMA/CA on message priority as medium access method.

Controller Area Network

OSI model

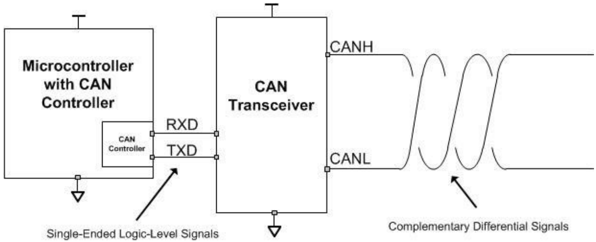


Figure: CAN Node (courtesy of Texas Instruments).

Controller Area Network

OSI model

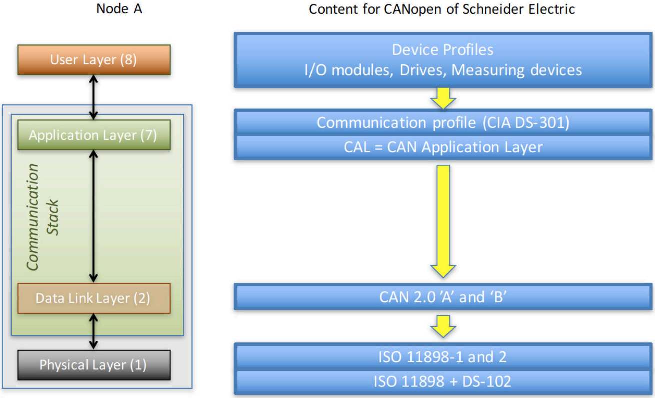


Figure: CAN and the OSI model.

Controller Area Network

Physical layer

Physical layer

Controller Area Network

Physical Layer

Medium: Shielded twisted pair 2 or 4-wire (if power supply)
Topology: Bus type with short tap links and 120 ohm line termination resistor
Maximum distance: 1000 m
Speed: 9 possible speeds from 1 Mbps to 10 Kbps. Speed depends on bus length and cable type: 25 m at 1 Mbps, 1000 m at 10Kbps
Max. no. of devices: 128, that are 1 master and 127 slaves
Max. size of useful data: 8 bytes per frame
Transmission security: One of the best local industrial networks. Numerous signalling and error detection devices ensure high transmission security.

Controller Area Network

Signalling

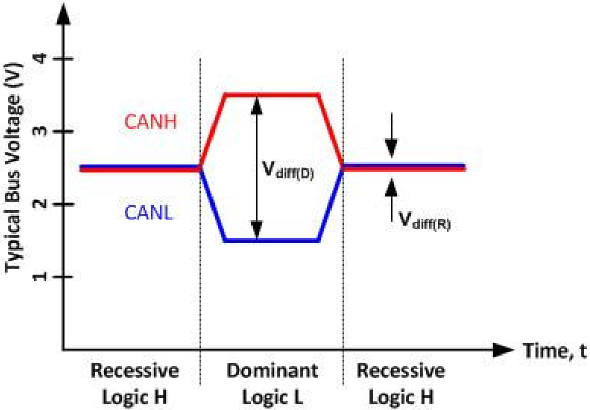


Figure: Signals at the *Physical layer* (courtesy of Texas Instruments).

Controller Area Network

Data-Link layer

Data-Link layer

Controller Area Network

Data-Link layer: Random access with collision arbitration

The protocol is *message oriented*, and each message has a *specific priority* that is used to arbitrate access to the bus in case of simultaneous transmissions.
The bit stream of a transmission is *synchronised* on the start bit, and the *arbitration* is performed on the *following message identifier*, in which a logic zero is dominant over a logic one.

Control Area Network

Data-Link layer: Sending field

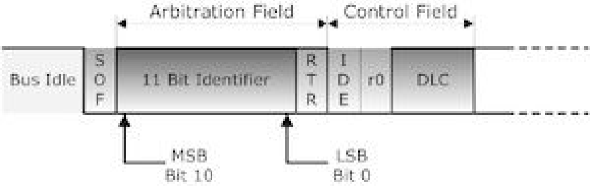


Figure: CAN sending field (courtesy of Wilfried F. Voss, "Controller Area Network (CAN) Bus Arbitration Principle").

SOF: Start of Frame; *RTR*: Remote Transmission Request.

Control Area Network

Data-Link layer: Arbitration

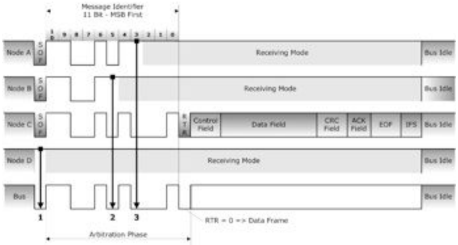


Figure: Arbitration (courtesy of Wilfried F. Voss, “Controller Area Network (CAN) Bus Arbitration Principle”).

The nodes in this example have the following message IDs: A) 1100101100 (32C hex); B) 1100110000 (330 hex); C) 1100101000 (328 hex); D) no significance (not requesting bus access).

Controller Area Network

Application layer

Application layer

Controller Area Network

Application Layer

Four types of standardised services using device profiles:

Network administration: Parameter settings, start-up, monitoring (master-slaves)

Transmission of low-volume process data ($<= 8$ bytes) in real time: PDO = Process Data Object (*producer-consumer*)

Transmission of high-volume parameter data (> 8 bytes) by segmentation without time restrictions: SDO = Service Data Object (*client-server*)

Predefined messages for managing synchronisation (SYNC), time-based references, fatal errors: SFO= Special Function Object

Outline

1. Automation protocols
2. Fieldbuses
   - FF: H1 Protocol
   - CAN bus
   - PROFIBUS
3. Take home message

Profibus

History

In 1987, the German federal minister for technological research and development creates a *Fieldbus* working group comprising 13 organisations including SIEMENS and 5 research institutes, which released *Profibus*, i.e., *PRO*cess *FI*eld*BUS*

PROFIBUS is managed by a user group which includes manufacturers, users and researchers: the *PROFIBUS CLUB*.

User clubs in 20 of the world's most industrialised countries provide support in native languages. These centres of competence are governed by the *PROFIBUS International* (PI) organisation, which has more than 750 members (*http://www.profibus.com/*).

PROFIBUS

Three versions

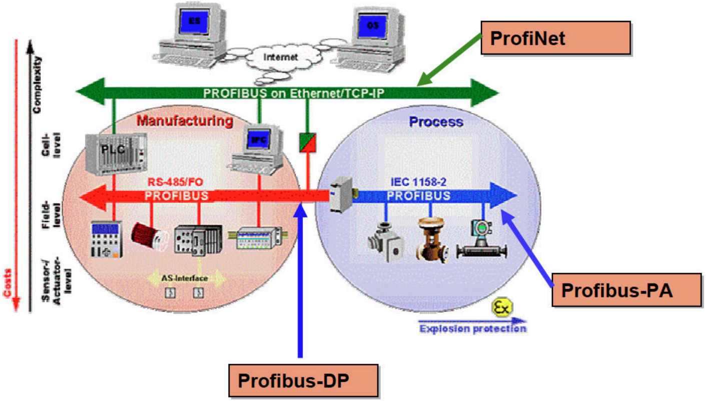


Figure: The three versions of PROFIBUS (courtesy of Schneider Electric).

PROFIBUS

PROFIBUS family

There are three different PROFIBUS family:

(1) *PROFIBUS-FMS*: Fieldbus Message Specification;

(2) *PROFIBUS-DP*: Dezentrale Pheripherie;

(3) *PROFIBUS-PA*: Process Automation.

PROFIBUS-DP

PROFIBUS-DP (utilising typically RS-485) is aimed at *time-critical communications* between automation and distributed peripherals and is based around DIN 19245-Part 3 since 1993.
It does not implement the *Application layer*. Therefore, at the *User layer* it describes the devices with profiles.

PROFIBUS-DP

OSI model

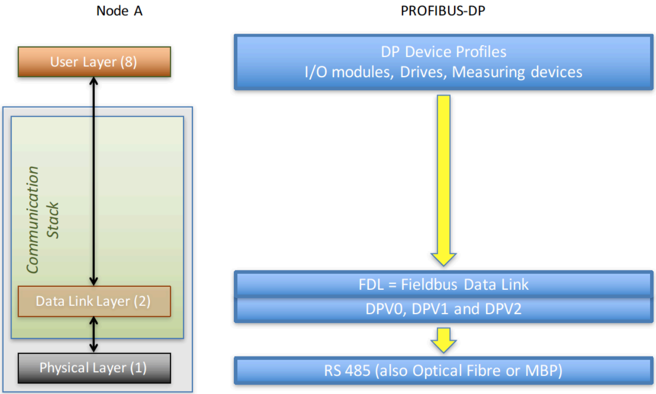


Figure: *PROFIBUS-DP* and the OSI model.

PROFIBUS-DP

Description

PROFIBUS DP is a widespread fieldbus based on a *master-slave cyclic* approach.

Implements *real-time* data exchange between controllers (masters) and field devices (slaves).

PROFIBUS DP cycles around the masters using a *non real-time* exchange based on *token-passing*.

Hence, at the *Data-Link layer*, the medium access is granted exclusively to master stations via a token passing scheme.

PROFIBUS-DP

Token ring and master-slave

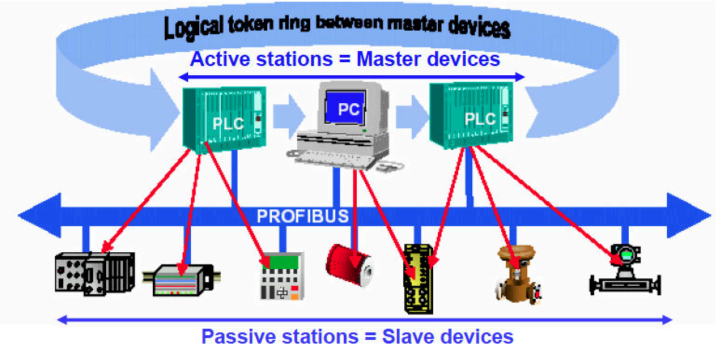


Figure: PROFIBUS: the token ring and master-slave (courtesy of Schneider Electric).

Outline

Fieldbuses

The *fieldbus* is the main communication solution for industrial plants.

In general, only the *Physical*, the *Data Link* and the *Application* layers are implemented in the communication stack.

There are specific fieldbuses that n can be used in *hazardous areas*.

The *Application* layer implements the *function blocks* idea.

Two main examples have been shown with some details: the *CAN bus* and the *PROFIBUS-DP*.