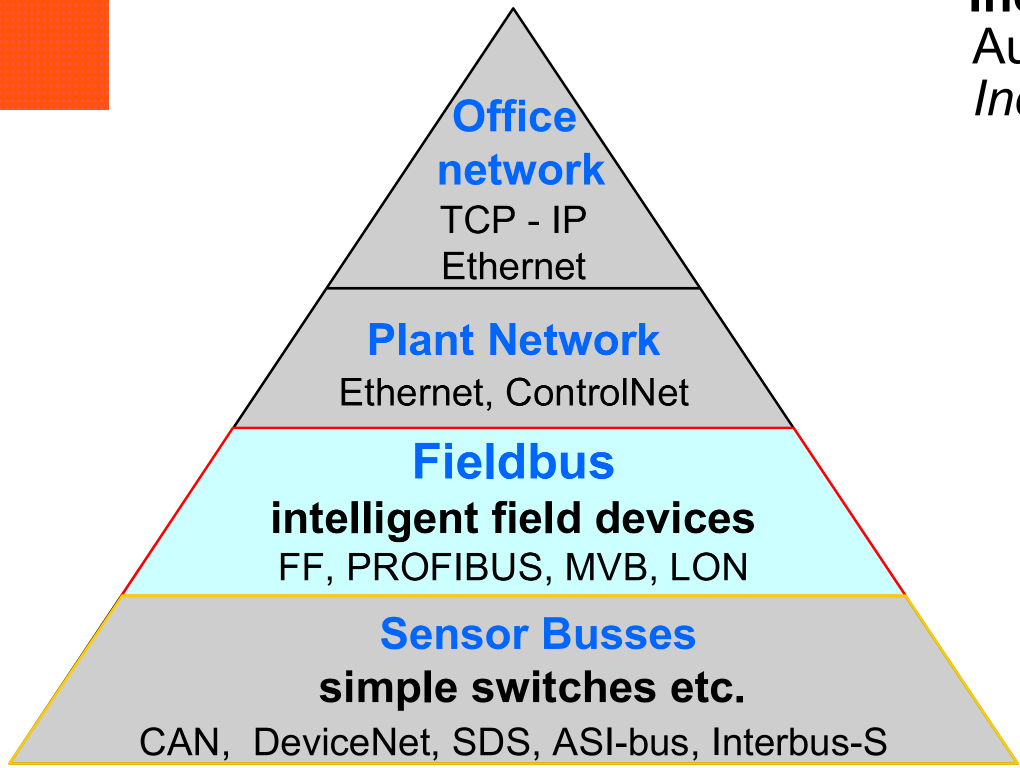
Industrial Automation
Automation Industrielle
Industrielle Automation



3 Industrial Communication Systems

3.1 Field Bus: principles
Bus de terrain: principes
Feldbusse: Grundlagen

Prof. Dr. H. Kirrmann
EPFL / ABB Research Center, Baden, Switzerland

2005 March, HK

Field bus: principles

3.1 Field bus principles

Classes

Physical layer

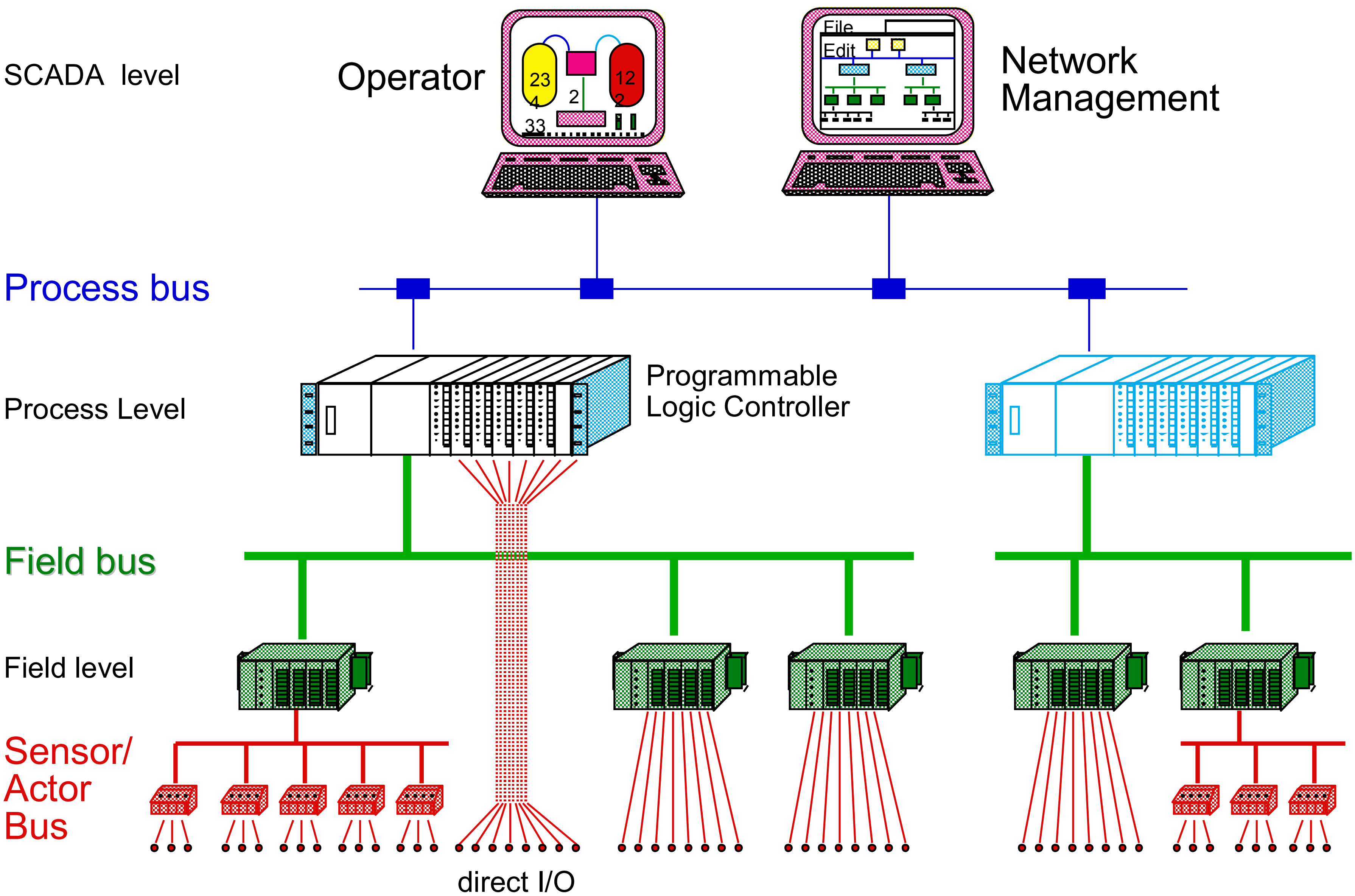
3.2 Field bus operation

Centralized - Decentralized

Cyclic and Event Driven Operation

3.3 Standard field busses

Location of the field bus in the plant hierarchy



What is a field bus ?

A data network, interconnecting a control system, characterized by:

- transmission of numerous small data items (process variables) with bound delay (1ms..1s)
- harsh environment (temperature, vibrations, EM-disturbances, water, salt,…)
- robust and easy installation by skilled people
- high integrity (no undetected errors)
- high availability (redundant layout)
- clock synchronization (milliseconds down to a few microseconds)
- continuous supervision and diagnostics
- low attachment costs (€ 5.- / node)
- moderate data rates (50 kbit/s … 5 Mbit/s) but large distance range (10m .. 4 km)
- non-real-time traffic for commissioning (e.g. download) and diagnostics
- in some applications intrinsic safety (oil & gas, mining, chemicals,..)

Expectations

- reduce cabling

- increased modularity of plant (each object comes with its computer)

- easy fault location and maintenance

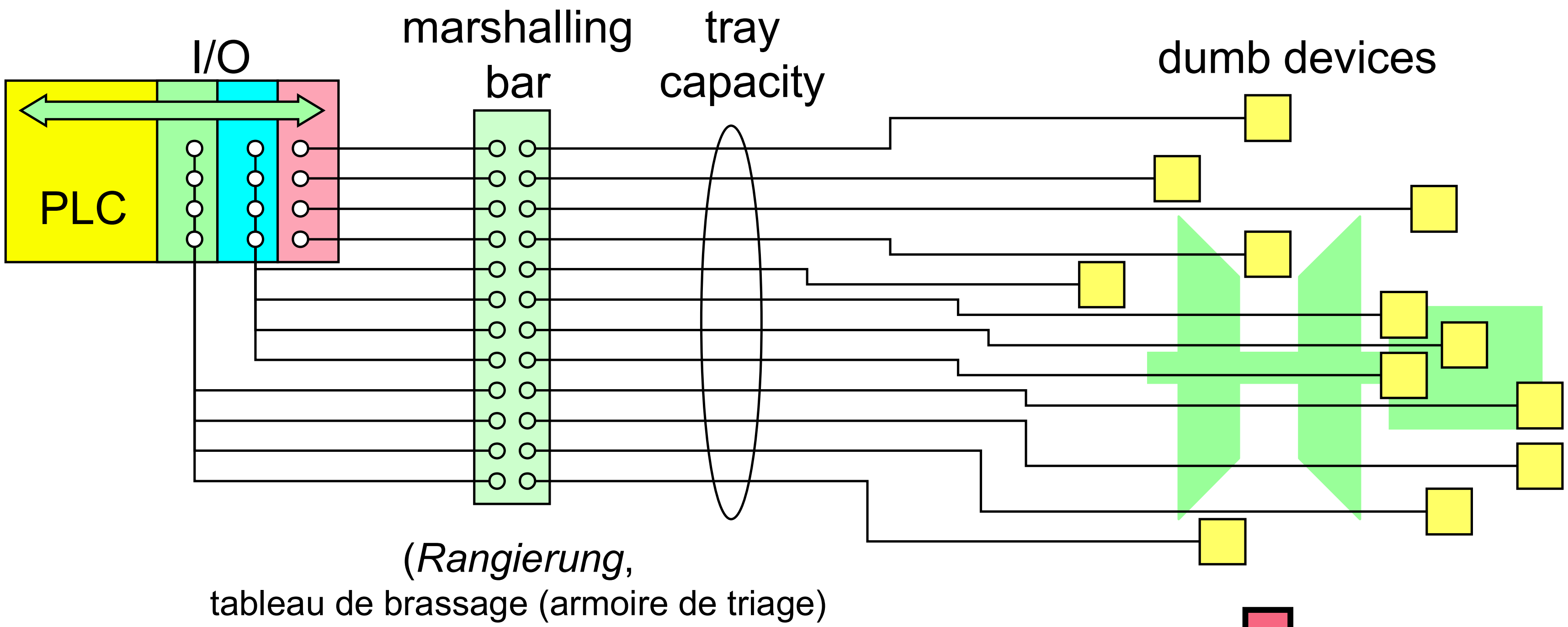
- simplify commissioning (mise en service, IBS = *Inbetriebssetzung*)

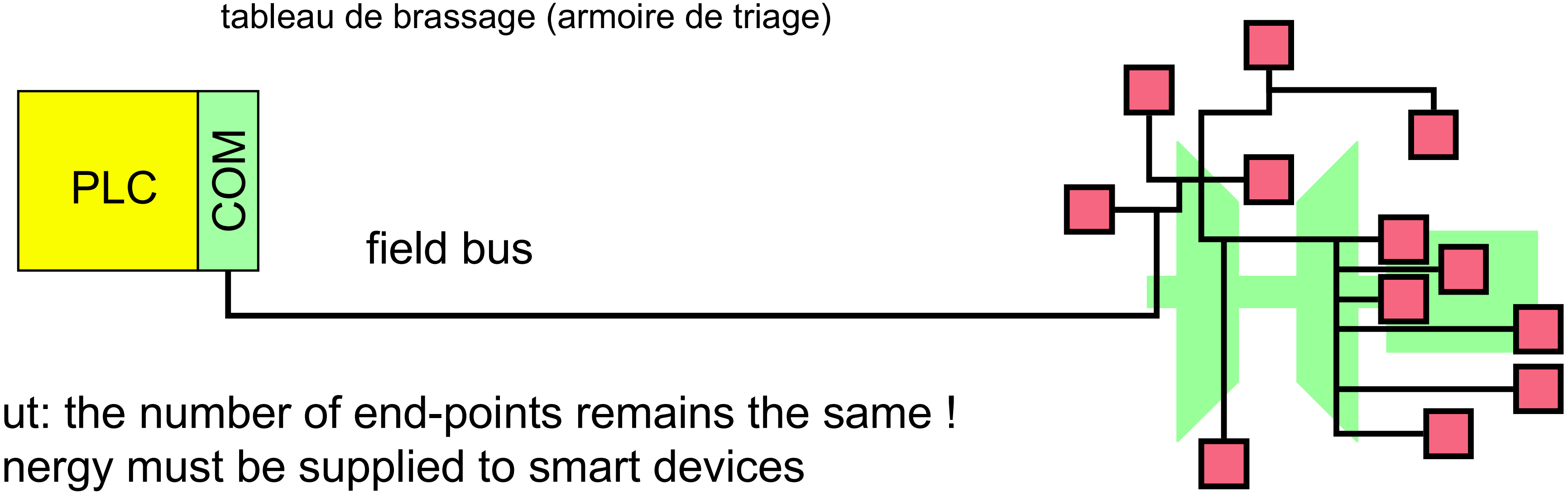
- simplify extension and retrofit

- large number of off-the-shelf standard products to build “Lego”-control systems

- possibility to sell one’s own developments (if based on a standard)

The original idea: save wiring

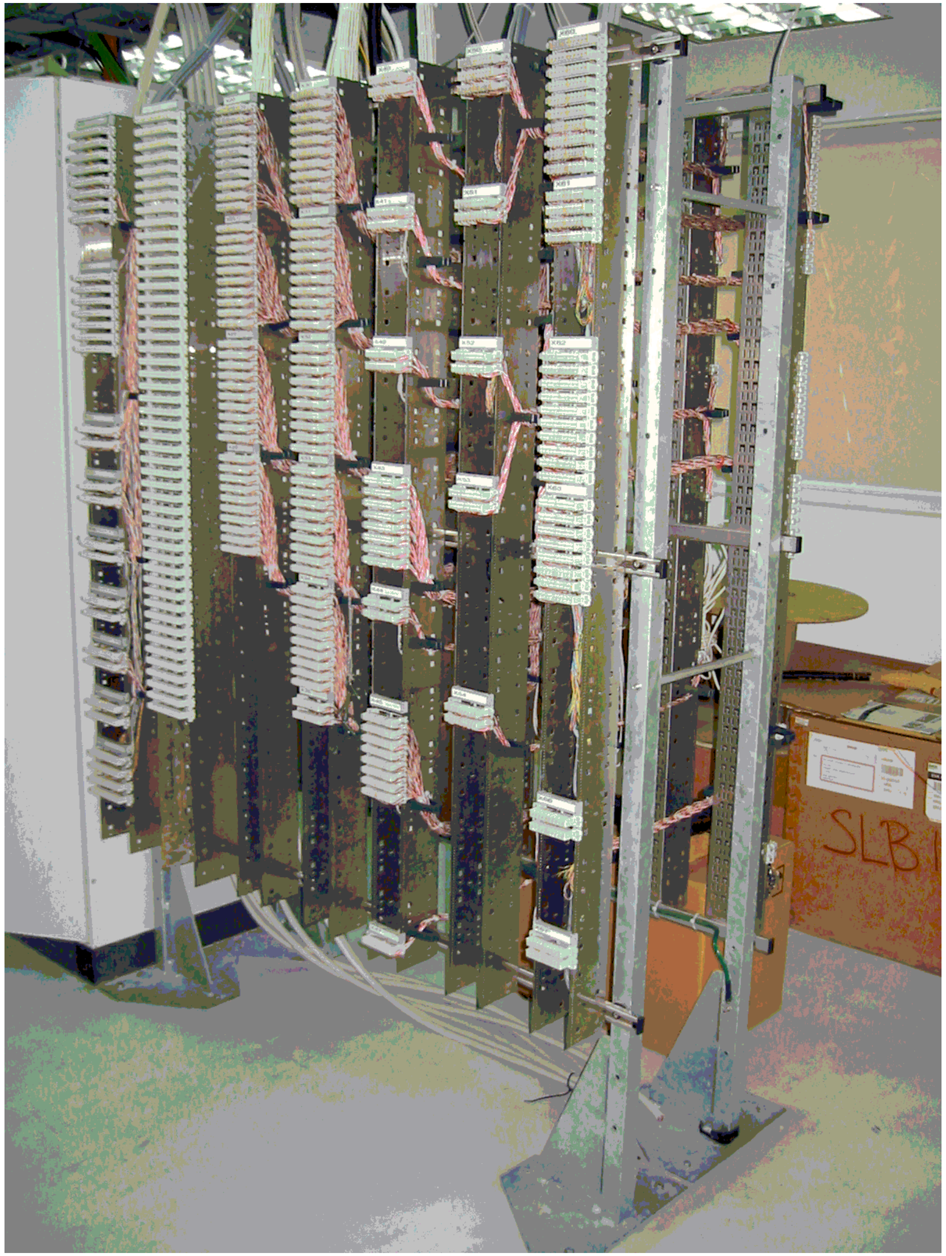




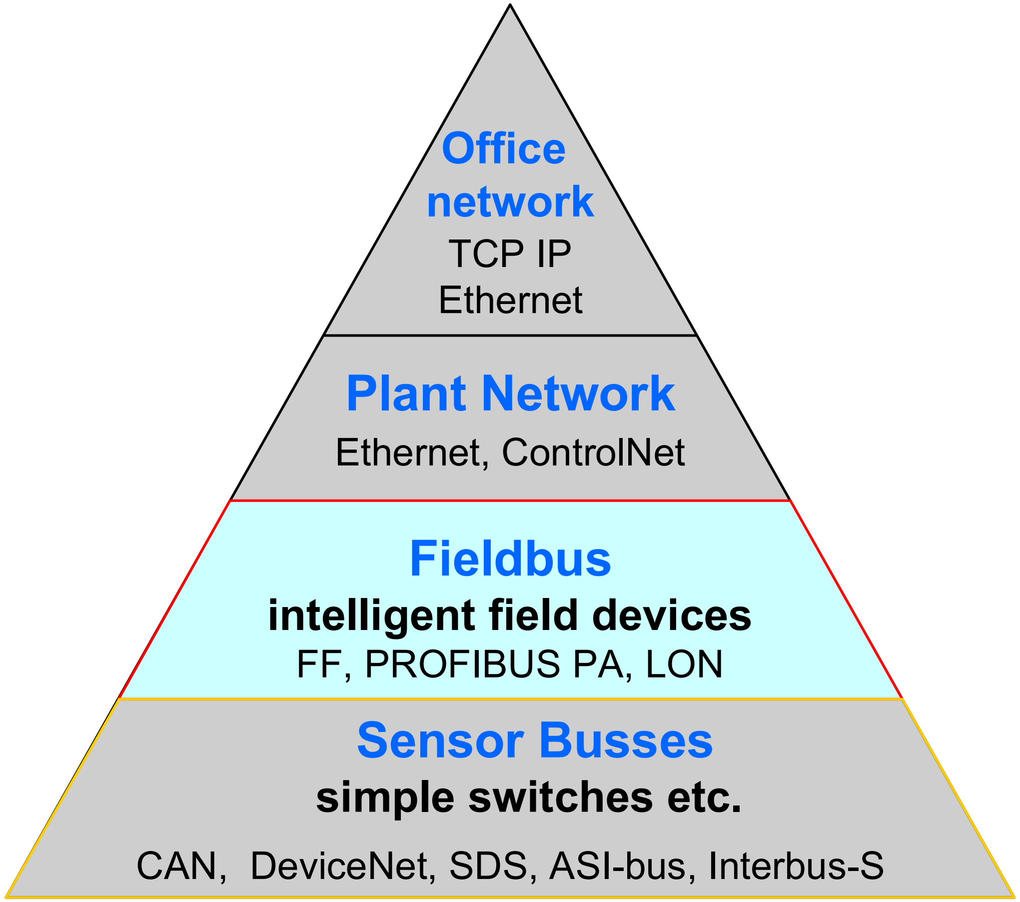
but: the number of end-points remains the same !
energy must be supplied to smart devices

Marshalling (Rangierschiene, Barre de rangement)

The marshalling is the interface between the PLC people and the instrumentation people.



Field busses classes



The field bus depends on:

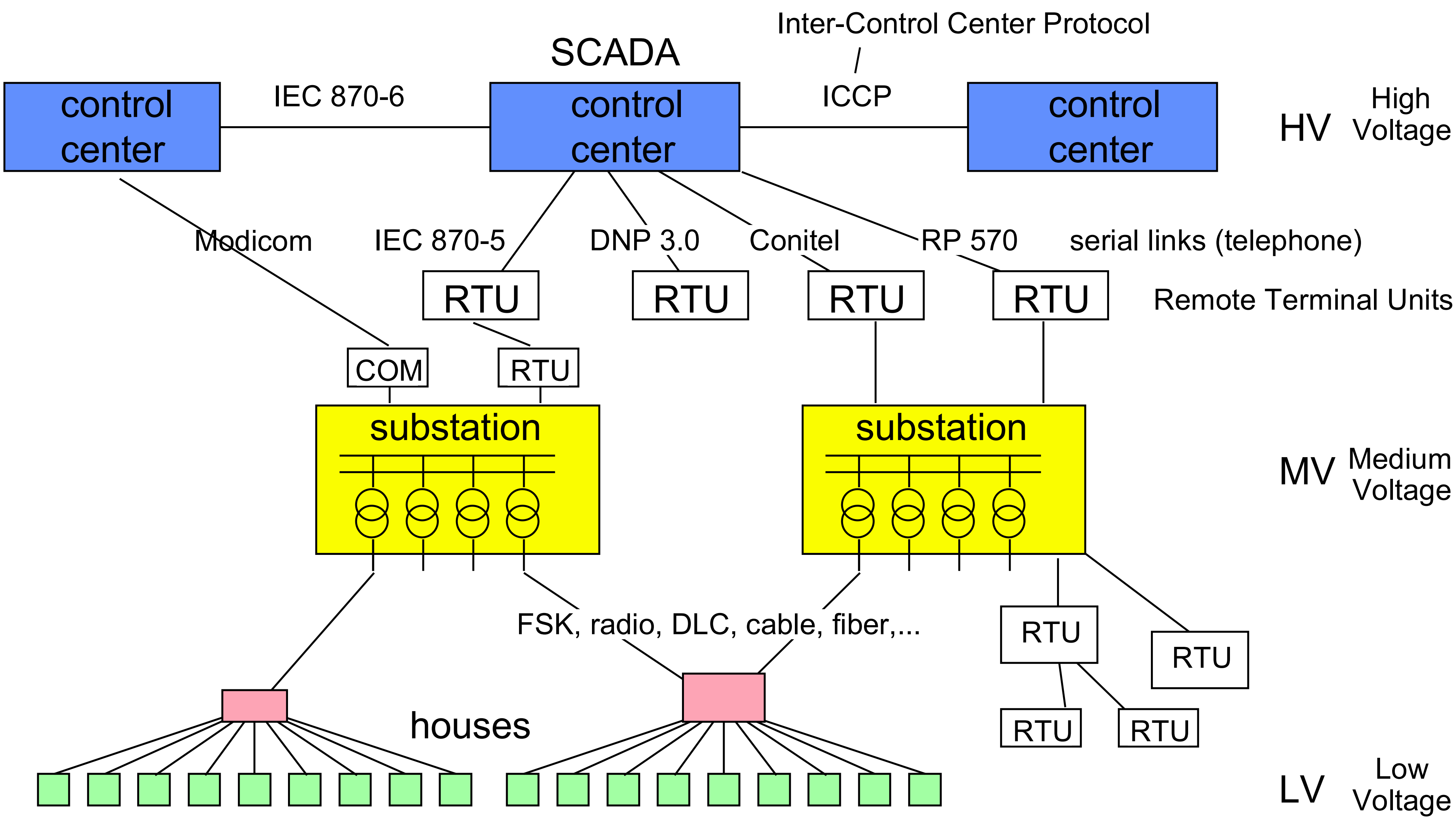
- its function in the hierarchy
- the distance it should cover
- the data density it should gather

Geographical extension of industrial plants

The field bus suits the physical extension of the plant

1 km .. 1000 km	**Transmission & Distribution**	Control and supervision of large distribution networks: • water - gas - oil - electricity - ...
1 km .. 5 km	**Power Generation**	Out of primary energy sources: • waterfalls - coal - gas - oil - nuclear - solar - ...
50 m .. 3 km	**Industrial Plants**	Manufacturing and transformation plants: • cement works - steel works - food silos - printing - paper pulp processing - glass plants - harbors - ...
500m .. 2 km	**Building Automation**	• energy - air conditioning - fire - intrusion - repair - ...
1 m .. 1 km	**Manufacturing**	flexible manufacturing cells - robots
1 m .. 800 m	**Vehicles**	• locomotives - trains - streetcars - trolley buses - vans - buses - cars - airplanes - spacecraft - ...

Networking busses: Electricity Network Control



low speed, long distance communication, may use power lines or telephone modems.
Problem: diversity of protocols, data format, semantics...

Substation (air isolated)

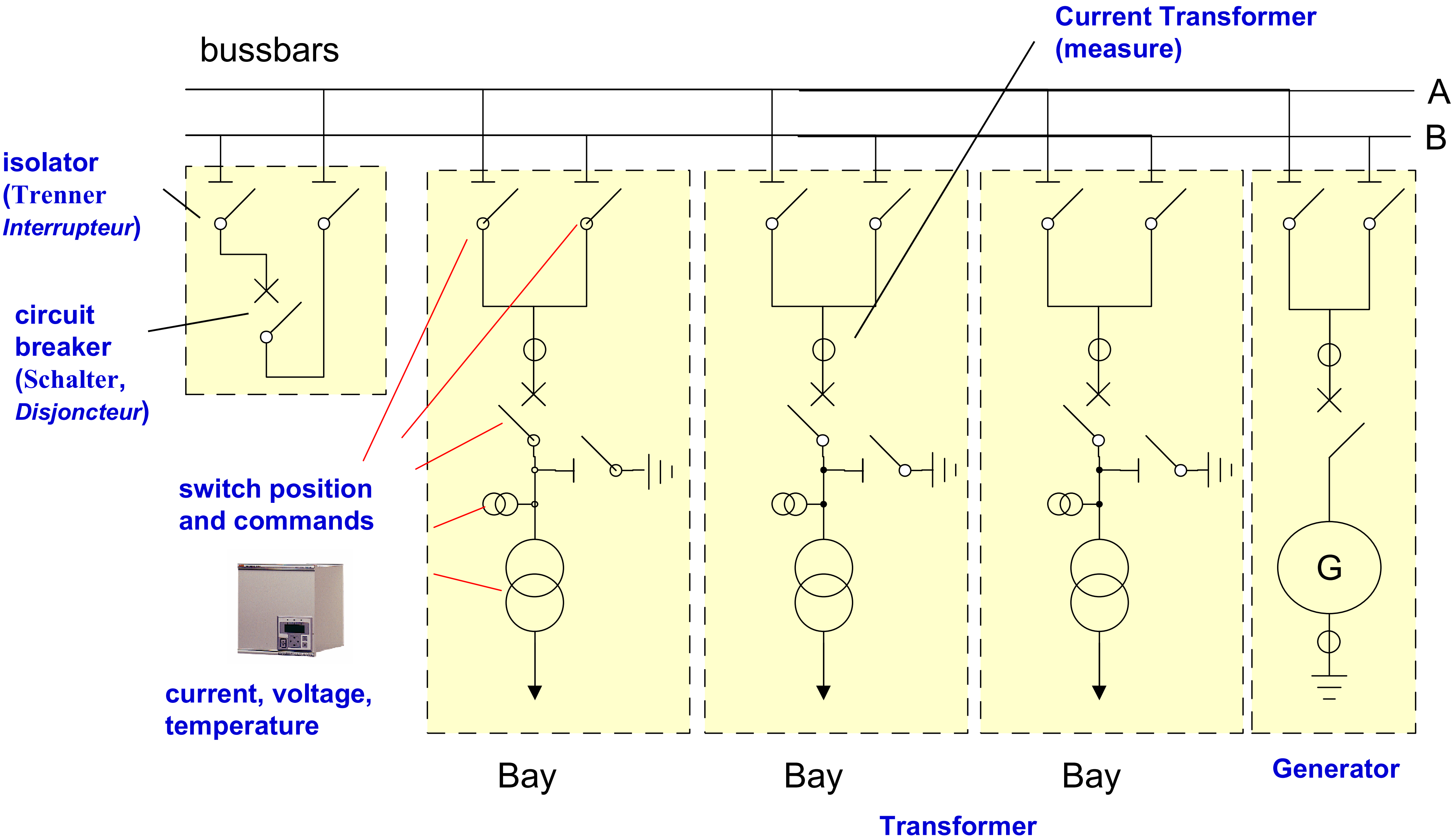
Substations (indoor)

Gas Isolated high voltage

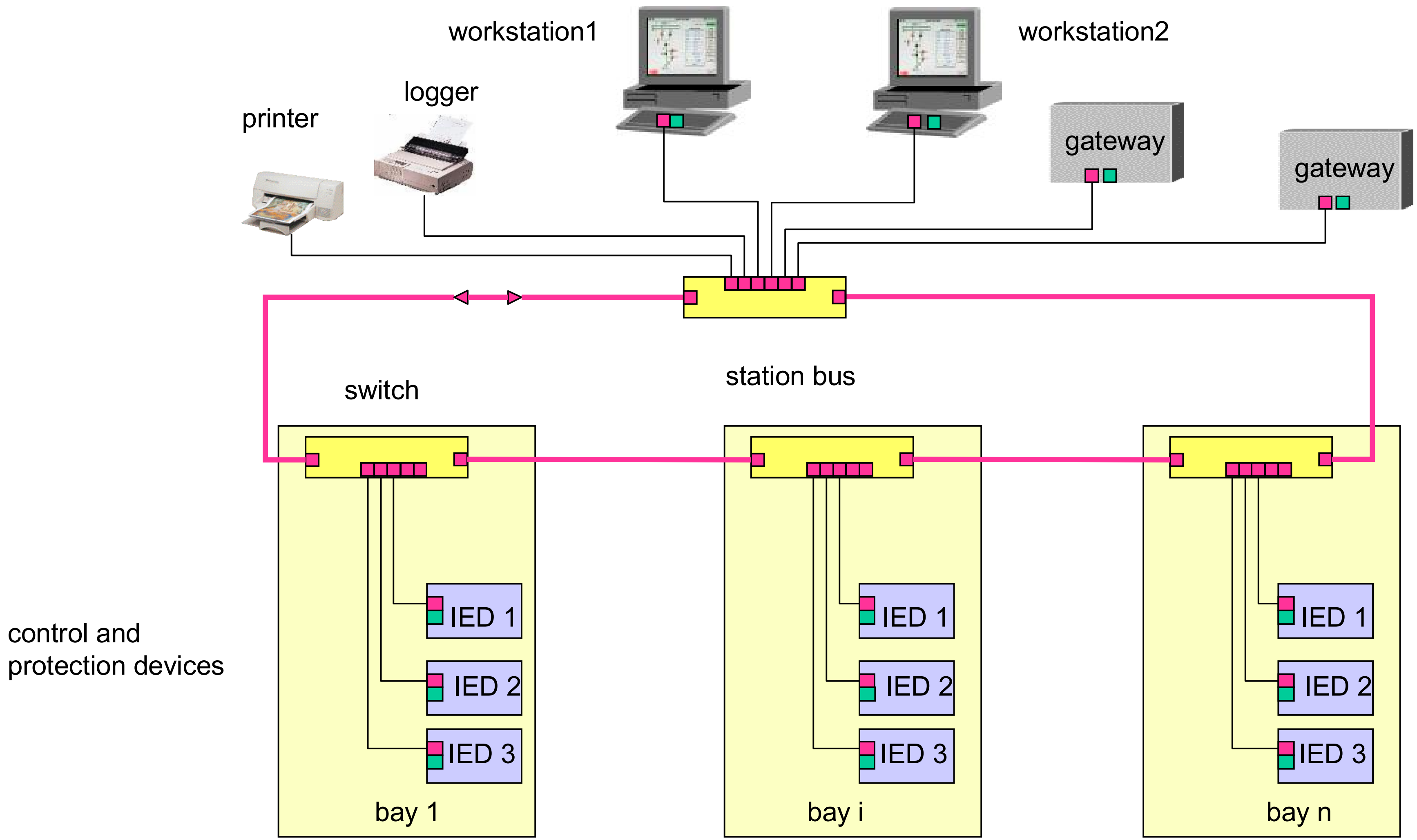
medium voltage

consist of bays (*départs*, Abgang), interconnected by a buss bar (*barre*, Sammelschiene)

Substation electrical busses



Substation data busses



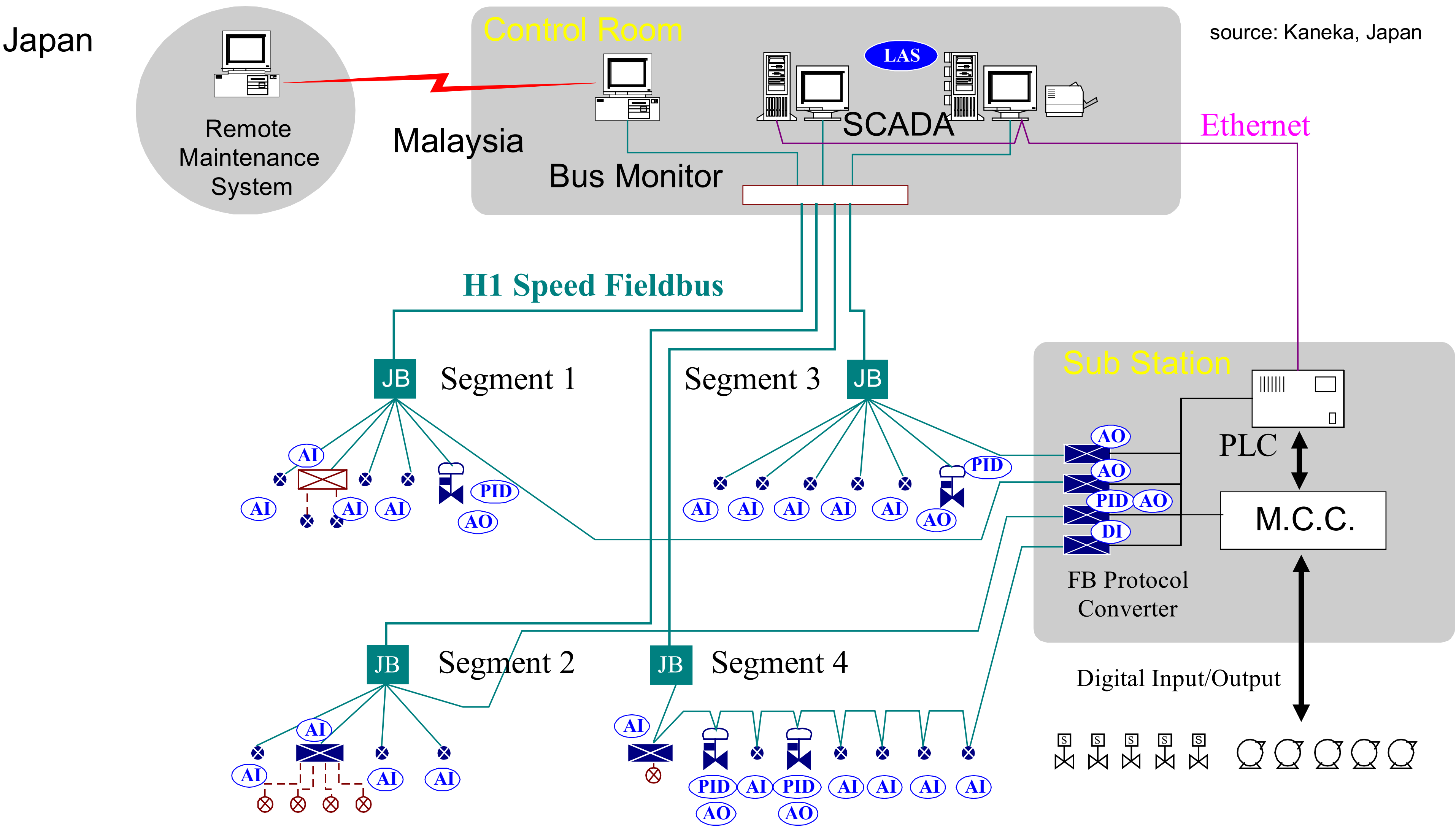
the structure of the data busses reflects the substation structure

Fieldbus Application: wastewater treatment

Pumps, gates, valves, motors, water level sensors, flow meters, temperature sensors, gas meters (CH_4), generators, … are spread over an area of several km^2
Some parts of the plant have explosive atmosphere.

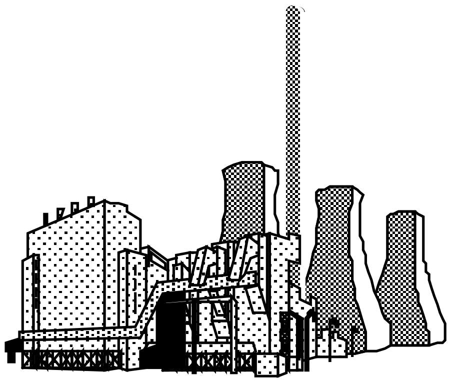
Wiring is traditionally 4..20 mA, resulting in long threads of cable (several 100 km).

Process Industry Application: Water treatment plant



Numerous analog inputs (AI),
low speed (37 kbit/s) segments merged to 1 Mbit/s links.

Data density (Example: Power Plants)



Acceleration limiter and prime mover: 1 kbit in 5 ms

Burner Control: 2 kbit in 10 ms

per each 30 m of plant: 200 kbit/s

Fast controllers require at least 16 Mbit/s over distances of 2 m

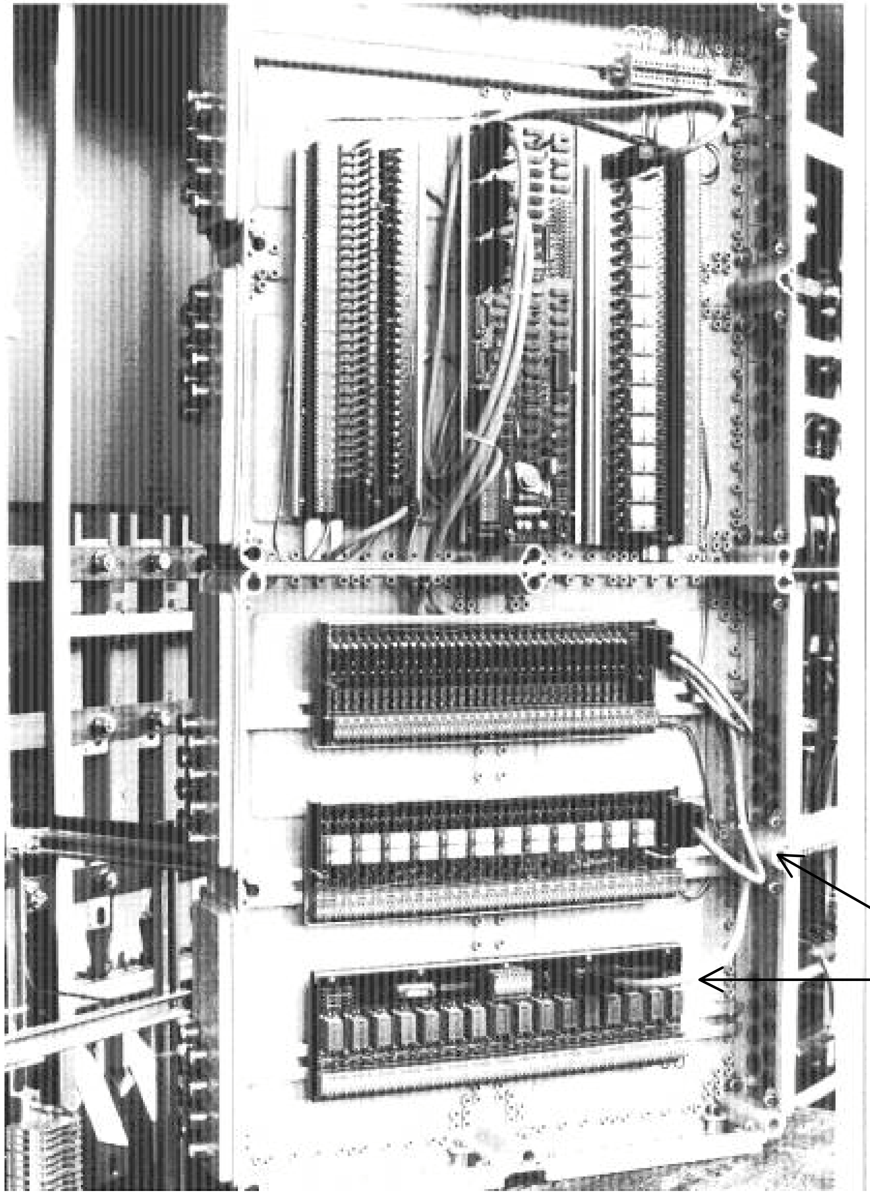
Data are transmitted from the periphery or from fast controllers to higher level, but slower links to the control level through field busses over distances of 1-2 km.

The control stations gather data at rates of about 200 kbit/s over distances of 30 m.

The control room computers are interconnected by a bus of at least 10 Mbit/s, over distances of several 100 m.

Planning of a field bus requires to estimate the data density per unit of length (or surface) and the requirements in response time and throughput over each link.

Distributed peripherals

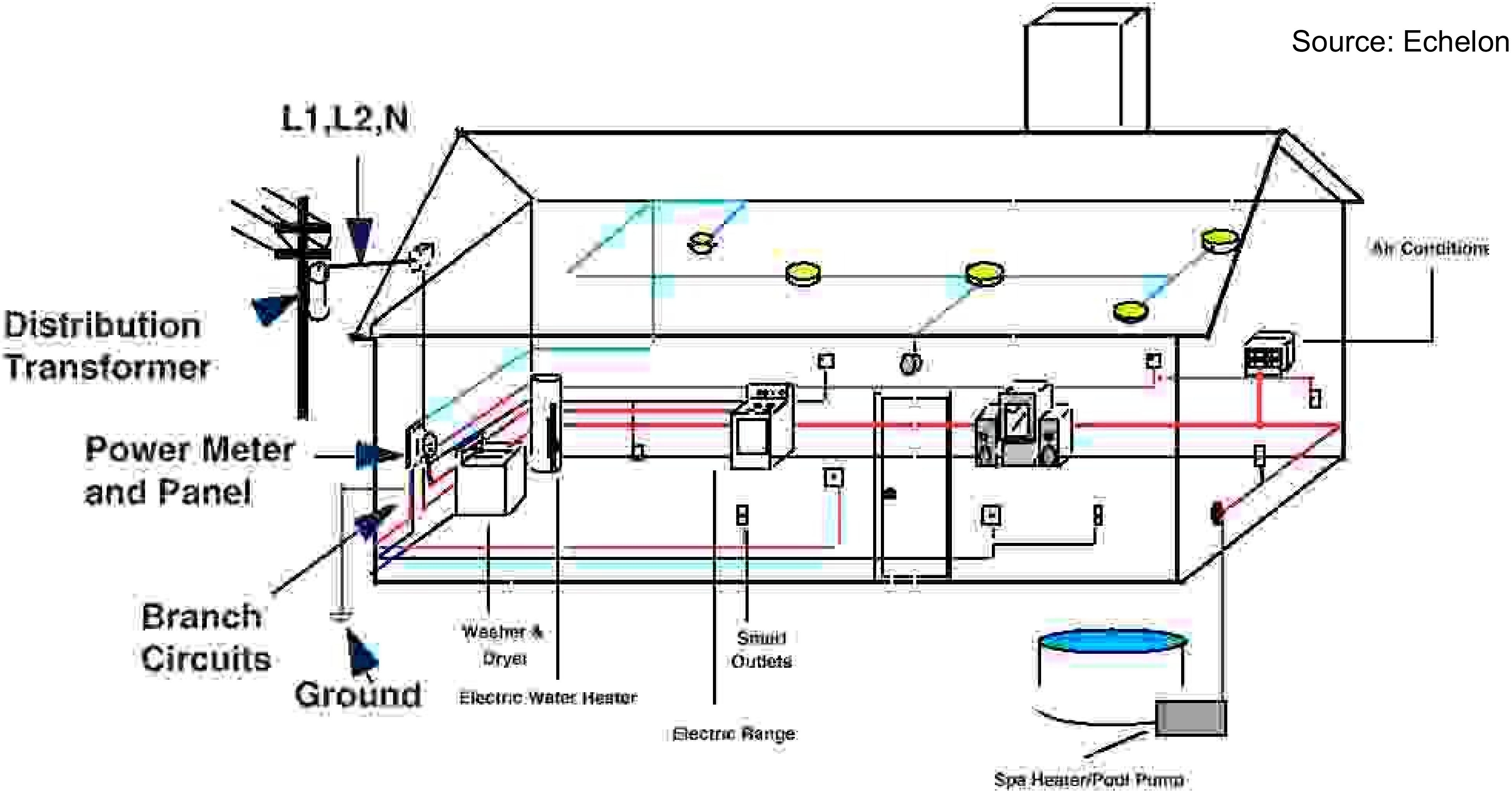


Many field busses are just extensions of the PLC's Inputs and Outputs,
field devices are data concentrators.
Devices are only visible to the PLC that controls them

relays and fuses

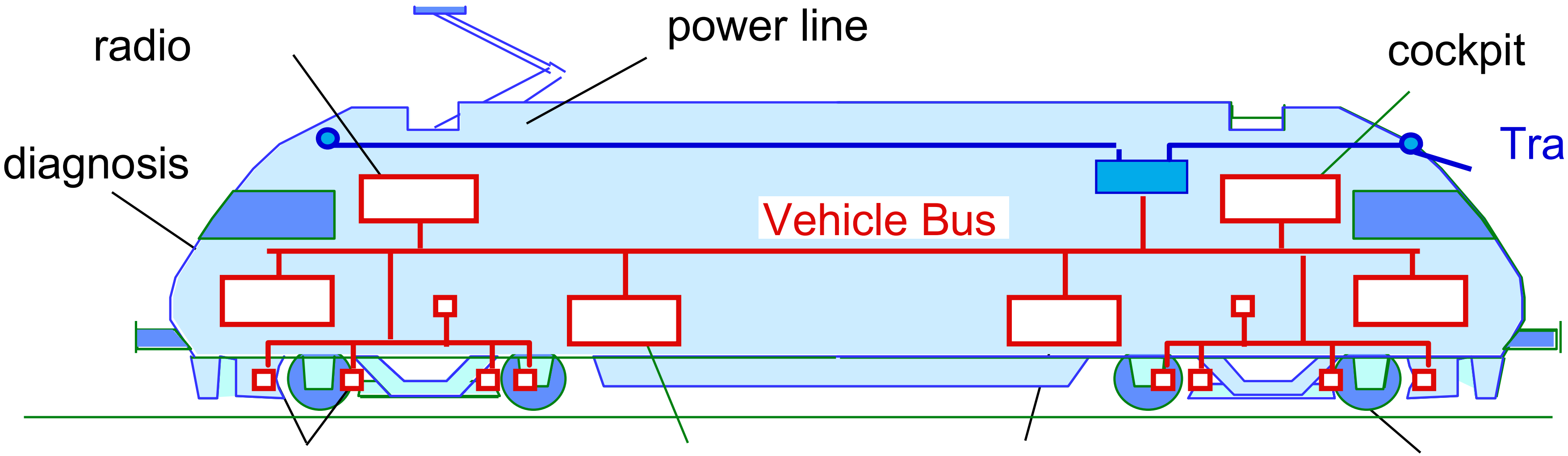
Application: Building Automation

Source: Echelon



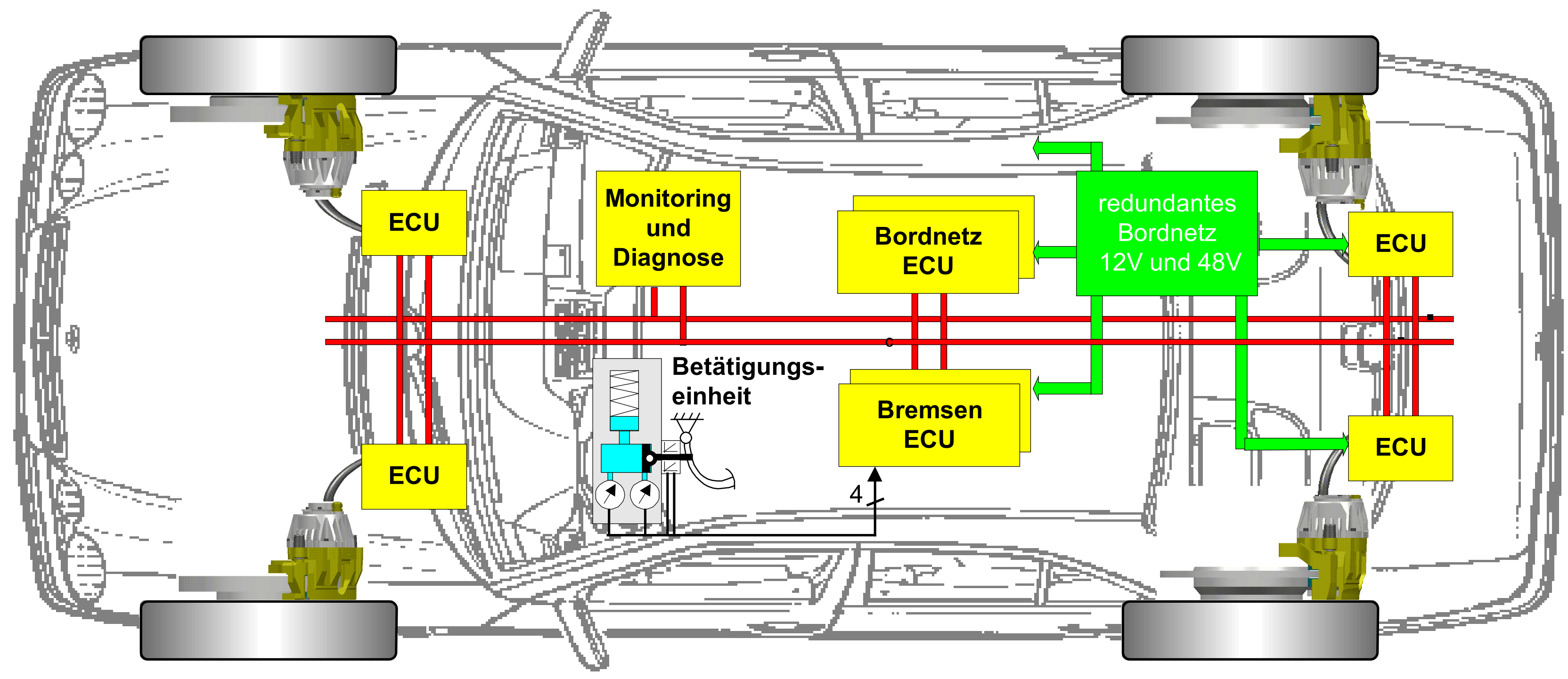
low cost, low data rate (78 kbit/s), may use power lines (10 kbit/s)

Application: Field bus in locomotives



data rate	1.5 Mbit/second
delay	1 ms (16 ms for skip/slip control)
medium	twisted wire pair, optical fibers (EM disturbances)
number of stations	up to 255 programmable stations, 4096 simple I/O
integrity	very high (signaling tasks)
cost	engineering costs dominate

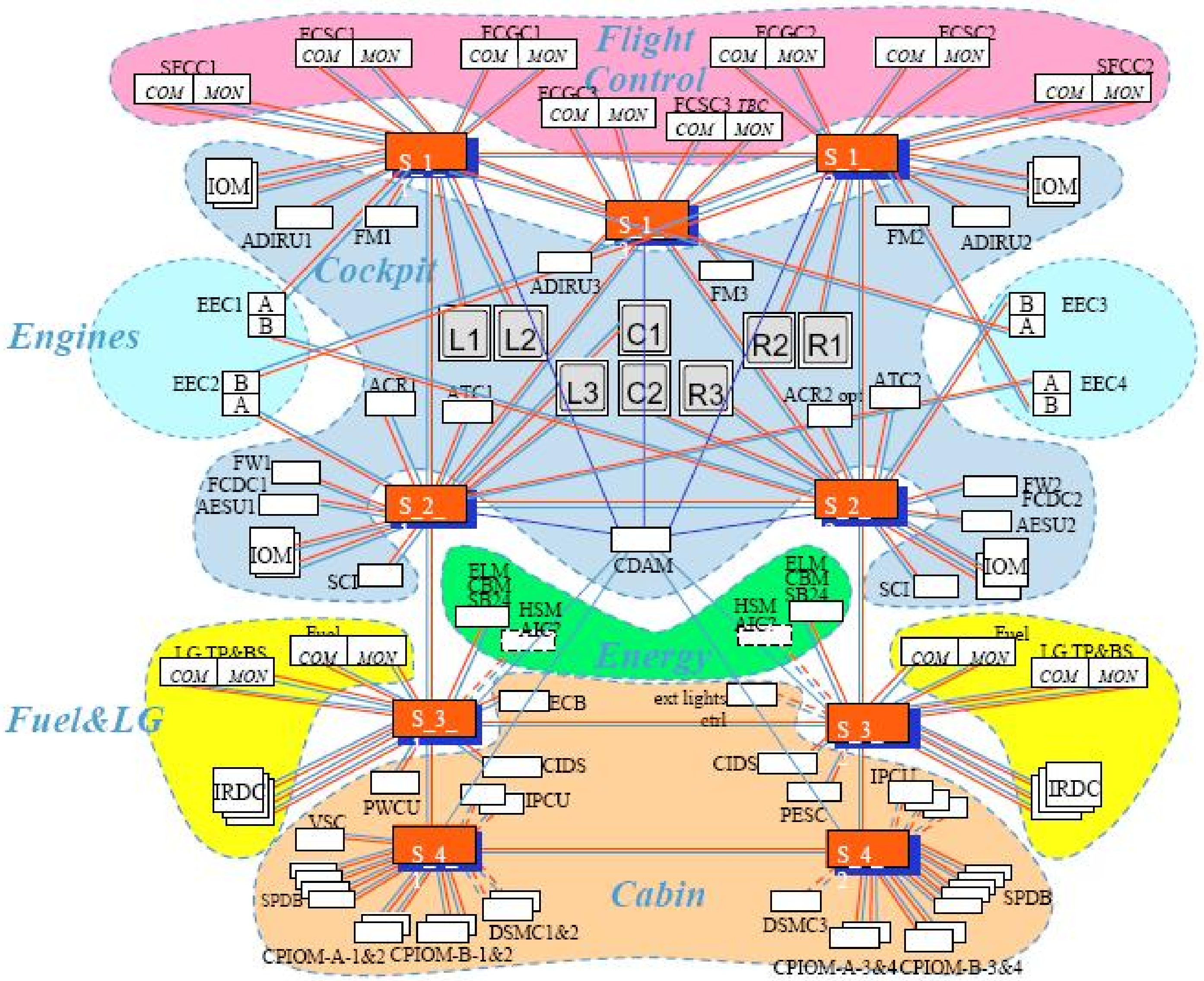
Application: automobile



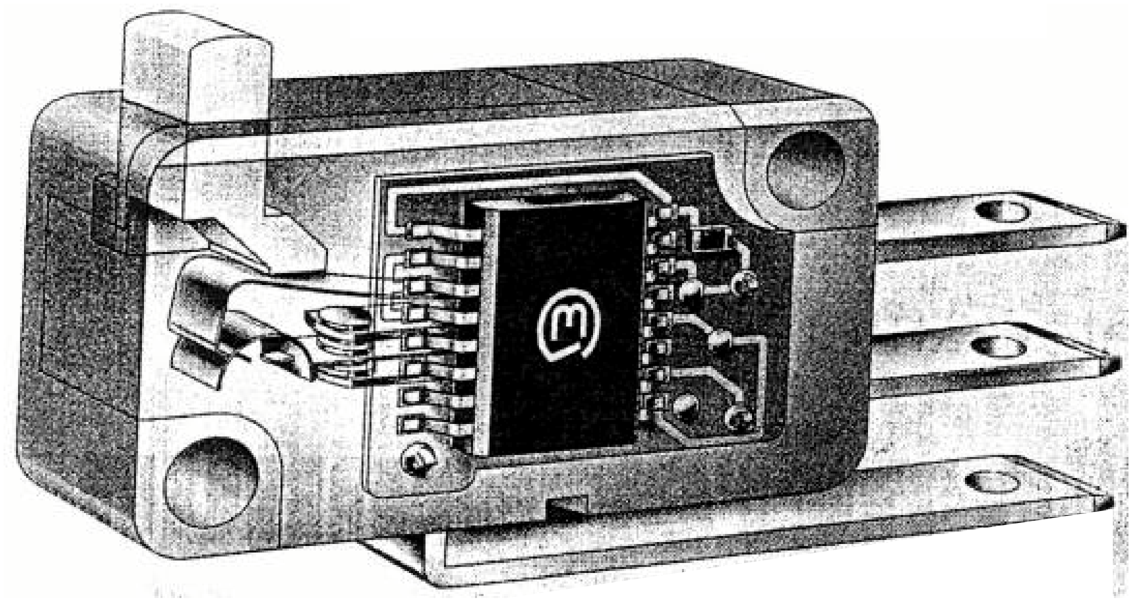
- 8 nodes
- 4 electromechanical wheel brakes
- 2 redundant Vehicle Control Unit
- Pedal simulator
- Fault-tolerant 2-voltage on-board power supply
- Diagnostic System

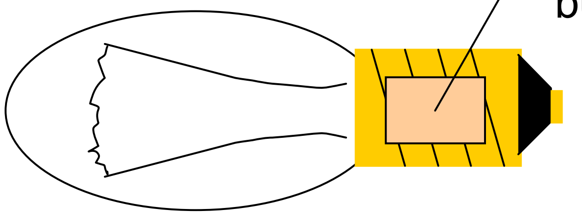
DAIMLERBENZ
AKTIENGESELLSCHAFT

Application: Avionics (Airbus 380)



The ultimate sensor bus





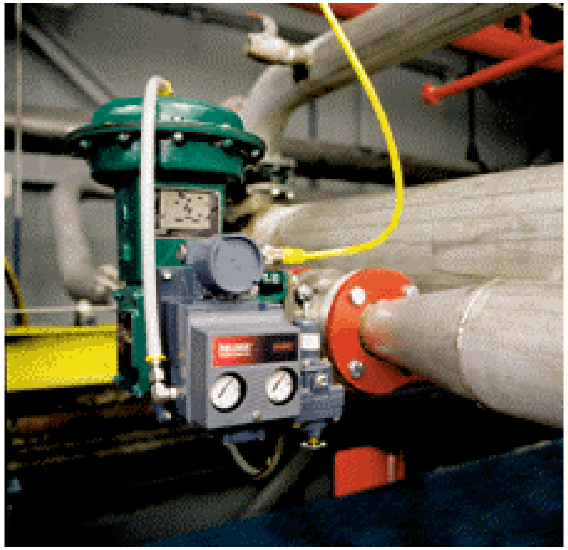
requires integration of power electronics and communication at very low cost.

Assessment

- What is a field bus ?
- How does a field bus supports modularity ?
- What is the difference between a sensor bus and a process bus ?
- Which advantages are expected from a field bus ?

EPFL

Industrial Automation
Automation Industrielle
Industrielle Automation



3 Industrial Communication Systems

Field Bus Operation
3.2 *Bus de terrain: mode de travail*
Feldbus: Arbeitsweise

Prof. Dr. H. Kirrmann
ABB Research Center, Baden, Switzerland

2005 April, HK

Fieldbus - Operation

3.1 Field bus types
- Classes
- Physical layer

3.2 Field bus operation
- Data distribution
- Cyclic Operation
- Event Driven Operation
- Real-time communication model
- Networking

3.3 Standard field busses

Objective of the field bus

Distribute to all interested parties process variables, consisting of:

•accurate process value and units

•source identification: requires a naming scheme

•quality indication: good, bad, substituted,

•time indication: how long ago was the value produced

•(description)

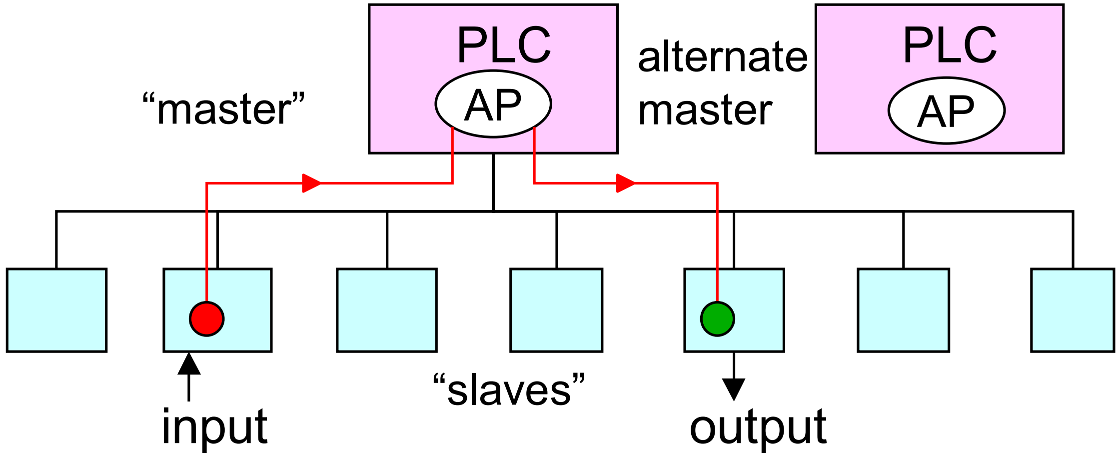
source	value	quality	time	description

Master or peer-to-peer communication

communication in a control system is evolving from hierarchical to distributed

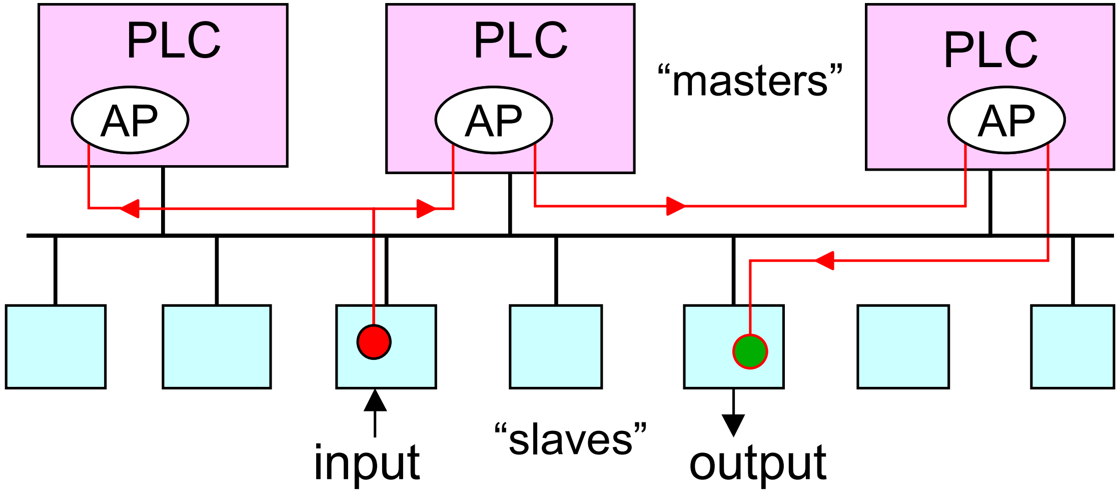
central master: hierarchical

all traffic passes by the master (PLC);
adding an alternate master is difficult
(it must be both master and slave)



peer-to-peer: distributed

PLCs may exchange data,
share inputs and outputs
allows redundancy
and "distributed intelligence"
devices talk directly to each other

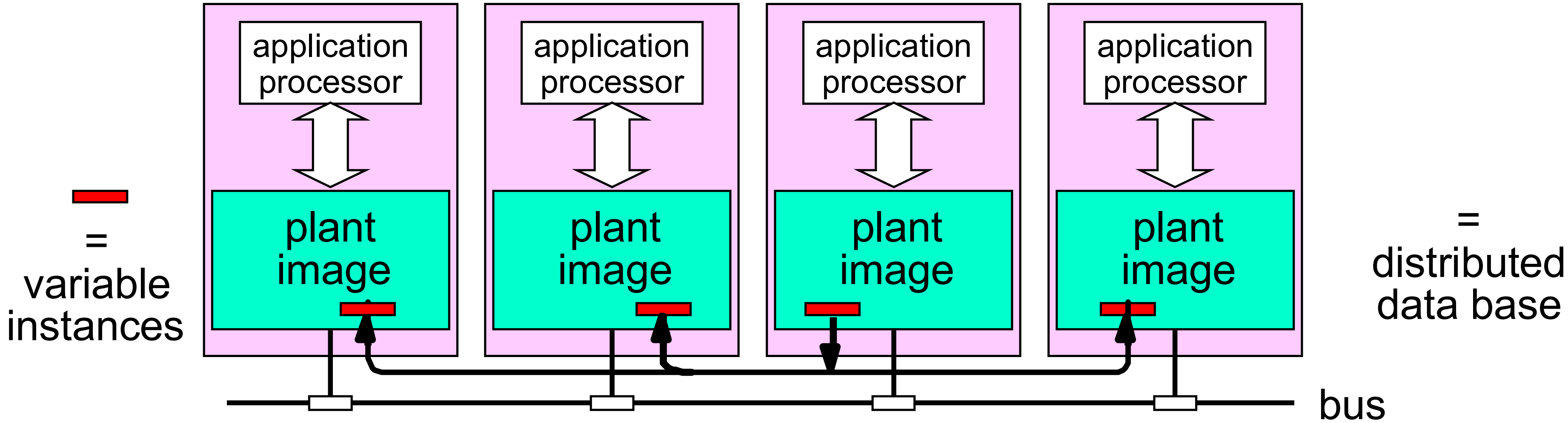


separate bus master from application master !

Broadcasts

A variable is read on the average in 1..3 different places

Broadcasting messages identified by their source (or contents) increases efficiency.



Each device is subscribed as source or as sink for a number of process variables

Only one device may be source of a certain process data (otherwise, collision).

The bus refreshes the plant image in the background, it becomes an on-line database

The replicated traffic memories can be considered as "caches" of the plant state (similar to caches in a multiprocessor system), representing part of the plant image.

Each station snoops the bus and reads the variables it is interested in.

Data format

source	value	quality	time

In principle, the bus could transmit the process variable in clear text, possibly using XML.

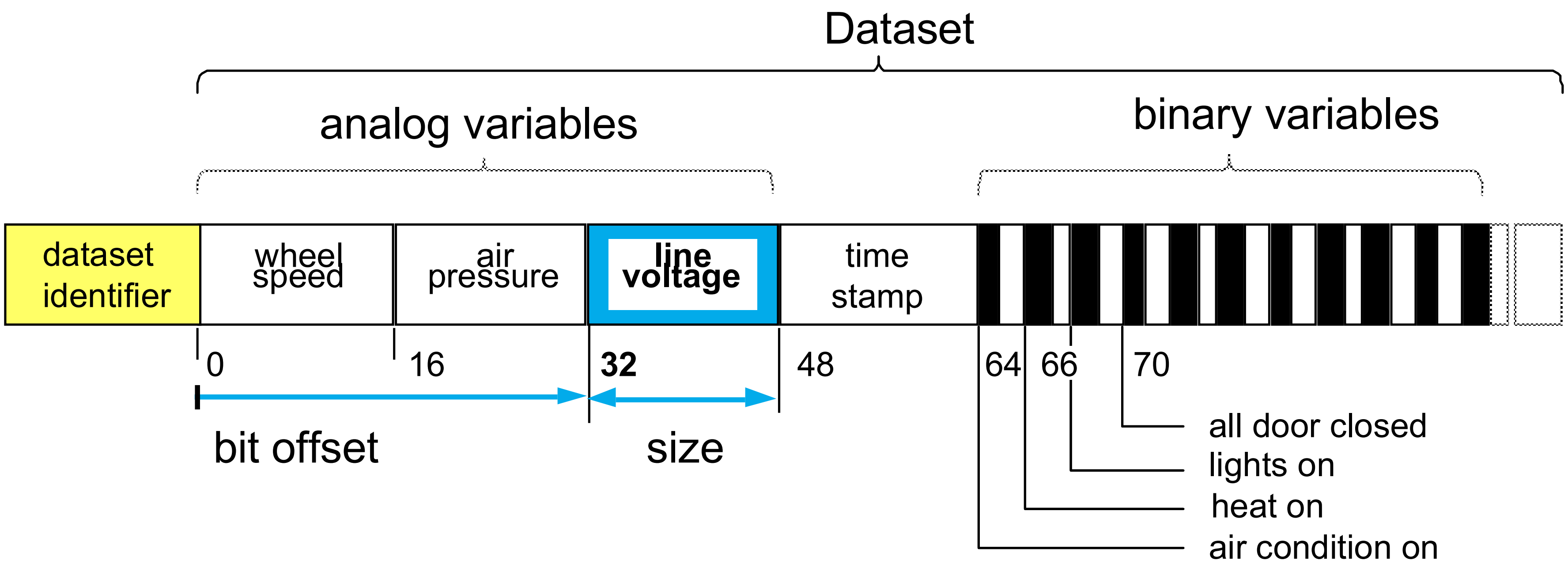
However, this is quite expansive and only considered when the communication network offers some 100 Mbit/s and a powerful processor is available to parse the message

More compact ways such as ASN.1 have been used in the past with 10 Mbit/s Ethernet.

Field busses are still slow (1Mbit/s ..12 Mbits/s) and therefore more compact encodings are used.

Datasets

Field busses devices had a low data rate and transmit over and over the same variables.
It is economical to group variables of a device in the same frame as a dataset.
A dataset is treated as a whole for communication and access.
A variable is identified within a dataset by its offset and its size
Variables may be of different types, types can be mixed.

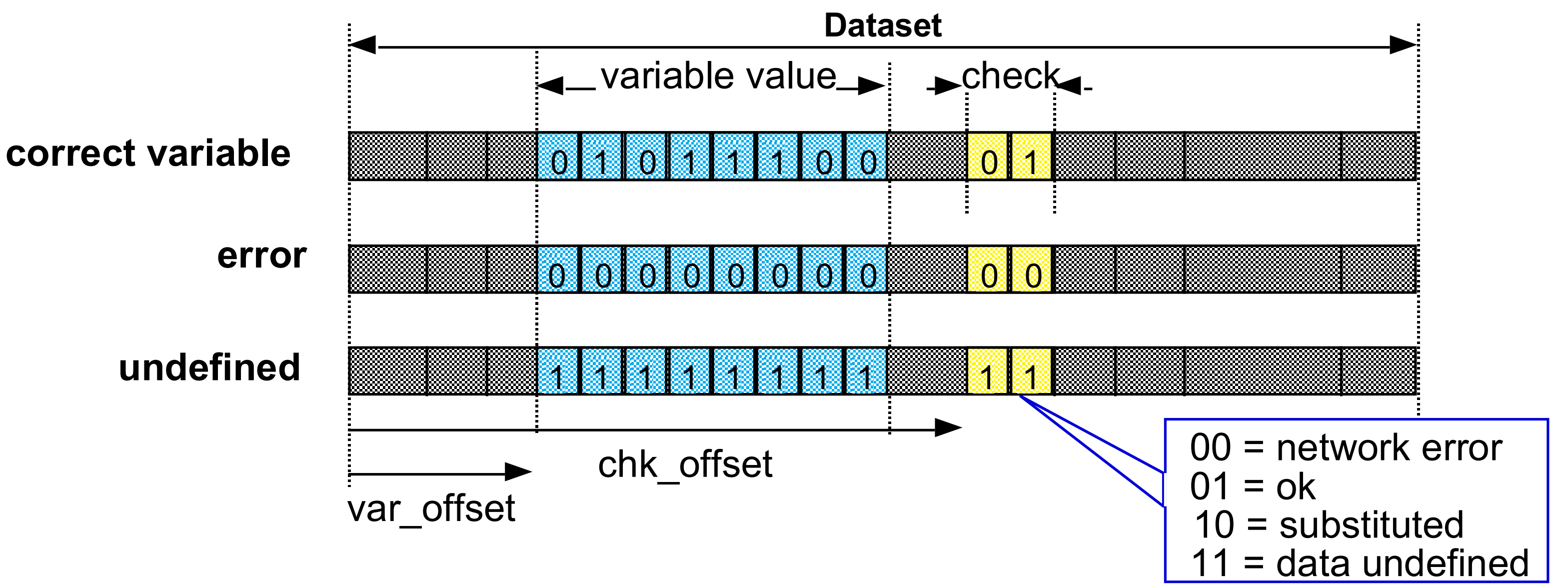


Dataset extension and quality

To allow later extension, room is left in the datasets for additional variables.
Since the type of these future data is unknown, unused fields are filled with '1".

To signal that a variable is invalid, the producer overwrites the variable with "0".

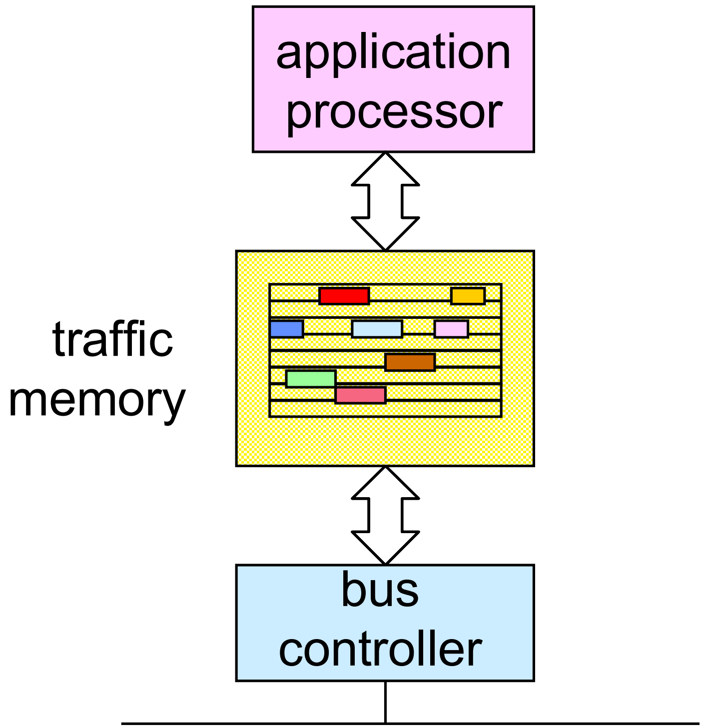
Since both an "all 1" and an "all 0" word can be a meaningful combination, each variable can be supervised by a check variable, of type ANTIVALENT2:



A variable and its check variable are treated indivisibly when reading or writing
The check variable may be located anywhere in the same data set.

Decoupling Application and Bus traffic

decoupled (asynchronous):

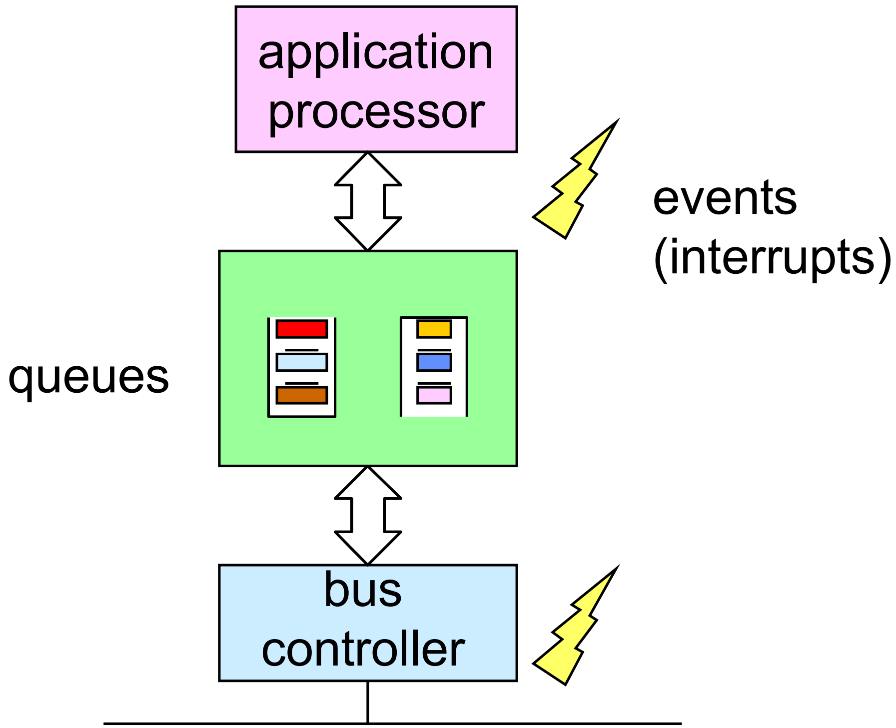


sending: application writes data into memory

receiving: application reads data from memory

the bus controller decides when to transmit
bus and application are not synchronized

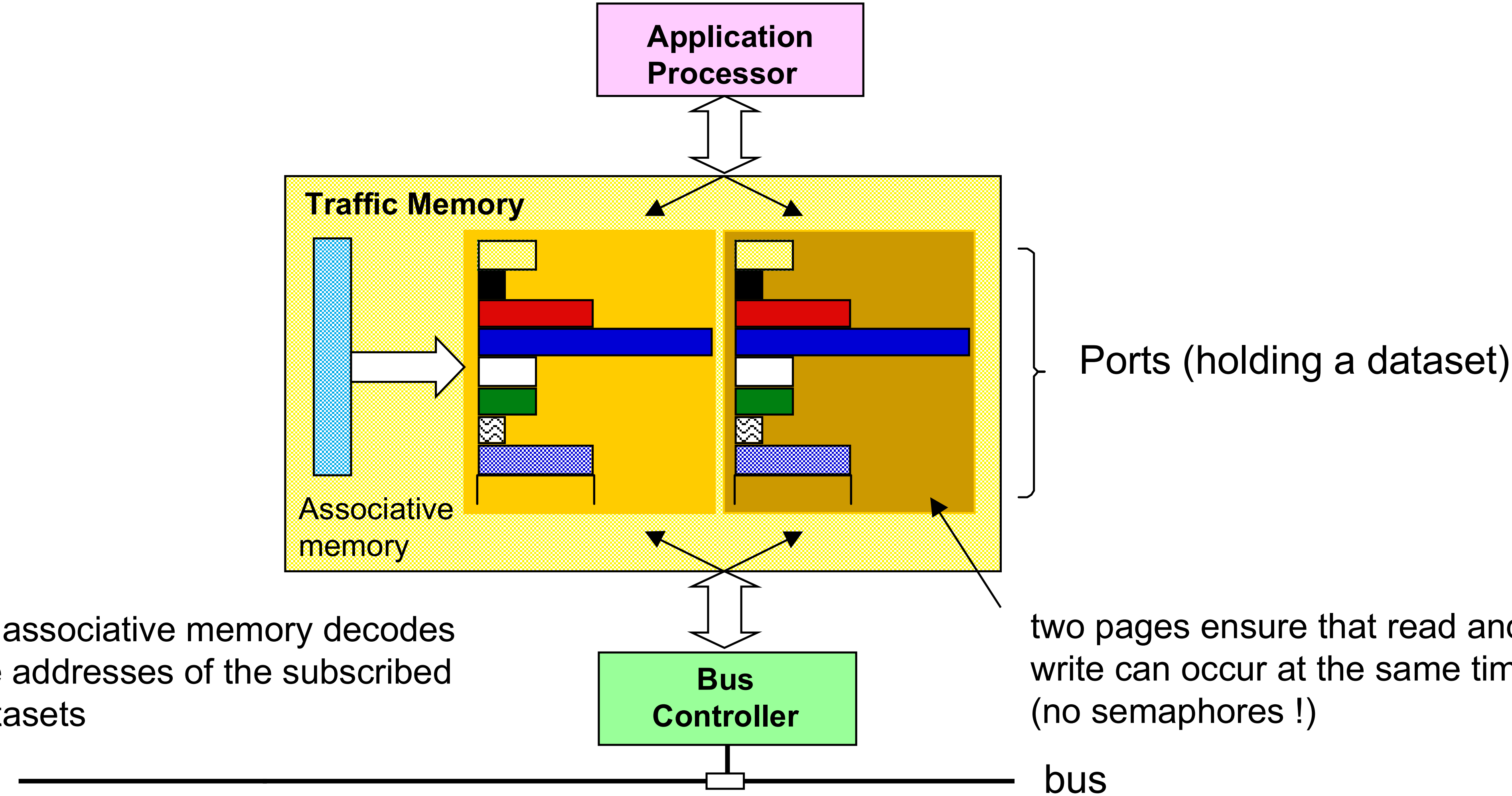
coupled (event-driven):



sending: application inserts data into queue
and triggers transmission,
bus controller fetches data from queue
receiving: bus controller inserts data into queue
and interrupts application to fetch them,
application retrieves data

Traffic Memory: implementation

Bus and Application are (de)coupled by a shared memory, the *Traffic Memory,* where process variables are directly accessible to the application.



an associative memory decodes the addresses of the subscribed datasets

two pages ensure that read and write can occur at the same time (no semaphores !)

Freshness supervision

It is necessary to check that the data in the traffic memory is still up-to-date, independently of a time-stamp (simple devices do not have time-stamping)

Applications tolerate an occasional loss of data, but no stale data.

To protect the application from using obsolete data, each Port in the traffic memory has a freshness counter.

This counter is reset by writing to that port. It is incremented regularly, either by the application processor or by the bus controller.

The application should always read the value of the counter before using the port data and compare it with its tolerance level.

The freshness supervision is evaluated by each reader independently, some readers may be more tolerant than others.

Bus error interrupts in case of severe disturbances are not directed to the application, but to the device management.

Process Variable Interface

Access of the application to variables in a traffic memory is very easy:

ap_put (variable_name, variable value)

ap_get (variable_name, variable value, variable_status, variable_freshness)

Rather than fetch and store individual variables, access is done by clusters (predefined groups of variables):

ap_put_cluster (cluster_name)

ap_get (cluster_name)

The cluster is a table containing the names and values of several variables.

The clusters can correspond to "segments" in the function block programming.

Note: Usually, only one variable is allowed to raise an interrupt when received: the one carrying the current time (sent by the common clock)

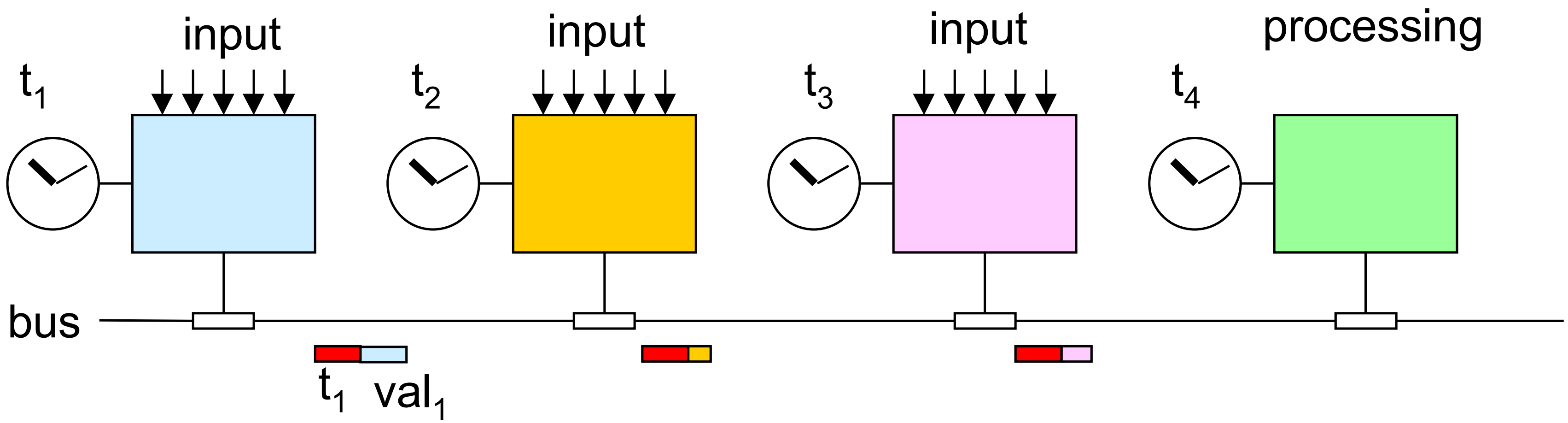
Time-stamping and clock synchronisation

In many applications, such as disturbance logging and sequence-of-events, the exact sampling time of a variable must be transmitted together with its value.

To this purpose, the devices are equipped with a clock that records the creation date of the value (not the transmission time).

To reconstruct events coming from several devices, clocks must be synchronized. considering transmission delays over the field bus (and in repeaters,....)

A field bus provides means to synchronize clocks in spite of propagation delays and failure of individual nodes. Protocols such as IEEE 1588 can be used.



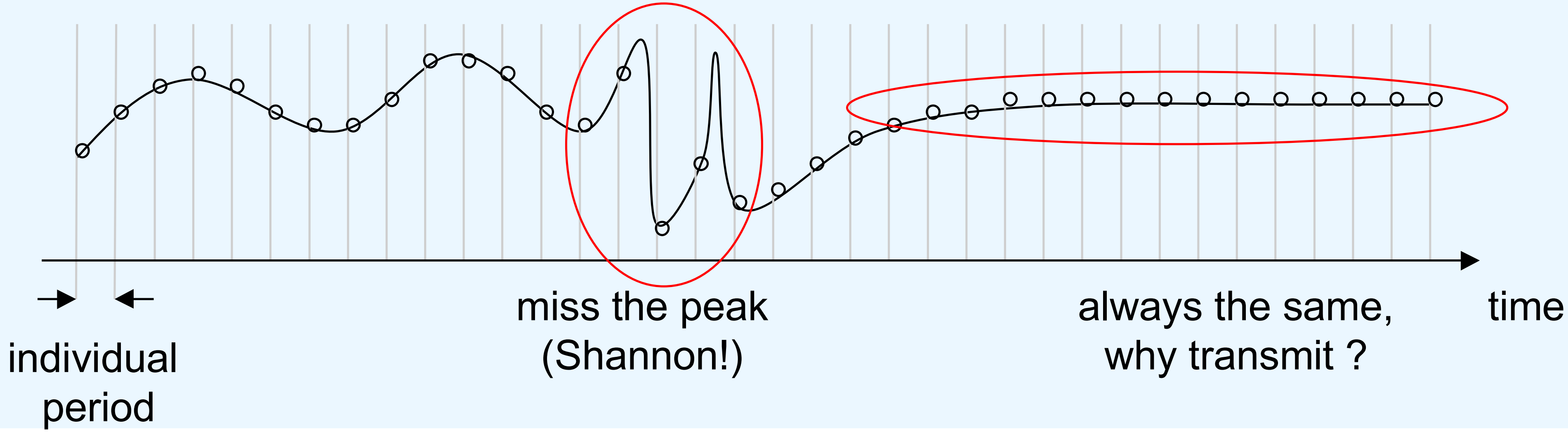
Transmission principle

The previous operation modes made no assumption, how data are transmitted.

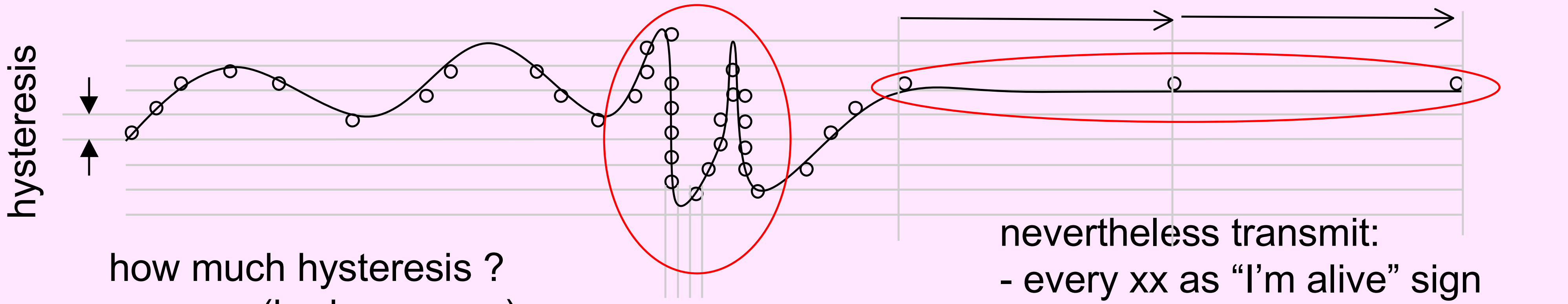
The actual network can transmit data
cyclically (time-driven) or
on demand (event-driven),
or a combination of both.

Cyclic and Event-Driven transmission

cyclic: send value every xx milliseconds



event-driven: send when value change by more than x% of range



how much hysteresis ?
- coarse (bad accuracy)
- fine (high frequency)

limit update
frequency !,
limit hysteresis

nevertheless transmit:
- every xx as “I’m alive” sign
- when data is internally updated
- upon quality change (failure)

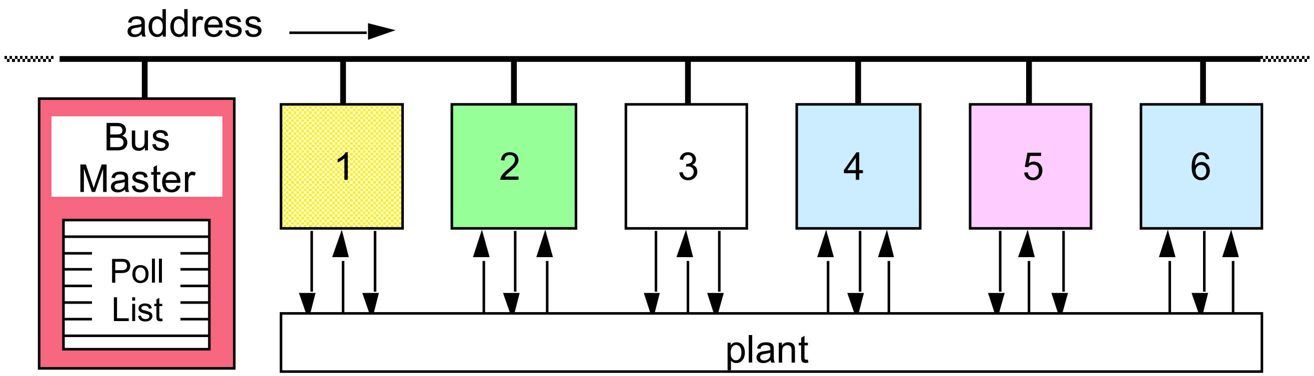
Fieldbus: Cyclic Operation mode

3.1 Field bus types
- Classes
- Physical layer

3.2 Field bus operation
- Data distribution
- Cyclic Operation
- Event Driven Operation
- Real-time communication model
- Networking

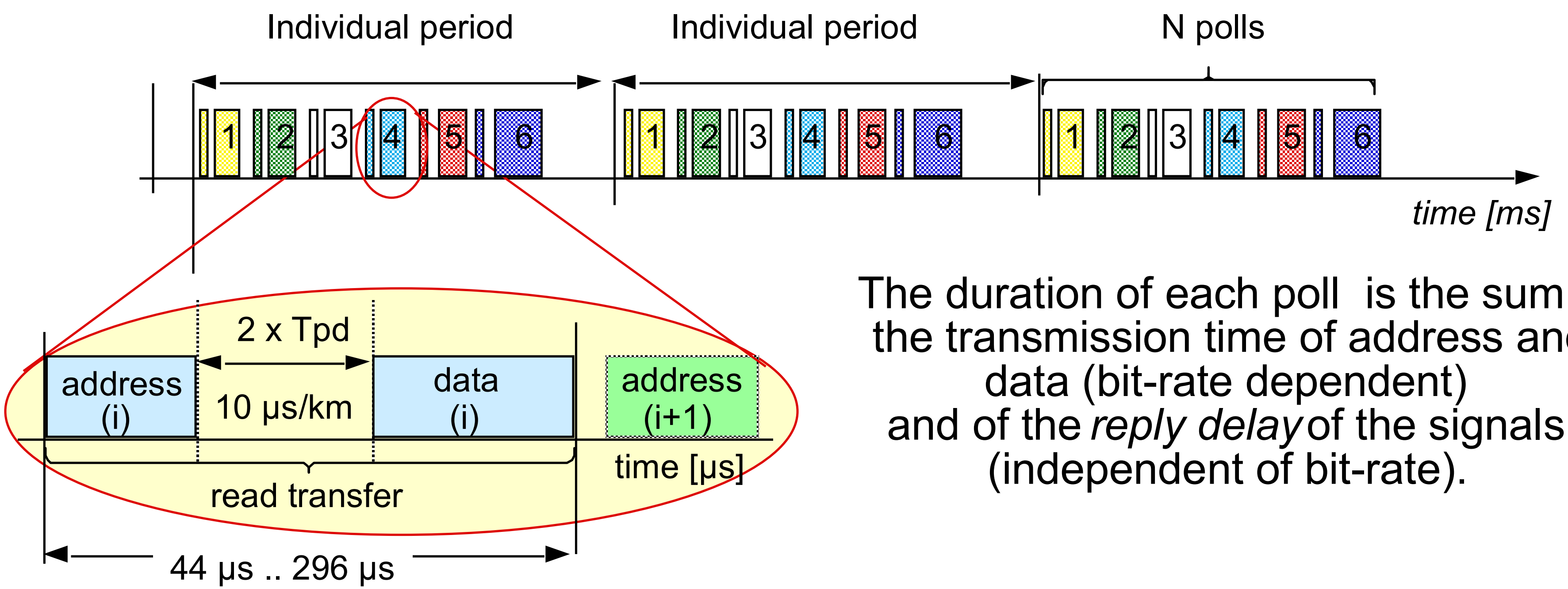
3.3 Standard field busses

Cyclic Data Transmission



devices (slaves)

The master polls the addresses in a fixed sequence, according to its poll list.



The duration of each poll is the sum of the transmission time of address and data (bit-rate dependent) and of the *reply delay* of the signals (independent of bit-rate).

Cyclic operation principle

Data are transmitted at fixed intervals, whether they changed or not.

The delivery delay (refresh rate) is deterministic and constant.

The bus is under control of a central master (or distributed time-triggered algorithm).

No explicit error recovery needed since a fresh value will be transmitted in the next cycle.

Only states may be transmitted, not state changes.

Cycle time is limited by the product of the number of data transmitted by the duration of each poll (e.g. 100 µs / point x 100 points => 10 ms)

To keep a low poll time, only small data items may be transmitted (< 256 bits)

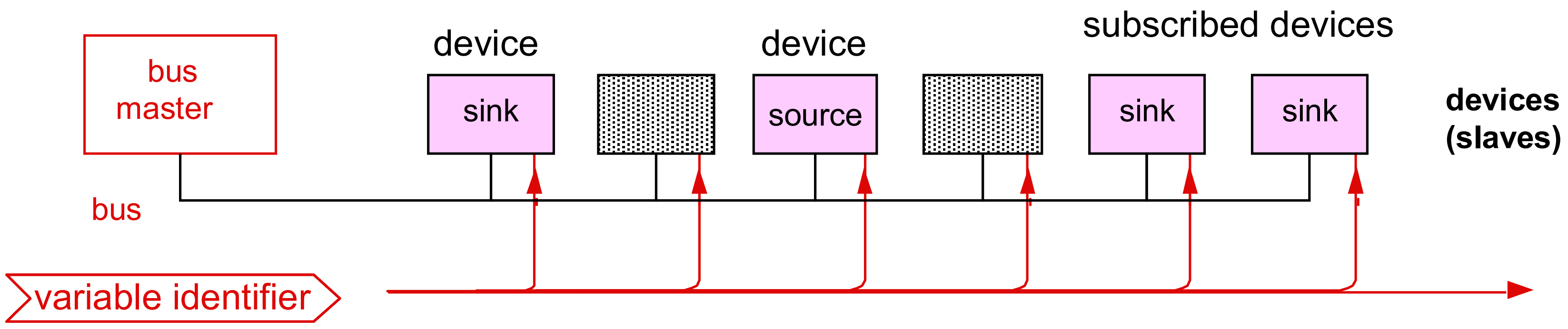
The bus capacity must be configured beforehand.
Determinism gets lost if the cycles are modified at run-time.

Cyclic operation is used to transmit the state variables of the process.
These are called Process Data (or Periodic Data)

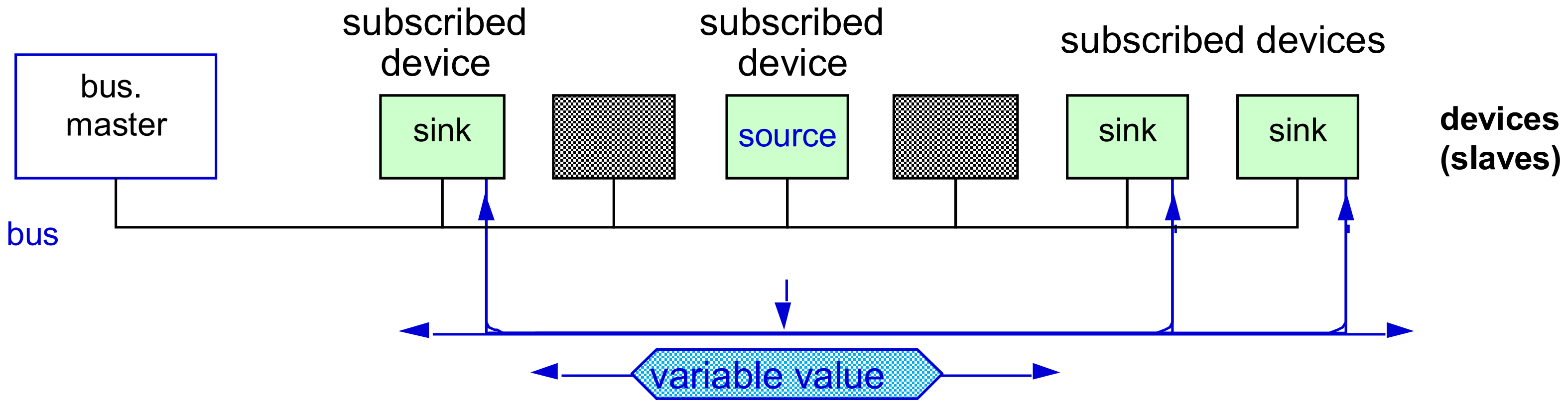
Source-Addressed Broadcast

Process Data are transmitted by *source-addressed broadcast.*

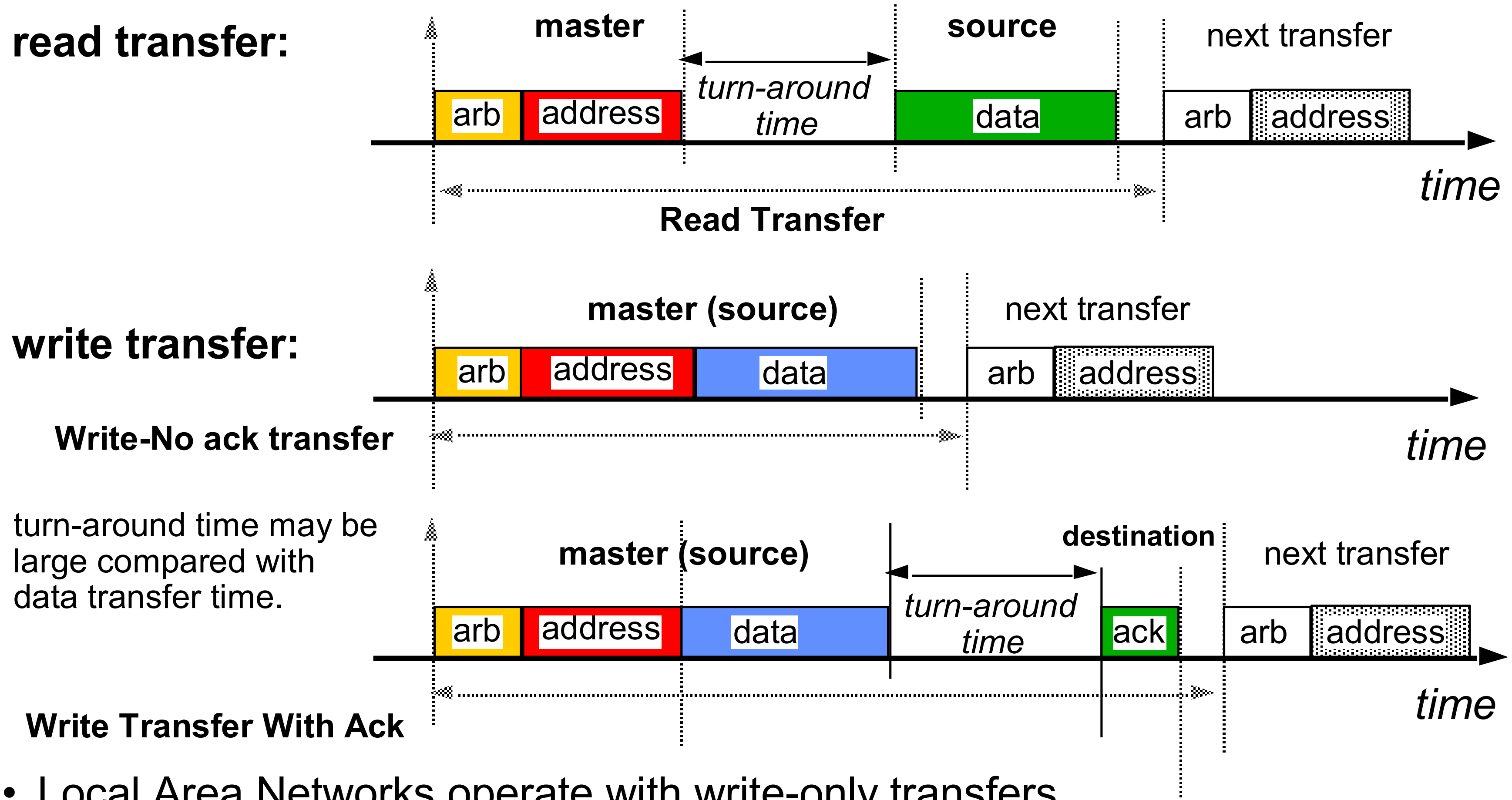
Phase1: The bus master broadcasts the identifier of a variable to be transmitted:



Phase 2: The device that sources that variable responds with a slave frame containing the value, all devices subscribed as sink receive that frame.



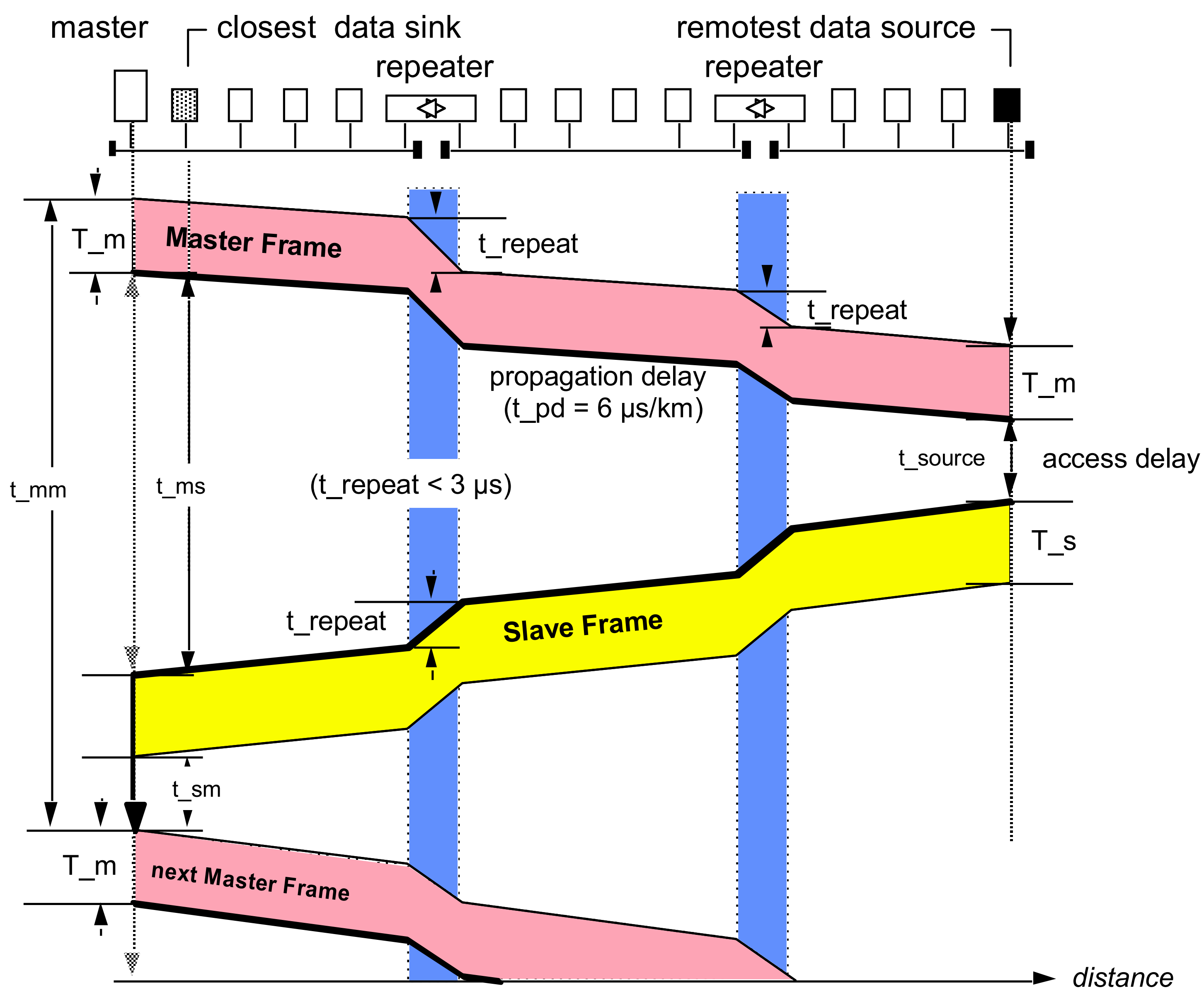
Read And Write Transfers



- Local Area Networks operate with write-only transfers.
  Their link layer or transport layer provides acknowledgements by another write-only transfer
- Parallel busses use read and write-ack transfers
- Most field busses operate with read cycles only.

Round-tip Delay

The round-trip delay limits the extension of a read-only bus



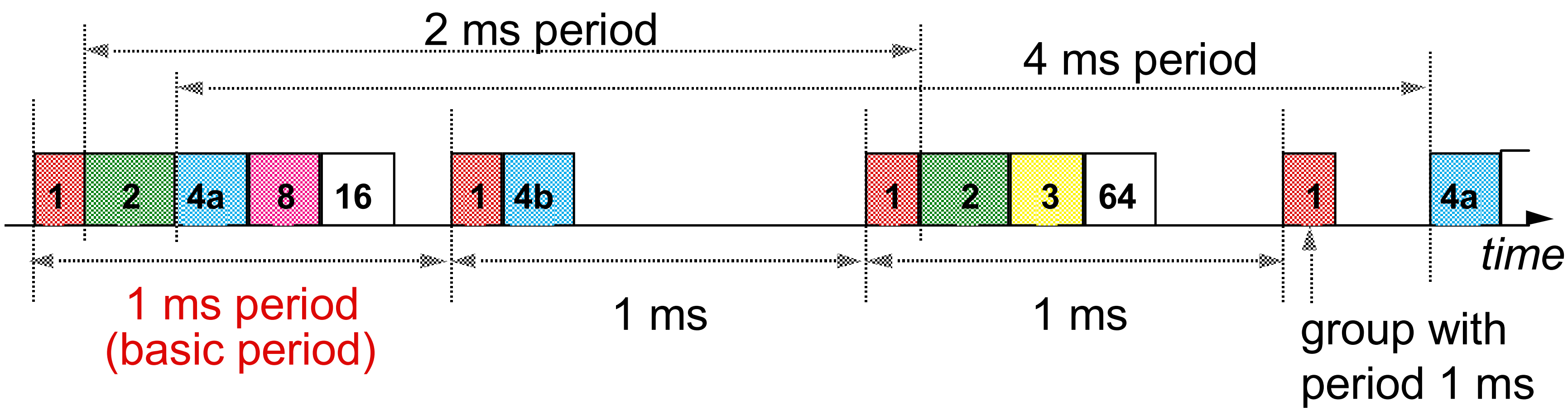
Optimizing Cyclic Operation

Cyclic operation uses a fixed portion of the bus's time

The poll period increases with the number of polled items

The response time slows down accordingly

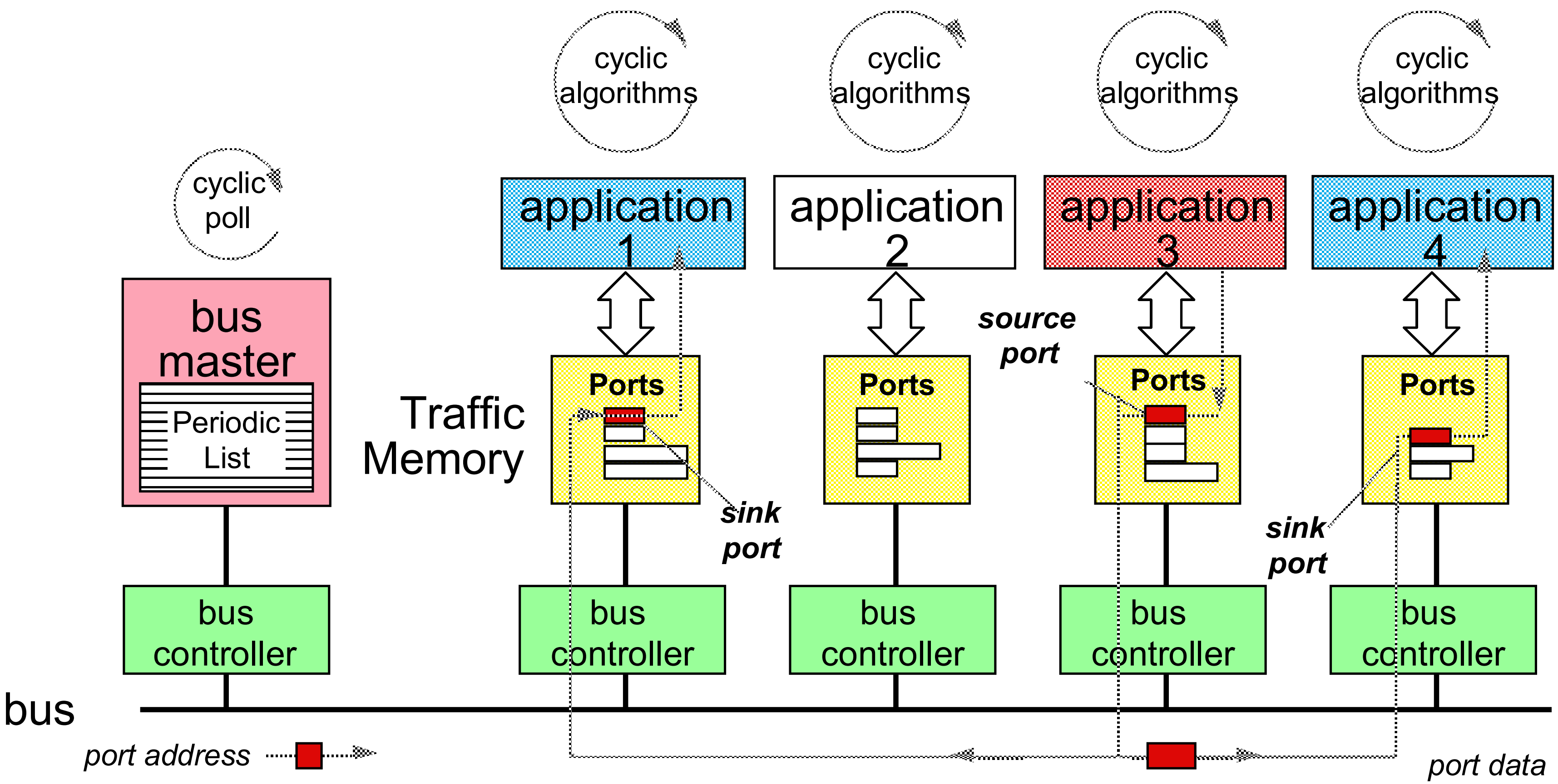
Solution: introduce sub-cycles for less urgent periodic variables:



Cyclic polling need tools to configure the poll cycles.
The poll cycles should not be modified at run-time (non-determinism)

A device exports many process data (state variables) with different priorities.
If there is only one poll type per device, a device must be polled at the frequency required by its highest-priority data.
To reduce bus load, the master polls the process data, not the devices

Cyclic Transmission and Application



The bus traffic and the application cycles are asynchronous to each other.

The bus master scans the identifiers at its own pace.

Bus and applications are decoupled by a shared memory, the *traffic memory*, which acts as distributed database actualized by the network.

Application Of Cyclic Bus

The principle of cyclic operation, combined with source-addressed broadcast, has been adopted by most modern field busses

It is currently used for power plant control, rail vehicles, aircrafts, etc...

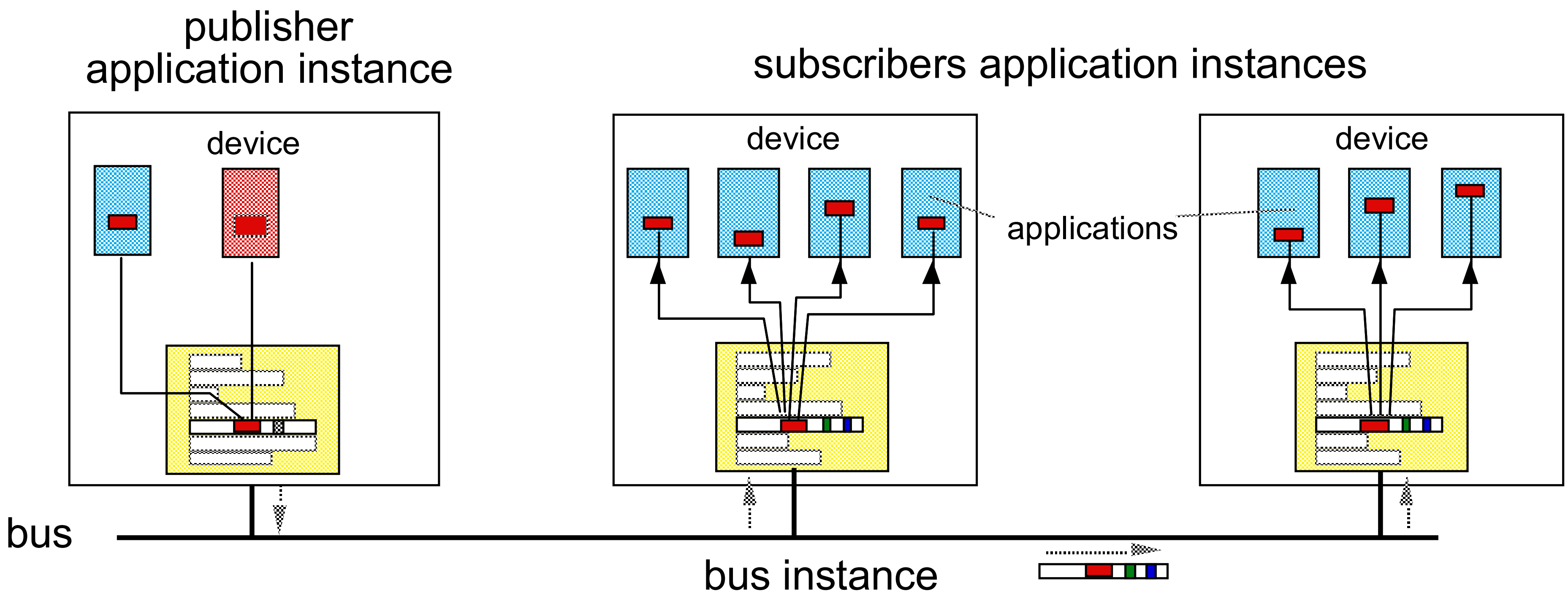
This method gives the network a deterministic behavior, at expenses of a reduced bandwidth and geographical extension.

The poll scan list located in the central master (which may be duplicated for availability purposes) determines the behavior of the bus.

It is configured for a specific project by a single tool, which takes into account the transmission wishes of the applications.

This guarantees that no application can occupy more than its share of the bus bandwidth and gives control to the project leader.

Example: delay requirement

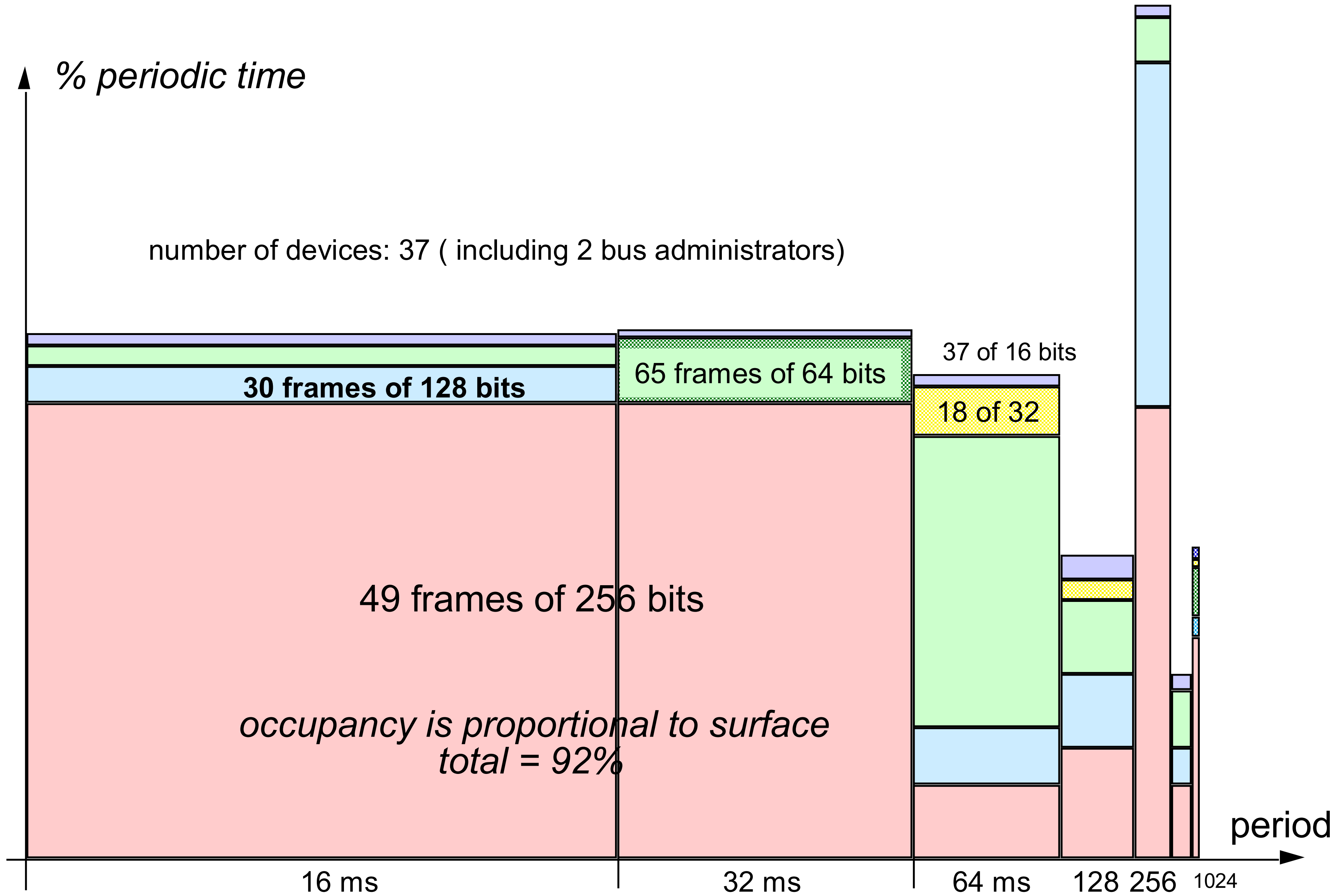


Worst-case delay for transmitting all time critical variables is the sum of:

Source application cycle time	8 ms
Individual period of the variable	16 ms
Sink application cycle time	8 ms

= 32 ms

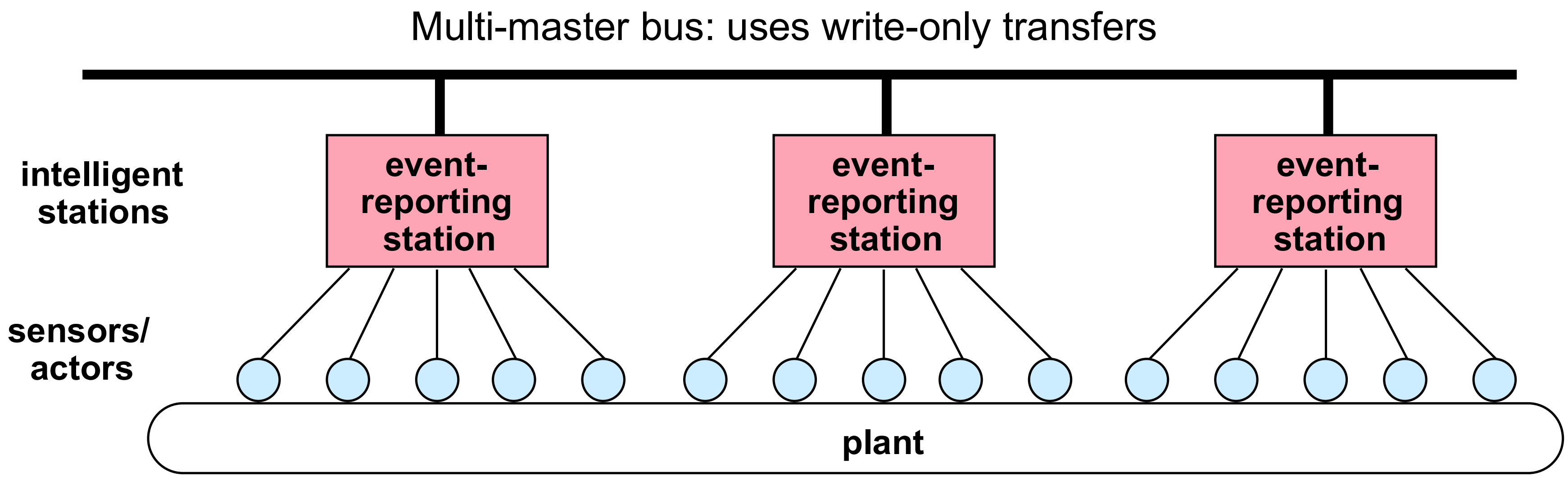
Example: traffic pattern in a locomotive



Fieldbus: Event-driven operation

Event-driven Operation

- Events cause a transmission only when an state change takes place.
- Bus load is very low on the average, but peaks under exceptional situations since transmissions are correlated by the process (christmas-tree effect).

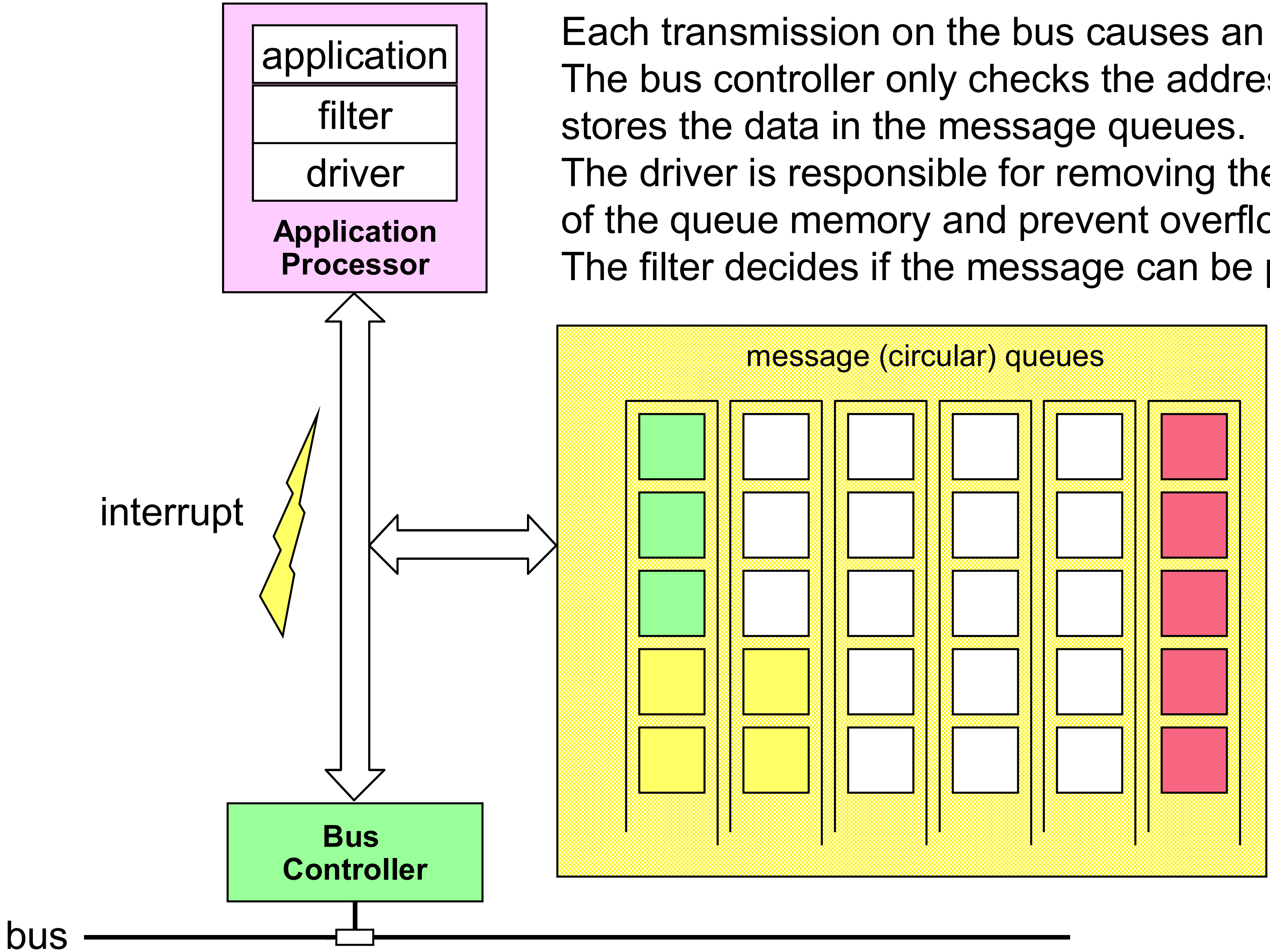


Detection of an event is an intelligent process:

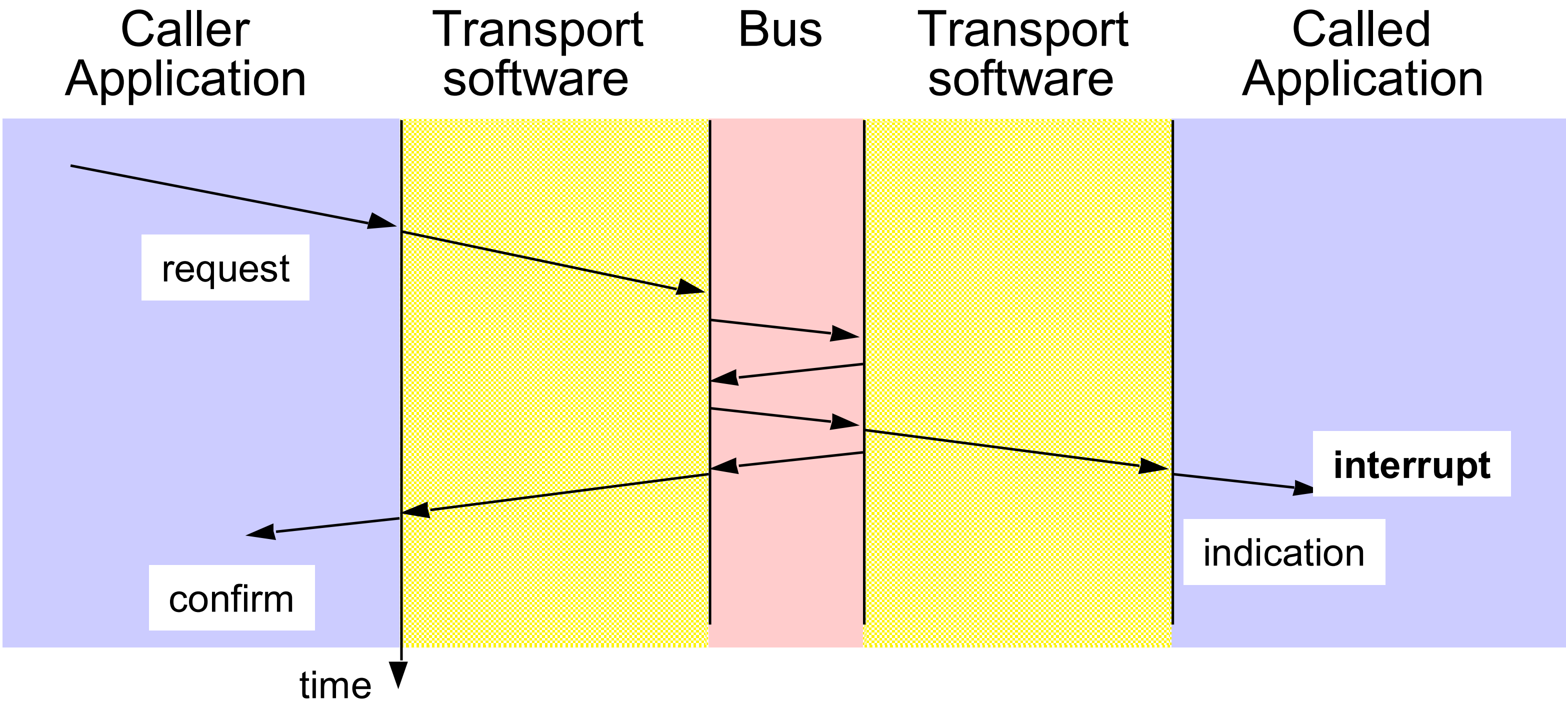
- Not every change of a variable is an event, even for binary variables.
- Often, a combination of changes builds an event.
- Only the application can decide what is an event, since only the application programmer knows the meaning of the variables.

Bus interface for event-driven operation

Each transmission on the bus causes an interrupt.
The bus controller only checks the address and stores the data in the message queues.
The driver is responsible for removing the messages of the queue memory and prevent overflow.
The filter decides if the message can be processed.



Response of Event-driven operation



Since events can occur anytime on any device, stations communicate by spontaneous transmission, leading to possible collisions

Interruption of server device at any instant can disrupt a time-critical task.

Buffering of events cause unbound delays

Gateways introduce additional uncertainties

Determinism and Medium Access In Busses

Although the moment an event occurs is not predictable, the communication means should transmit the event in a finite time to guarantee the reaction delay.

Events are necessarily announced spontaneously: this requires a multi-master medium like in a LAN.

The time required to transmit the event depends on the medium access (arbitration) procedure of the bus.

Medium access control methods are either deterministic or not.

Non-deterministic

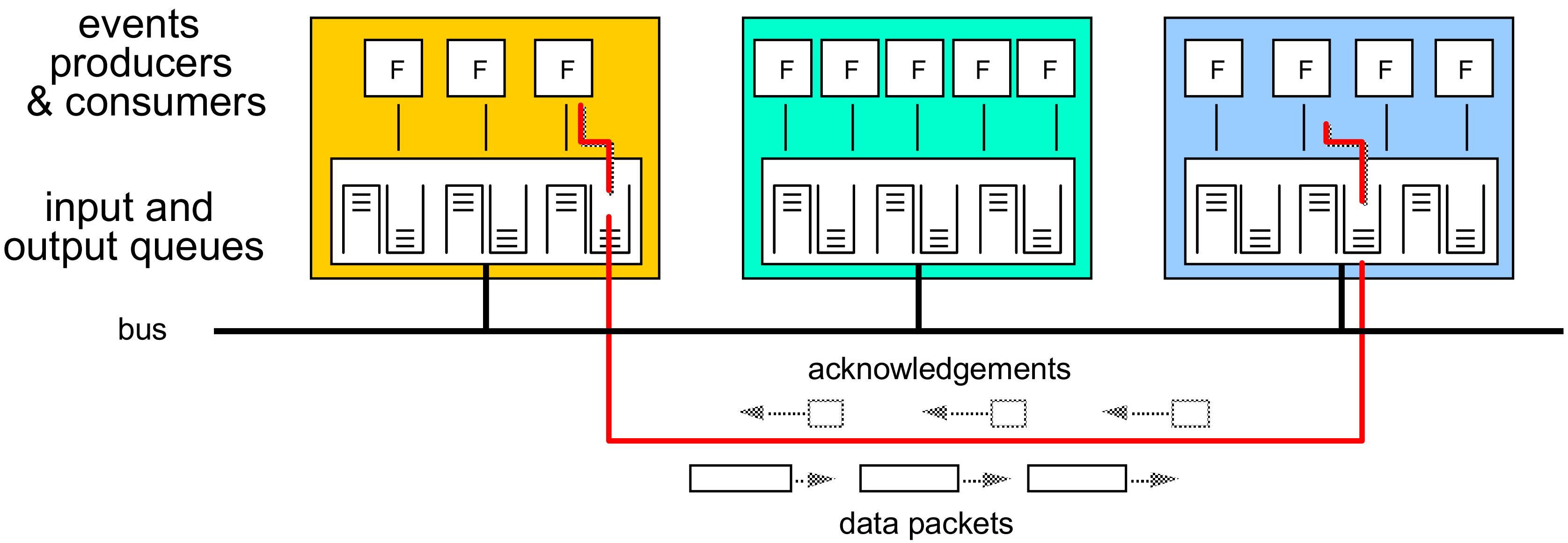
Collision
(Ethernet)

Deterministic

Central master,
Token-passing (round-robin),
Binary bisection,
Collision with winner.

Events and Determinism

Although a deterministic medium access is the condition to guarantee delivery time, it is not sufficient since events messages are queued in the devices.



The average delivery time depends on the length of the queues, on the bus traffic and on the processing time at the destination.
Often, the computers limit far more the event delay than the bus does.

Real-time Control = Measurement + Transmission + Processing + Acting

Events Pros and Cons

In an event-driven control system, there is only a transmission or an operation when an event occurs.

Advantages:
- Can treat a large number of events - if not all at the same time
- Supports a large number of stations
- System idle under steady - state conditions
- Better use of resources
- Uses write-only transfers, suited for LANs with long propagation delays
- Suited for standard (interrupt-driven) operating systems (Unix, Windows)

Drawbacks:
- Requires intelligent stations (event building)
- Needs shared access to resources (arbitration)
- No upper limit to access time if some component not deterministic
- Response time difficult to estimate, requires analysis
- Limited by congestion effects: process correlates events
- A background cyclic operation is needed to check liveliness

Fieldbus: real-time communication model

3.1 Field bus types
- Classes
- Physical layer

3.2 Field bus operation
- Centralized - Decentralized
- Cyclic Operation
- Event Driven Operation
- Real-time communication model
- Networking

3.3 Standard field busses

Mixed Data Traffic

Process Data

represent the state of the plant

short and urgent data items

... motor current, axle speed, operator's commands, emergency stops,...

-> Periodic Transmission of Process Variables

Since variables are refreshed periodically, no retransmission protocol is needed to recover from transmission error.

Message Data

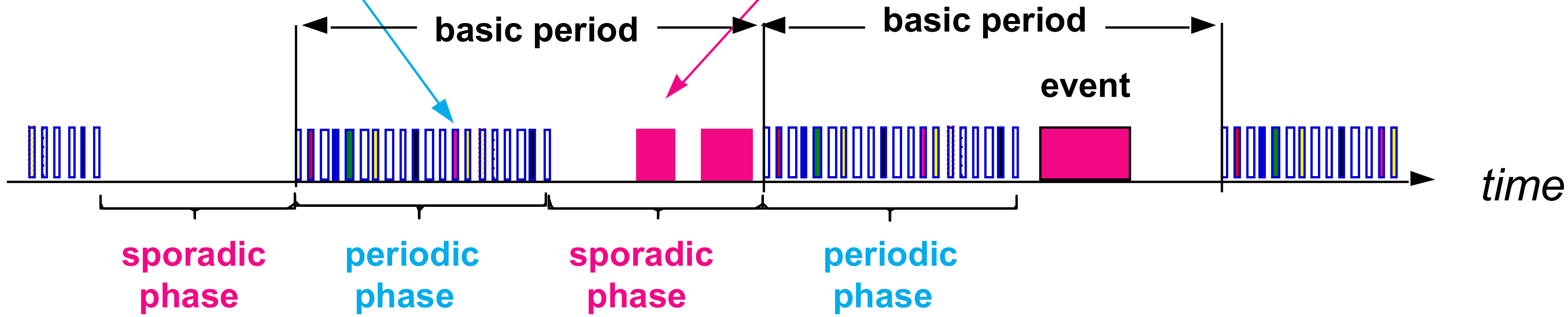
represent state changes of the plant

infrequent, sometimes lengthy messages reporting events, for:

- Users: set points, diagnostics, status
- System: initialisation, down-loading, ...

-> Sporadic Transmission of Process Variables and Messages

Since messages represent state changes, a protocol must recover lost data in case of transmission errors



Mixing Traffic is a configuration issue

Cyclic broadcast of source-addressed variables is the standard solution in field busses for process control.

Cyclic transmission takes a large share of the bus bandwidth and should be reserved for really critical variables.

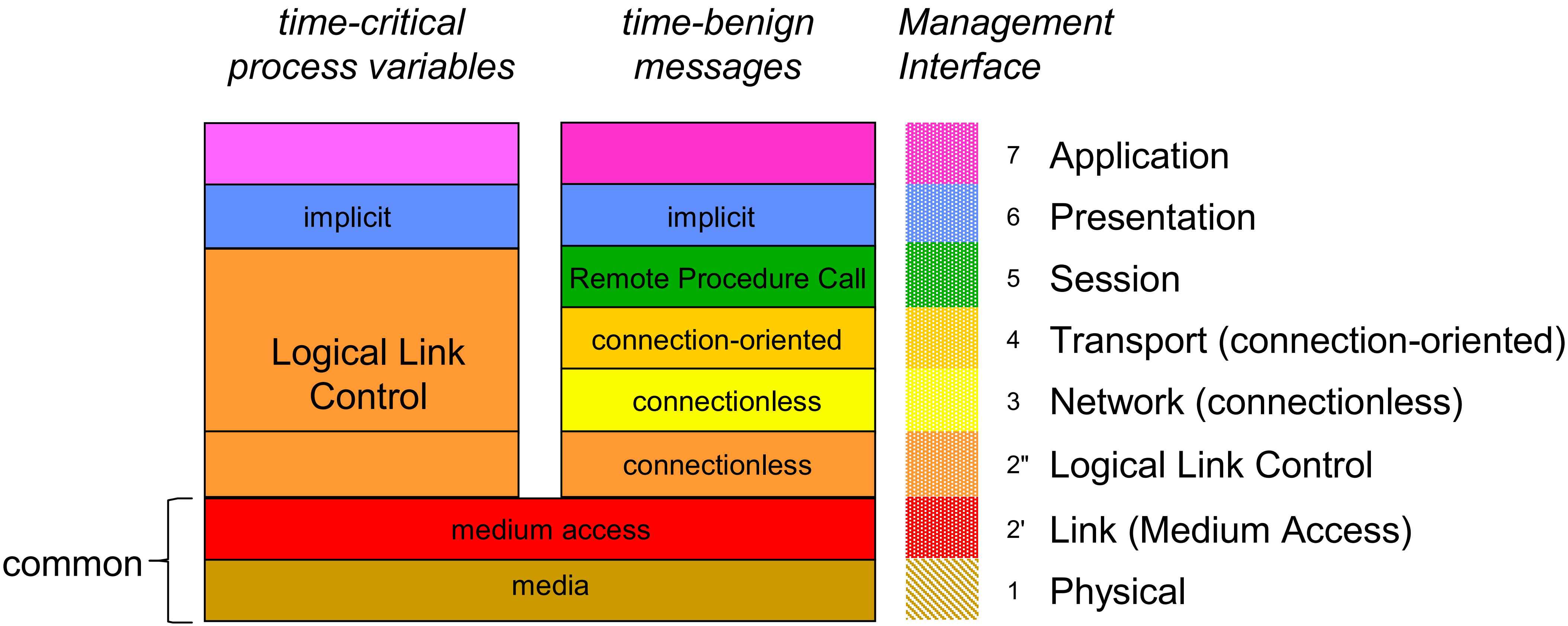
The decision to declare a variable as cyclic or event-driven can be taken late in a project, but cannot be changed on-the-fly in an operating device.

A message transmission scheme must exist alongside the cyclic transmission to carry not-critical variables and long messages such as diagnostics or network management

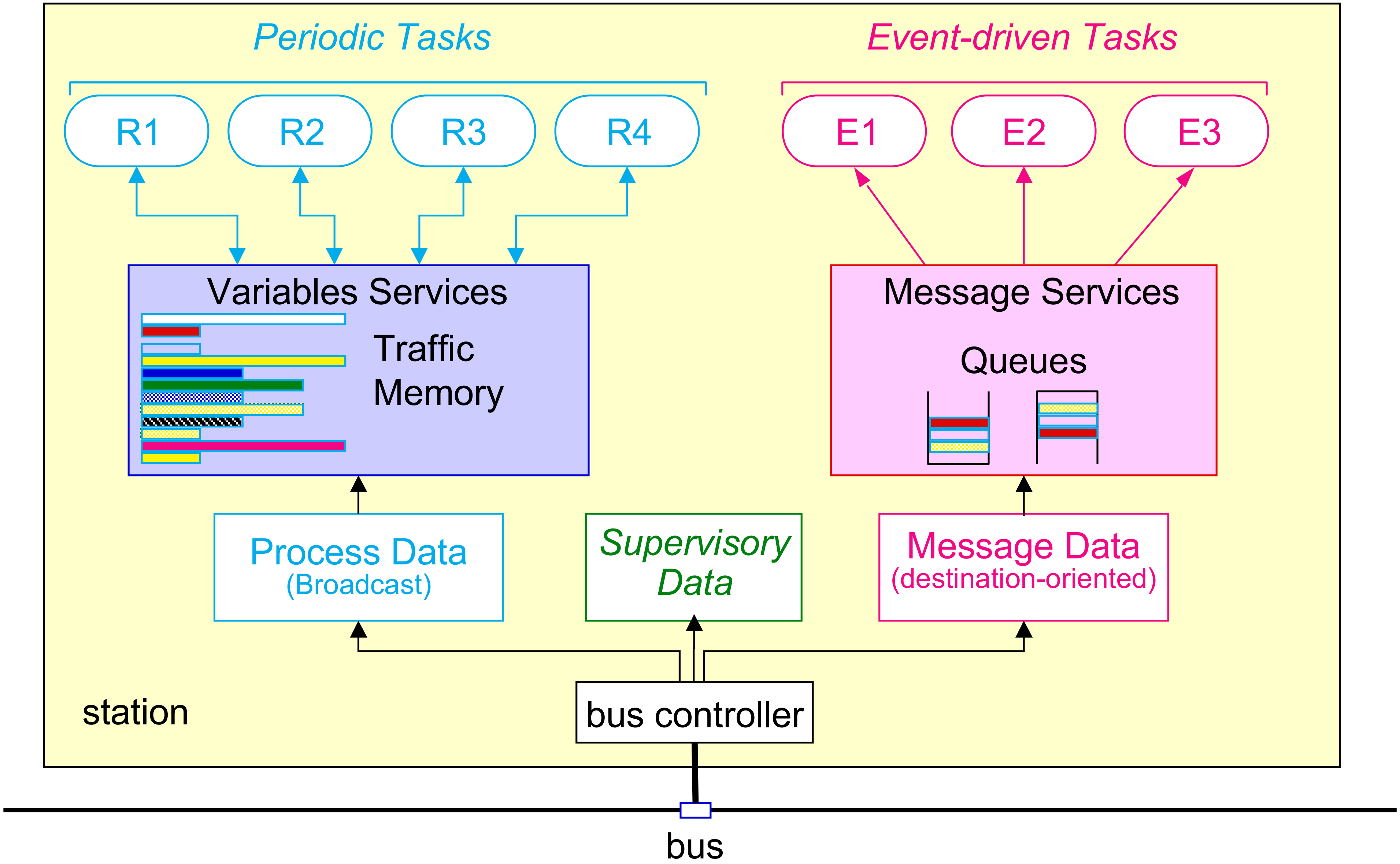
An industrial communication system should provide both transmission kinds.

Real-Time communication stack

The real-time communication model uses two stacks, one for time-critical variables and one for messages



Application Sight Of Communication



Field - and Process bus

Fieldbus	Process Bus
strictly deterministic	non - deterministic
controlled by a central master (redundant for availability)	multi-master bus (Arbitration) deterministic arbitration -> Token
cyclic polling	event-driven
number of participants limited by maximum period	large number of participants
call/reply in one bus transfer (read-cycle) ("fetch principle")	call/reply uses two different messages. both parties must become bus master ("bring - principle")
cheap connection (dumb)	costly connection (intelligent)
only possible over a limited geographical extension	also suited for open systems

Cyclic or Event-driven Operation For Real-time ?

The operation mode of the communication exposes the main approach to conciliate real-time constrains and efficiency in a control systems.

cyclic operation	event-driven operation
Data are transmitted at fixed intervals, whether they changed or not.	Data are only transmitted when they change or upon explicit demand.
Deterministic: delivery time is bound	Non-deterministic: delivery time vary widely
Worst Case is normal case	Typical Case works most of the time
All resources are pre-allocated	Best use of resources
(periodic, round-robin)	(aperiodic, demand-driven, sporadic)
object-oriented bus	**message-oriented bus**
Fieldbus Foundation, MVB, FIP, ..	Profibus, CAN, LON, ARCnet

Fieldbus: Networking

Networking field busses

Networking field busses is not done through bridges or routers, because normally, transition from one bus to another is associated with:

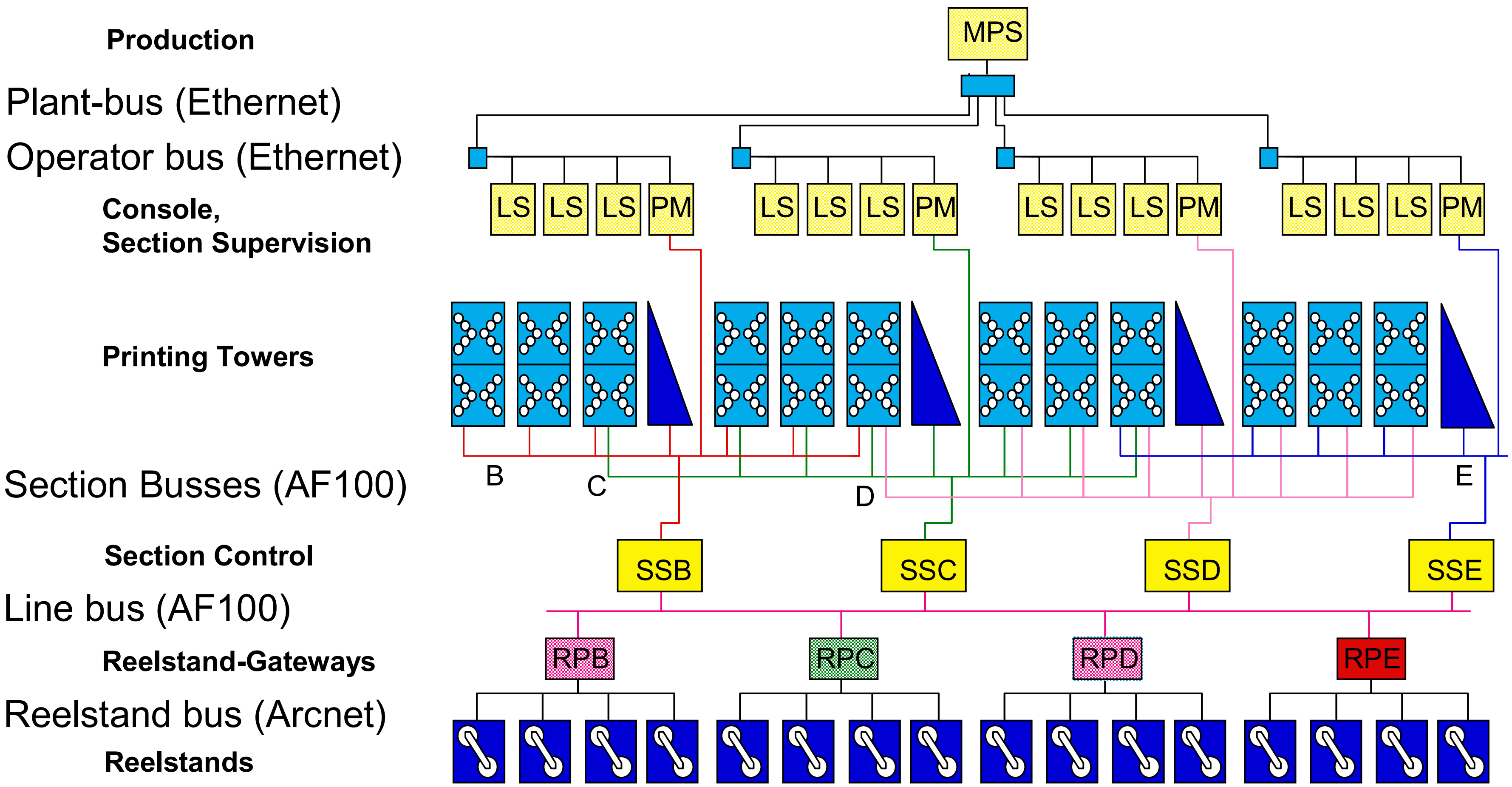
- data reduction (processing, sum building, alarm building, multiplexing)

- data marshalling (different position in the frames)

- data transformation (different formats on different busses)

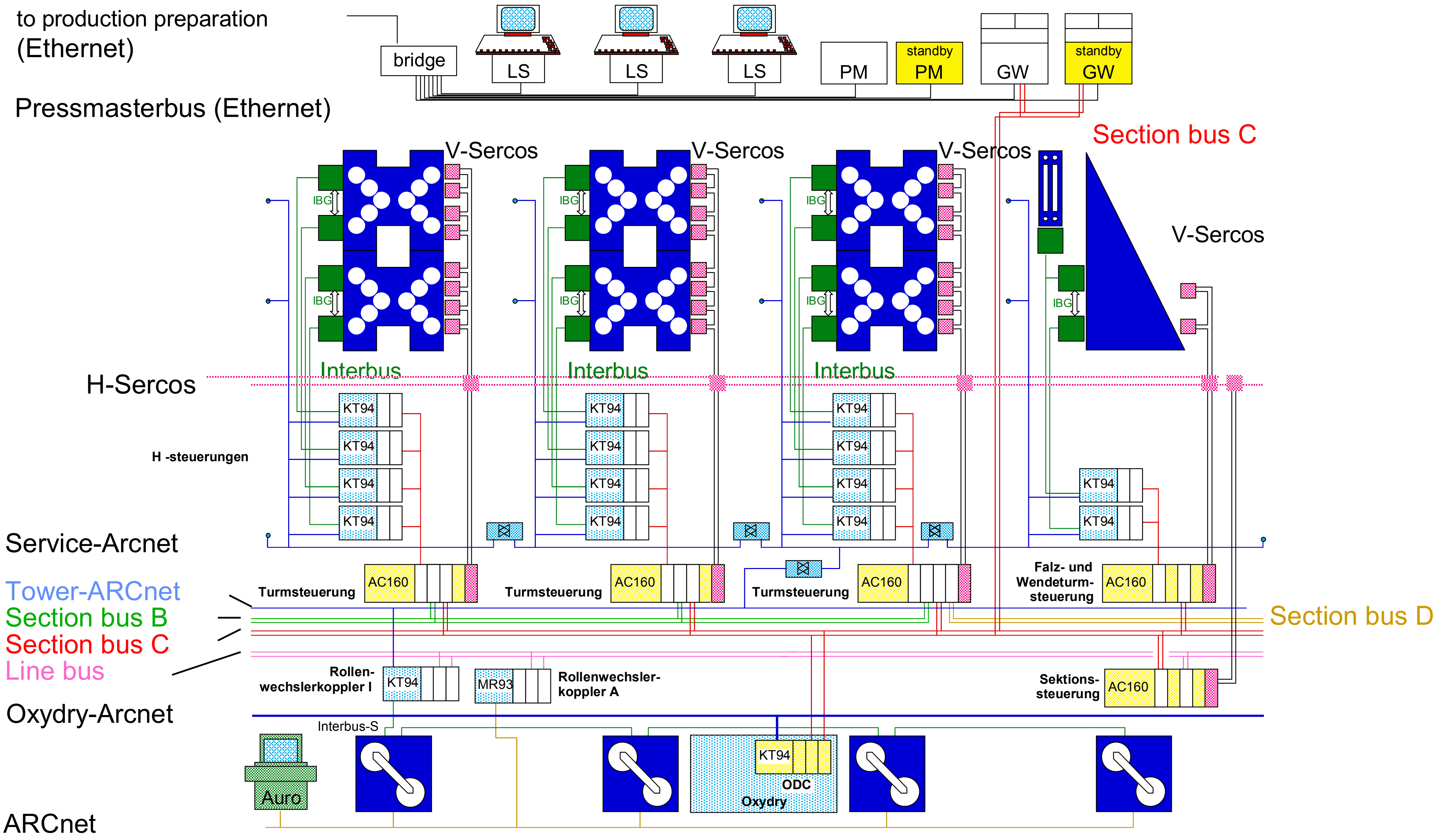
Only system management messages could be threaded through from end to end, but due to lack of standardization, data conversion is today not avoidable.

Networking: Printing machine (1)

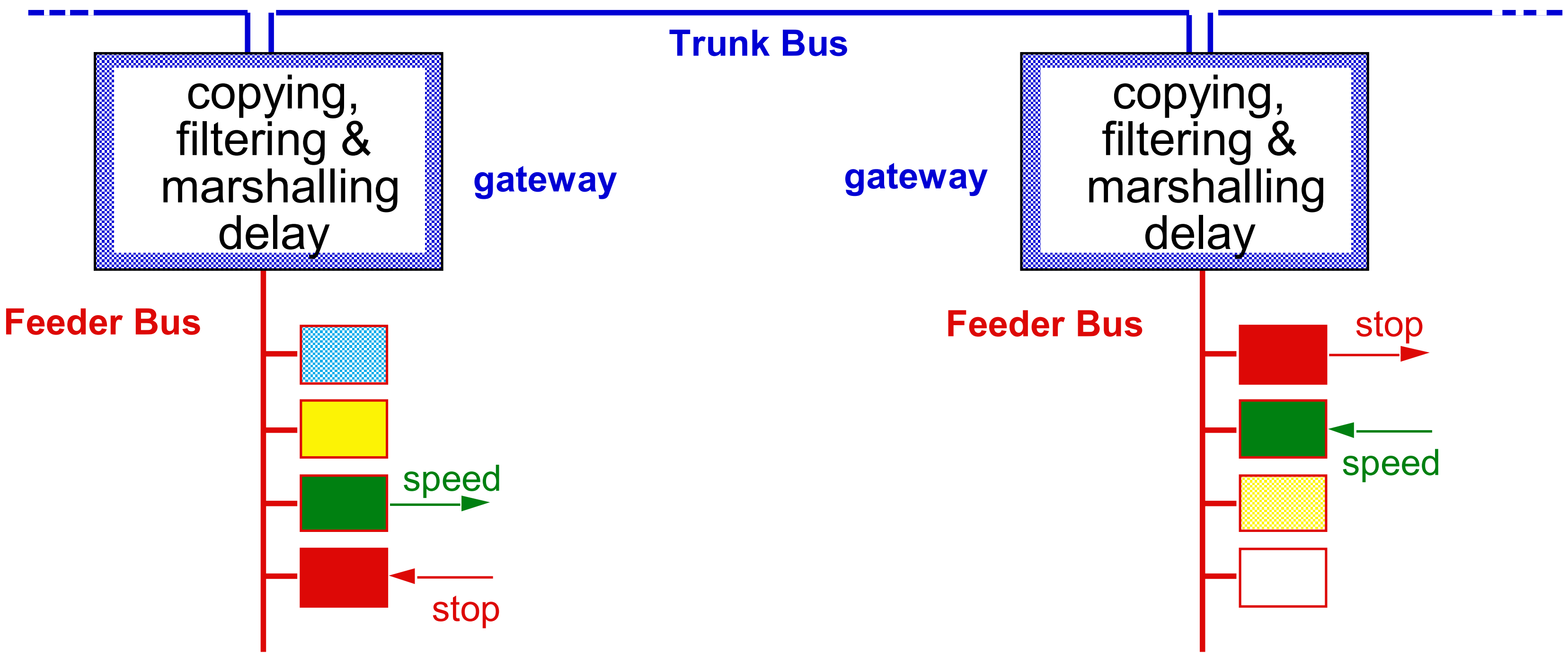


multiplicity of field busses with different tasks, often associated with units.
main task of controllers: gateway, routing, filtering, processing data.
most of the processing power of the controllers is used to route data

Networking: Printing Section (2)



Transmission delay over a Trunk Bus (cyclic bus)



The actual delay is non-deterministic, but bounded

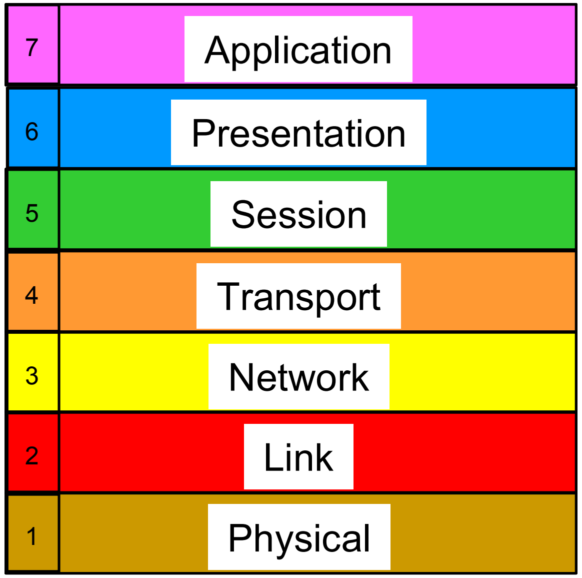
The worst-case delay for the transmission of all variables is the sum of 5 delays:

• feeder bus delay	32 ms	**= 100 ms**
• gateway marshalling delay	16 ms	
• trunk bus delay	25 ms	
• gateway marshalling delay	10 ms (synchronized)	
• feeder bus delay	32 ms	

Assessment

- What is the difference between a centralized and a decentralized industrial bus ?
- What is the principle of source-addressed broadcast ?
- What is the difference between a time-stamp and a freshness counter ?
- Why is an associative memory needed for source-addressed broadcast ?
- What are the advantages / disadvantages of event-driven communication ?
- What are the advantages / disadvantages of cyclic communication ?
- How are field busses networked ?

Industrial Automation
Automation Industrielle
Industrielle Automation



3 Industrial Communication Systems

3.3.1 Open System Interconnection (OSI) model
Modèle OSI d'interconnexion
OSI-Modell

Prof. Dr. H. Kirrmann
ABB Research Center, Baden, Switzerland

The OSI model

The Open System Interconnection (OSI) model is a standard way to structure communication software that is applicable to any network.

- was developed to structure telecommunication protocols in the ‘70 (Pouzin & Zimmermann)

- standardized by CCITT and ISO as ISO / IEC 7498

- all communication protocols (TCP/IP, Appletalk or DNA) can be mapped to the OSI model.

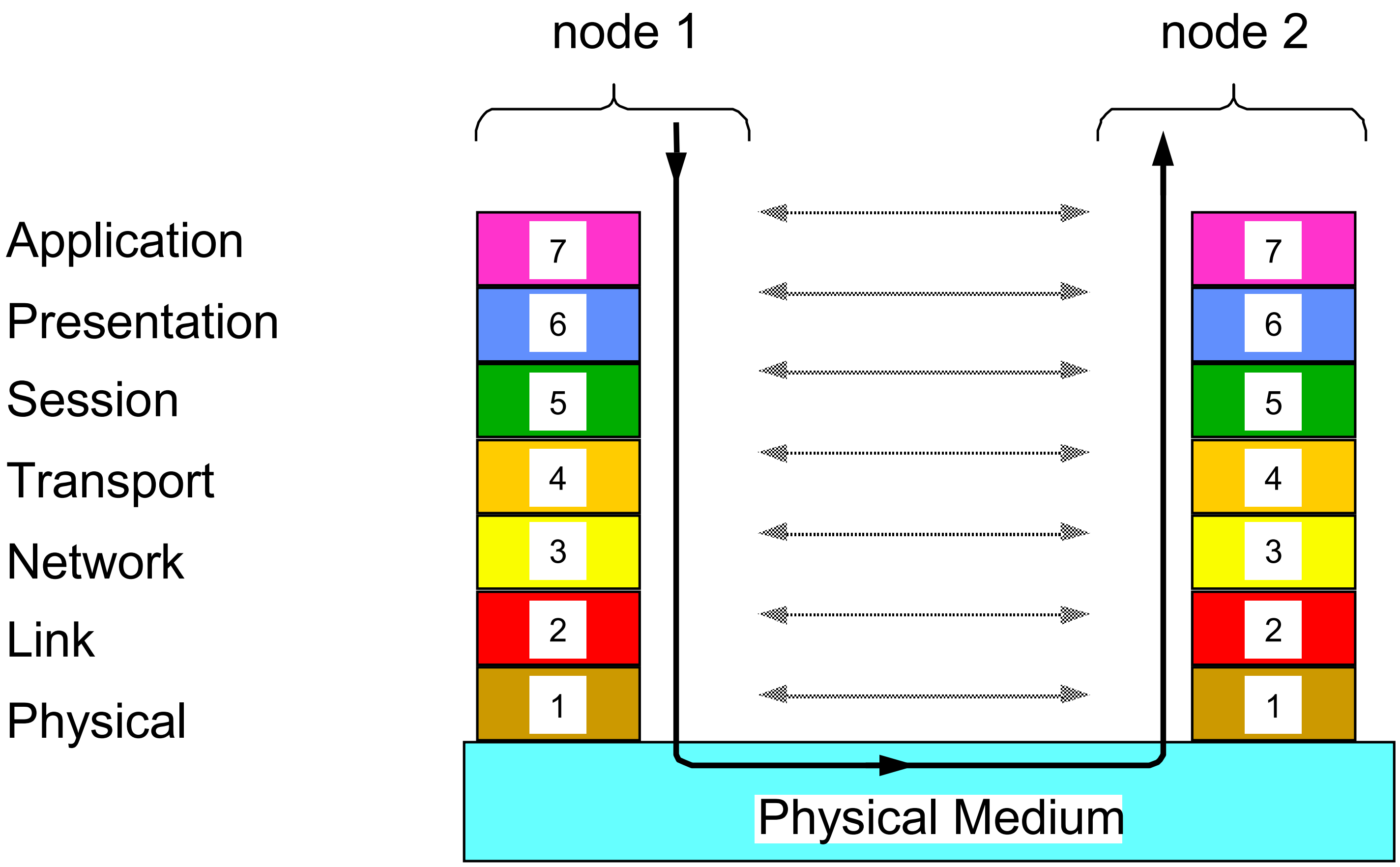
- is a model, not a standard protocol, but a suite of protocols with the same name has been standardized by UIT / ISO / IEC for open systems data interconnection (but with little success)

- mapping of OSI to industrial communication requires some additions

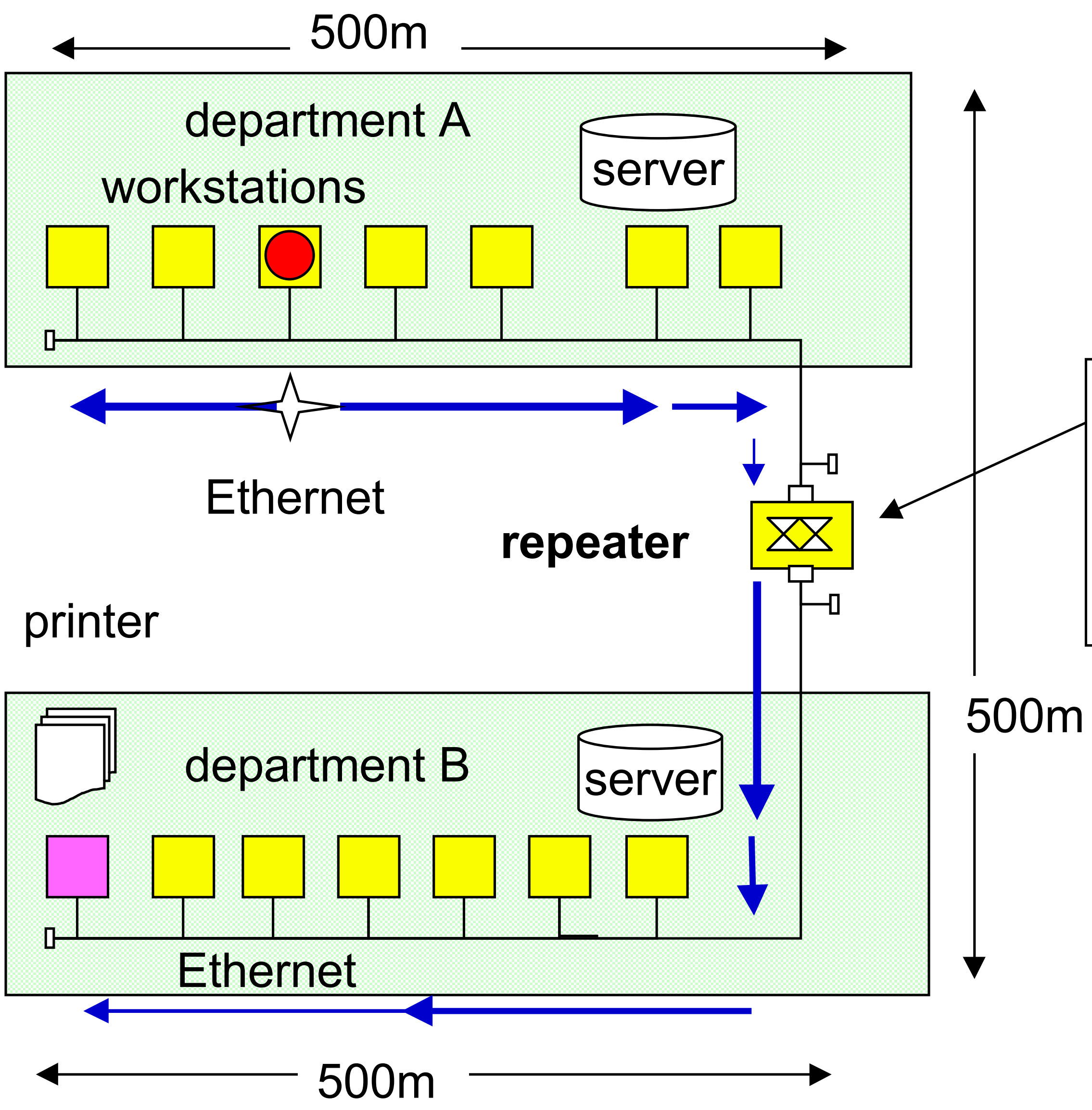
OSI-Model (ISO/IEC standard 7498)

"Application" protocols	7	Application	All services directly called by the end user (Mail, File Transfer,...) e.g. Telnet, SMTP
	6	Presentation	Definition and conversion of the data formats (e.g. ASN 1)
	5	Session	Management of connections (e.g. ISO 8326)
"Transport" protocols	4	Transport	End-to-end flow control and error recovery (e.g. TP4, TCP)
	3	Network	Routing, possibly segmenting (e.g. IP, X25)
	2	Link	Error detection, Flow control and error recovery, medium access (e.g. HDLC)
	1	Physical	Coding, Modulation, Electrical and mechanical coupling (e.g. RS485)

OSI Model with two nodes



Repeater

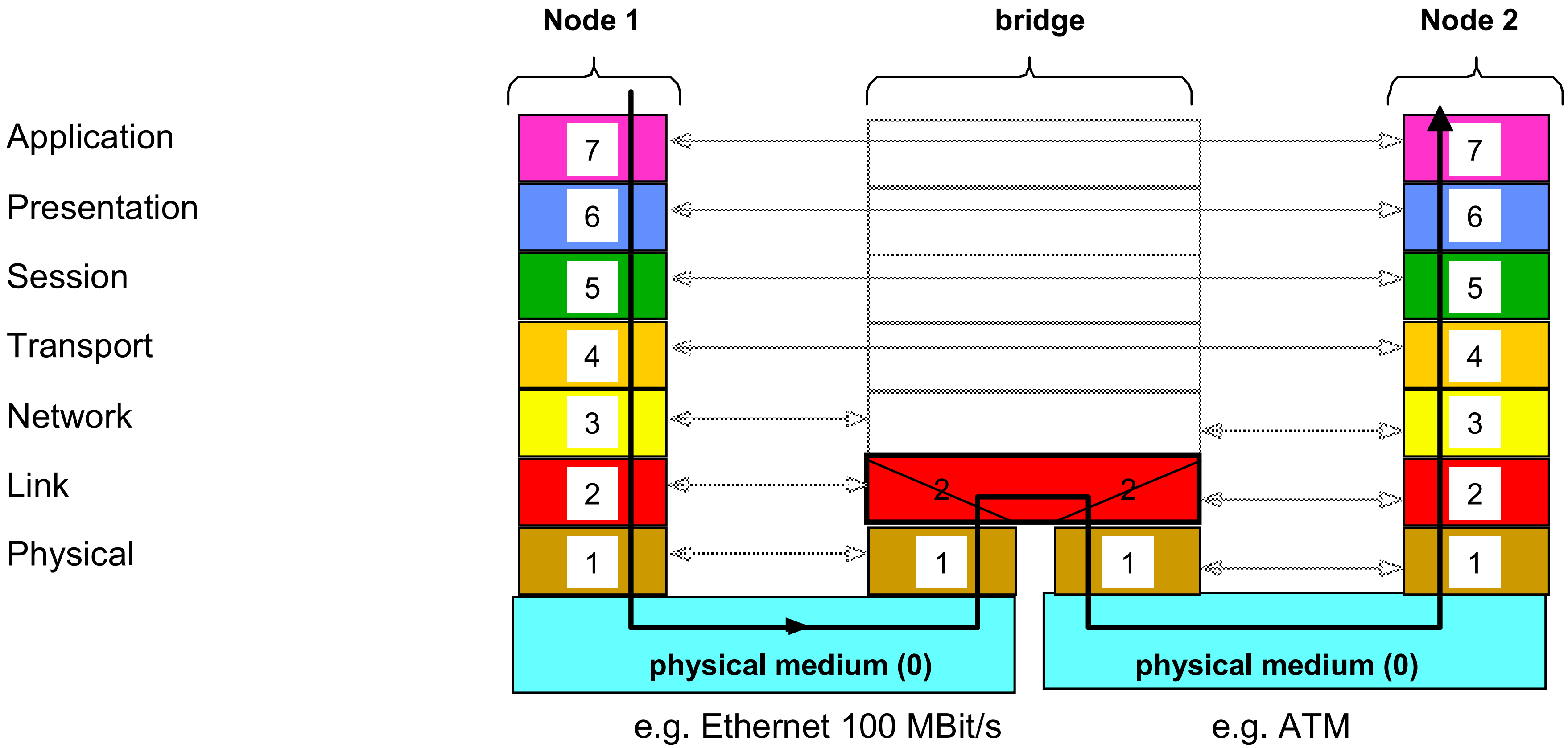


To connect a workstation of department A to the printer of department B, the cable becomes too long and the messages are corrupted.

The repeater restores signal levels and synchronization.
It introduces a signal delay of about 1..4 bits

Physically, there is only one bus carrying both department's traffic, only one node may transmit at a time.

OSI model with three nodes (bridge)



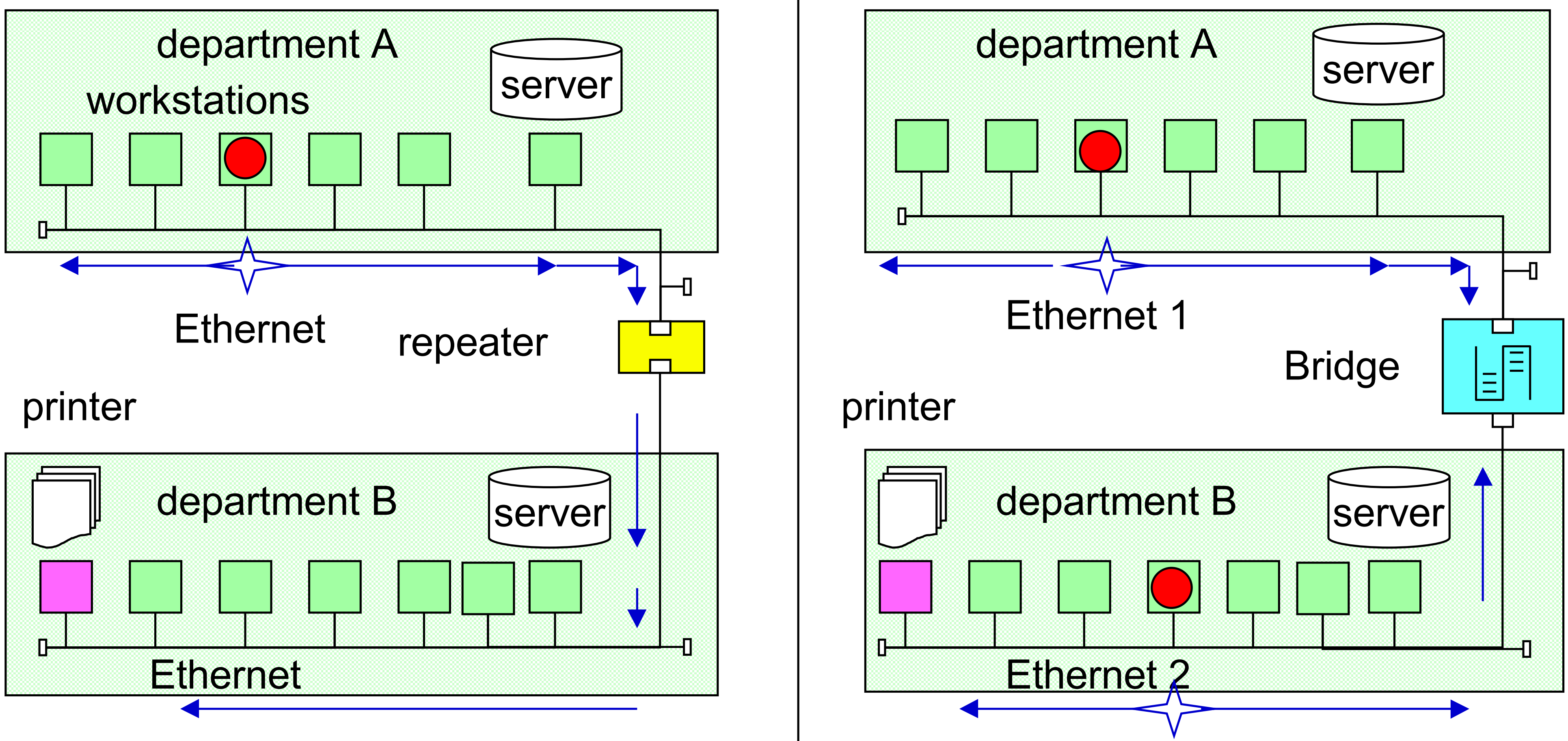
The subnet on both sides of a bridge have:

- the same frame format (except header),
- the same address space (different addresses on both sides of the bridge)
- the same link layer protocol (if link layer is connection-oriented)

Bridges filter the frames on the base of their link addresses

Bridge example

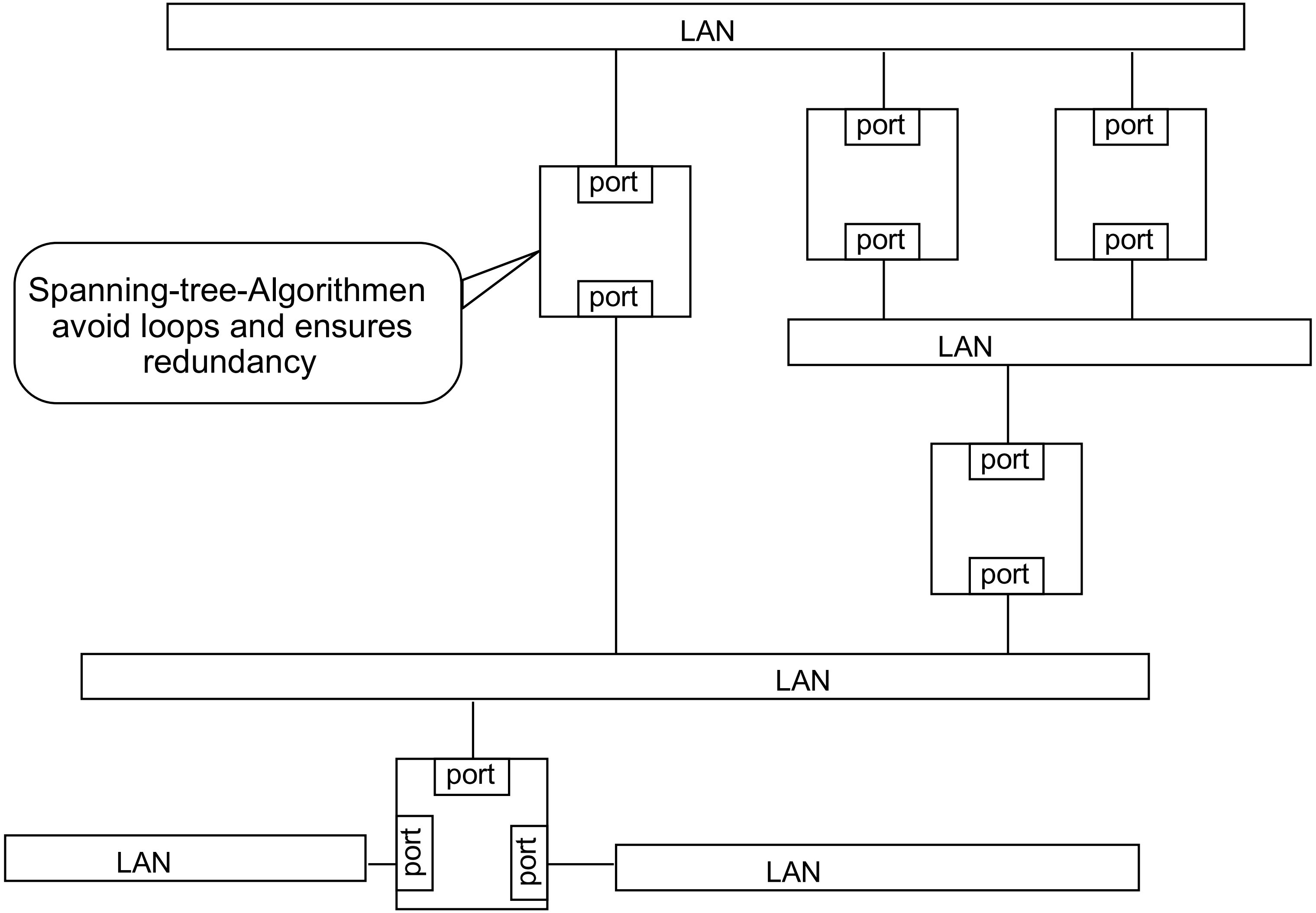
In this example, most traffic is directed from the workstations to the department server, there is little cross-department traffic



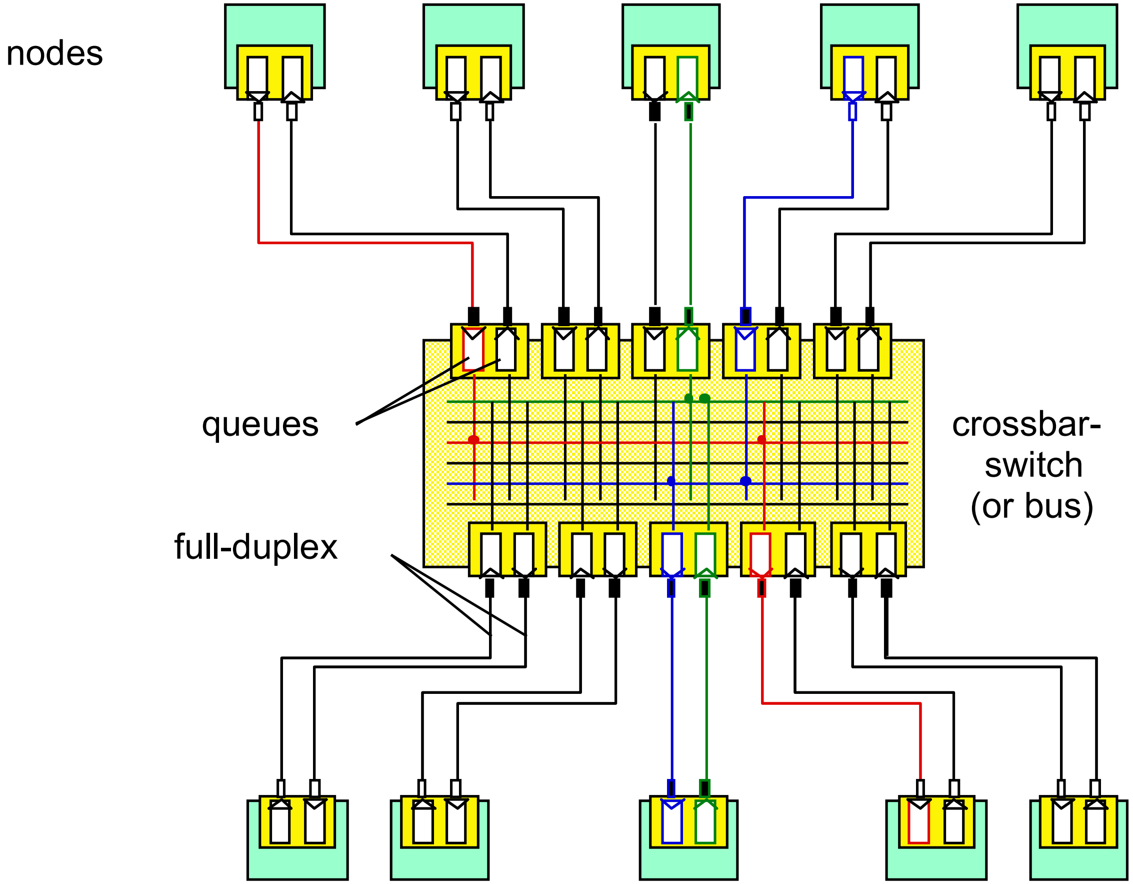
There is only one Ethernet which carries both department's traffic

There are now two Ethernets and only the cross-department traffic burdens both busses

Networking with bridges

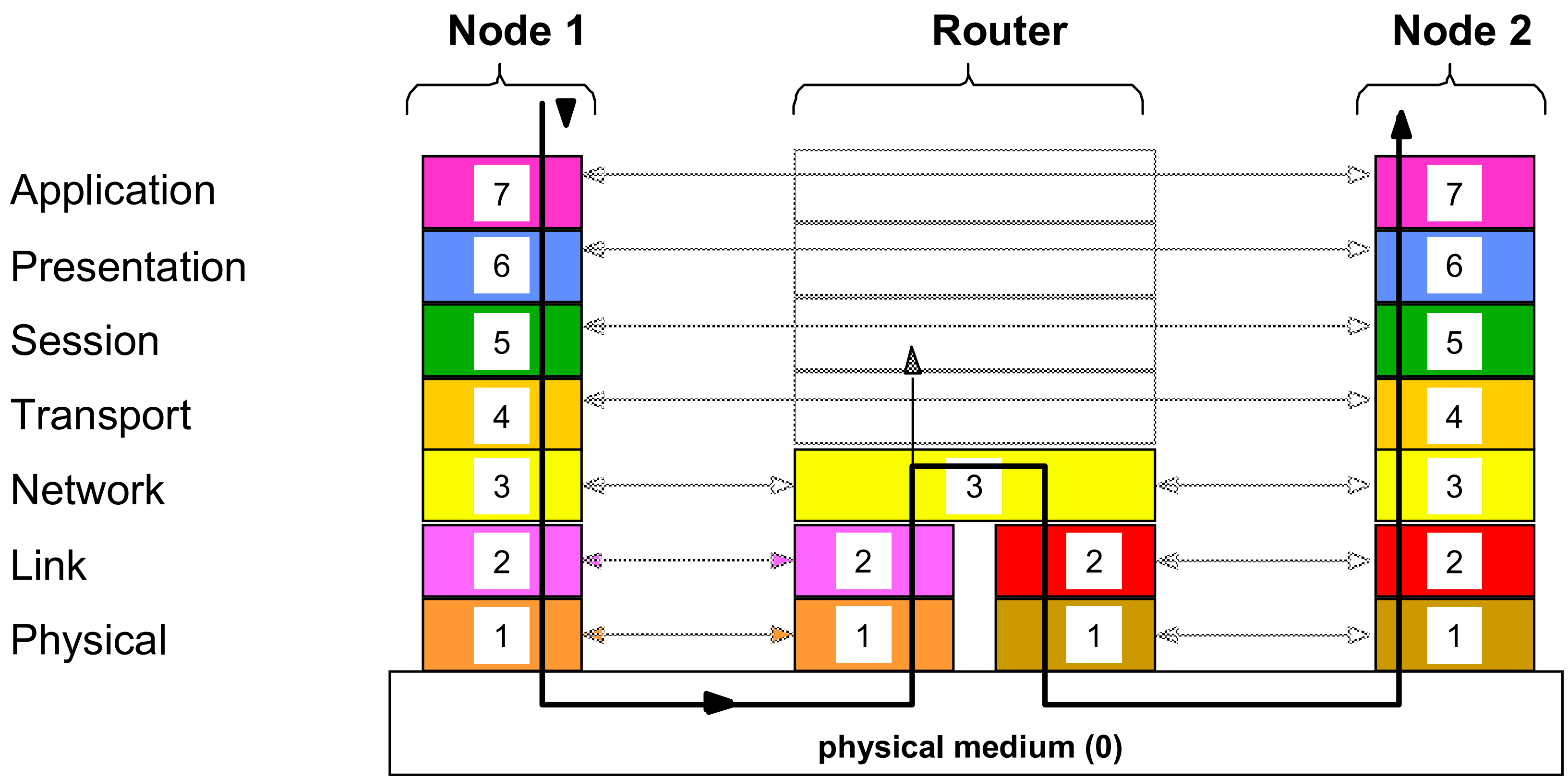


Switch



a switch is an extension of a hub that allows store-and-forward.

OSI Model with three nodes (router)

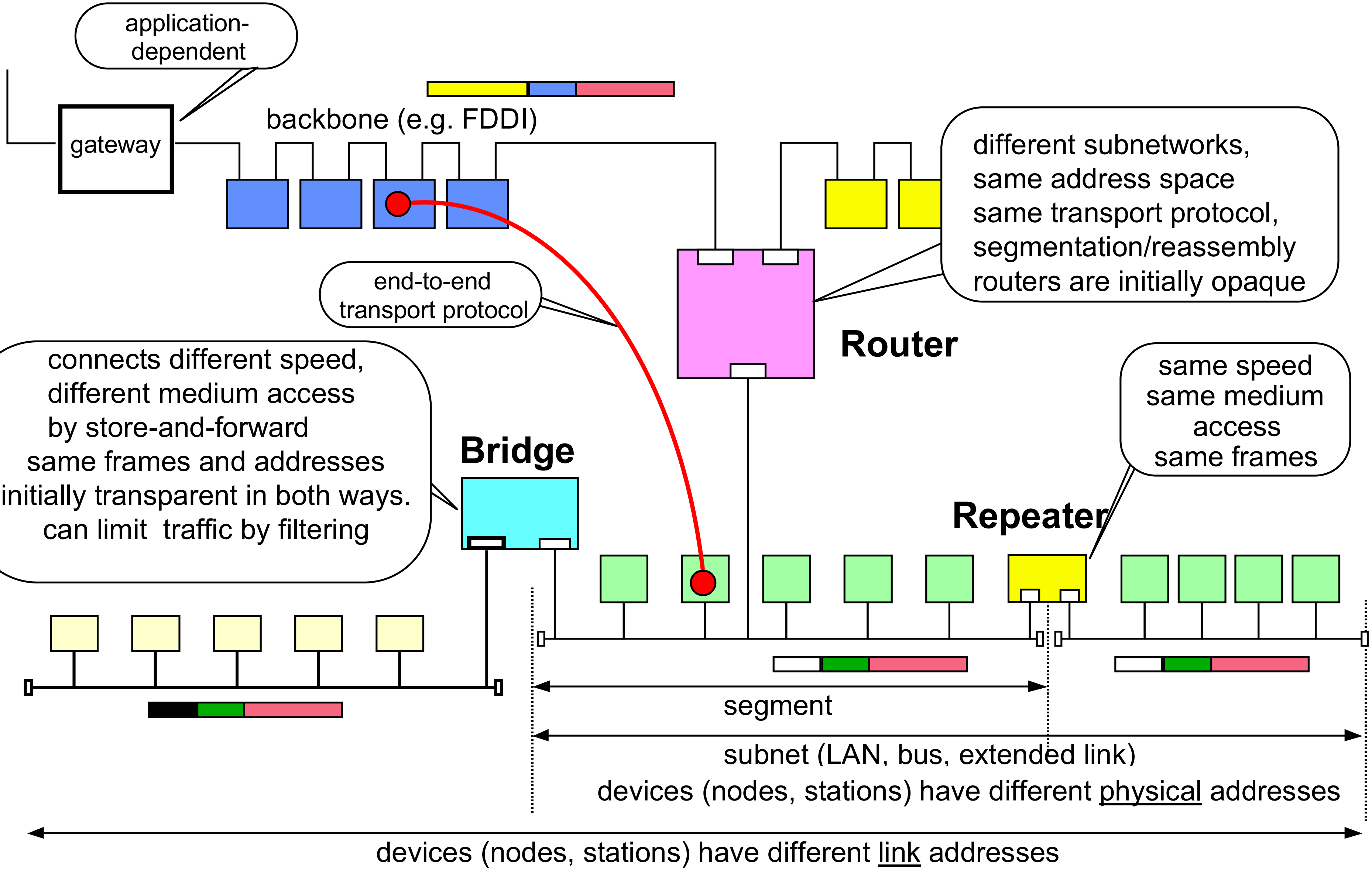


The router routes the frames on the base of their network address.

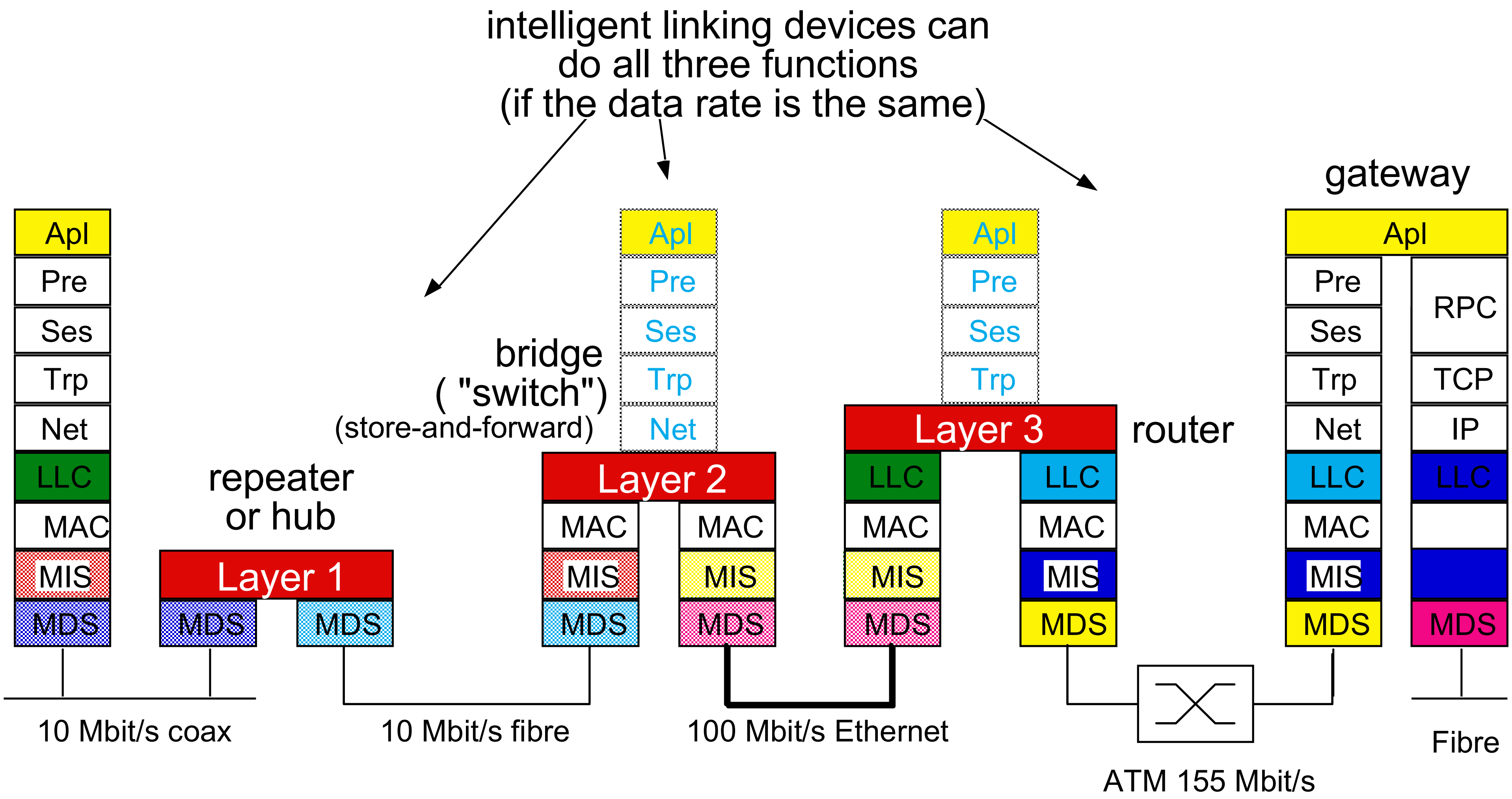
The subnets may have different link layer protocols

Frames in transit are handled in the network layer .

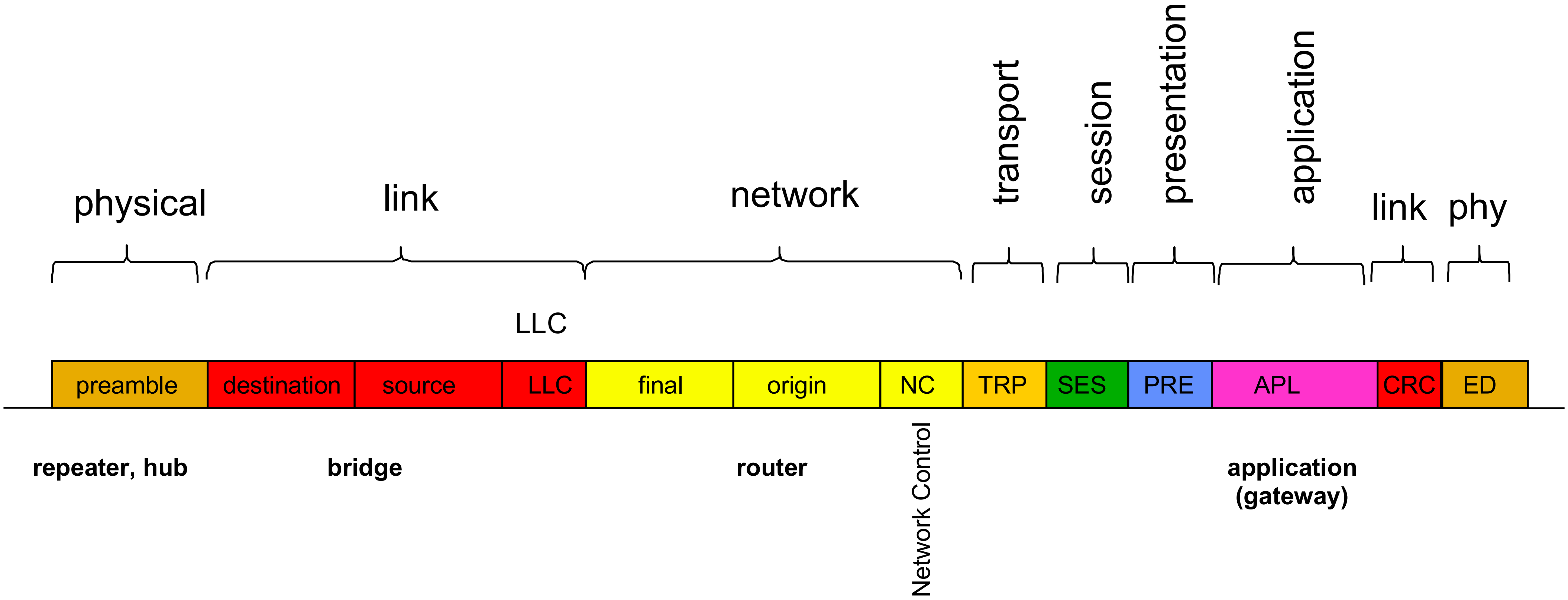
Repeater, Bridge, Router, Gateway: Topography



Repeaters, Bridges, Routers and Gateways: OSI model

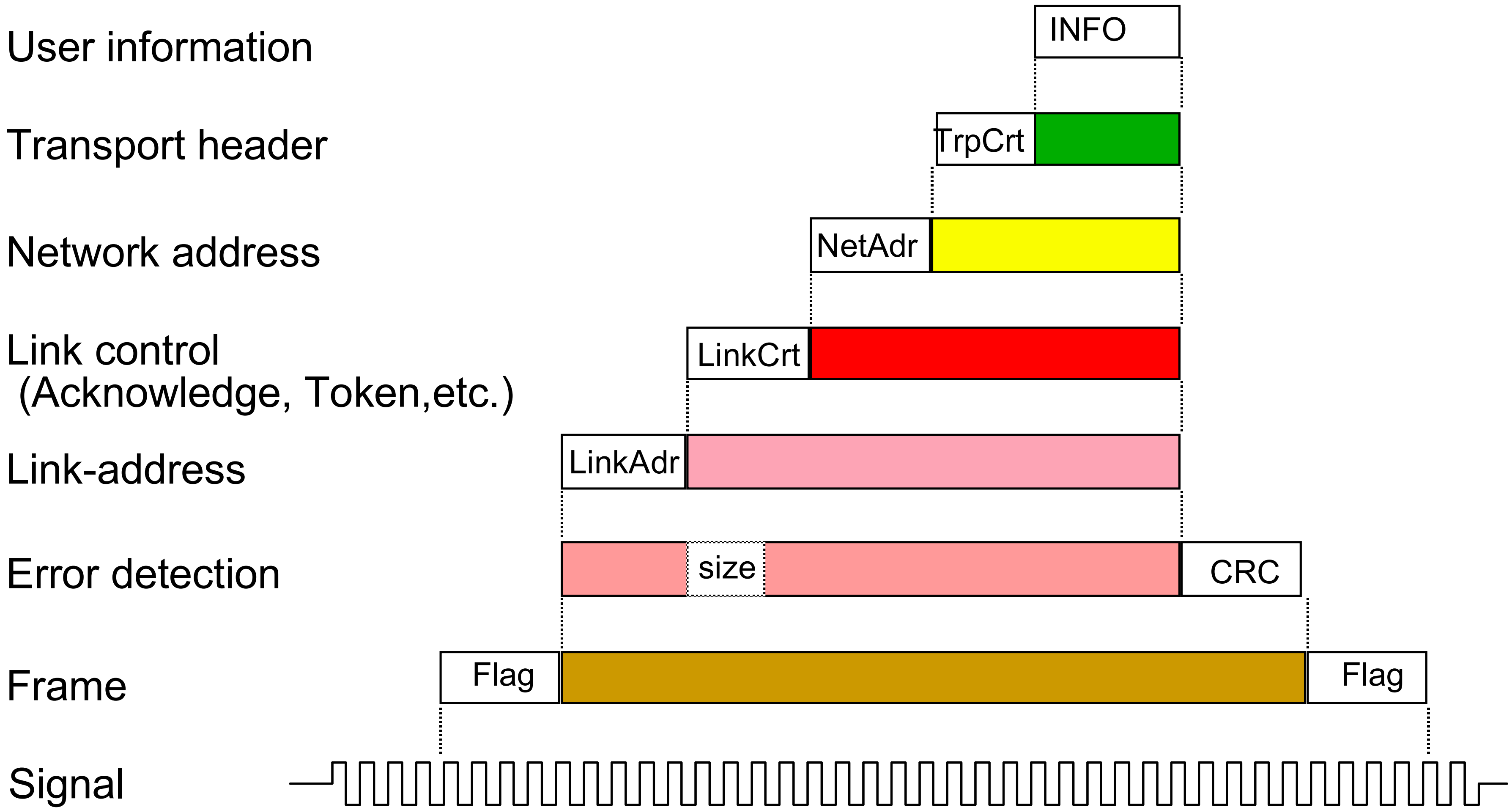


To which level does a frame element belong ?



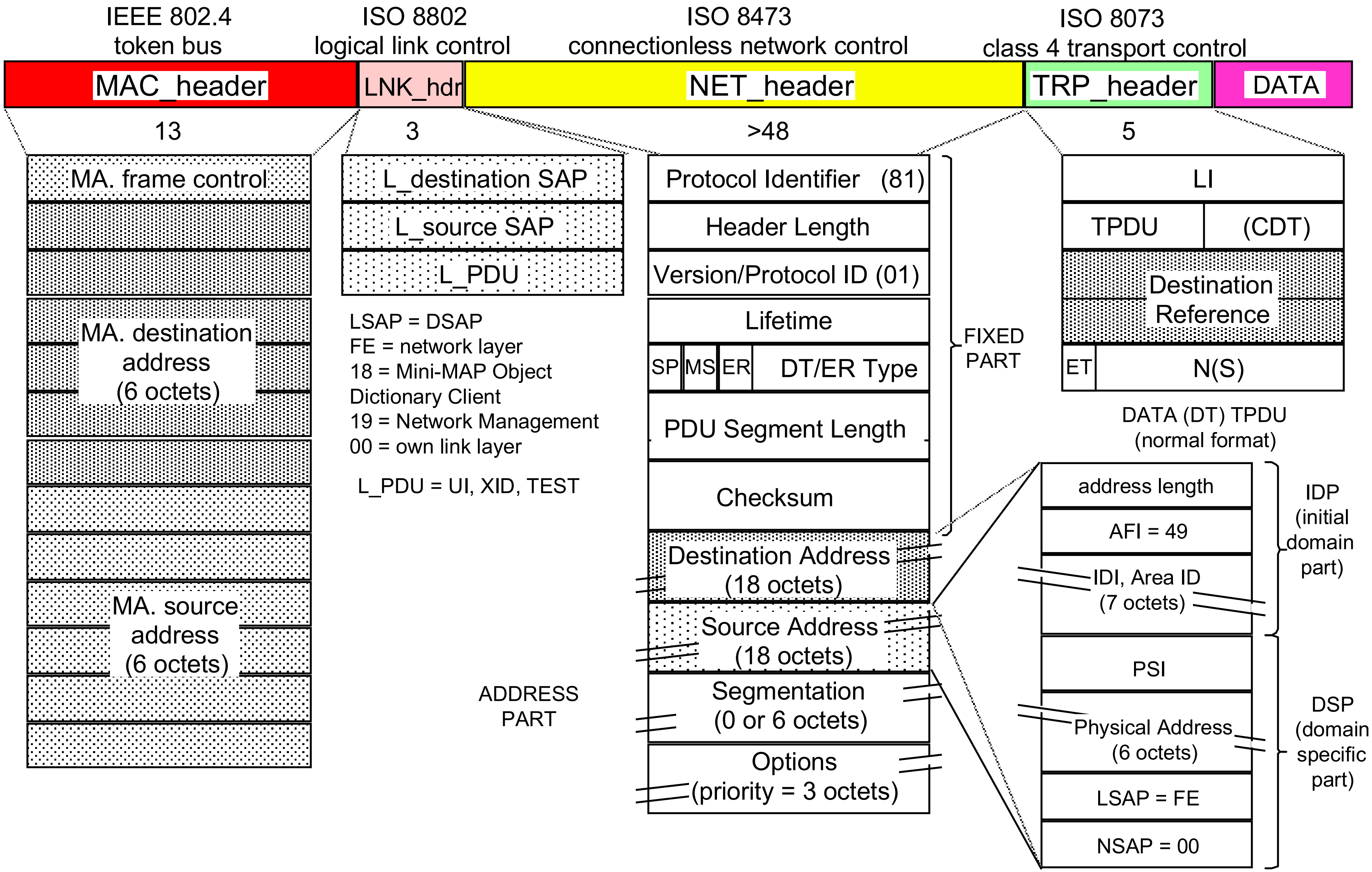
A frame is structured according to the ISO model

Encapsulation

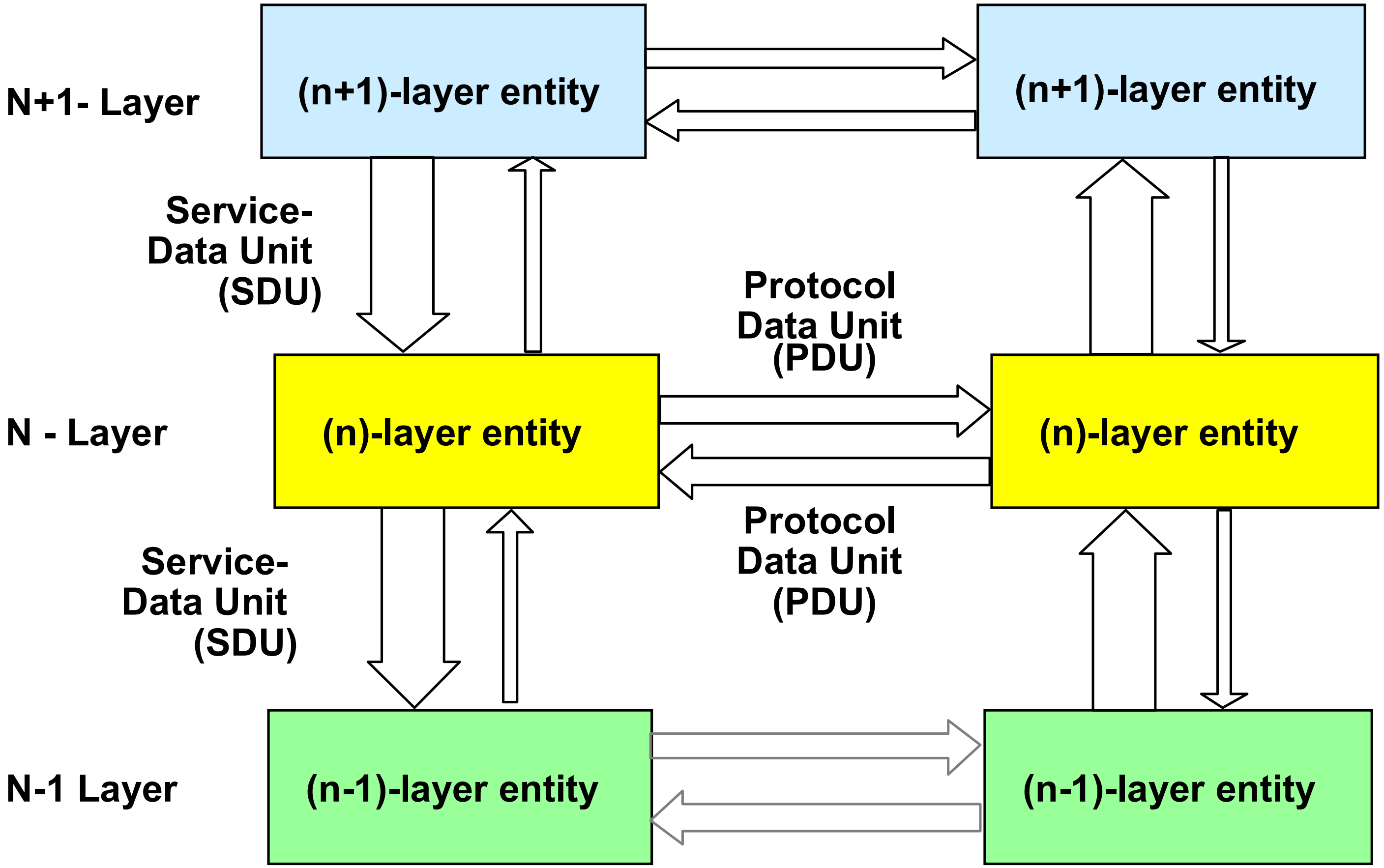


Each layer introduces its own header and overhead

Example: OSI-Stack frame structure

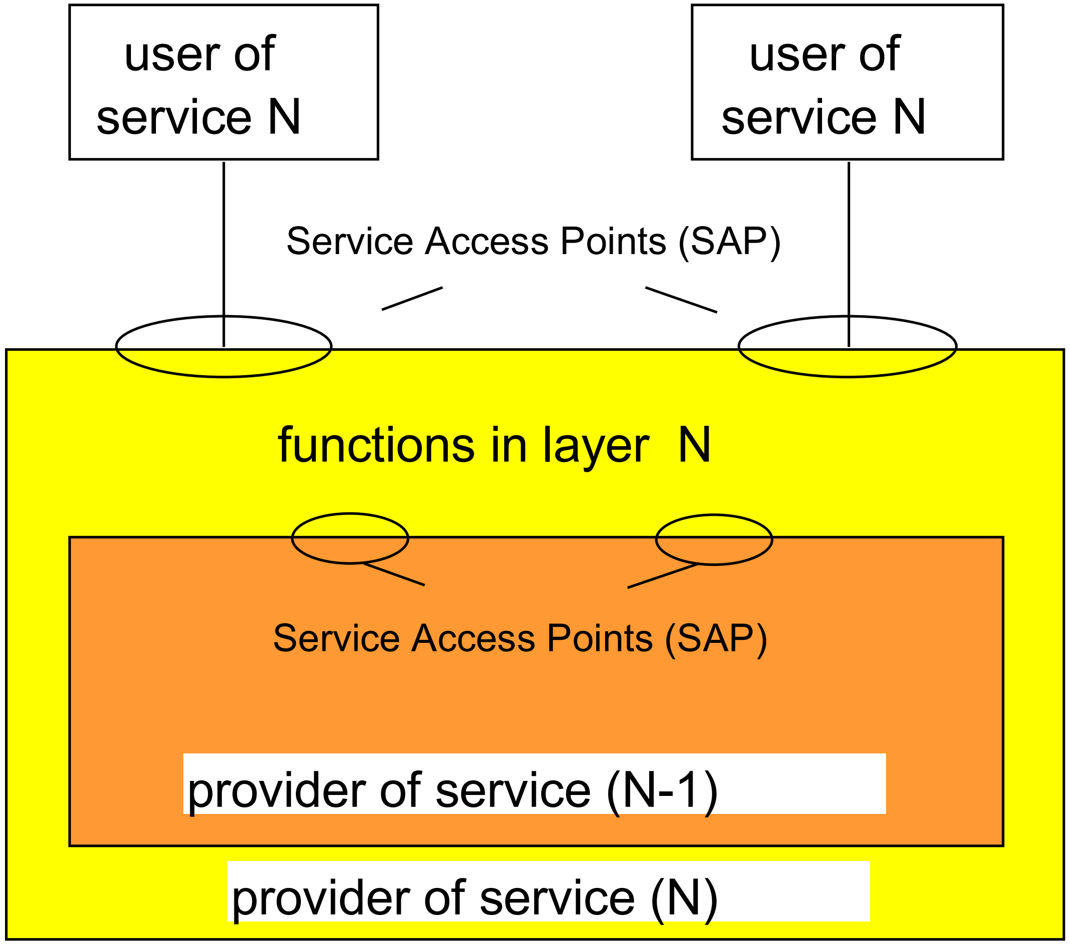


Protocol Data Units and Service Data Units



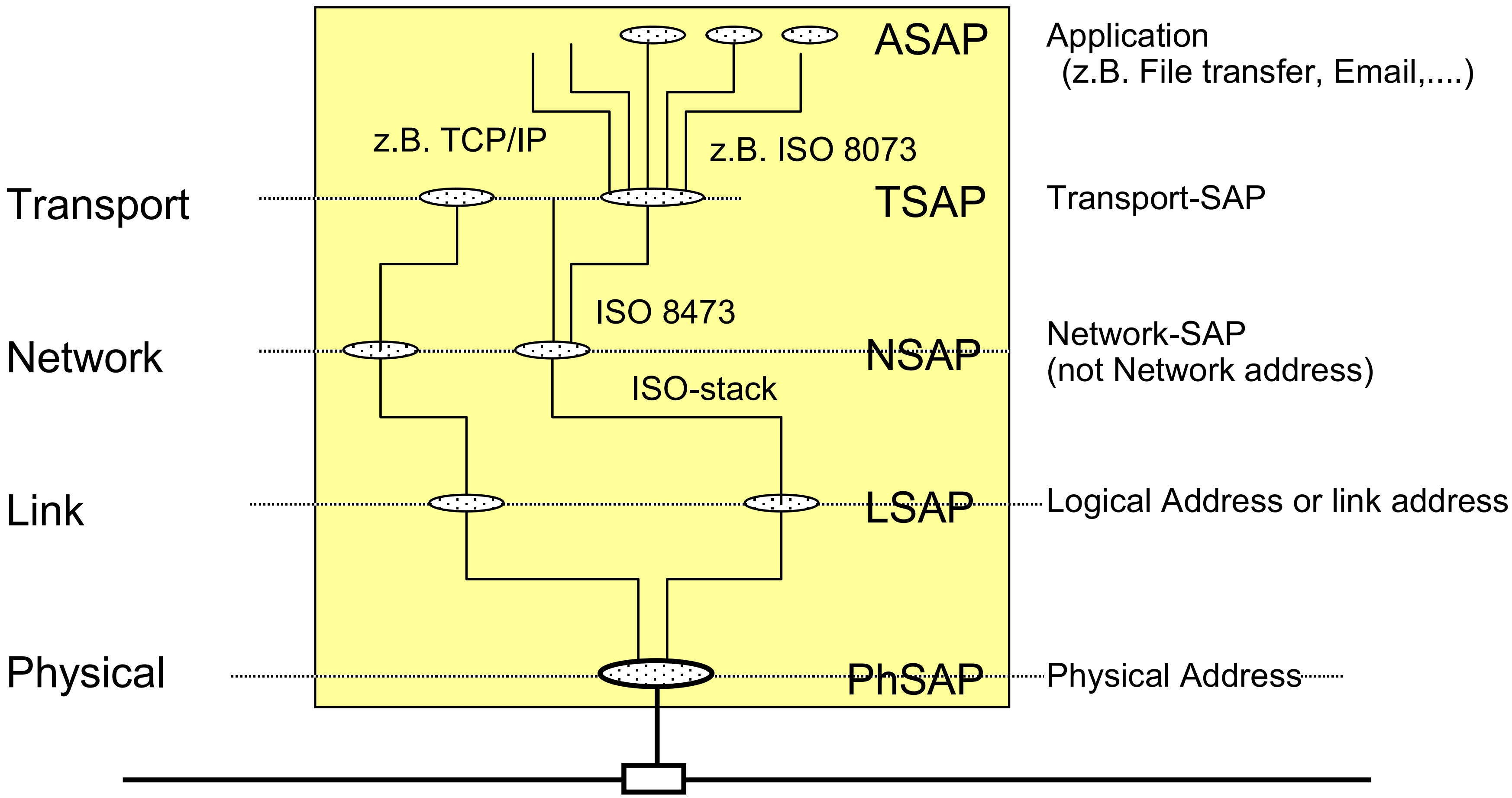
Layer N provides services to Layer N+1;
Layer N relies on services of Layer n-1

Service Access Points

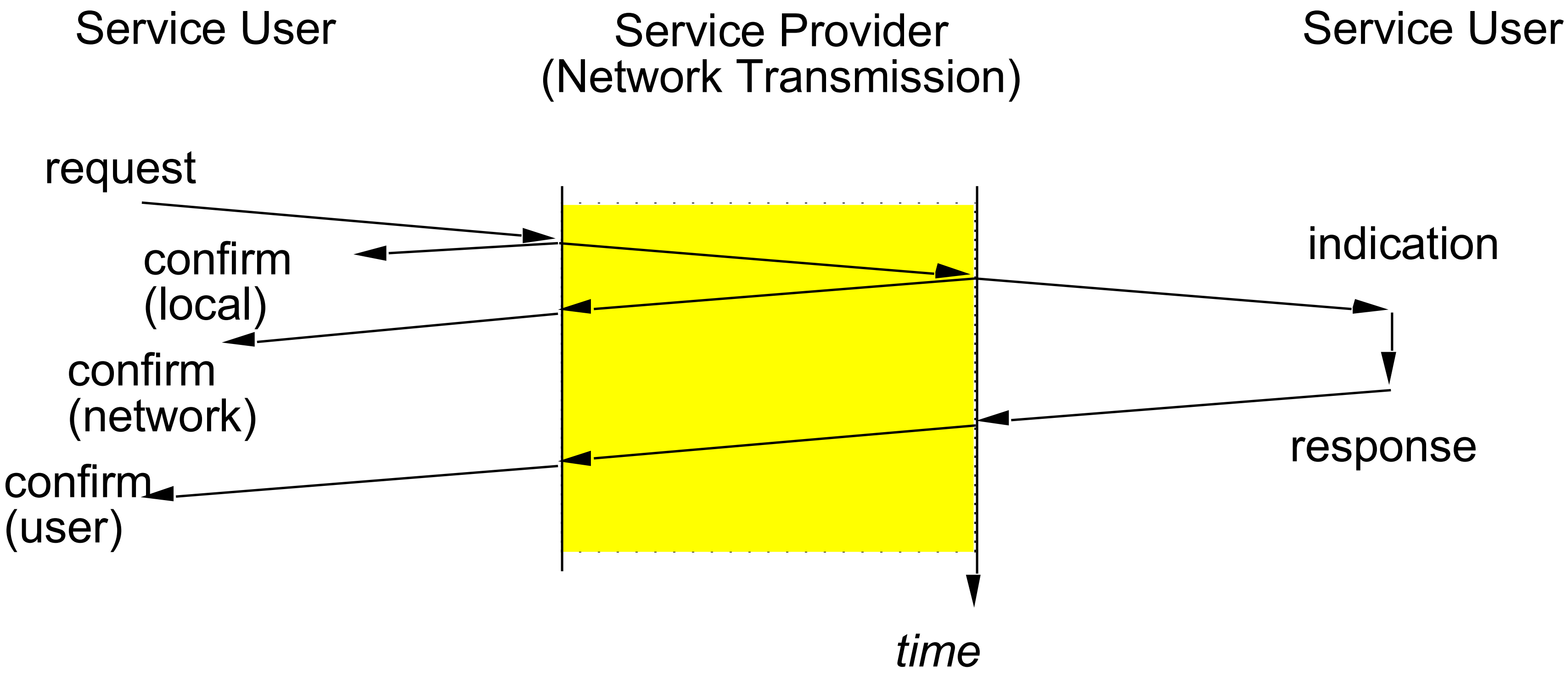


Service Access Points represent the interface to a service (name, address, pointer,...)

Address and SAPs in a device



Procedure call conventions in ISO



OSI implementation

OSI should be considered as a model, not as an implementation guide

The idea of independent layers is a useful as a way of thinking, not the best implementation.

Even if many claim to have "OSI"-conformant implementation, it cannot be proven.

IEC published about 300 standards which form the "OSI" stack, e.g.:

ISO/IEC 8473-2:1996 Information technology -- Protocol for providing the connectionless-mode network service --

ISO/IEC 8073:1997 Information technology -- Open Systems Interconnection -- Protocol for providing the connection-mode transport service

ISO/IEC 8327-1:1996 Information technology -- Open Systems Interconnection -- Connection-oriented Session protocol: Protocol specification

ISO/IEC 8649:1996 Information technology -- Open Systems Interconnection -- Service definition for the Association Control Service Element

ISO 8571-2:1988 Information processing systems -- Open Systems Interconnection -- File Transfer, Access and Management

Former implementations, which implemented each layer by an independent process, caused the general belief that OSI is slow and bulky.

OSI stack has not been able to establish itself against TCP/IP

OSI protocols in industry

Theory:

ISO-OSI standards should be used since they reduce specification and conformance testing work and commercial components exist

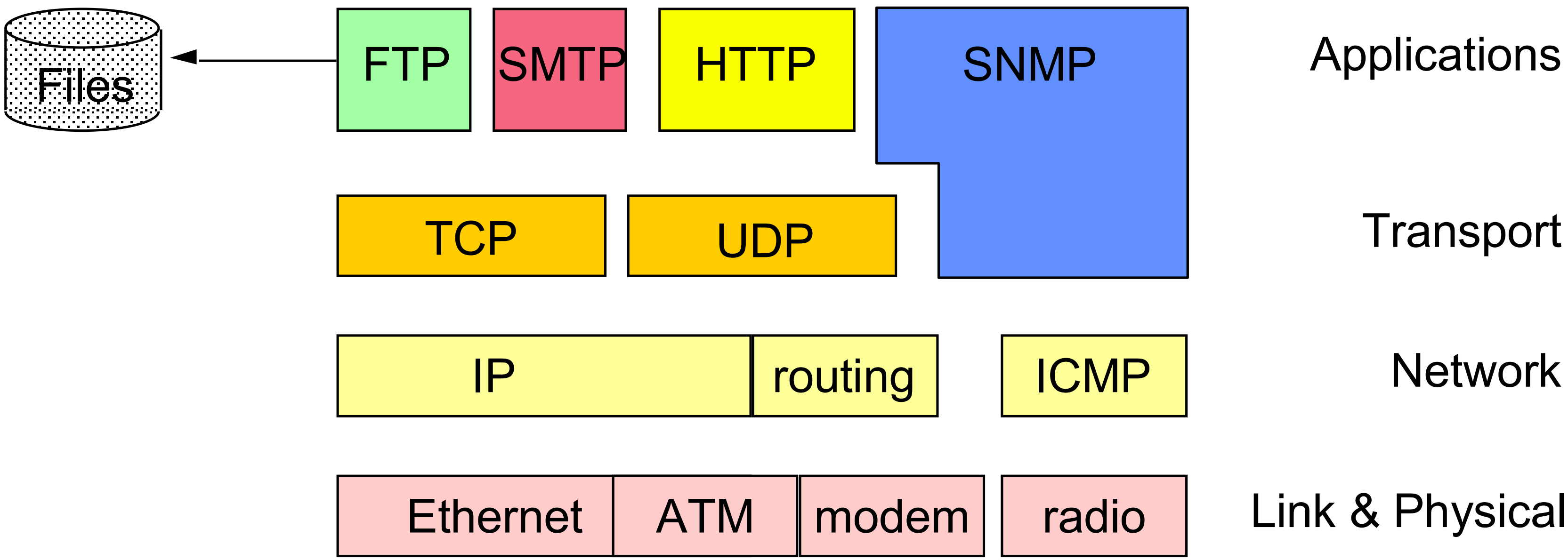
Reality:

- the OSI model is a general telecommunication framework - implementations considers feasibility and economics.
- the overhead of the ISO-OSI protocols (8073/8074) is not bearable with low data rates under real-time conditions.
- the OSI-conformant software is too complex:
  simple devices like door control or air-condition have limited power.
- the OSI model does not consider transmission of real-time data

Therefore:

- industrial busses use for real-time data a fast response access and for messages a simplified OSI communication stack
- the devices must be plug compatible: there are practically no options.
- Communication is greatly simplified by adhering to conventions negotiating parameters at run-time is a waste in closed applications.

TCP / IP structure



The TCP/IP stack is lighter than the OSI stack, but has about the same complexity

TCP/IP was implemented and used before being standardized.

Internet gave TCP/IP a decisive push

Conclusions

The OSI model is the reference for all industrial communication

Even when some layers are skipped, the concepts are generally implemented

Real-Time extensions to OSI are under consideration

TCP/IP however installs itself as a competitor to the OSI suite, although some efforts are made to integrate it into the OSI model

TCP/IP/UDP is becoming the backbone for all non-time critical industrial communication

TCP/IP/UDP is quickly displacing proprietary protocols.

Next generation TCP/IP (V6) is very much like the OSI standards.

Many embedded controllers come with an integrated Ethernet controller, an the corresponding real-time operating system kernel offers TCP/IP services

Like OSI, TCP protocols have delays counted in tens or hundred milliseconds, often unpredictable especially in case of disturbances.

For further reading: Motorola Digital Data Communication Guide

Assessment

1) Name the layers of the OSI model and describe their function

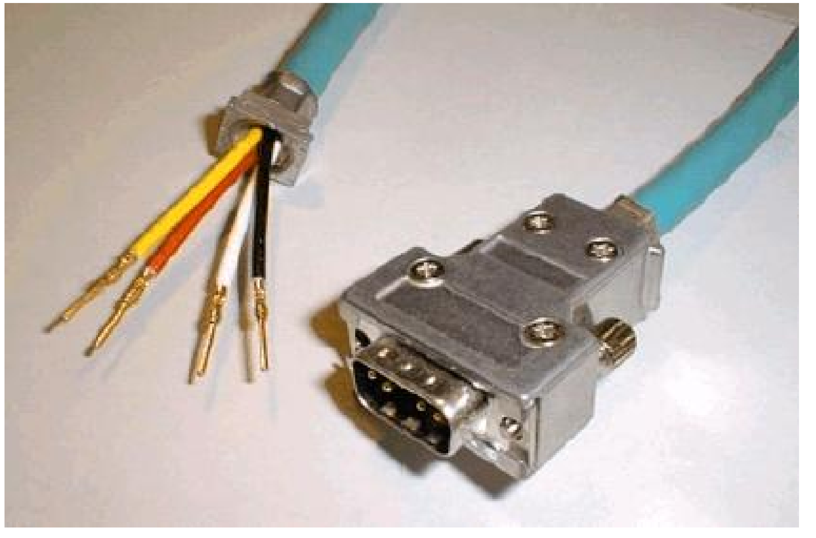
2) What is the difference between a repeater, a bridge and a router ?

3) What is encapsulation ?

4) By which device is an Appletalk connected to TCP/IP ?

5) How successful are implementations of the OSI standard suite ?

Industrial Automation
Automation Industrielle
Industrielle Automation



3. Industrial Communication Systems
Physical Layer
3.3.2 *Niveau physique*
Physische Schicht

Prof. Dr. H. Kirrmann
ABB Research Center, Baden, Switzerland

2004 June, HK

Physical Layer Outline

1. Layering

2. Topology

3. Physical media

4. Electric Signal coupling

5. Optical Fibres

6. Modulation

7. Synchronization

8. Encoding

9. Repeaters

OSI Model - location of the physical level

Application protocols	7	Application	All services directly called by the end user (Mail, File Transfer,...)
	6	Presentation	**Definition and conversion of the data formats (e.g. ASN 1)**
	5	Session	**Management of connections (e.g. ISO 8326)**
Transport protocols	4	Transport	**End-to-end flow control and error recovery (z.B. TP4, TCP)**
	3	Network	Routing, possibly segmenting (e.g. IP, X25)
	2	Link	Error detection, Flow control and error recovery, medium access (e.g. HDLC)
	1	Physical	Coding, Modulation, Electrical and mechanical coupling (e.g. V24)

Subdivisions of the physical layer

Physical Layer	medium-independent signalling	same for different media (e.g. coax, fibre, RS485)
	medium-dependent signalling	applies to one media (e.g. optical fibres)
	electrical / optical specifications	applies to one media type (e.g. 200µm optical fibres)
	mechanical specifications	defines the mechanical interface (e.g. connector type and pin-out)

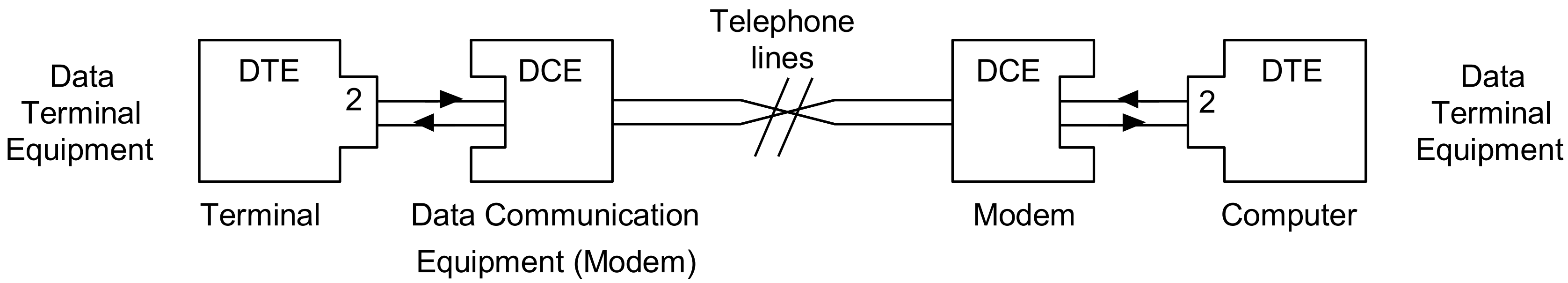
Concepts relevant to the physical layer

Topology	Ring, Bus, Point-to-point
Mechanical	Connector, Pin-out, Cable, Assembly
Medium	signals, transfer rate, levels
Channels	Half-duplex, full-duplex, broadcast
Control	Send, Receive, Collision
Modulation	Baseband, Carrier band, Broadband
Coding/Decoding	Binary, NRZ, Manchester,..
Synchronisation	Bit, Character, Frame
Flow Control	Handshake
Interface	Binary bit, Collision detection [multiple access] Signal quality supervision, redundancy control

Example: RS-232 - Mechanical-Electrical Standard

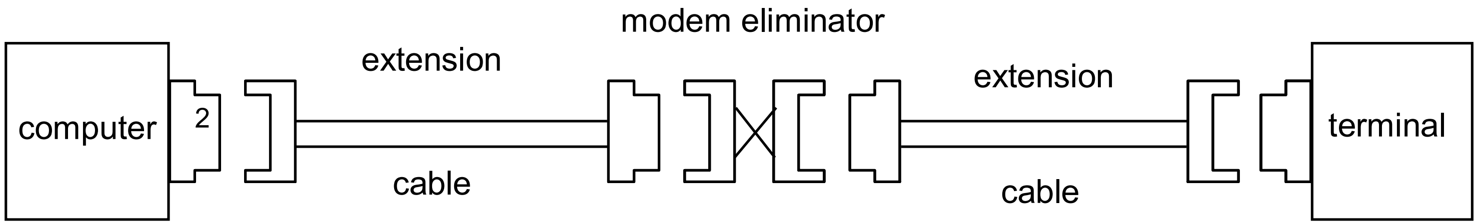
Originally developed for modem communication, now serial port in IBM-PCs

Topology:

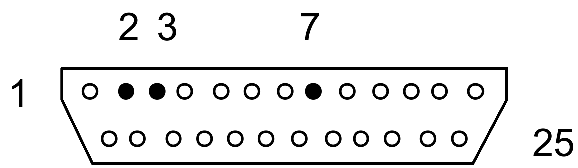


Cabling rules

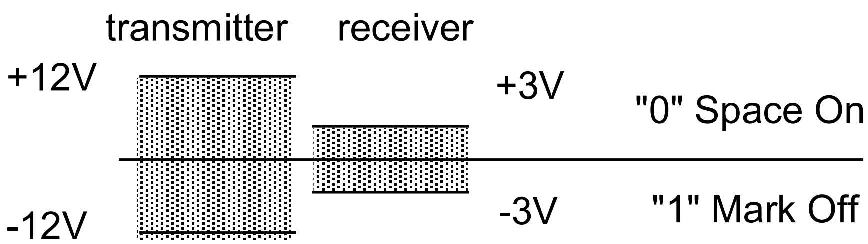
Tip: Do not use Modem cables, only Extension cables



Mechanical



Electrical:



Physical Layer Outline

1. Layering

2. Topology

3. Physical media

4. Electric Signal coupling

5. Optical Fibers

6. Modulation

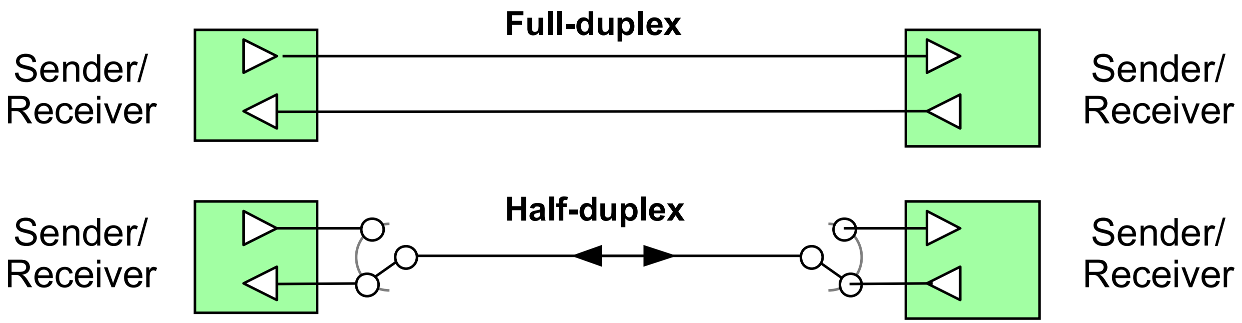
7. Synchronization

8. Encoding

9. Repeaters

Topology: Simplex, Half and Full Duplex

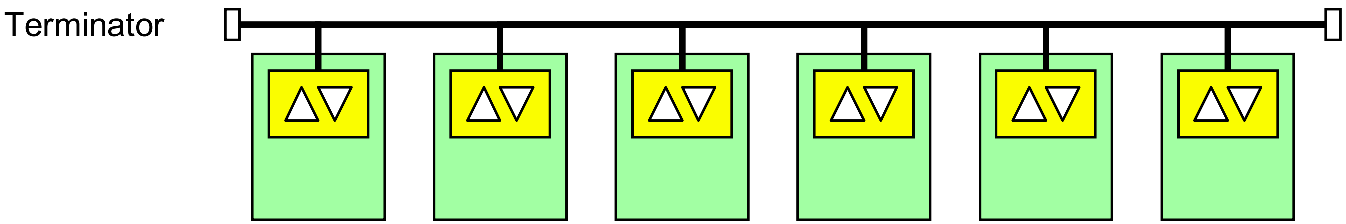
Link (Point -To-Point)



Examples:
RS232

Examples:
RS485

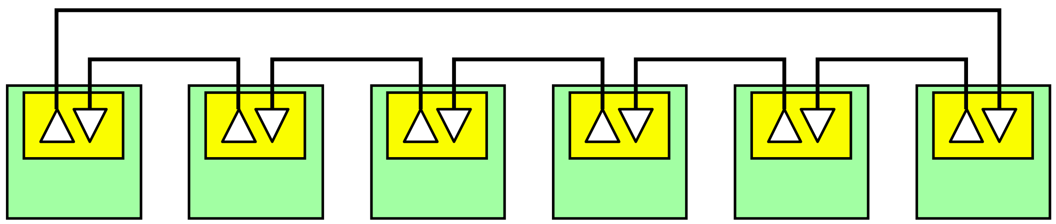
Bus (Half-Duplex, except when using Carrier Frequency over multiple bands)



Examples:

Ethernet, Profibus

Ring (Half-Duplex, except double ring)



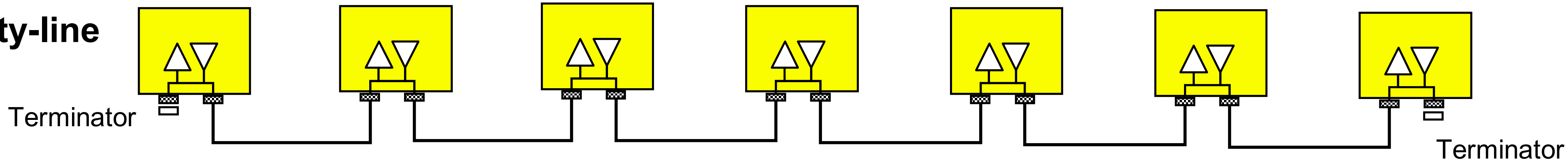
consists of point-to-point links

Examples:

SERCOS, Interbus-S

Bus topologies

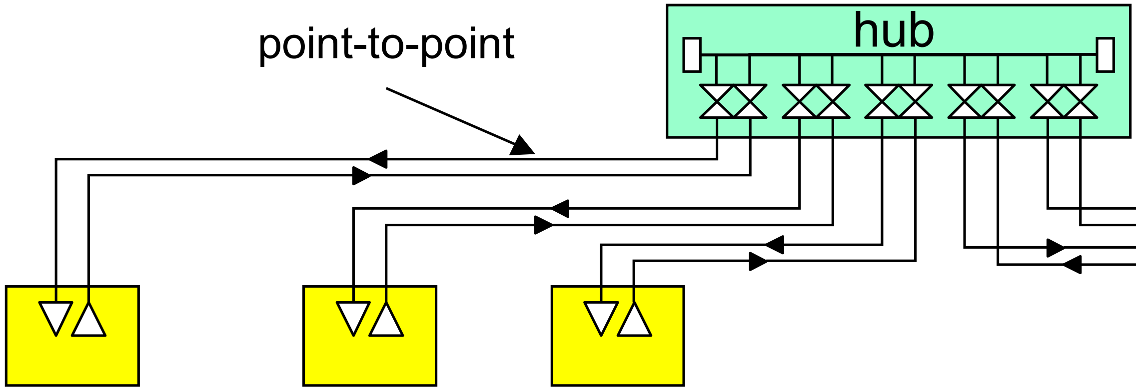
party-line



advantage: little wiring

disadvantages: easy to disrupt, high attenuation and reflections, no fibres

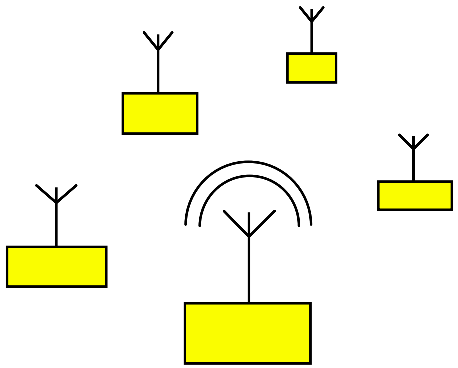
star



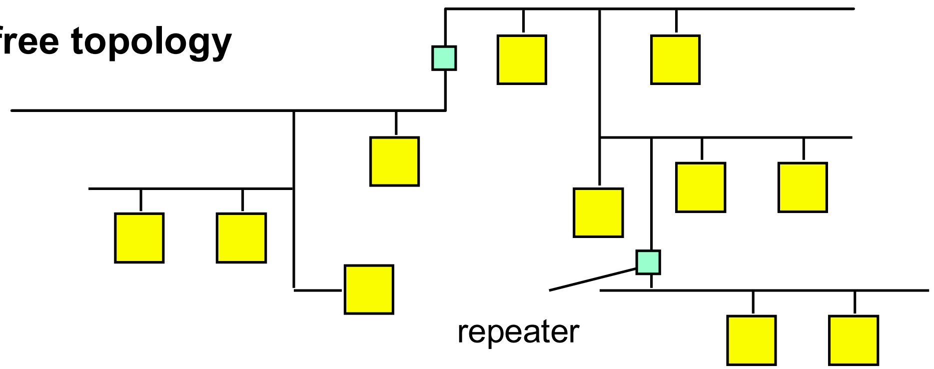
advantage: robust point-to-point links, can use fibres

disadvantage: requires hub, more wiring

radio

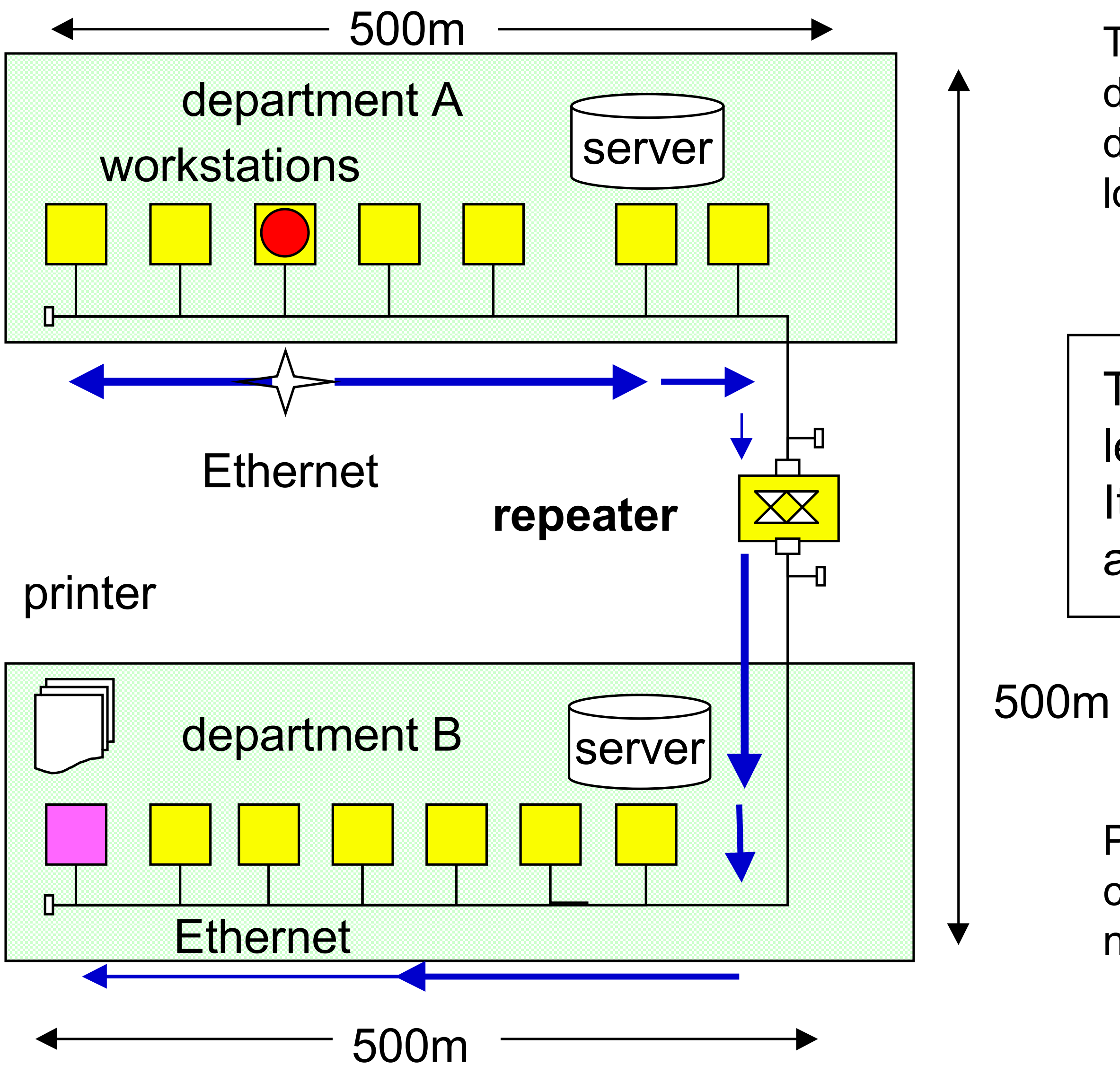


free topology



a bus is a broadcast medium (delays come from propagation and repeaters)

Repeater

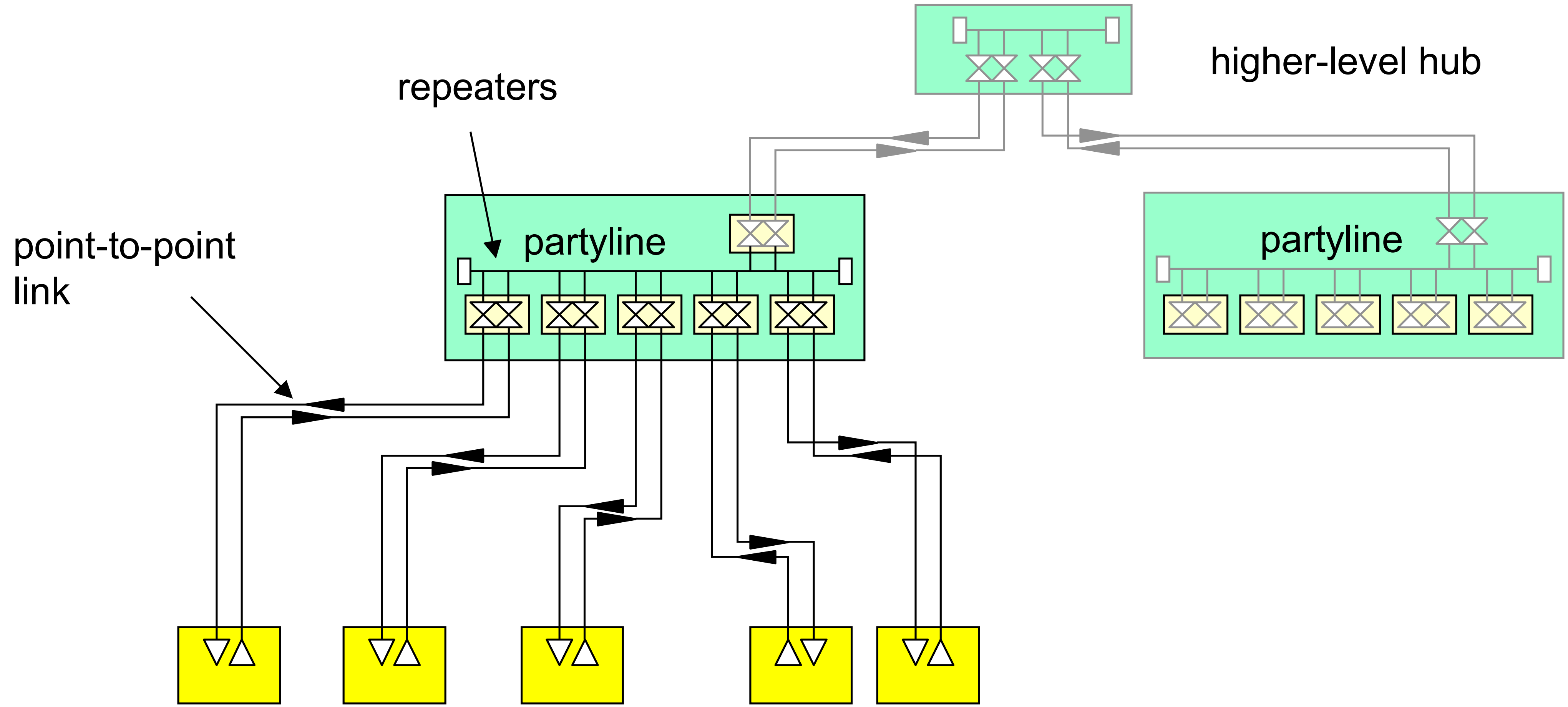


To connect a workstation of department A to the printer of department B, the cable becomes too long and the messages are corrupted.

The repeater restores signal levels and synchronization.
It introduces a signal delay of about 1..4 bits

Physically, there is only one Ethernet carrying both department's traffic, only one node may transmit at a time.

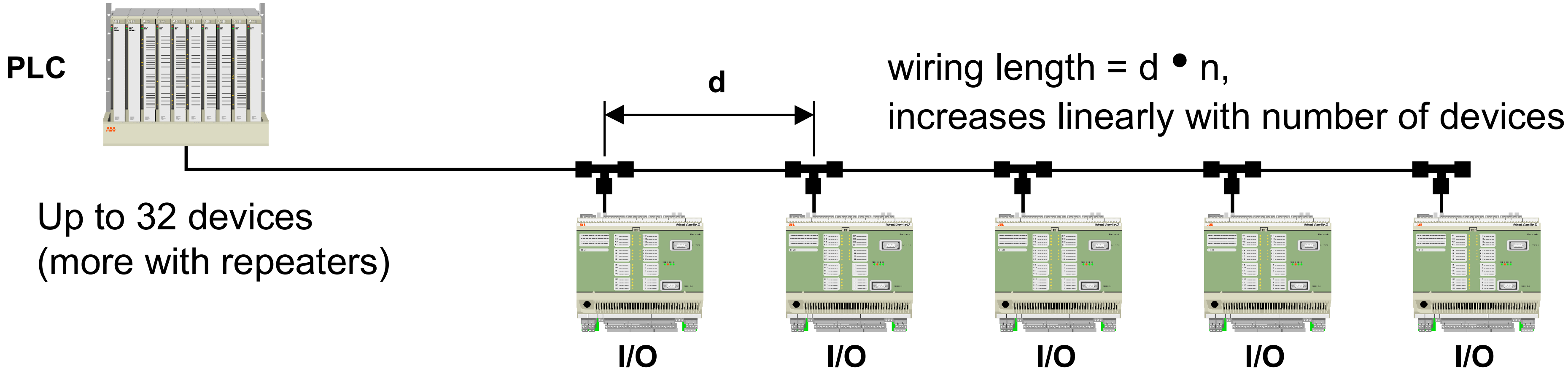
Bus: repeaters and hubs



hubs assemble point-to-point links to form a broadcast medium (bus)

Party-line (bus) and star wiring

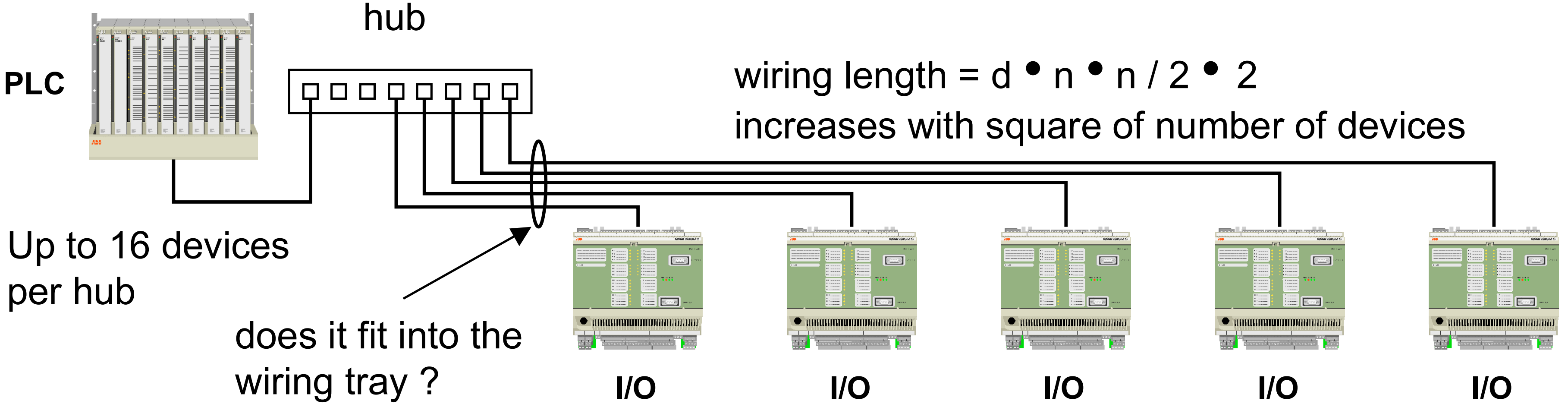
d = average distance between devices



wiring length = d • n,
increases linearly with number of devices

Up to 32 devices
(more with repeaters)

party-line wiring is well adapted to the varying topography of control systems



wiring length = d • n • n / 2 • 2
increases with square of number of devices

Up to 16 devices
per hub

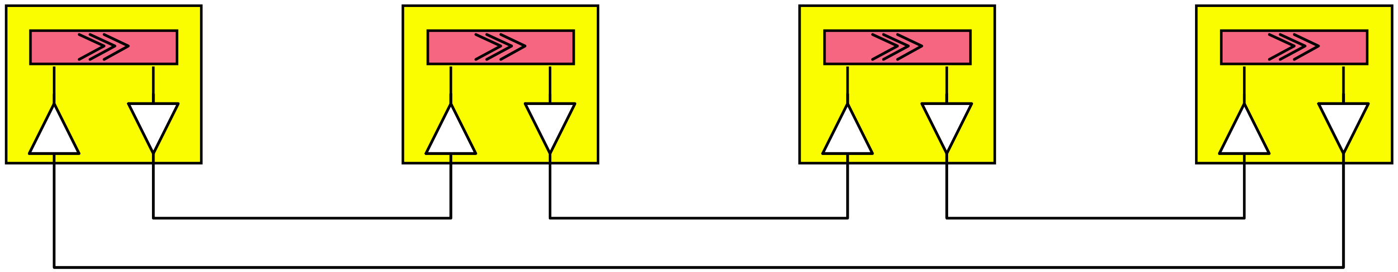
does it fit into the
wiring tray ?

star wiring may more than offset the advantage of field busses (reduced wiring) and leads to more concentration of I/O on the field devices.

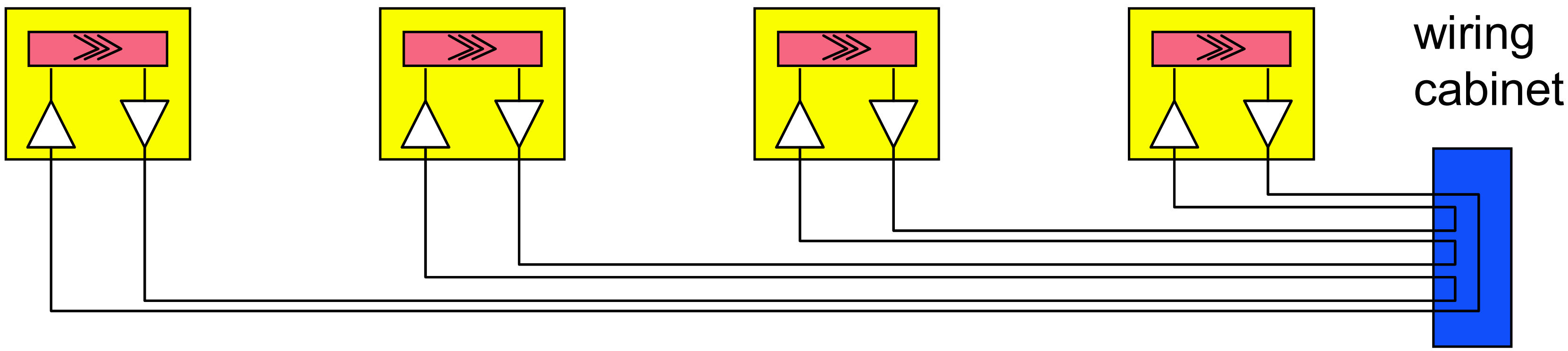
Rings

a ring consists only of point-to-point links
Each node can interrupt the ring and introduce its own frames

classical ring



ring in floor wiring



The wiring amount is the same for a bus with hub or for a ring with wiring cabinet.
Since rings use point-to-point links, they are well adapted to fibres

Physical Layer Outline

1. Layering

2. Topology

3. Physical media

4. Electric Signal coupling

5. Optical Fibres

6. Modulation

7. Synchronization

8. Encoding

9. Repeaters

Media (bandwidth x distance)

		Transfer rate (Mbit/s)			Costs (Fr/m)	Electromagnetic Compatibility
		200m	700m	2000m		
optical fibres	single mode	2058	516	207	5.5	very good
	multimode	196	49	20	6.5	very good
	plastic	1	0.5	-	6.5	very good
coaxial cables	50 Ohm	20	8	1	1.2	good
	75 Ohm TV 1/2"	12	2.5	1.0	2.2	good
	93-100 Ohm	15	5	0.8	2.5	good
twisted wire	twinax	8	0.9	0.2	3.5	very good
	individually shielded (STP)	2	0.35	0.15	.5	very good
	group shielding (UTP)	1	0.3	0.1	1	good (crosstalk) regular (foreign)
	Telephone cable	0.2	0.1	0.05	0.2	good (crosstalk) bad (foreign)
others	Power line carrier	1	0.05	0.01	-	very bad
	Radio	1	1	1	-	bad
	Infrared	0.02	0	0	-	good
	ultrasound	0.01	0	0	-	bad

the bandwidth x distance is an important quality factor of a medium

Physical Layer Outline

1. Layering

2. Topology

3. Physical media

4. Electric Signal coupling

5. Optical Fibres

6. Modulation

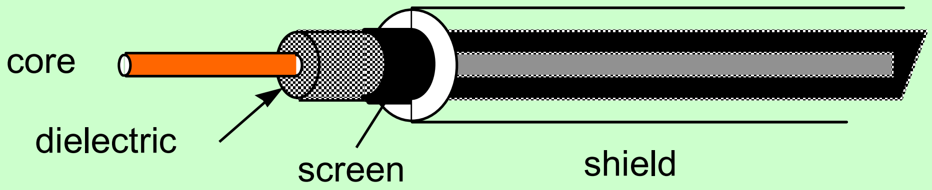
7. Synchronization

8. Encoding

9. Repeaters

Electrical: Transmission media

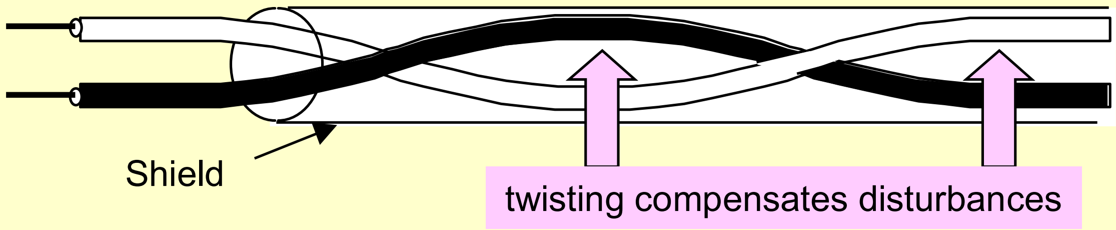
Coaxial cable



Zw = 50Ω ... 100Ω

inflexible, costly,
low losses
10 MHz..100 MHz

Shielded twisted wire (Twinax)



Zw = 85Ω..120Ω

flexible, cheap,
medium attenuation
~1 MHz..12 MHz

Unshielded twisted wire

Telephone

very cheap,
sensible to perturbations

Uncommitted wiring (e.g. powerline com.)

numerous branches, not terminated,
except possibly at one place

very cheap,
very high losses and disturbances,
very low speed (~10 ..100 kbit/s)

+

1) Classical wiring technology,
2) Well understood by electricians in the field
3) Easy to configure in the field
4) Cheap (depends if debug costs are included)

−

1) low data rate
2) costly galvanic separation (transformer, optical)
3) sensible to disturbances
4) difficult to debug, find bad contacts
5) heavy

Electrical: Twisted wire pair

characteristic impedance most used in industrial environment:
120 Ohm for bus, 150 Ohm for point-to-point.

Standard from the telecommunication world: ISO/IEC 11801

Cat 5 (class D): 100 MHz, RJ 45 connector
Cat 6 (class E): 200 MHz, RJ 45 connector
Cat 7 (class F): 600 MHz, in development

These are only for point-to-point links ! (no busses)

Electrical: What limits transmission distance ?

All parameters

- Characteristic impedance
- Attenuation
- Linear resistance
- Linear capacitance
- Cross talk
- Common-mode
- Shield protection

are frequency-dependent

Attenuation: copper resistance, dielectric loss.

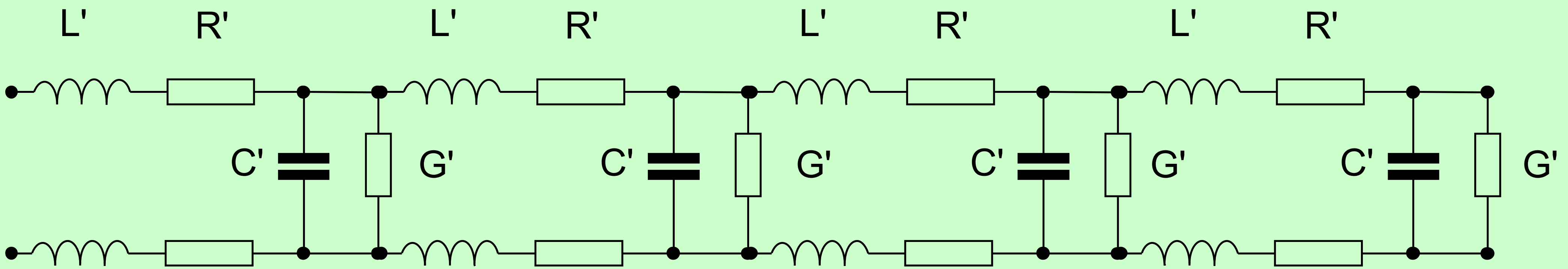
Frequency dependent losses cause dispersion (edges wash-out):

Signal reflection on discontinuities (branches, connectors) cause self-distortions

Consider in cables

- characteristic impedance (Zw) (must match the source impedance)
- attenuation (limits distance and number of repeaters)
- bending radius (layout of channels)
- weight
- fire-retardant isolation

lumped line model



specific inductance (H/m) specific capacitance (F/m)
specific resistance (Ω/m) specific conductance (S/m)

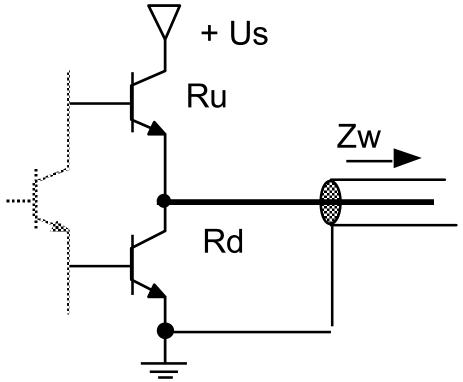
$$Zw = \sqrt{\frac{L'}{C'}}$$

Electrical: Signal Coupling Types

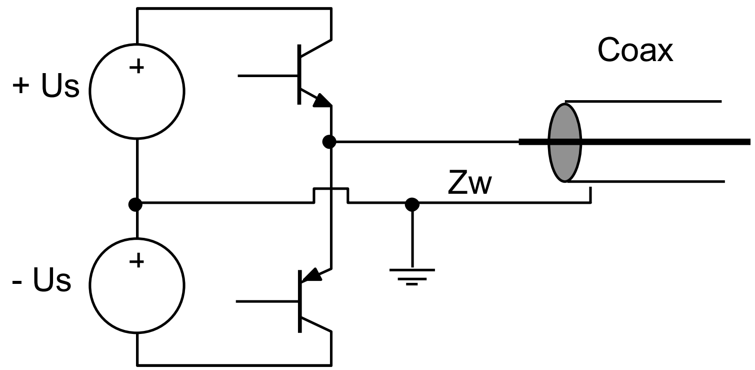
Resistive	direct coupling
	driver on line without galvanic coupling
	collision possible when several transmitters active
	Wired-OR combination possible
	cheap as long as no galvanic separation is required (opto-coupler)
	good efficiency
Inductive	transformer-coupling
	galvanic separation
	good electromagnetic compatibility (filter)
	retro-action free
	good efficiency
	signal may not contain DC-components
	bandwidth limited
Capacitive	capacitor as coupler
	weak galvanic separation
	signal may not contain DC components
	cheap
	not efficient

Electrical: Resistive (direct) coupling

Unipolar, unbalanced

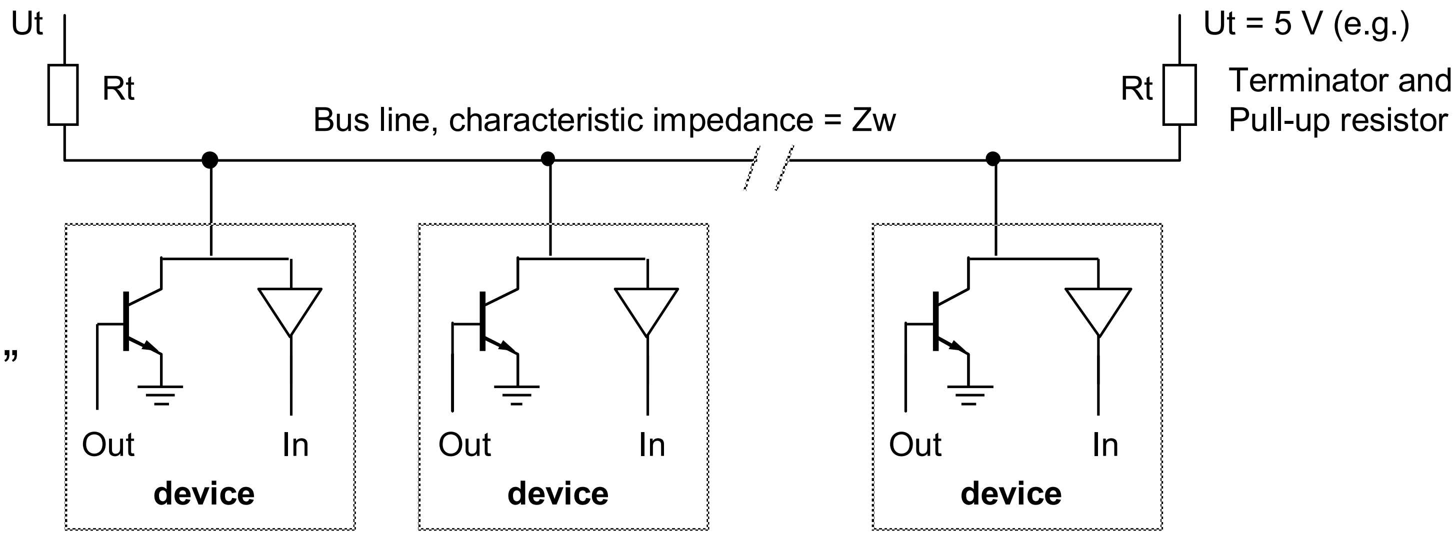


Bipolar, unbalanced



Open Collector
(unbalanced)

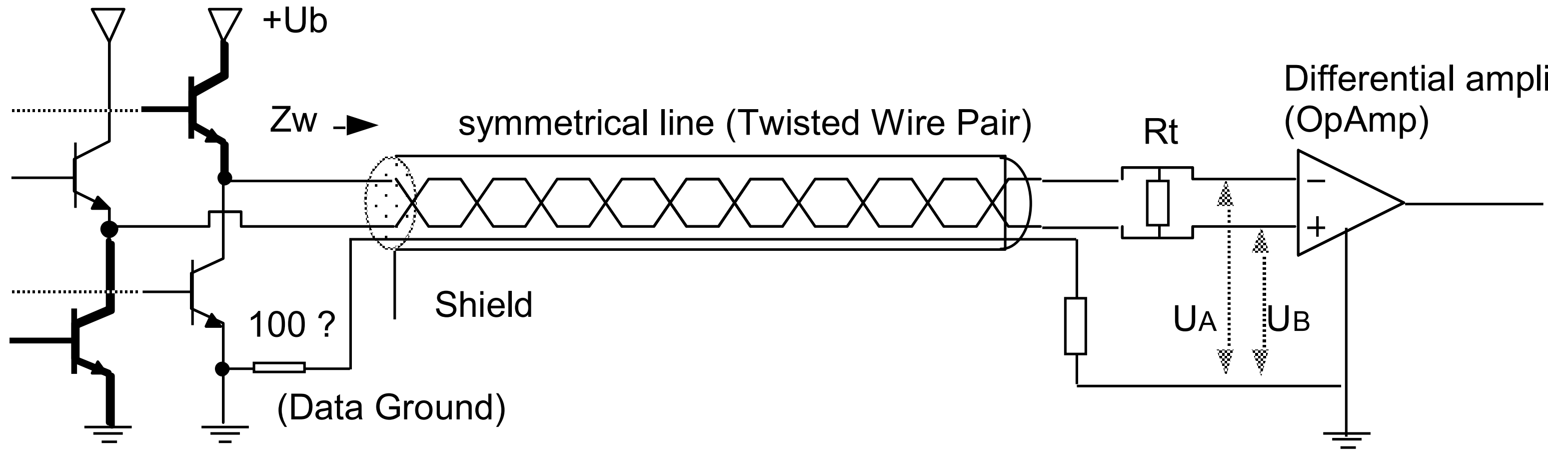
wired-OR behaviour
("Low" wins over "High")



Electrical: Balanced Transmission

Differential transmitter and receiver

+ good rejection of disturbances on the line and common-mode
- double number of lines



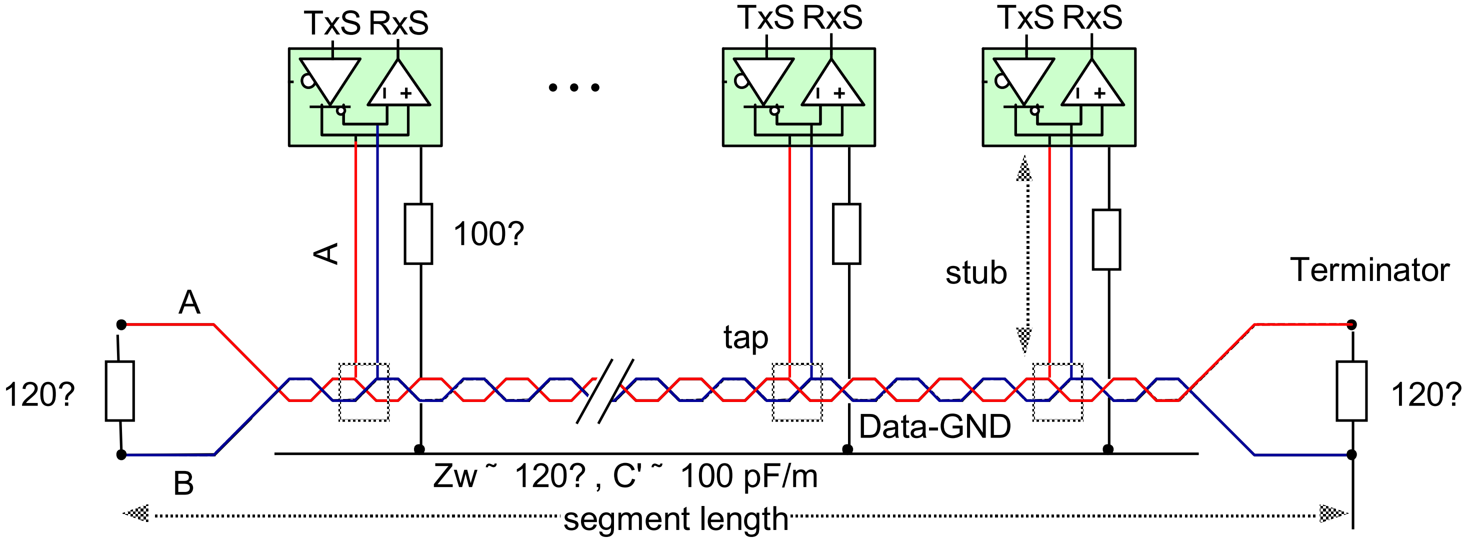
Used for twisted wire pairs (e.g. RS422, RS485)

Common mode rejection: influence of a voltage which is applied simultaneously on both lines with respect to ground.

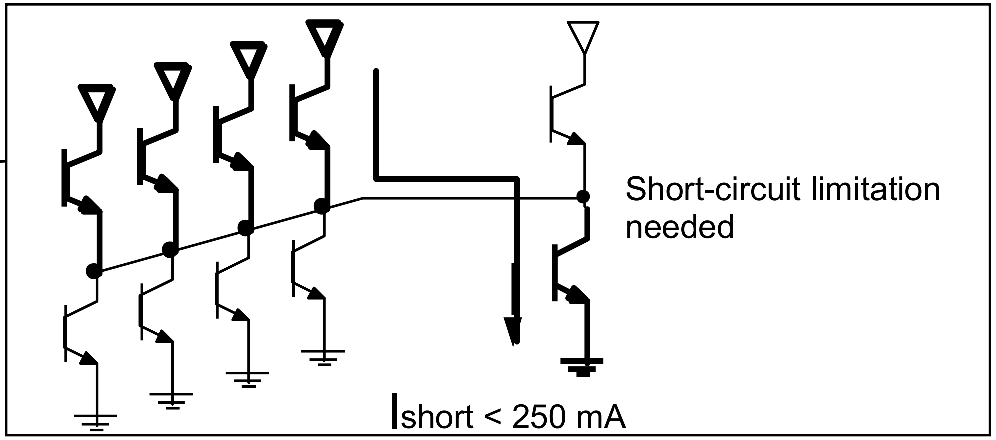
The shield should not be used as a data ground (inductance of currents into conductors)

Electrical: RS-485 as an example of balanced transmission

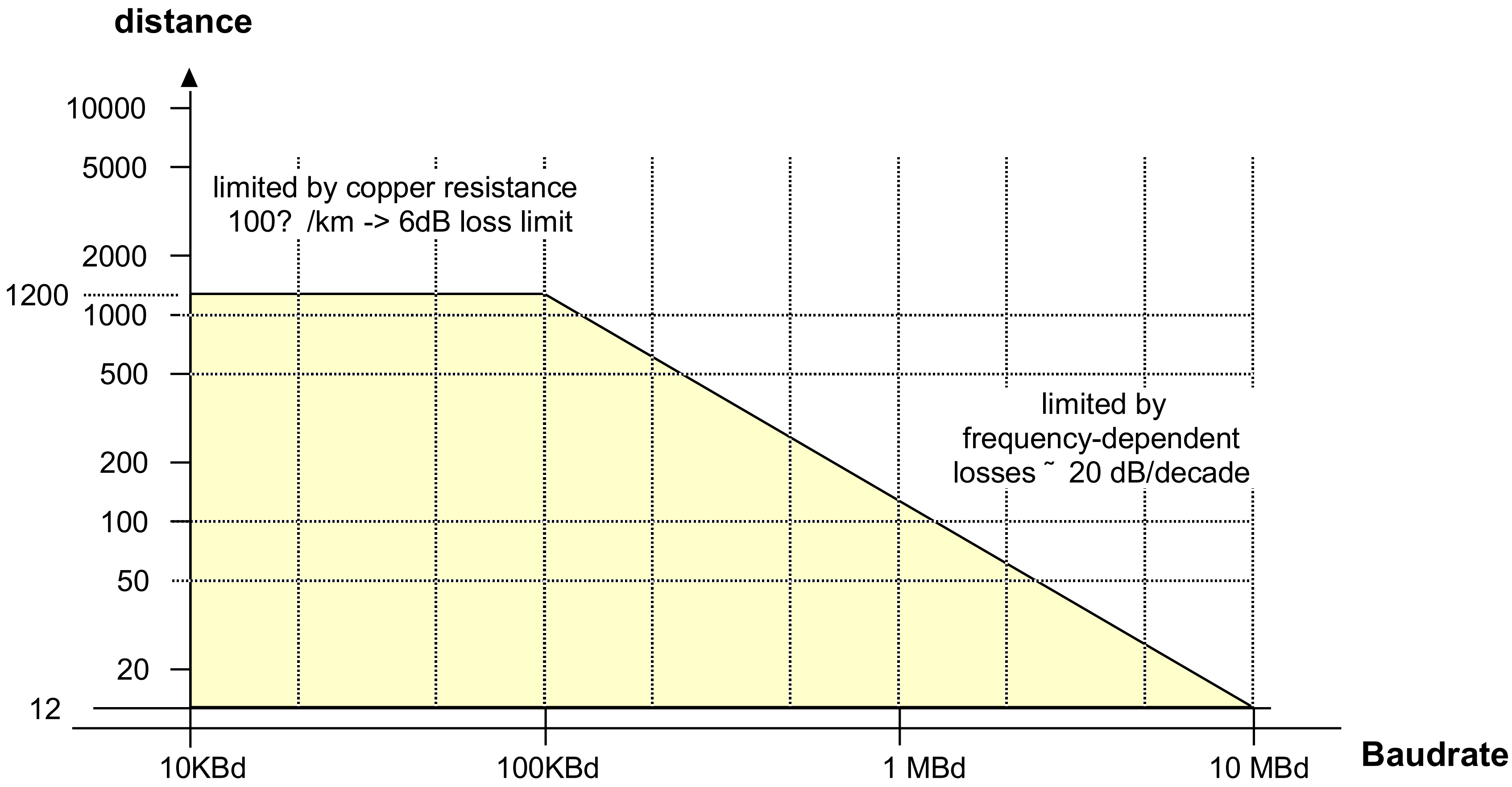
The most widely used transmission for busses over balanced lines (not point-to-point)



multiple transmitter allowed



Electrical: RS-485 Distance x Baudrate product



limited by: Cable quality: attenuation, capacitive loading, copper resistance

Signal/Noise ratio, disturbances

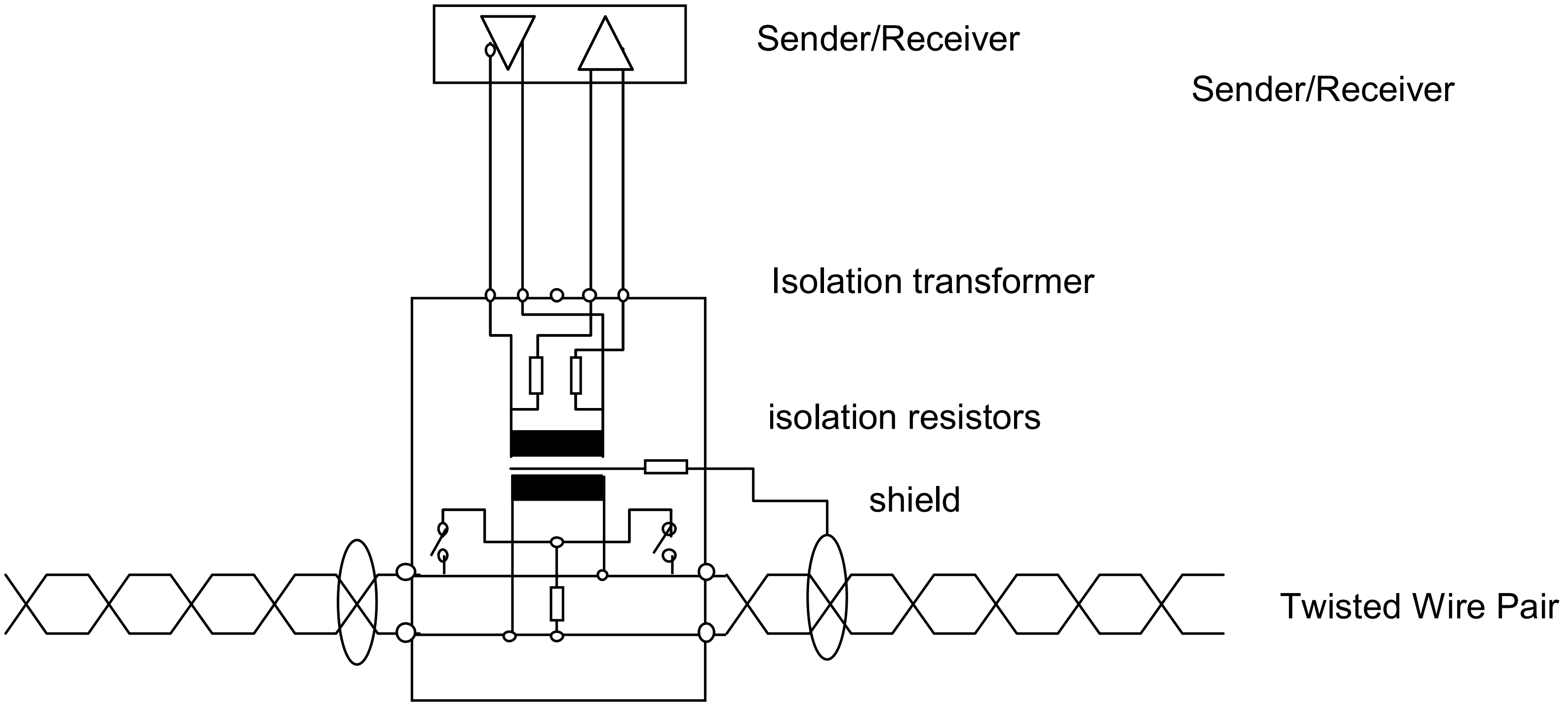
Receiver quality and decoding method

Electrical: Transformer Coupling

Provides galvanic separation, freedom of retro-action and impedance matching

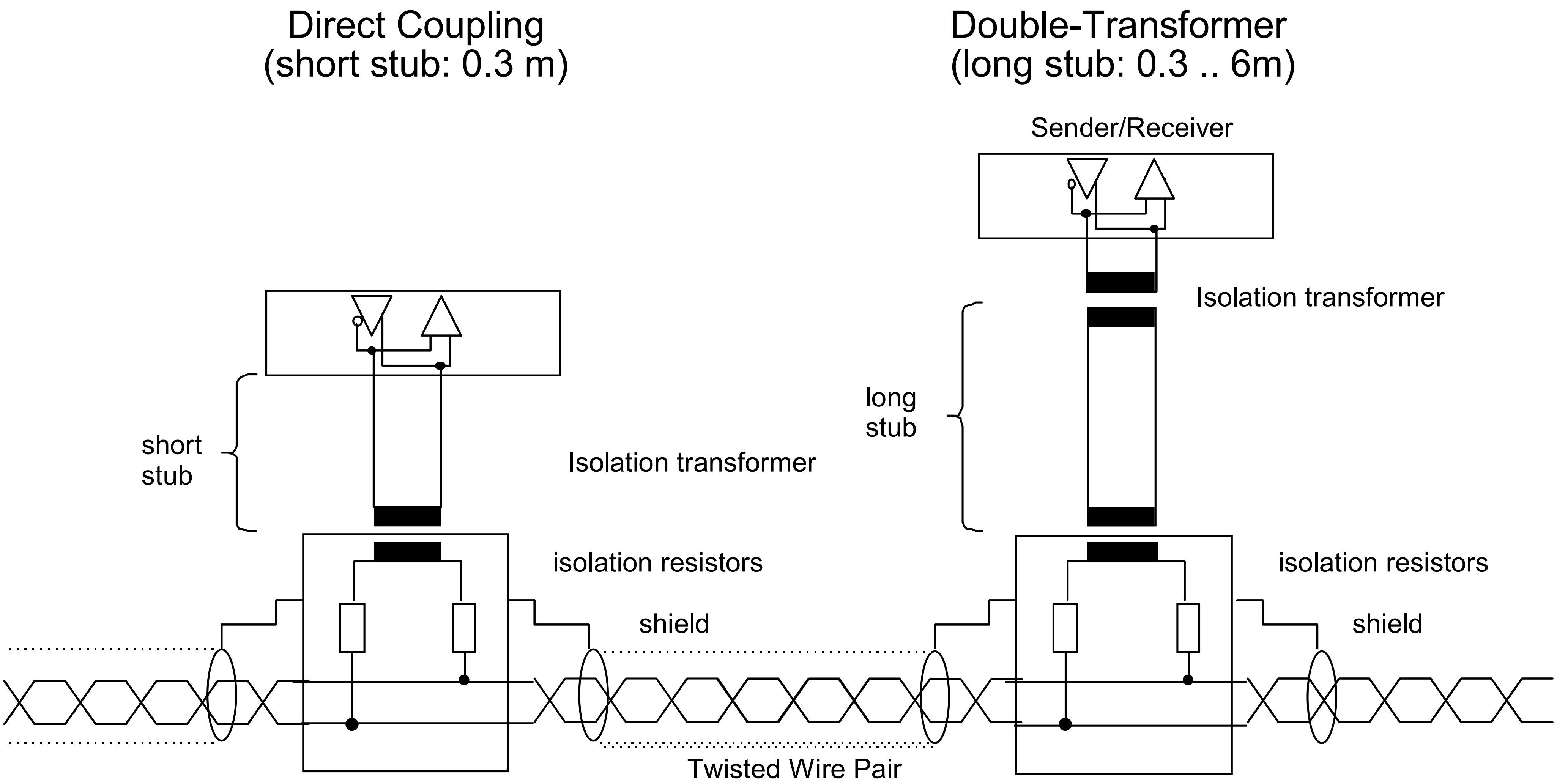
but: no DC-components may be transmitted.

cost of the transformer depends on transmitted frequency band (not center frequency)



Source: Appletalk manual

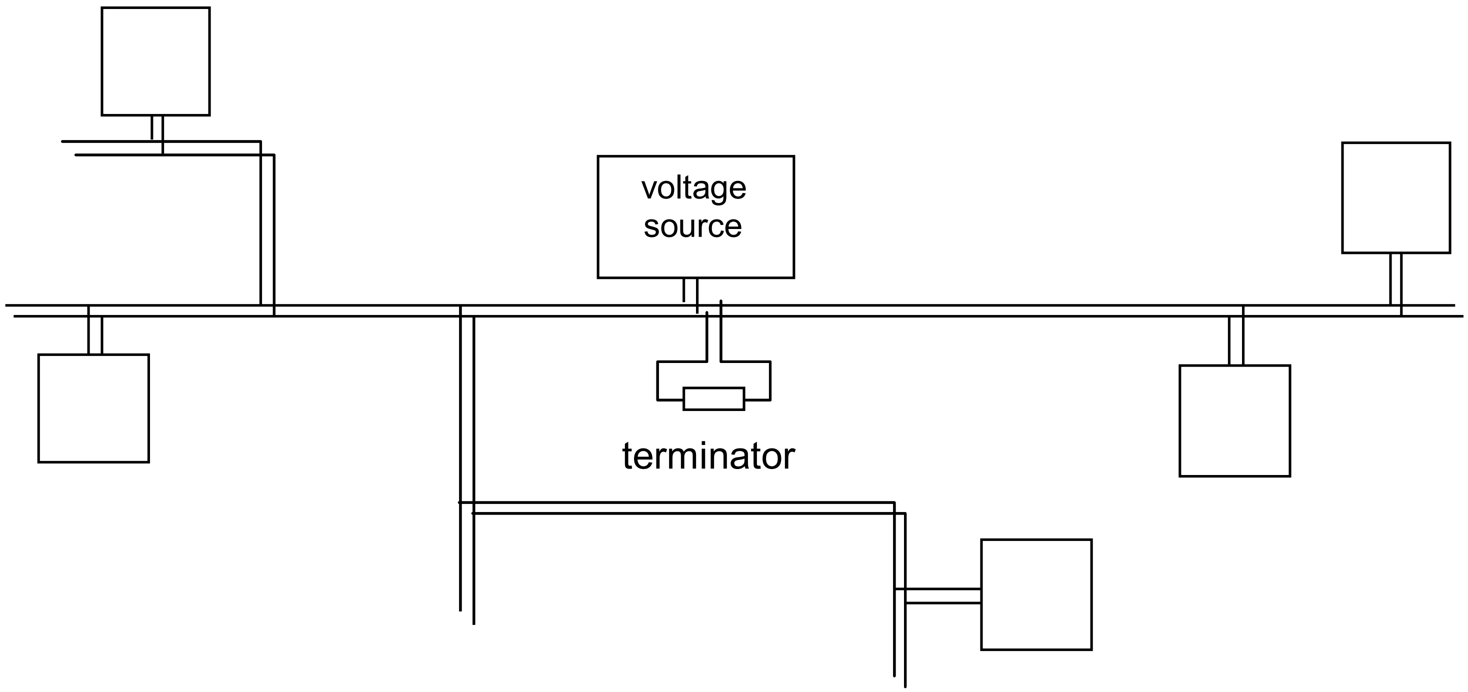
Electrical: MIL 1553 as an example of transformer coupling



Extract from: MIL-STD-1553

MIL 1553 is the standard field bus used in avionics since the years '60 - it is costly and obsolete

Electrical: Free topology wiring



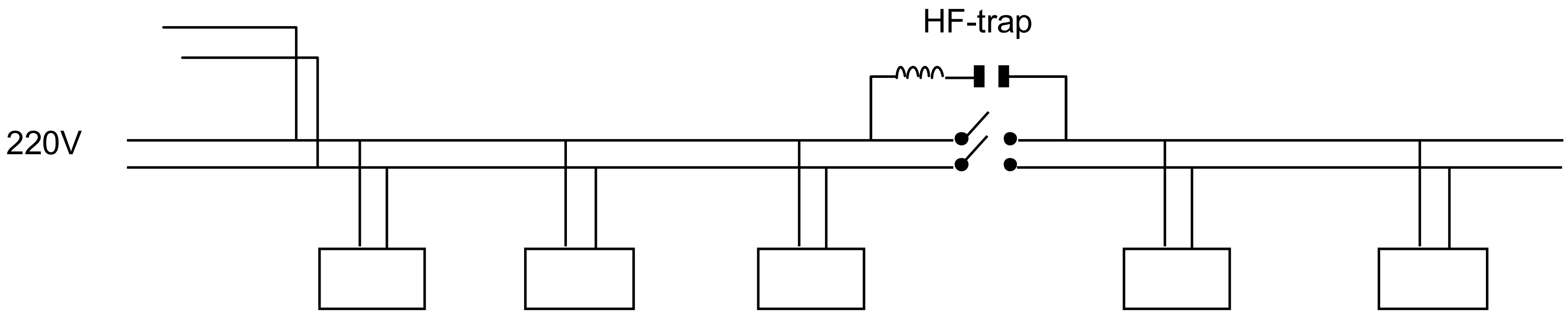
Free topology is used to connect scattered devices which are usually line-powered.
Main application: building wiring

Transmission medium is inhomogeneous, with many reflections and discontinuities.

Radio techniques such as echo cancellation, multiple frequency transmission (similar to ADSL) phase modulation, etc... are used.

Speed is limited by the amount of signal processing required (typically: 10 kbit/s)

Electrical: Power Line Carrier technology



A free-topology medium using the power lines as carrier.

Used for retrofit wiring (revamping old installations) and for minimum cabling

Capacitive or inductive coupling, sometimes over shield

Problems with disturbances, switches, transformers, HF-traps, EMC,..

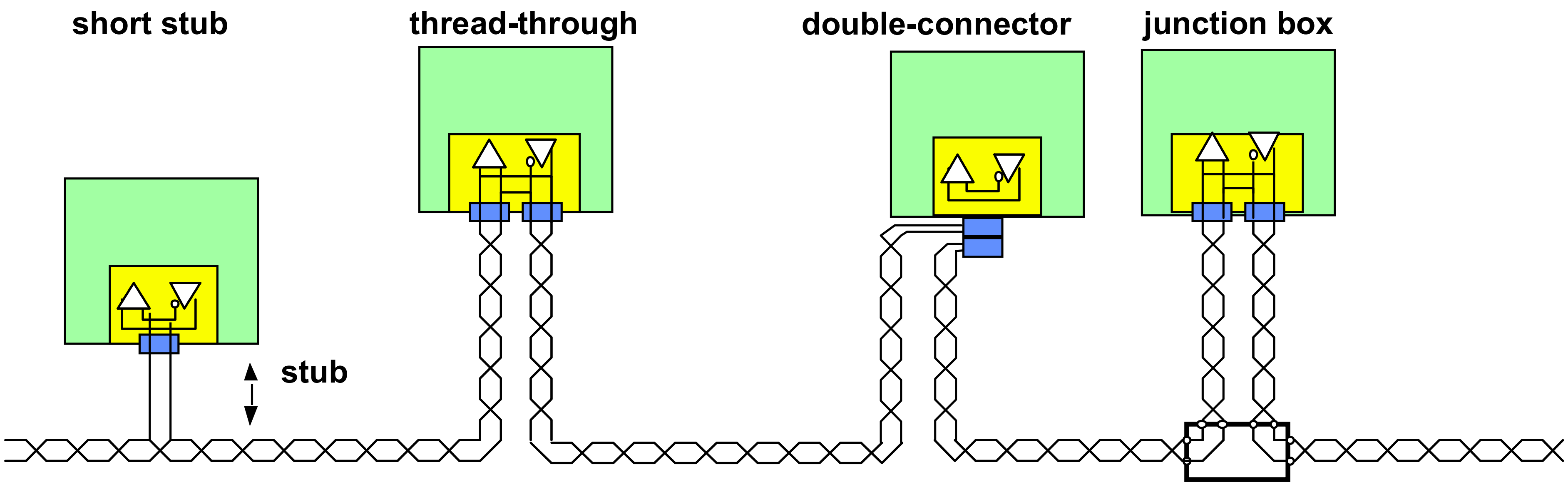
Proposed for voice communication over the last mile (ASCOM)

Difficult demodulation

Low data rates (< 10 kbit/s)

Applications: remote meter reading, substation remote control

Electrical: Mechanical Connecting devices to an electrical bus



1 connector live insertion	2 connectors no live insertion	2 connectors live insertion installation ?	1 connector live insertion (costly) junction box

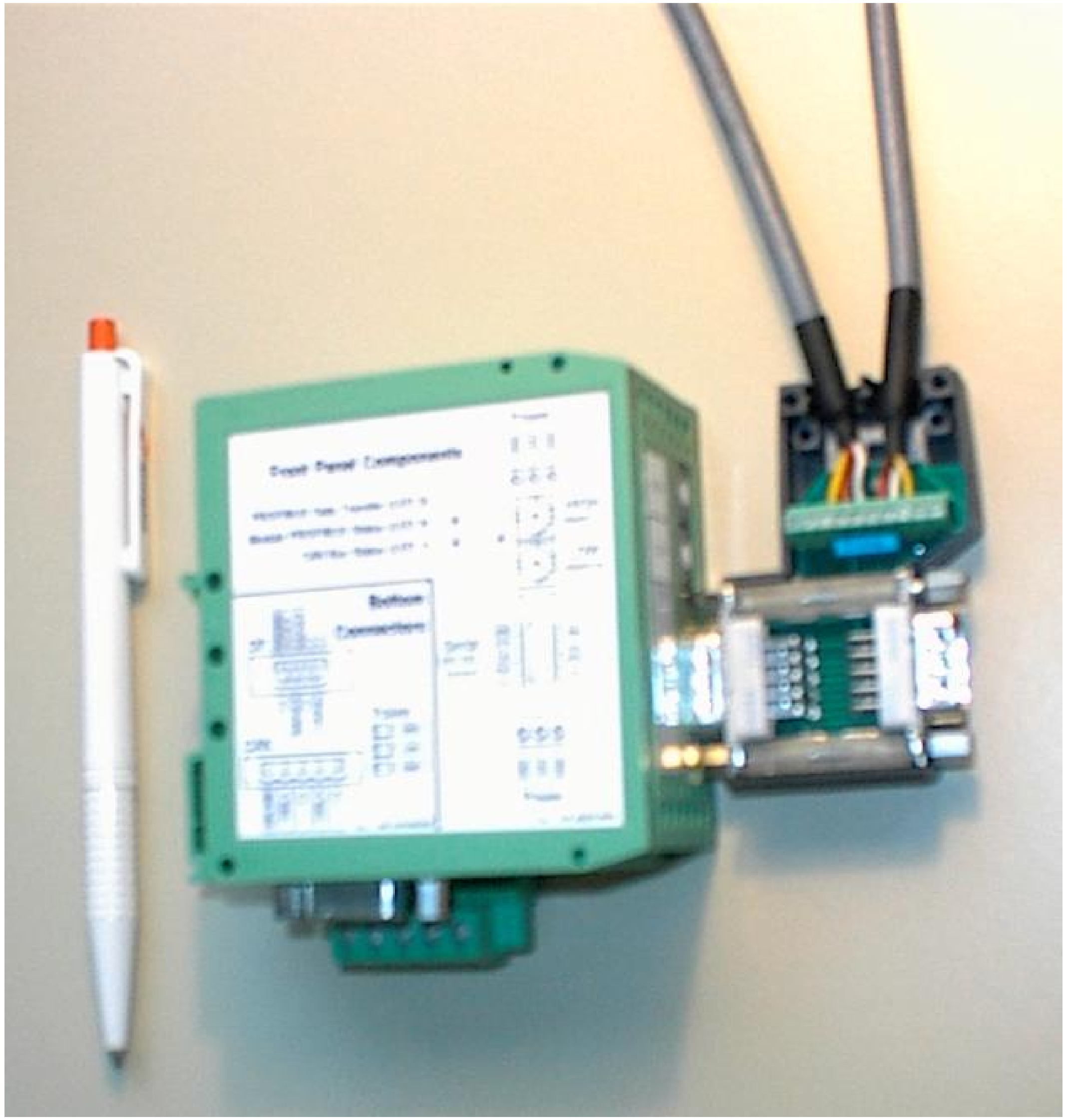
Electrical wiring at high speed requires careful layout
(reflections due to device clustering or other discontinuities, crosstalk, EM disturbances)

some applications require live insertion (power plants, substations)
time-outs (causing emergency stop) limit disconnection time

installation or operational requirements may prohibit screws (only crimping)

Practical solution to live insertion

Offers life insertion
but costs a lot
(also in place)



Electrical: Connectors

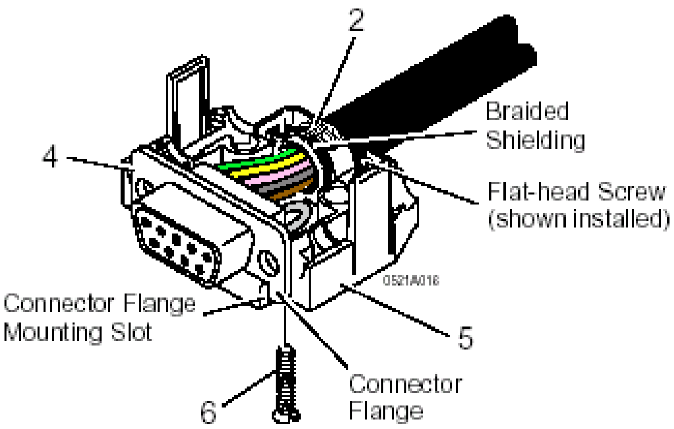
Field busses require at the same time low cost and robust connectors.

The cheapest connectors come from the automobile industry (Faston clips)
and from telephony (RJ11, RJ 45)

However, these connectors are fragile. They fail to comply with:
- shield continuity
- protection against water, dust and dirt (IP68 standard)
- stamping-proof (during commissioning, it happens that workers and vehicles pass over cables)

The most popular connector is the sub-D 9 (the IBM PC's serial port),
which exists in diverse rugged versions.

Also popular are Weidmann
and Phoenix connectors.

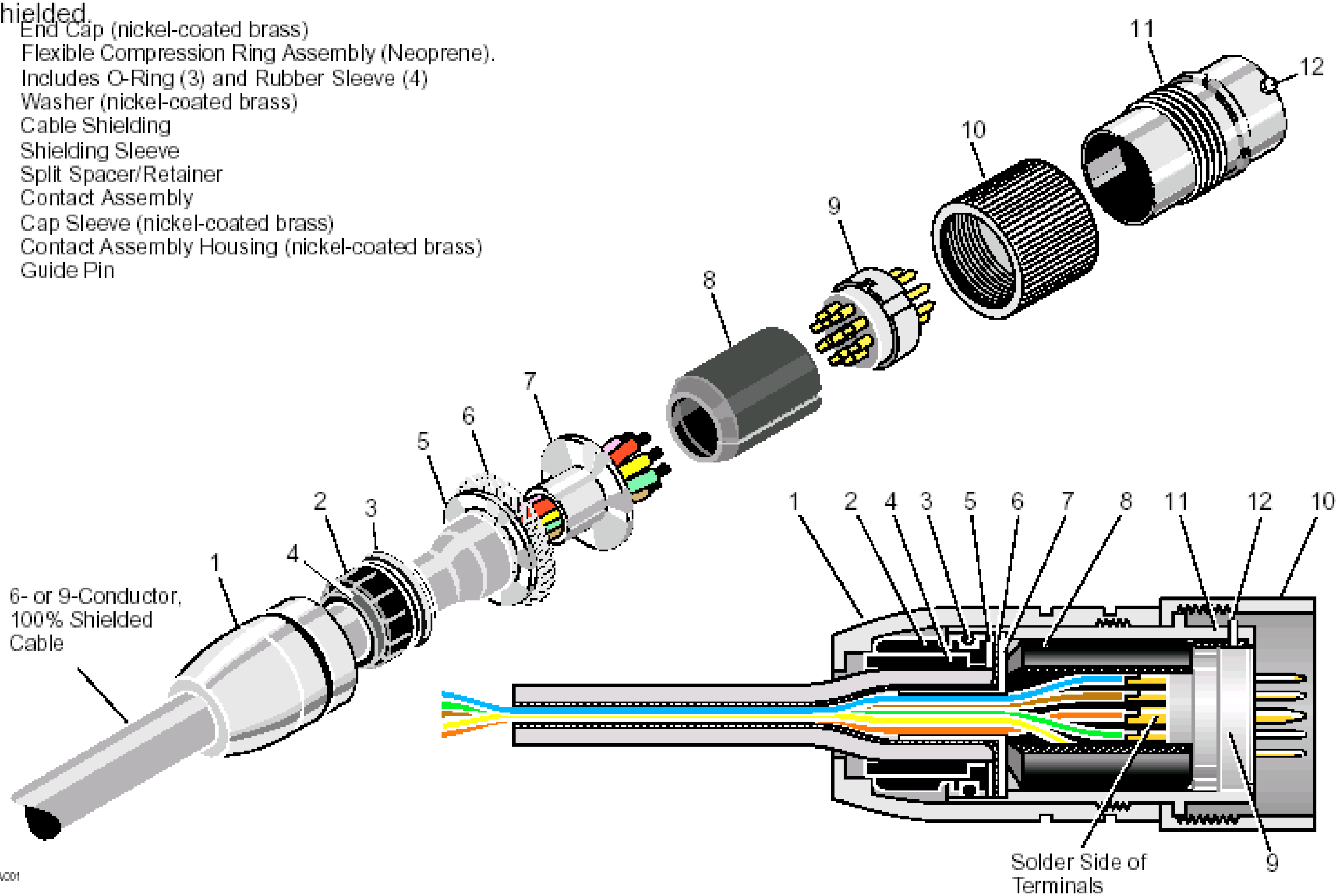


Electrical: Water-proof Connectors

Cable

For Installation Remote Bus (IRB) assemblies, use cable that contains 9 conductors and is 100% shielded.

1. End Cap (nickel-coated brass)
2. Flexible Compression Ring Assembly (Neoprene). Includes O-Ring (3) and Rubber Sleeve (4)
5. Washer (nickel-coated brass)
6. Cable Shielding
7. Shielding Sleeve
8. Split Spacer/Retainer
9. Contact Assembly
10. Cap Sleeve (nickel-coated brass)
11. Contact Assembly Housing (nickel-coated brass)
12. Guide Pin



connector costs can become the dominant cost factor…

Physical Layer Outline

1. Layering

2. Topology

3. Physical media

4. Electric Signal coupling

5. Optical Fibers

6. Modulation

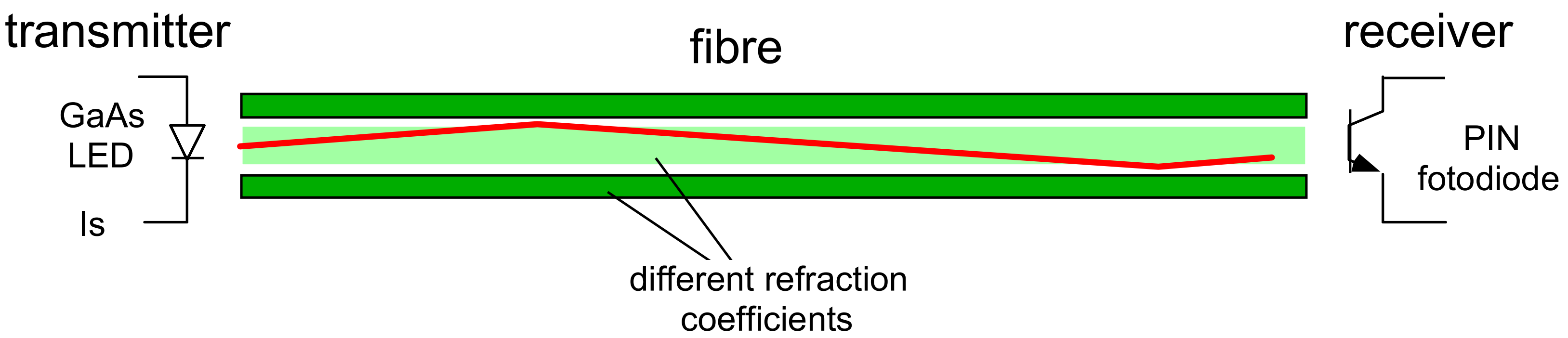
7. Synchronization

8. Encoding

9. Repeaters

Fiber: Principle

3 components:

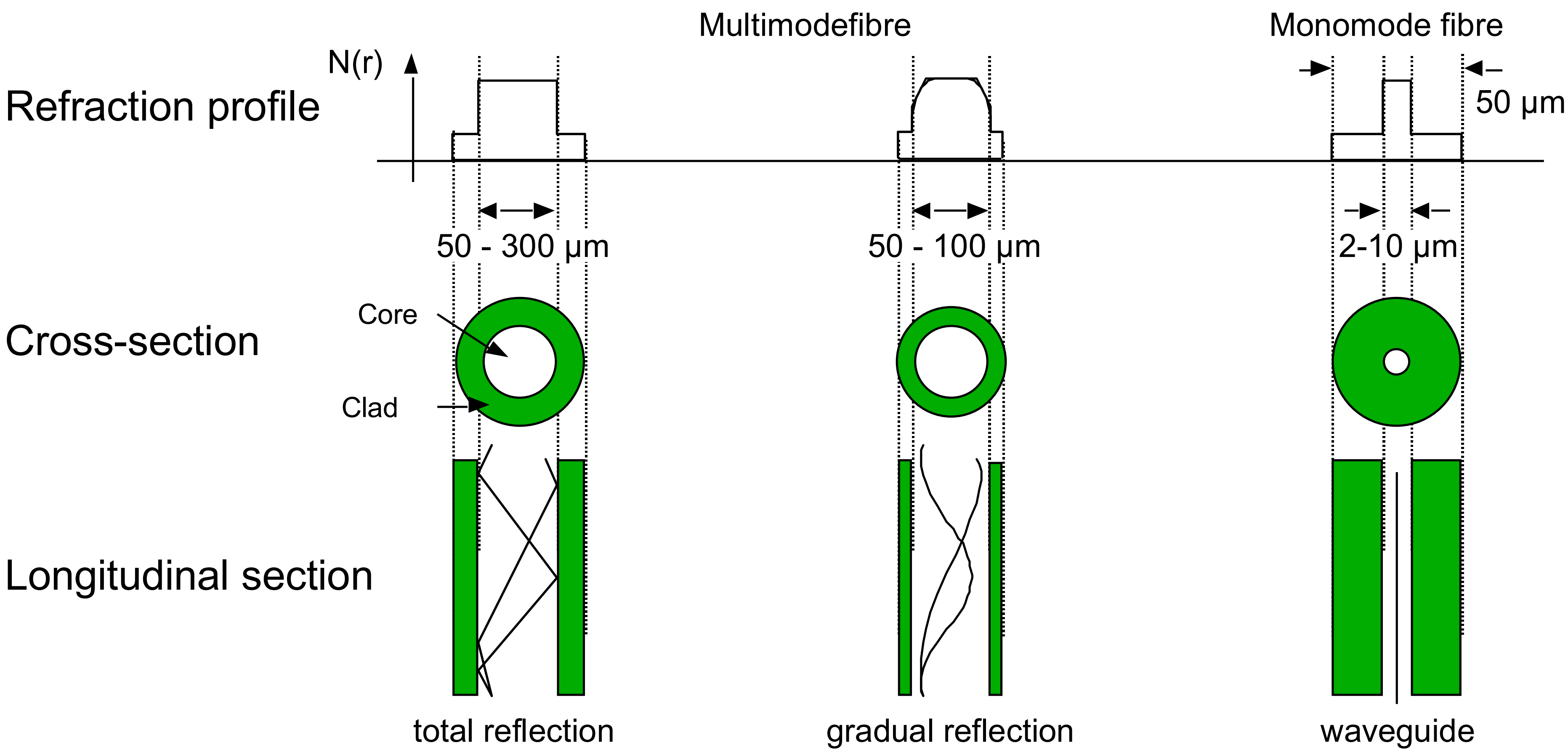


Transmitter, cable and receiver must be "tuned" to the same wavelength

Cable	glass (up to 100 km) or plastic (up to 30 m).
Transmitter	laser (power), laser-diode (GaAsP, GaAlAs, InGaAsP)
Receiver	PIN-diode
Wavelength	850 nm (< 3,5 dB/km, > 400 MHz x km) 1300 nm-window (Monomode)

light does not travel faster than electricity in a fiber (refraction index).

Fiber: Types



(red) 650nm	10 dB/km		
800nm	5dB/km	3 dB/km	2,3 dB/km
(infra-red) 1300nm		0,6 dB/km	0,4 dB/km
	20MHz·km	1 GHz·km	100 GHz·km
	HCS (Hard-Clad Silica) ø 200 µm, < 500m	50 or 62.5 µm LAN fibre	telecom - costly

Fibre: Use

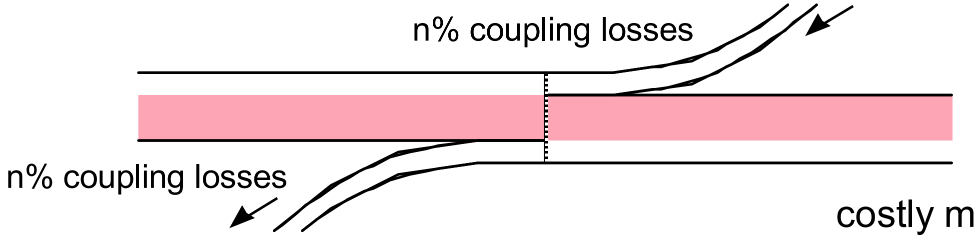
Type	POF	HCS/PCF	GOF
Material	plastic	glass / plastic	glass
distance	70m	400m	1km
Usage	local networking	remote networking	telephone
Connector	simple	precision	high-precision
Cost	cheap	medium	medium
aging	poor	very good	good
bending	very good	good	poor
bandwidth	poor	good	very good

POF: Plastic Optical Fibres
GOF: Glass Optical Fibres
HCS: silica fibre

in industry, fibers cost the same as copper - think about system costs !

Fiber: building an optical bus

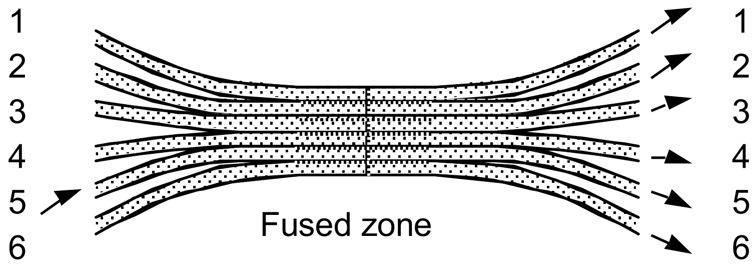
Passive coupler



Every branch costs a certain percentage of light

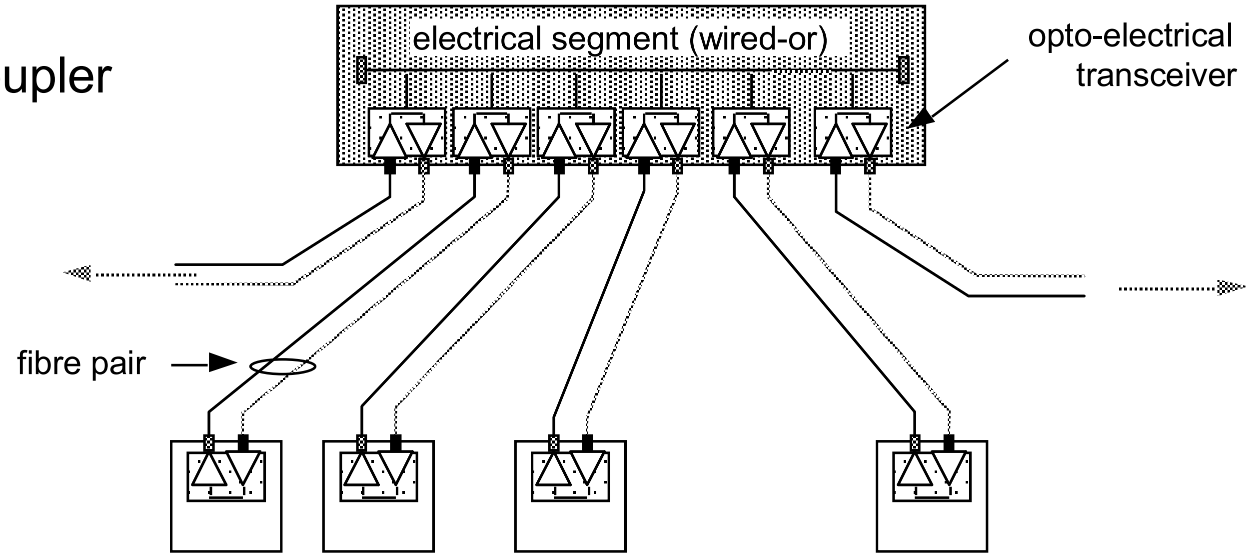
costly manufacturing (100 $ branches)

Passive star coupler



costly manufacturing (100 $ / 4 branches)

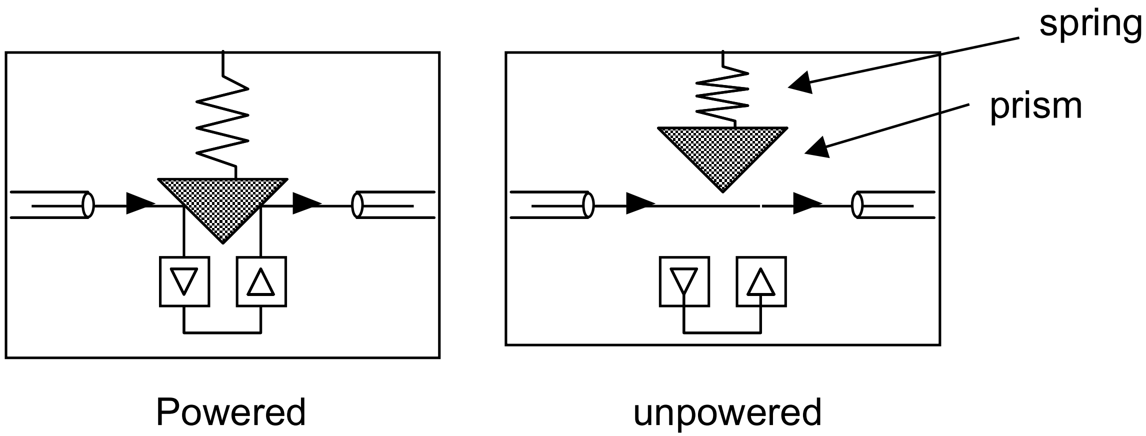
Active star coupler



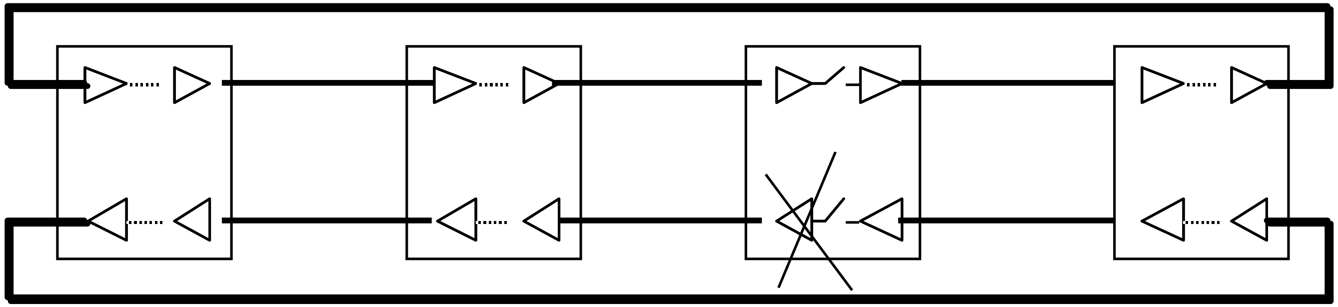
Fiber: building an optical ring and bridging

Mechanical bridging is difficult

example of solution



Double ring



This is why optical fibers are mostly used in rings (FDDI, Sercos)

Fiber: advantages

1) high bandwidth and data rate (400 MHz x km)

2) small, frequency-insensitive attenuation (ca. 3 dB/km)

3) cover long distances without a repeater

4) immune against electromagnetic disturbances (great for electrical substations)

5) galvanic separation and potential-free operation (great for large current environment)

6) tamper free

7) may be used in explosive environments (chemical, mining)

8) low cable weight (100 kg/km) and diameter, flexible, small cable duct costs

9) low cost cable

10) standardized

Fiber: Why are fibres so little used ?

1) In process control, propagation time is more important than data rate

2) Attenuation is not important for most distances used in factories (200m)

3) Coaxial cables provide a sufficiently high immunity

4) Reliability of optical senders and connections is insufficient (MTTF ≈ 1/power).

5) Galvanic isolation can be achieved with coaxial cables and twisted pairs through opto-couplers

6) Tapping is not a problem in industrial plants

7) Optical busses using (cheap) passive components are limited to a few branches (16)

8) In explosive environments, the power requirement of the optical components hurts.

9) Installation of optical fibres is costly due to splicing

10) Topology is restricted by the star coupler (hub) or the ring structure

Radio Transmission

Radio had the reputation to be slow, highly disturbed and range limited.

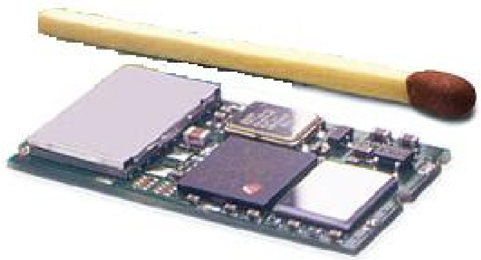
Mobile radio (GSM, DECT) is able to carry only limited rate of data (9.6 kbit/s) at high costs, distance being limited only by ground station coverage.

IEEE 802.11 standards developed for computer peripherals e.g. Apple’s AirPort allow short-range (200m) transmission at 11 Mbit/s in the 2.4 GHz band with 100mW.

Bluetooth allow low-cost, low power (1 mW) links in the same 2.4 GHz band, at 1 Mbit/s

Modulation uses amplitude, phase and multiple frequencies (see next Section)

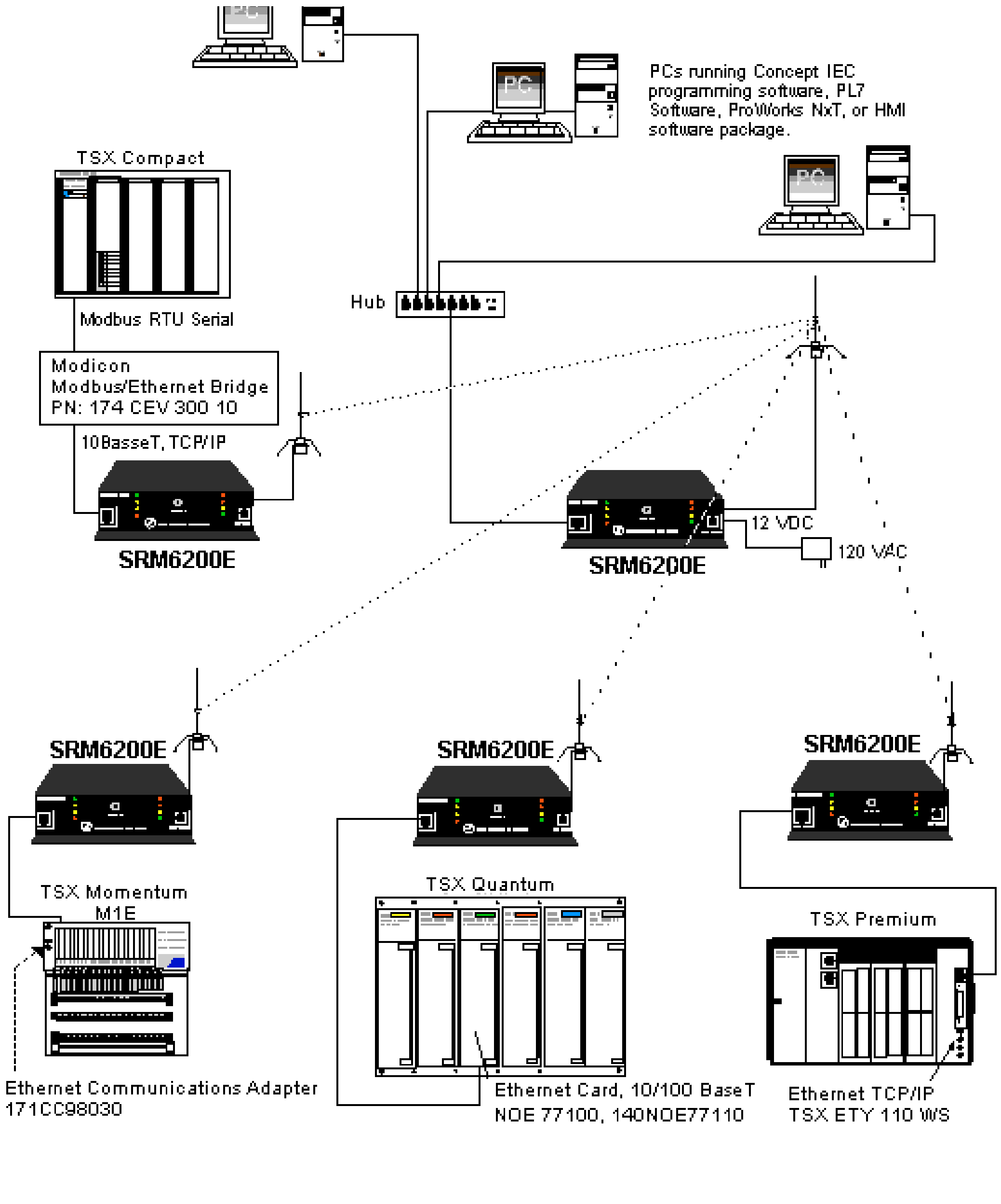
Higher-layer protocols (WAP, …) are tailored to packet radio communication.



bluetooth module

Radio == mobile -> power source (batteries) and low-power technologies.

Wireless Field busses

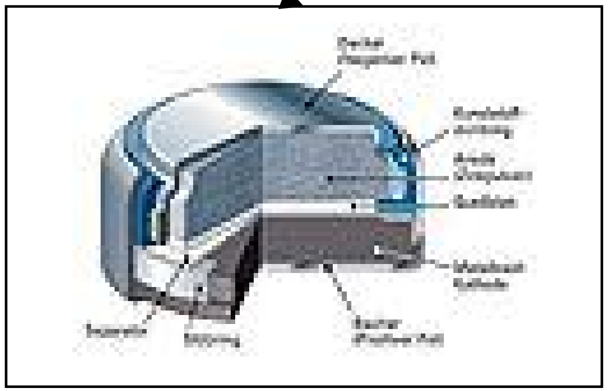


Transparent Factory is a registered trademark of Schneider Electric.

\+ no wiring,
mobile,
easy to install

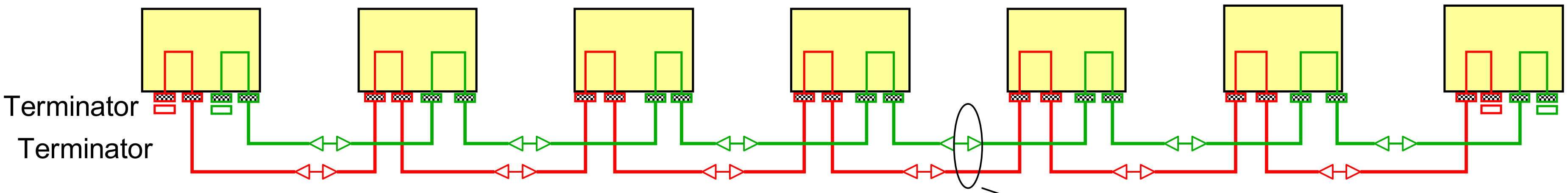
– short distance,
limited bandwidth,
area overlap and frequency limitations
not tamper-free,
difficult to power the devices
costs of base station

but: who changes the batteries ?



Redundancy at the physical layer

Party-Line

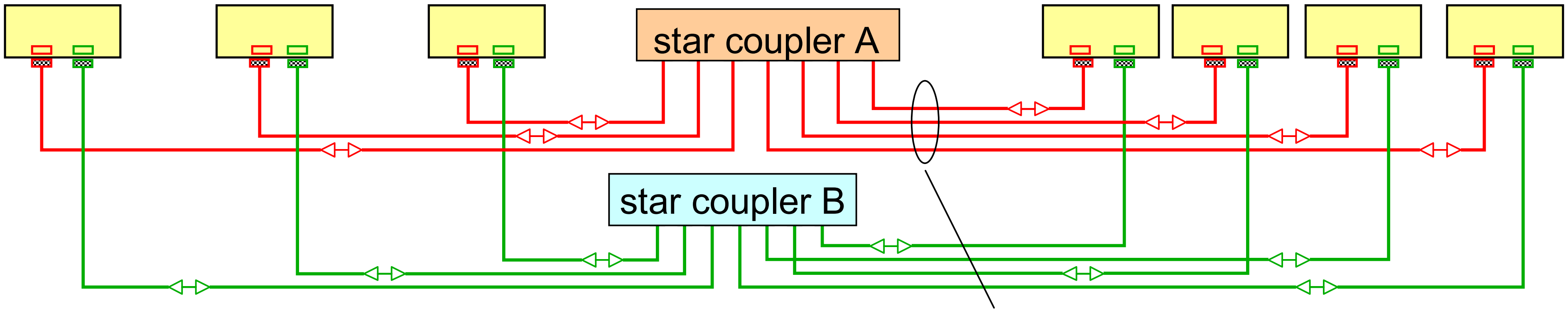


decentralized wiring

both cables can run in the same conduct where common mode failure acceptable

Star topology

star couplers should be separately powered



centralized wiring

cable come together at each device

common mode failures cannot be excluded since wiring has to come together at each device

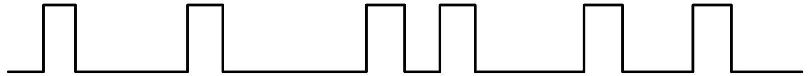
Physical Layer Outline

1. Layering
2. Topology
3. Physical media
4. Electric Signal coupling
5. Optical Fibers
6. **Modulation**
7. Synchronization
8. Encoding
9. Repeaters

Modulation

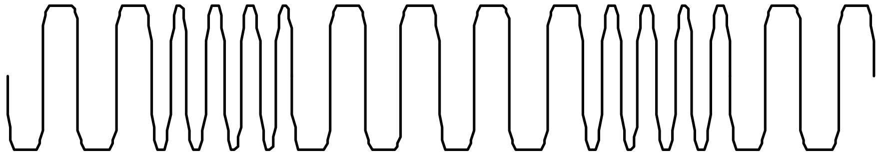
Base band

Signal transmitted as a sequence of binary states, one at a time (e.g. Manchester)



Carrier band

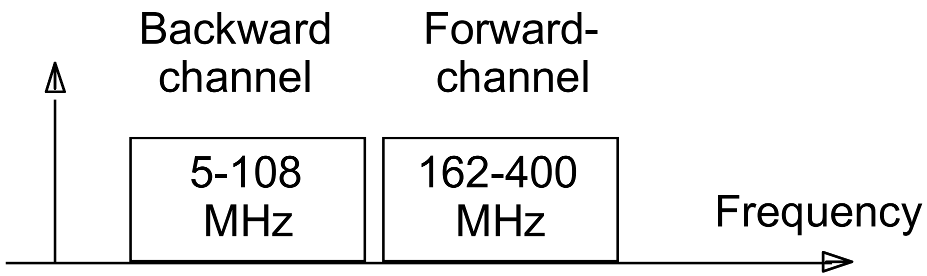
Signal transmitted as a sequence of frequencies, one at a time
(e.g. FSK = frequency shift keying = 2-phase Modulation.



Broadband

Signal transmitted as a sequence of frequencies, several at the same time.

Signals may be modulated on a carrier frequency (e.g. 300MHz-400MHz, in channel of 6 MHz)



Physical Layer Outline

1. Layering

2. Topology

3. Physical media

4. Electric Signal coupling

5. Optical Fibres

6. Modulation

7. Synchronization

8. Encoding

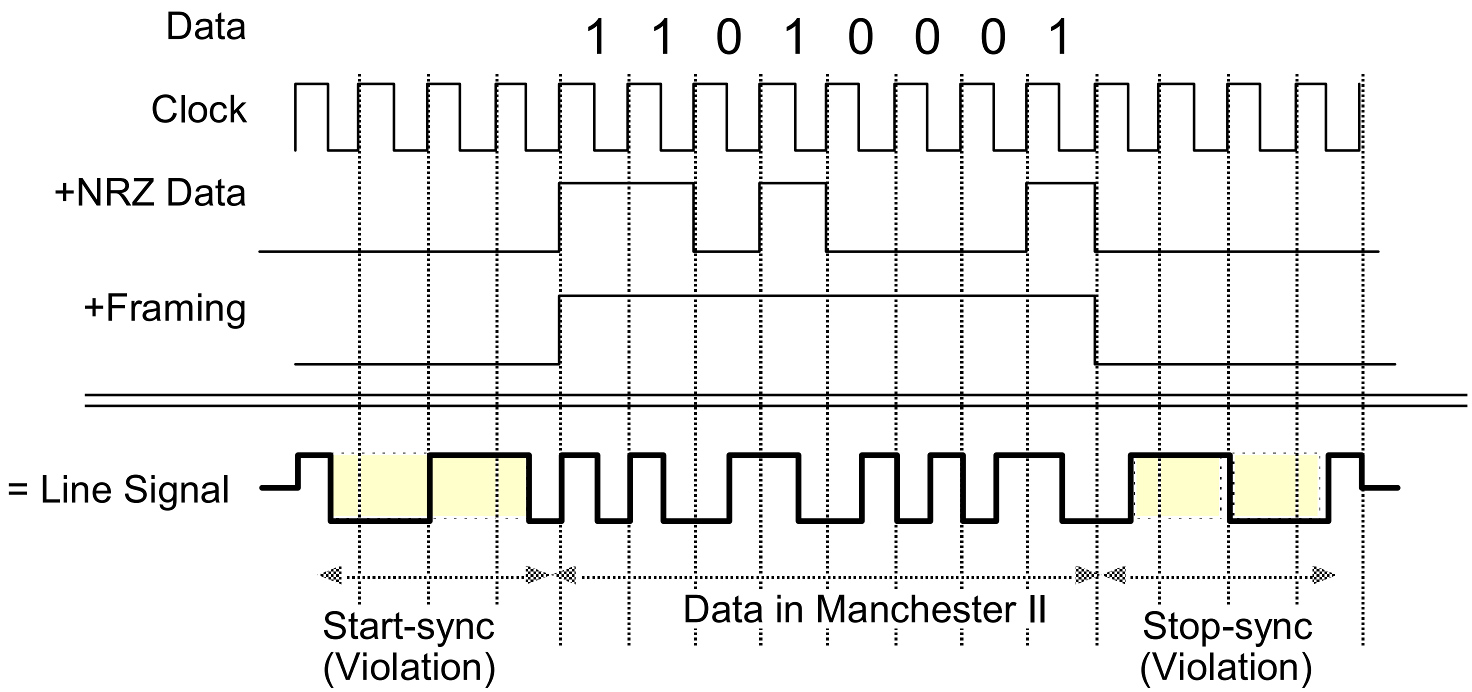
9. Repeaters

Synchronisation: where does it take place ?

"determine the beginning and the end of a data stream"

<u>Bit synchronisation</u>	Recognize individual bits
<u>Character synchronisation</u>	Recognize groups of (5,7,8,9,..) bits
<u>Frame synchronisation</u>	Recognize a sequence of bits transmitted as a whole
<u>Message synchronisation</u>	Recognize a sequence of frames
<u>Session synchronisation</u>	Recognize a sequence of messages

Example: Frame synchronisation using Manchester violation symbols



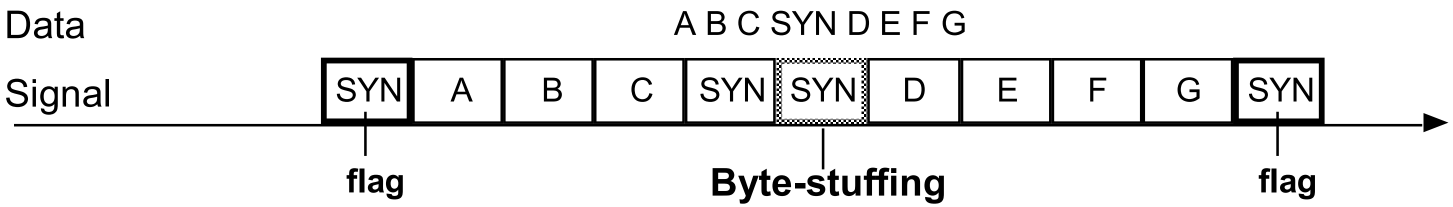
Frames: Synchronization

character-synchronous (e.g. bisync)

A character is used as synchronisation character

If this character appears in the data stream, it is duplicated

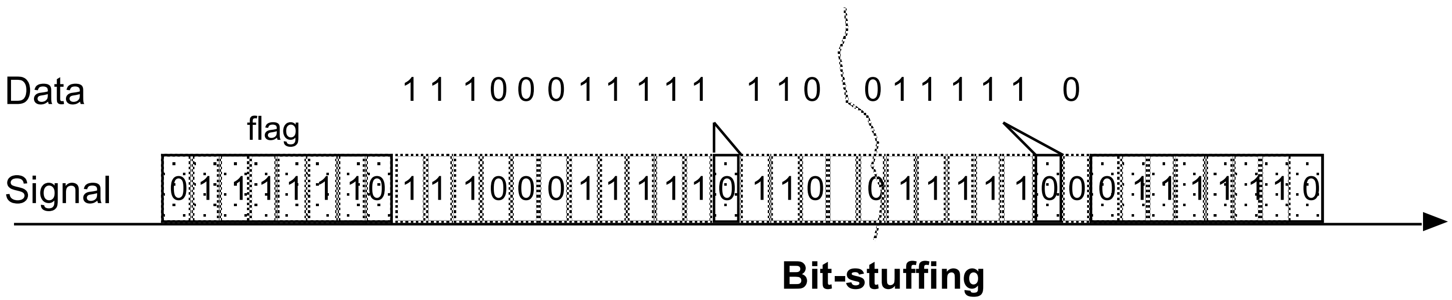
The receiver removes duplicated synchronisation characters



---

bit-synchronous (e.g. HDLC)

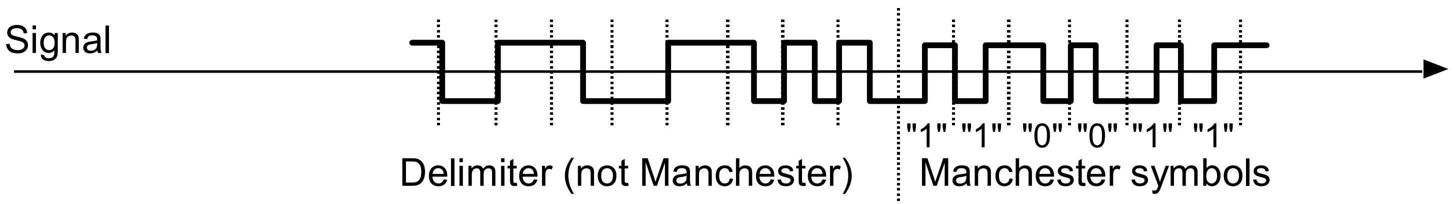
A bit sequence is used as a flag (e.g. 01111110).
To prevent this sequence in the bit-stream, the transmitter inserts a "0" after each group of 5 consecutive "1", which the receiver removes.



---

delimiter (e.g. IEC 61158)

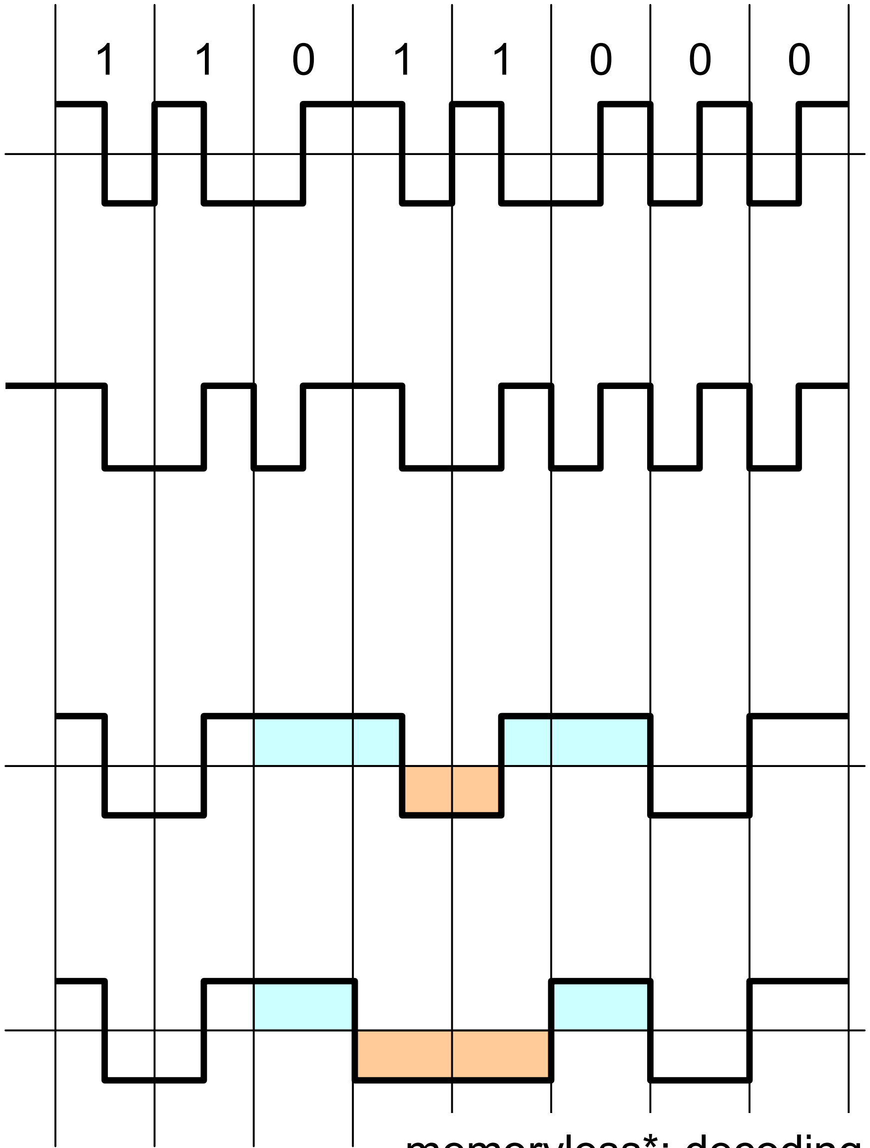
A symbol sequence is used as delimiter, which includes non-data symbols



Physical Layer Outline

1. Layering
2. Topology
3. Physical media
4. Electric Signal coupling
5. Optical Fibers
6. Modulation
7. Synchronization
8. **Encoding**
9. Repeaters

Encoding: popular DC-free encodings



user

Manchester
1: falling edge at midpoint
0: rising edge at midpoint
DC-free, memoryless*

Ethernet, FIP
IEC 61158,
MVB, MIL 1553

Differential Manchester
always transition at midpoint
1: no transition at start point
0: transition at start point
(polarity-insensitive, DC-free,
memoryless)

LON

Miller (MFM)
centre frequency halved
not completely DC-free
memory: two bits
(sequence 0110)

High-density
diskettes

Xerxes
replaces "101" sequence
by DC-balanced sequence
DC-free, memory: two bits

FlexRay

memoryless*: decoding does not depend on history

Encoding: DC-free coding for transformer coupling

DC-free encoding is a necessary, but not sufficient condition

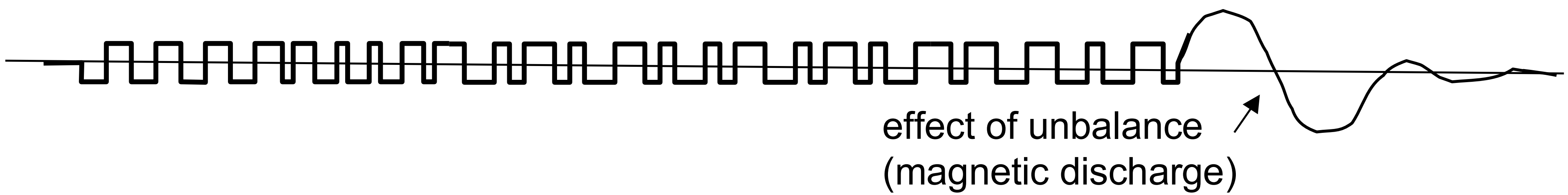
The drivers must be carefully balanced (positive and negative excursion |+U| = |-U|)

Slopes must be symmetrical, positive and negative surfaces must be balanced

Preamble, start delimiter and end delimiter must be DC-free
(and preferably not contain lower-frequency components)

Transformer-coupling requires a lot of experience.

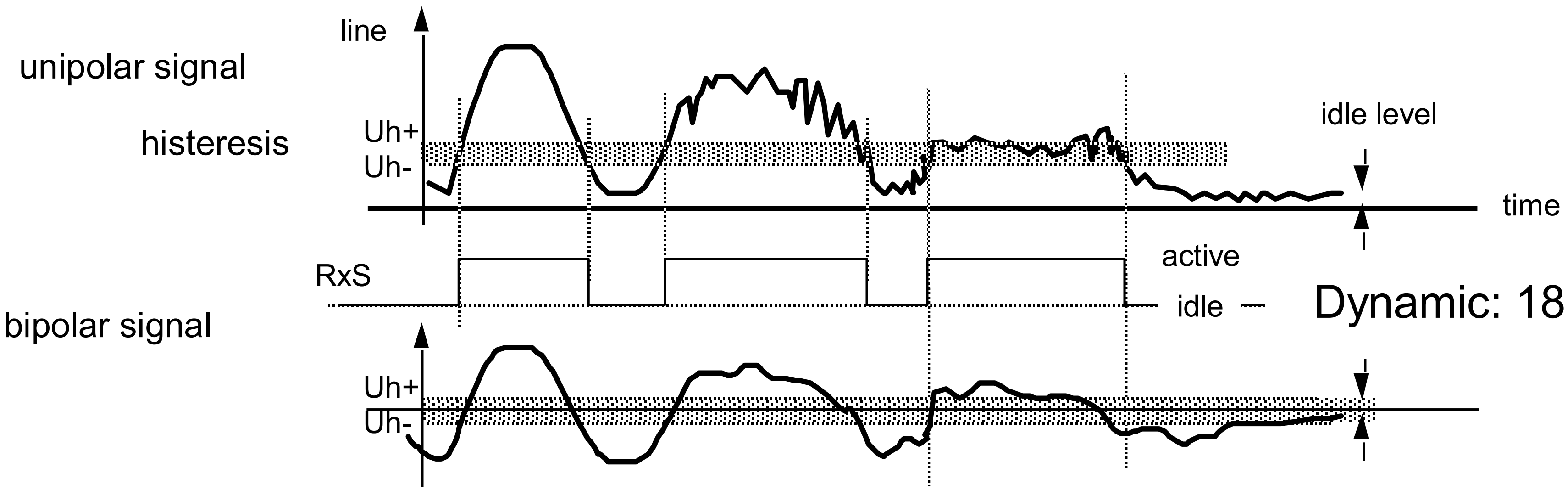
Many applications (ISDN…) accept not completely DC-free codes, provided that the DC component is statistically small when transmitting random data, but have to deal with large interframe gaps.



Decoding base-band signals

Zero-crossing detector

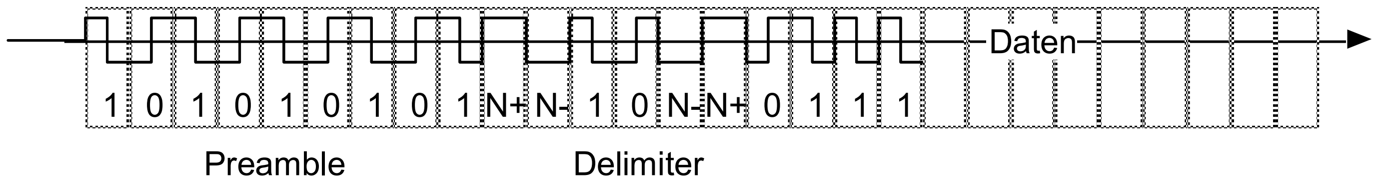
decoding depends on the distance between edges — Dynamic: 10 dB



Dynamic: 18 dB

Sampling of signal — Dynamic: 32 dB

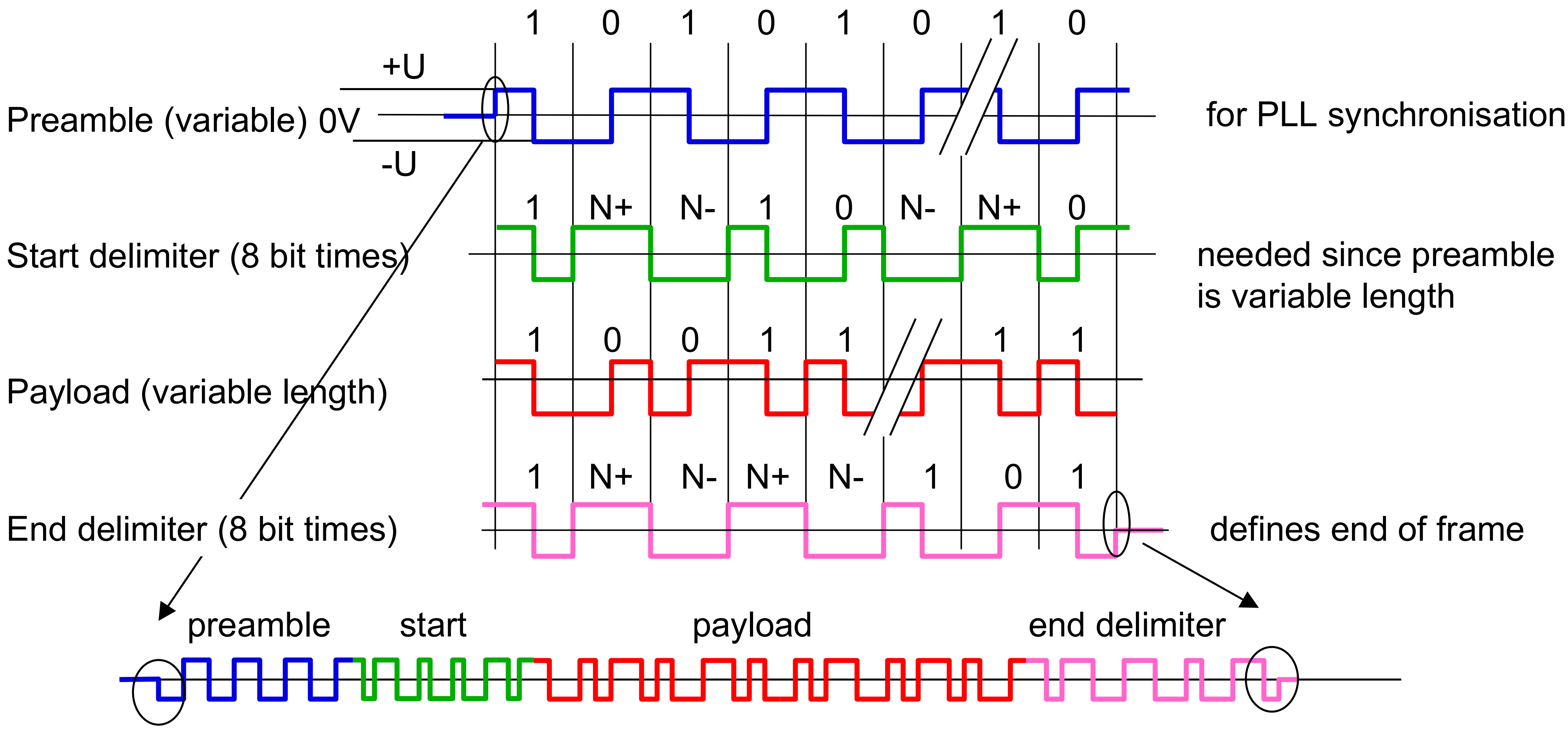
needs Phase-Locked Loop (PLL) and preamble (? delimiter)



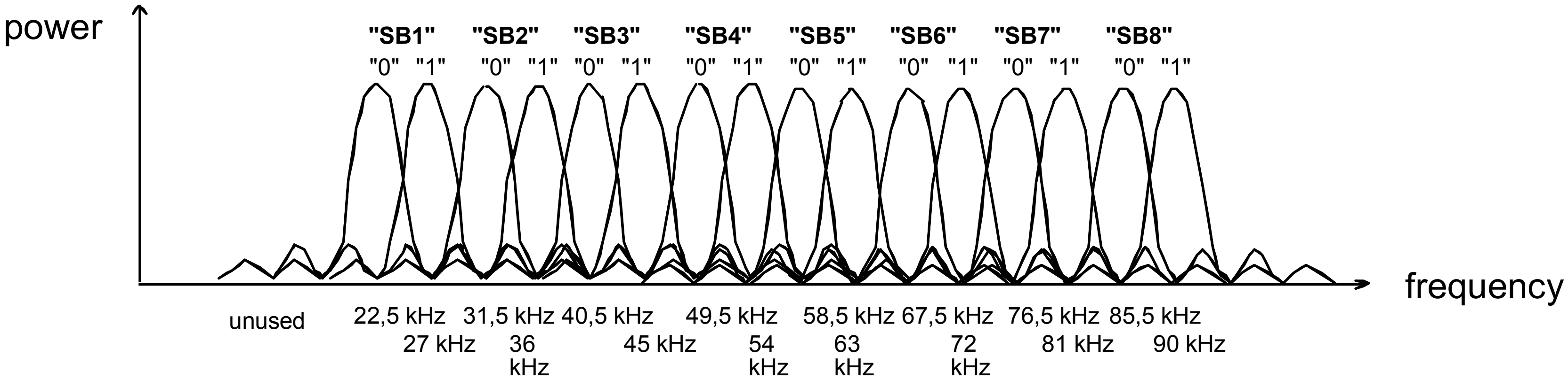
Signal Frequency Analysis — Dynamic: 38 dB

requires Signal Processor, Quadrature/Phase analysis

Encoding: Physical frame of IEC 61158-2

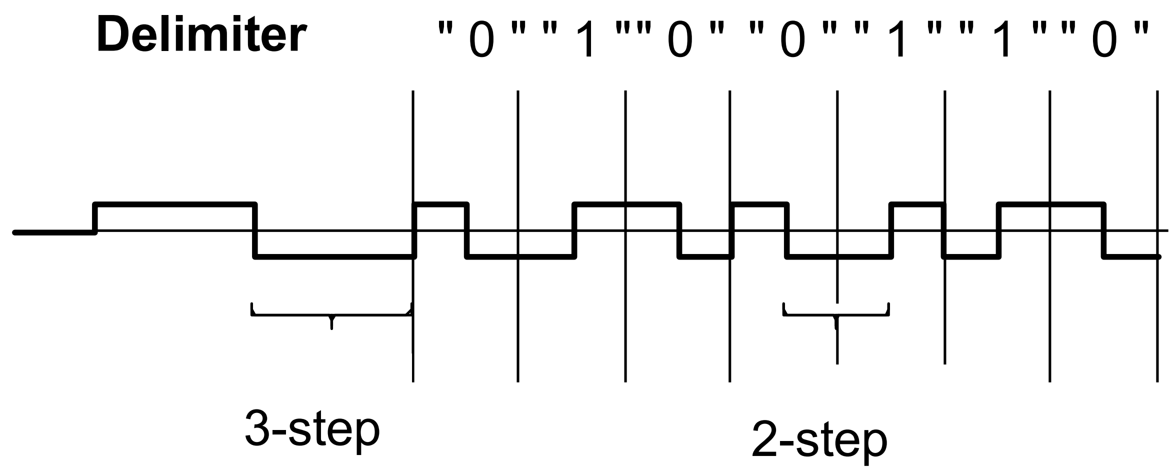


Encodings: Multi-frequency



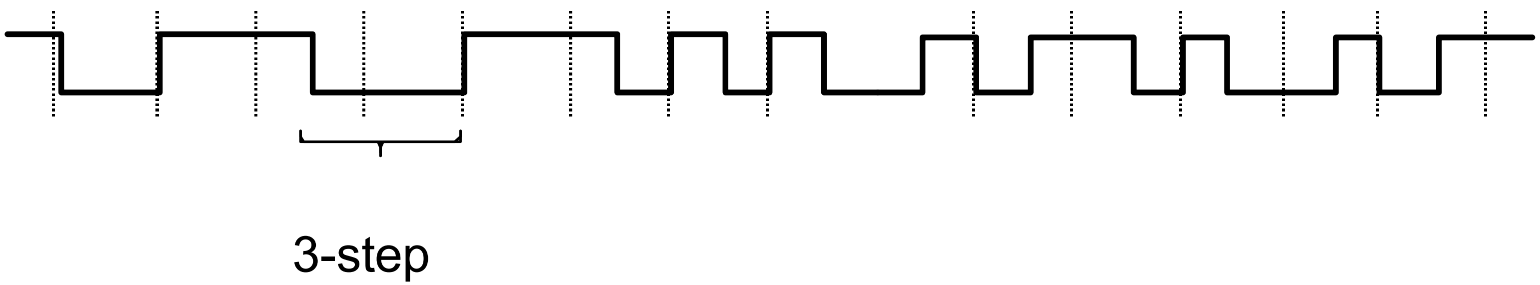
Best use of spectrum
Adaptive scheme (frequency-hopping, avoid disturbed frequencies, overcome bursts)
Base of ADSL, internet-over-power lines, etc...
Requires digital signal processor
Limited in frequency
EMC considerations

Bandwidth and Manchester Encoding



Non-data symbols may introduce a lower-frequency component which must go through a transformer.

The transformer must be able to transmit frequencies in a 1:20 ratio



Encoding: qualities

1) Self-clocking, Explicit clocking or asynchronous

Clocked: separate clock channel
Self-clocking: clock channel multiplexed with signal
Asynchronous: requires synchronisation at next higher level.
Code such as HDB3 insert "Blind Bits" to synchronize a random sequence.

2) Spectral efficiency

Which frequency components can be found in a random data sequence ?
especially: is there a DC-component ?
(Important for transformer and transceiver coupling)
Pseudo-DC-free codes such as AMI assume that "1" and "0" are equally frequent.

3) Efficiency: relation of bit rate to Baudrate

Bitrate = Information bits per second
Baudrate = Signal changes per second

4) Decoding ease

Spectral-efficient codes are difficult to decode
This is especially the case with memory-codes (value depends on former symbols)
(e.g. Miller, differential Manchester).

5) Integrity

For error detection, the type of error which can occur is important, and especially if a single disturbance can affect several bits at the same time (Differential Manchester).

6) Polarity

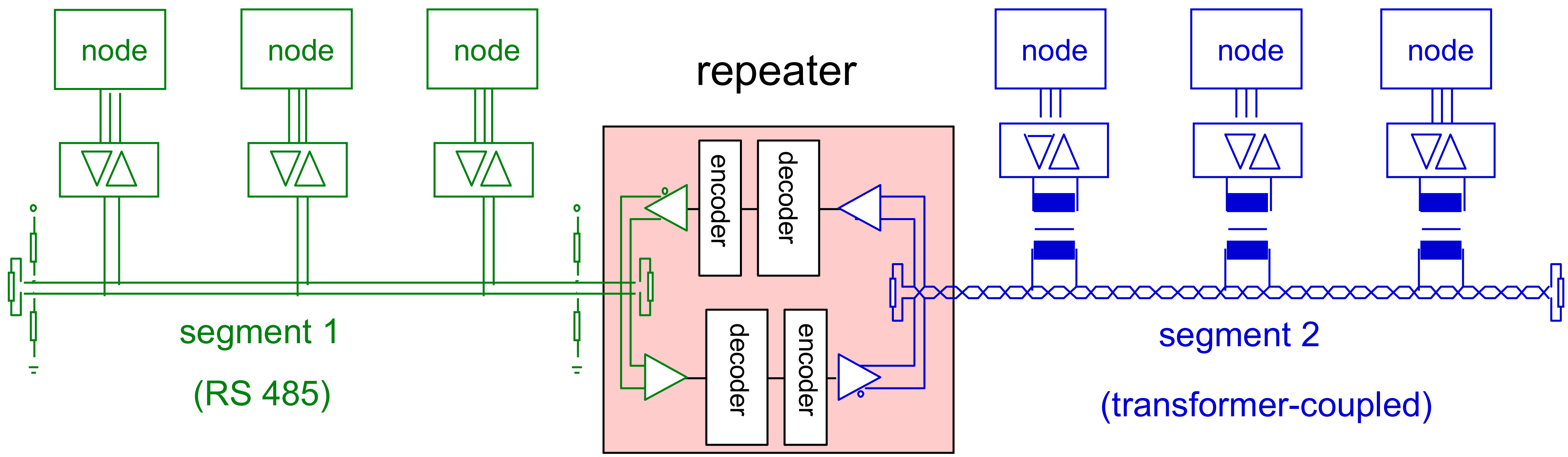
A polarity-insensitive electrical wiring simplifies installation

Physical Layer Outline

1. Layering
2. Topology
3. Physical media
4. Electric Signal coupling
5. Optical Fibres
6. Modulation
7. Synchronization
8. Encoding
9. **Repeaters**

Repeater

A repeater is used at a transition from one medium to another within the same subnet.



The repeater:

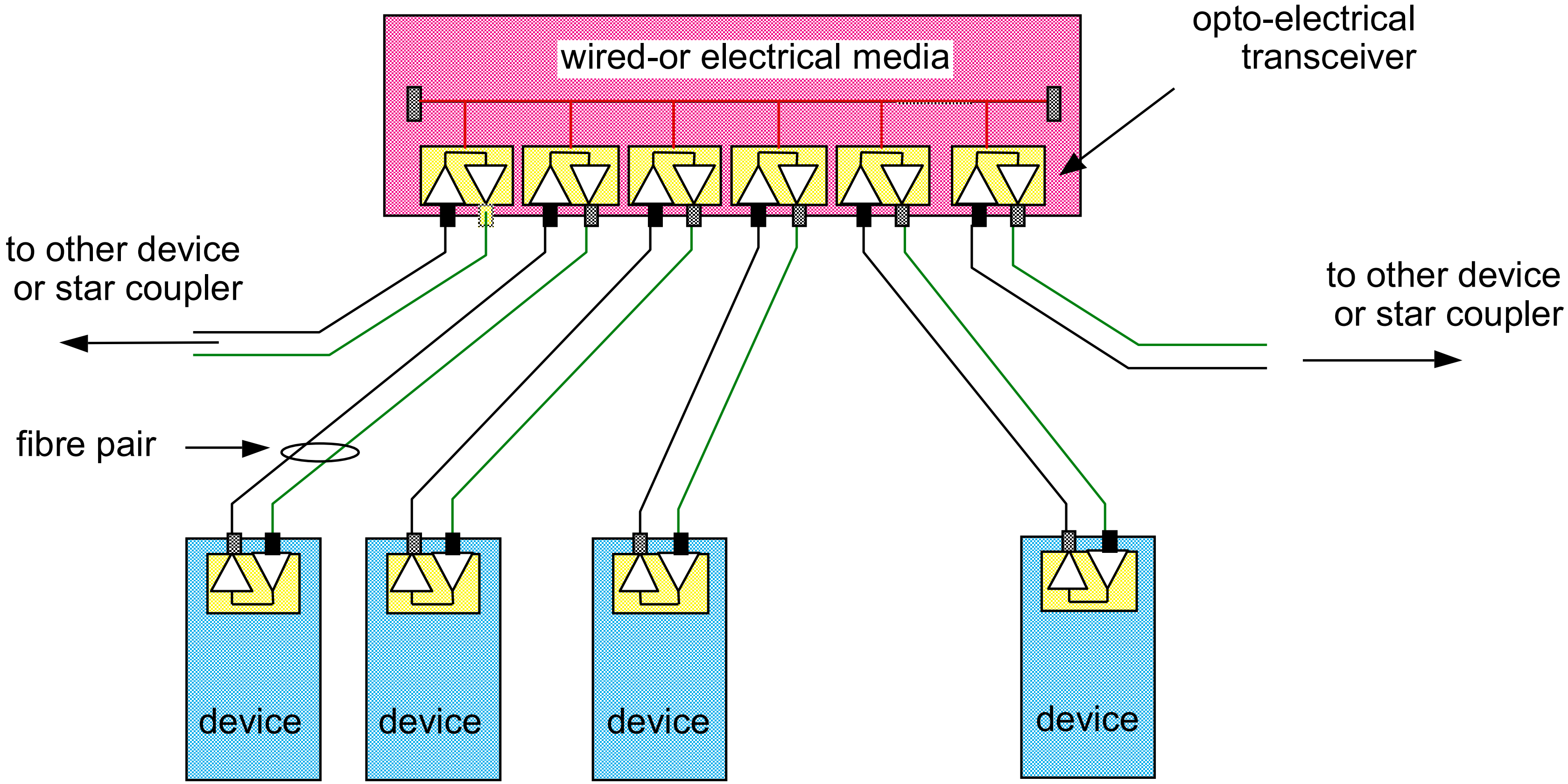
- decodes and reshapes the signal (knowing its shape)
- recognizes the transmission direction and forward the frame
- detects and propagates collisions

Repeater and time skew

Repeaters introduce an impredictable delay in the transmission since they need some time to synchronize on the incoming signal and resolve collisions.

Star coupler (hub)

A star coupler is a collection of repeaters that connect point-to-point links into a bus (e.g. for optical fibres). it is called "hub" in the Ethernet standard.
It is a star topology, but a bus structure



To probe further

Henri Nussbaumer, Téléinformatique 1, Presses polytechniques romandes

Fred Halsall, Data Communications, Computer Networks and Open Systems, Addison-Wesley

B. Sklar , “Digital Communications,” Prentice Hall, Englewood Cliffs, 1988

EIA Standard RS-485

Assessment

What is the function of the physical layer ?
What is the difference between a bus and a ring ?
How is a bus wired ?
Which electrical media are used in industry ?
How is the signal coupled to an electrical media ?
How is the signal decoded ?
What is an open-collector (open-drain) electrical media ?
What are the advantages and disadvantages of transformer coupling ?
What limits the distance covered by electrical signals and how is this to overcome ?
What are the advantages and disadvantages of optical fibres ?
When are optical fibers of 240 μm used rather than 62.5 μm ?
What is a broadband medium ?
What is DSL ?
What is the purpose of modulation ?
What is the purpose of encoding ?
What is the difference between bit rate and Baudrate and what does it say about efficiency?
What limits the theoretical throughput of a medium ?
What is the difference between Manchester encoding, Miller and differential Manchester ?
Which are the qualities expected from an encoding scheme ?

Industrial Automation
Automation Industrielle
Industrielle Automation

3. Industrial Communication Systems

3.3.3 Link Layer and Medium Access
Niveau de liaison et accès au médium
Verbindungsschicht und Mediumzugriff

Prof. Dr. H. Kirrmann
ABB Research Center, Baden, Switzerland

2004 June, HK

Link Layer Outline

Link Layer in the OSI model
- Stacks
- HDLC as example

Frame sub-layer
- Error detection
- Error correction

Medium Access control
- Quality Criteria
- Single Master
- Rings
- Ethernet
- Collision with winner
- Token Passing
- Comparison

Logical Link Control
- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

Link Layer function

The link layer implements the protocols used to communicate within the same *subnet*.

(subnet: same medium access, same bit rate)

- but different media may be interconnected by (bit-wise) repeaters

Tasks of the link layer:

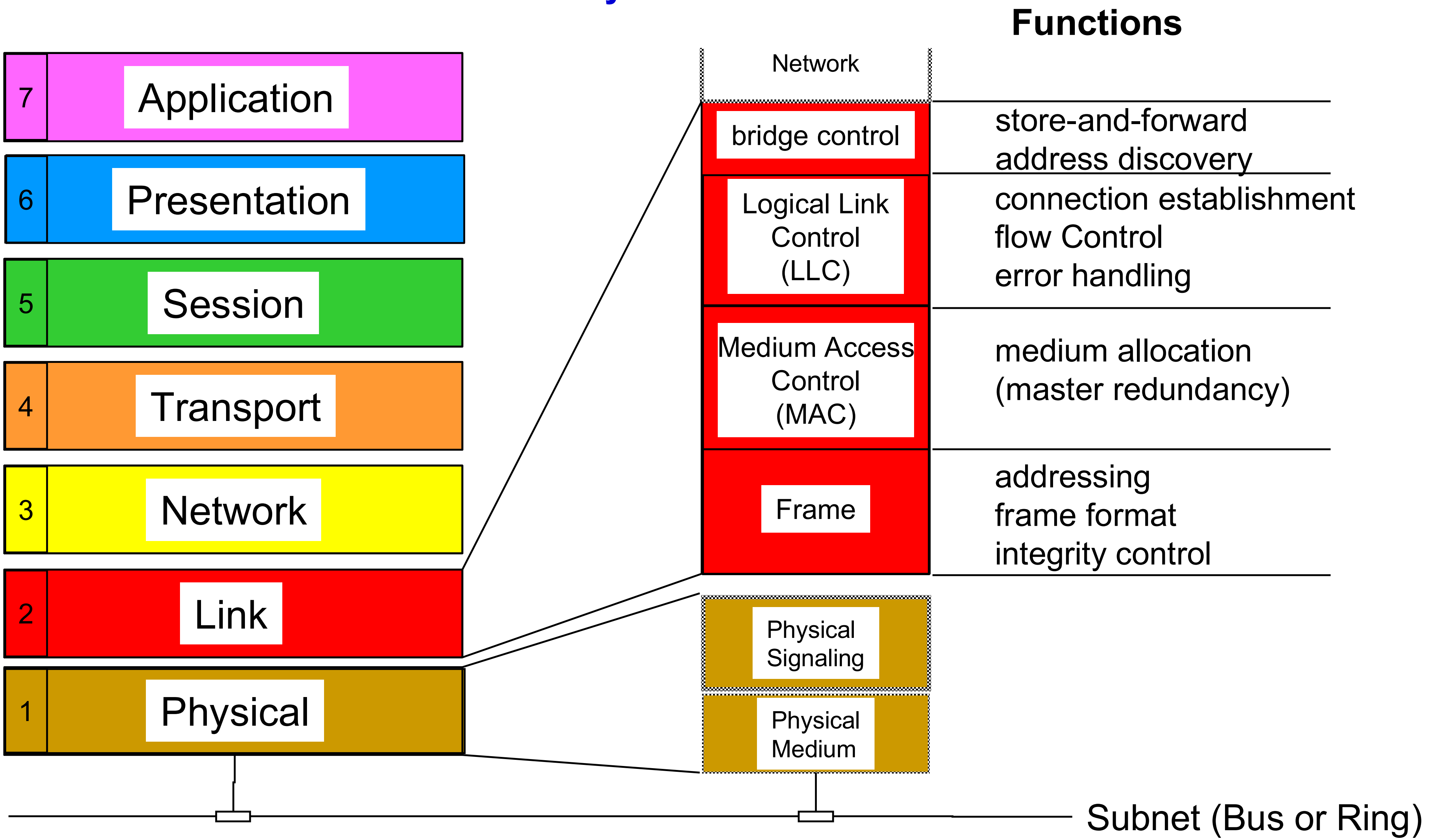
1) Data integrity

2) Medium Access

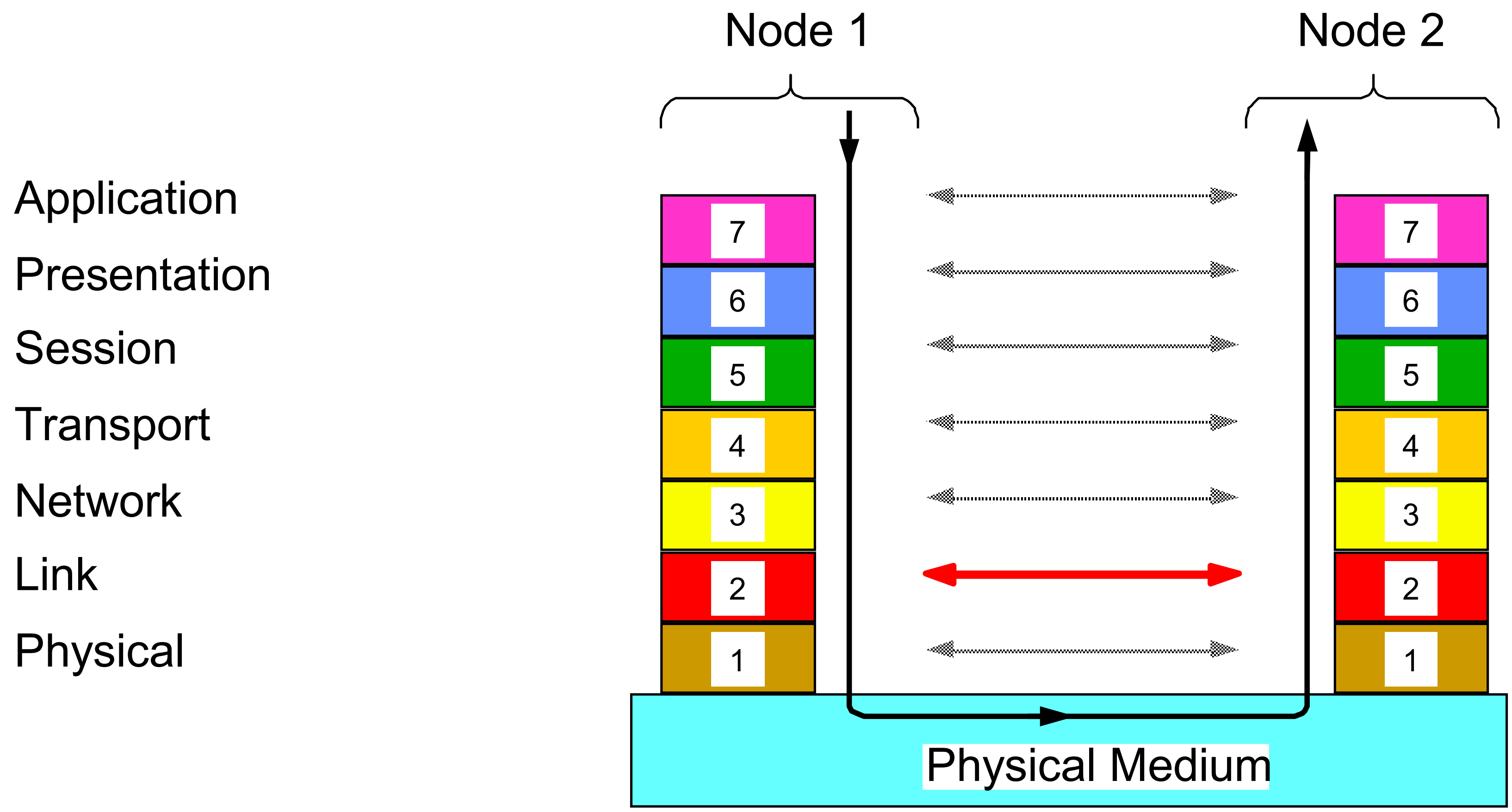
3) Logical Link Control

4) Link Layer Management

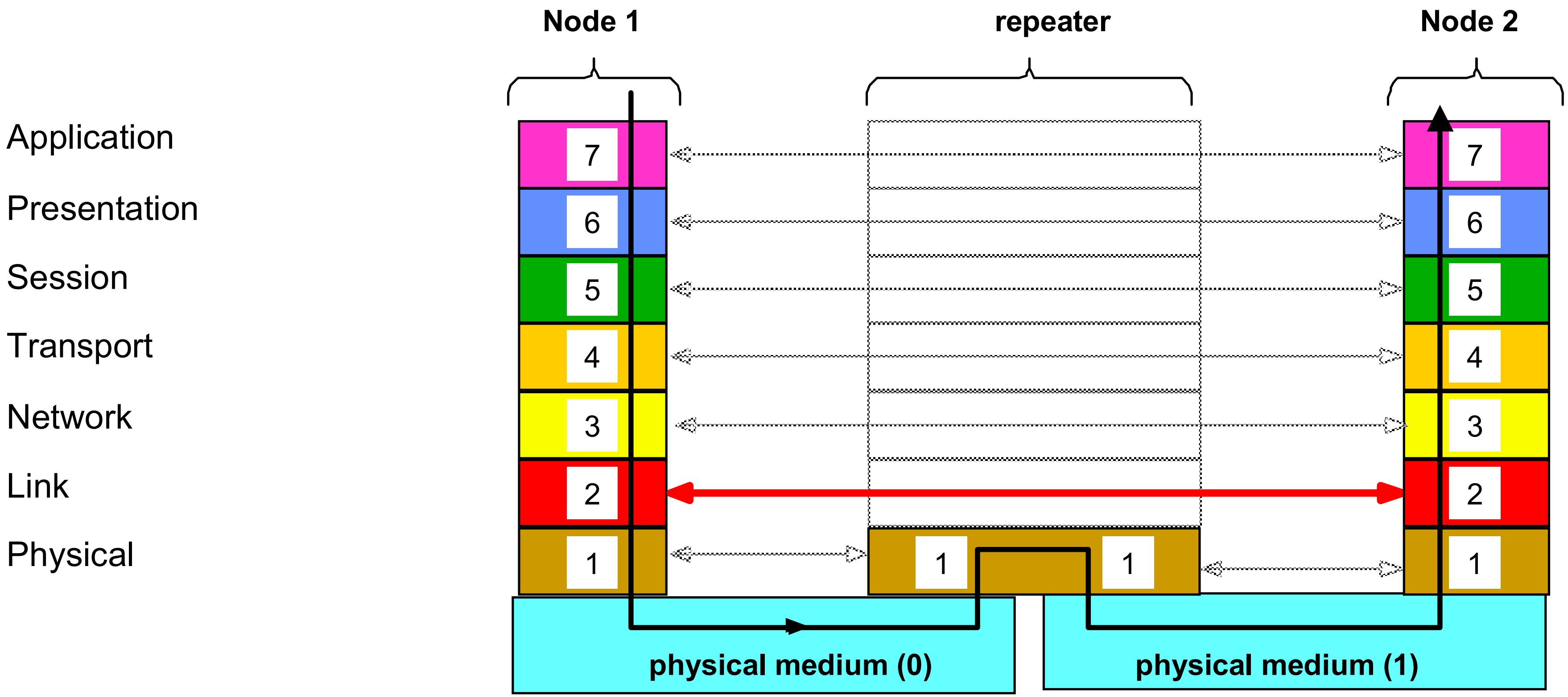
Link Layer in the OSI Model



OSI model with two nodes



OSI model with repeater (physical layer connection)

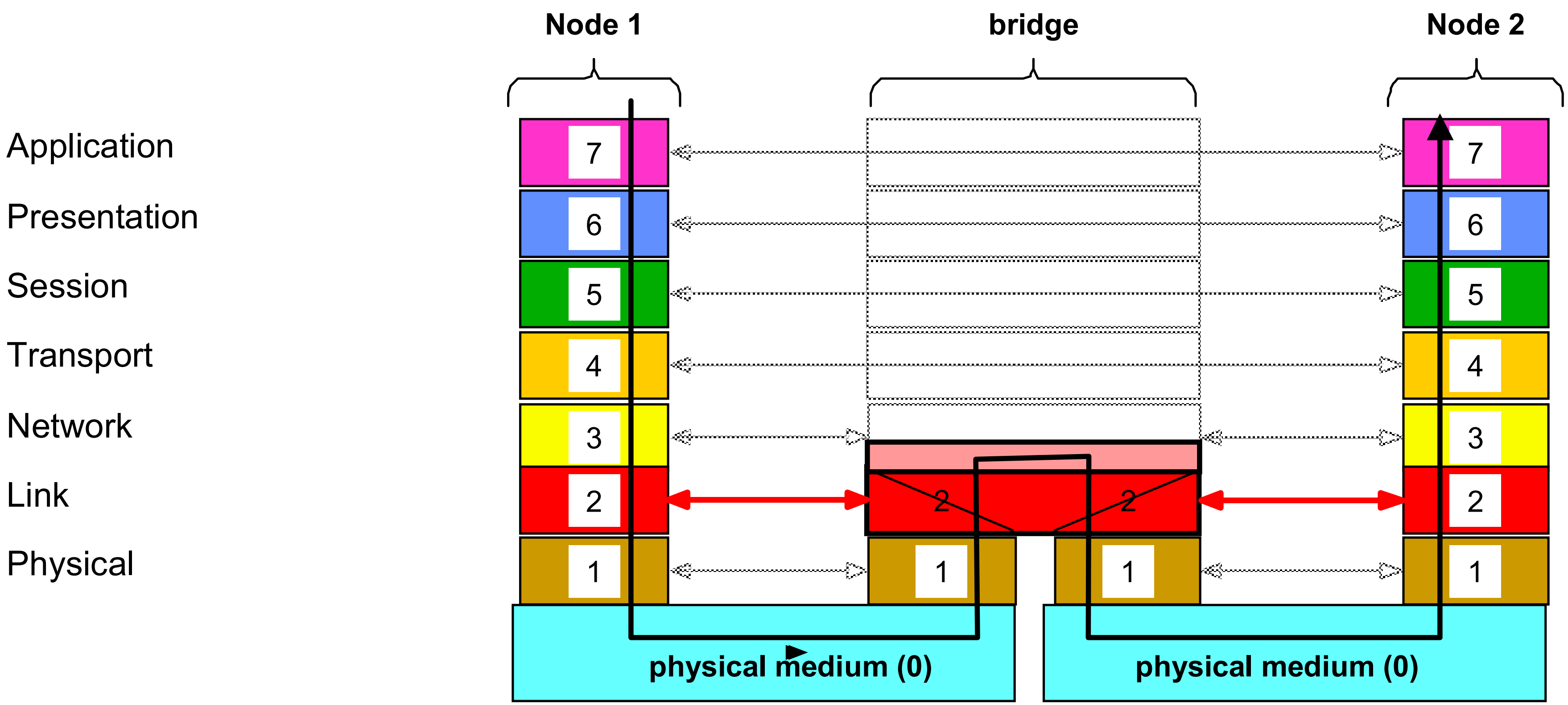


The two segments on each side of a repeater form a single *subnet*, identified by

- same speed (medium, modulation may differ)
- same frame format (except fringe effects)
- same medium access
- same address space (transparent on both side of the repeater)

Repeaters introduce a delay in the order of a few bit time.

OSI model with three nodes (bridge): link layer connection



The subnet on both sides of a bridge have:

- the same frame format (except header),
- the same address space (different addresses on both sides of the bridge)
- the same link layer protocol (if link layer is connection-oriented)

Bridges filter the frames on the base of their link addresses

Link Layer Outline

Link Layer in the OSI model
- Stacks
- **HDLC as example**

Frame sub-layer
- Error detection
- Error correction

Medium Access control
- Quality Criteria
- Single Master
- Rings
- Ethernet
- Collision with winner
- Token Passing
- Comparison

Logical Link Control
- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

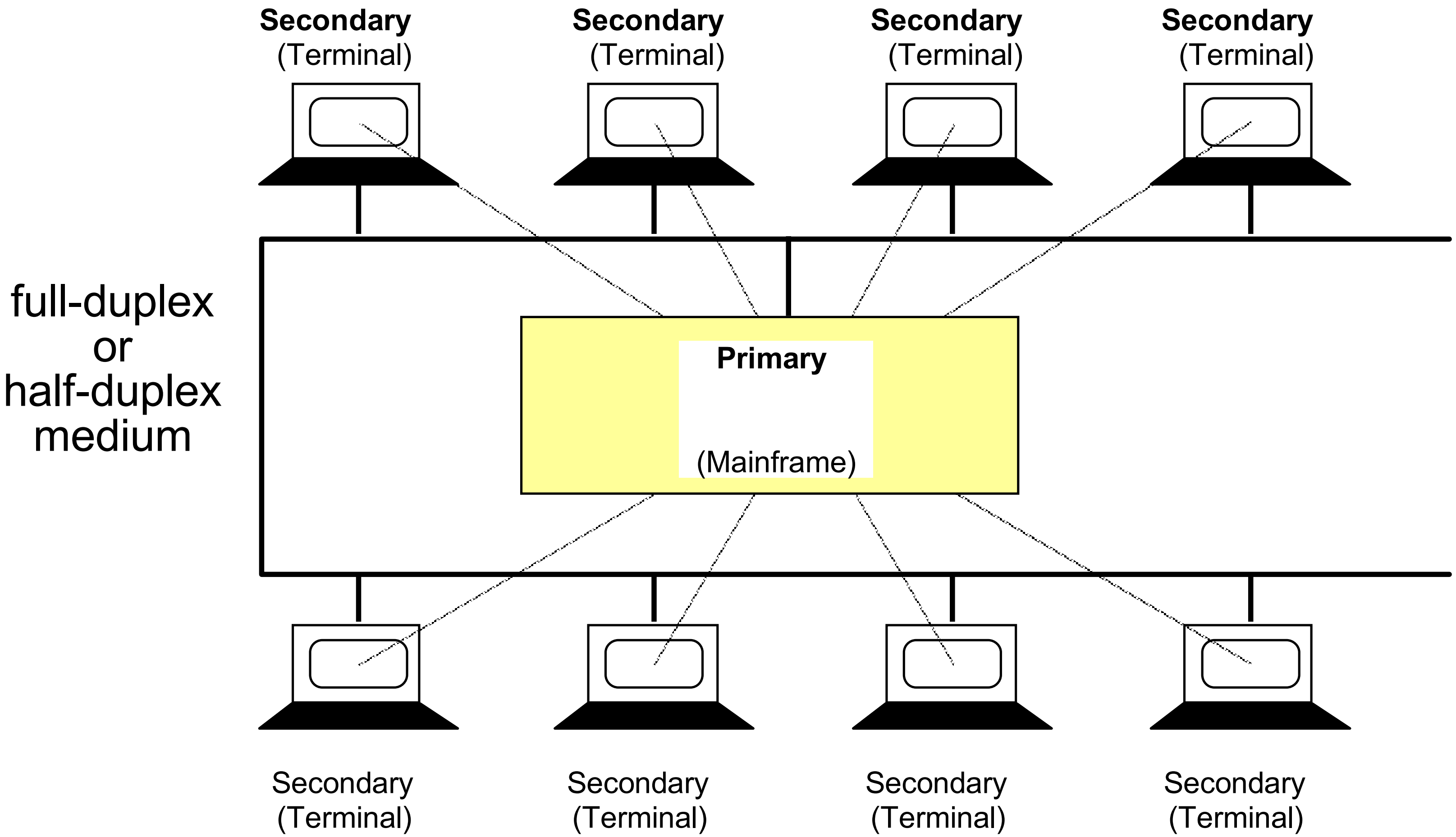
HDLC as example of a link layer protocol

Standard	HDLC (High Level Data Link)
Function	Reliable transmission between devices of a subnet
Objects	Frame structure according to ISO 3309.

Flag	Address	Control	Data	CRC	Flag
01111110	8 bit	8 bit	(n · 8)	16 bit	01111110

Integrity Check	16-bit Cyclic Redundancy Check
Medium Access	Master/Slave, (with slave initiative possible)
Error recovery	positive acknowledgement and retry
Flow control	7-frames (127 frames) credit system

HDLC Topology

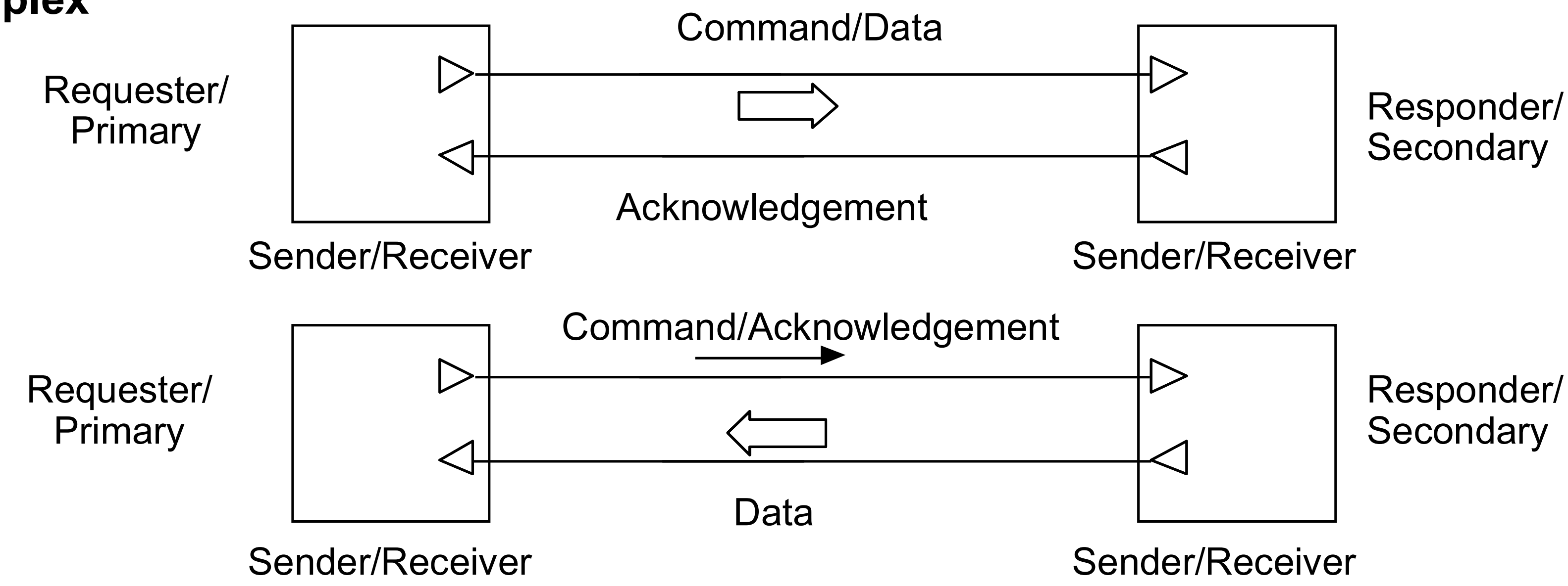


The Primary (Master) is connected with the Secondaries (Slaves) over a multidrop bus (e.g. RS 485) or over point-to-point lines

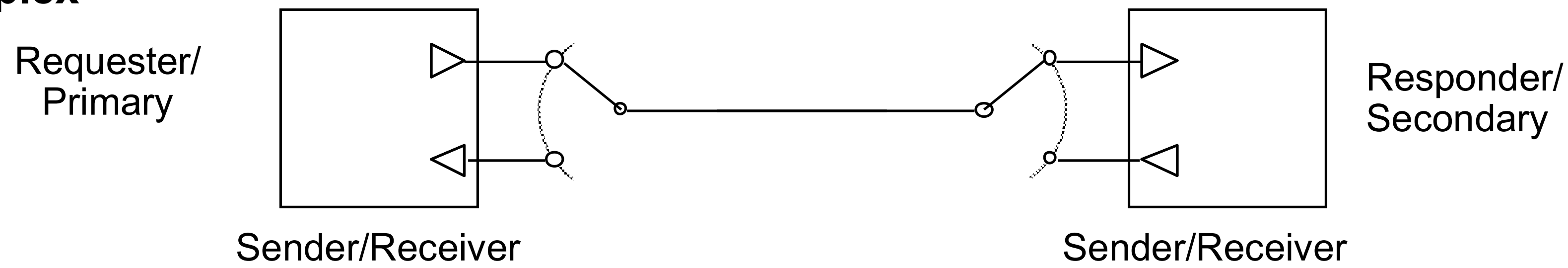
HDLC bases physically on a bus, but is logically a star network

HDLC - Full- and Half duplex operation

Full-Duplex



Half-Duplex



The Primary switches the Secondary to send mode, the Secondary sends until it returns control

Link Layer Outline

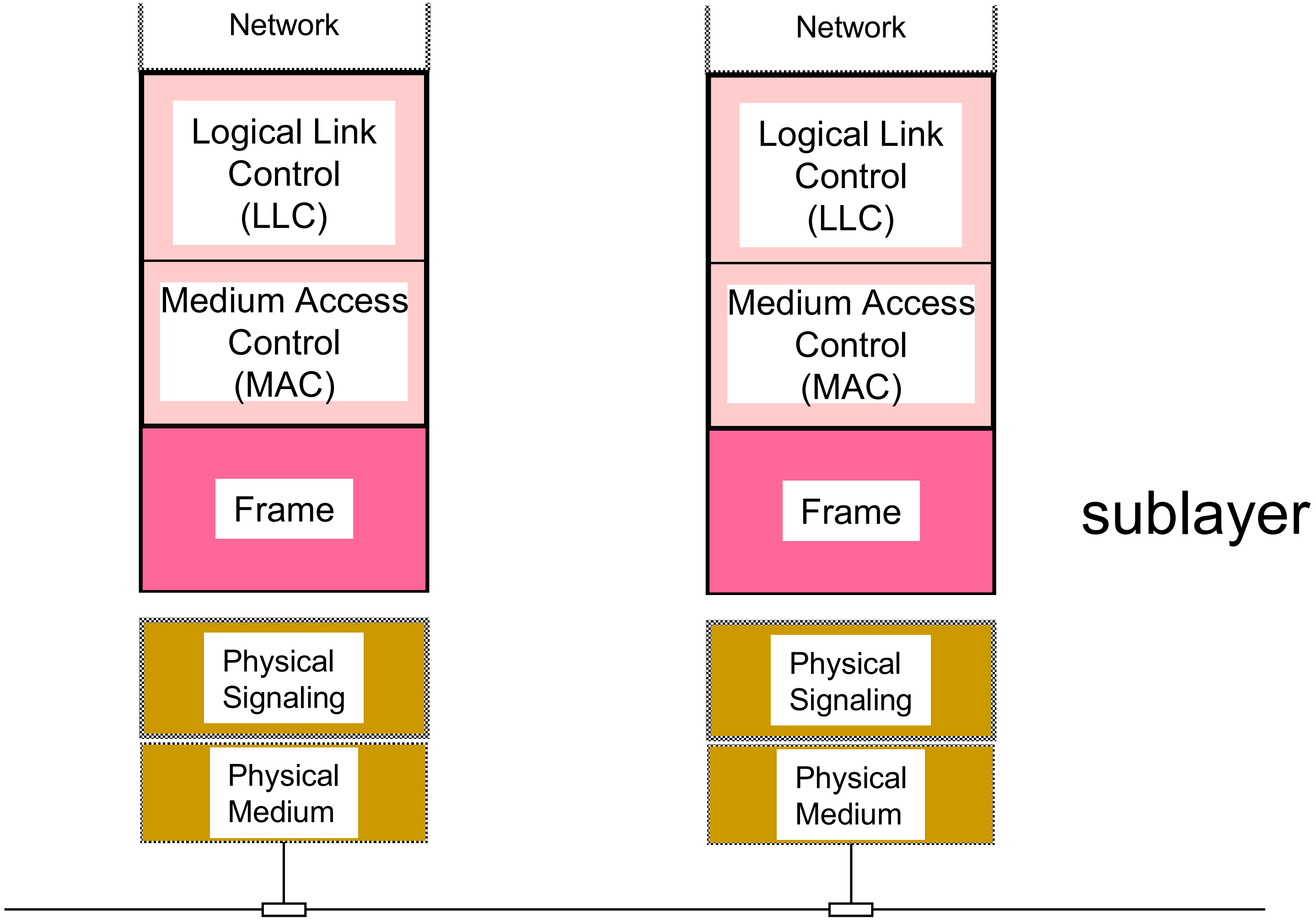
Link Layer in the OSI model
- Stacks
- HDLC as example

Frame sub-layer
- Error detection
- Error correction

Medium Access control
- Quality Criteria
- Single Master
- Rings
- Ethernet
- Collision with winner
- Token Passing
- Comparison

Logical Link Control
- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

Frame Sublayer



The frame layer is concerned with the correct frame format and contents, with (practically) no consideration of the medium or speed.

Error Handling

Industry imposes high standards regarding data integrity

Transmission quality is not very high , but consequences are severe.

Errors not detected at the link layer are very difficult to catch at higher layers

Error detection

Frame data are protected by redundant information, such as parity bits,checksum, CRC (Cyclic Redundancy Check)

Error recovery

Except when medium is very poor (Power Line Carrier, radio), error correcting codes are not used.

Erroneous frames are ignored, the potential sender of the error is not informed (the address of the sender is unknown if the frame is damaged)

The sender is informed of the lack of response when it does not receive the expected acknowledgement within a time-out.

Definition of the time-out has a strong impact on real-time behaviour

Link Layer Outline

Link Layer in the OSI model

- Stacks
- HDLC as example

Frame sub-layer

- **Error detection**
- Error correction

Medium Access control

- Quality Criteria
- Single Master
- Rings
- Ethernet
- Collision with winner
- Token Passing
- Comparison

Logical Link Control

- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

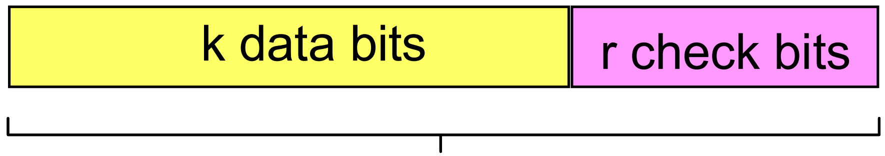
Error Detection

Error detection require redundancy in the transmitted information.

Signal redundancy: Signal quality supervision (squelch, jamming,..)

Coding redundancy: Timing-violations in decoder

Data redundancy: error detecting code



n-bit codeword = (n,k) block code

Quality criteria

- Code efficiency: CEF = k/n

- Hamming-Distance

- Residual Error Rate

Hamming Distance

The Hamming Distance between two *words* is the number of bits in which they differ:

Example

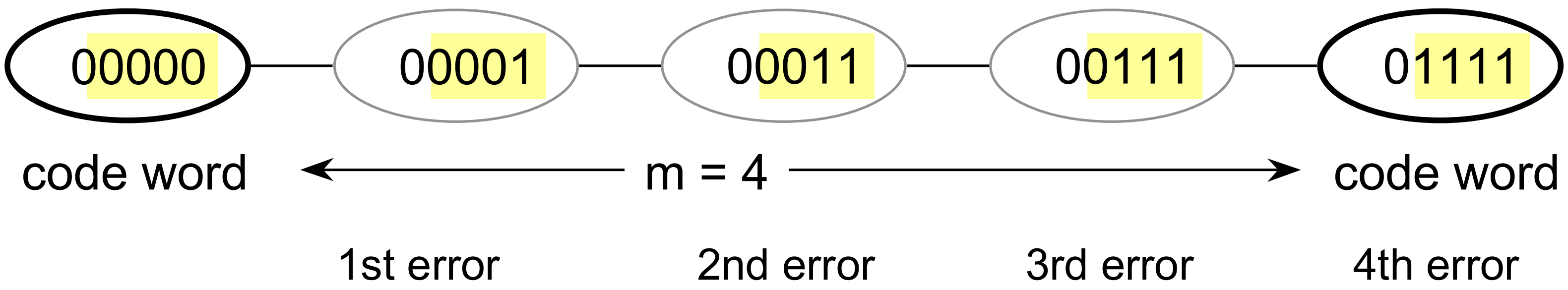
Word 1: 01100110
Word 2: 00101001 -> Hamming-Distance = 5

The Hamming Distance of a *code* is the minimum number of bits which need to be tilted in a valid codeword to produce another valid (but erroneous) codeword

Number of detectable bit errors: ZD = HD – 1

Numbers of correctable bit errors: ZC = (HD–1)/2

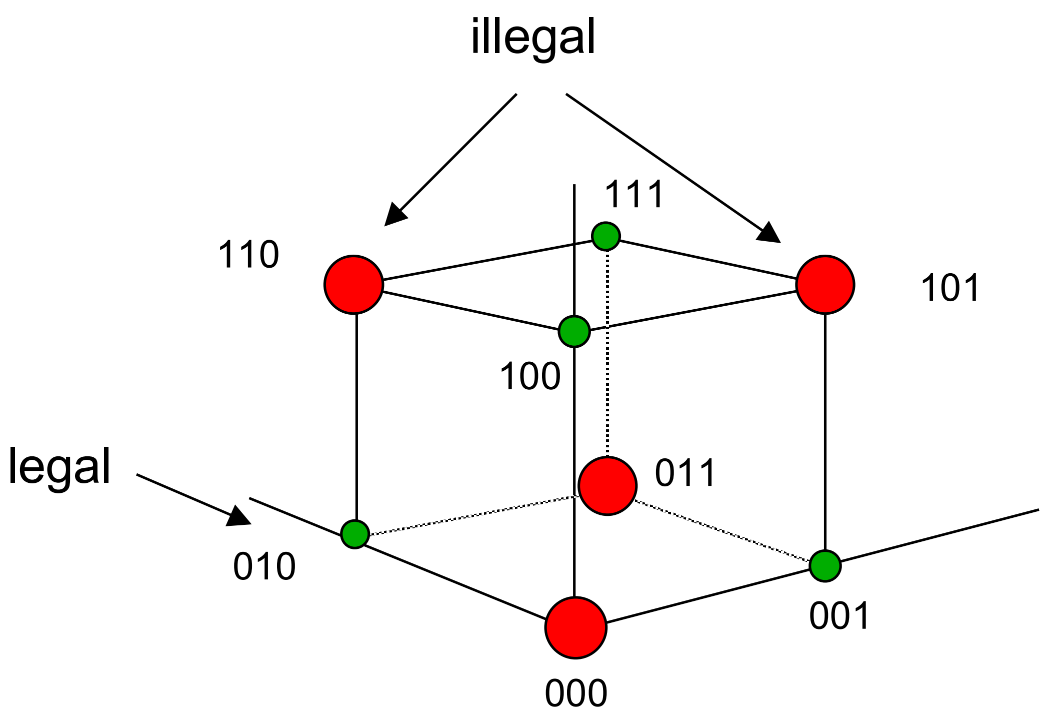
Example: HD = 4: ZD = 3, ZC = 1



Hamming distance in 3 dimensions: parity

1	0	1	1	0	0	0	0	1
par	D7	D6	D5	D4	D3	D2	D1	D0

Odd parity: sum Modulo-2 of all "1" in the codeword (including the parity bit) is odd
The parity bit is the last transmitted bit (->LSB first, a matter of convention)

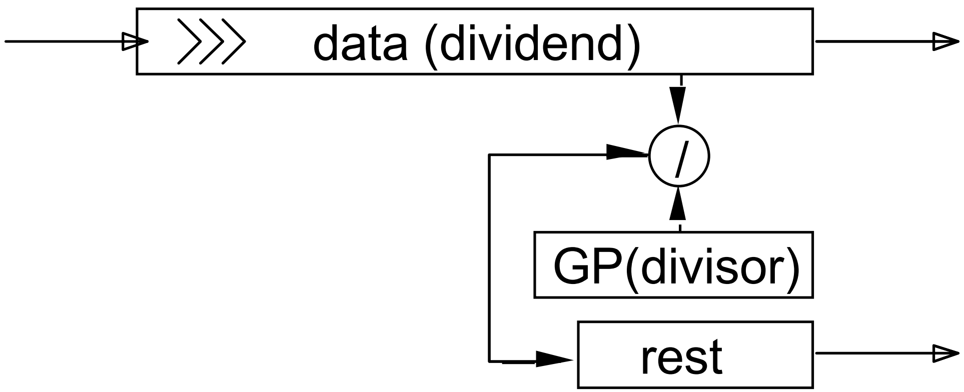


Error Detection through CRC

(Cyclic Redundancy Check)

Principle

The data stream consists of a sequence of n "0" and "1"
This sequence is treated as a polynomial of degree n.
This polynomial is divided by a Generator polynomial of degree m, m<n,
The rest of this division (which has (m-1) bits) is appended to the data stream.



At reception, the data stream is divided through the same generator-polynomial, the rest of that division should be identical to the transmitted rest.
To simplify the receiver circuit, the rest is also divided by the generator polynomial, the result should then be a known constant if the transmission was error-free.
The Generator Polynomial is chosen according to the expected disturbances: burst or scattered errors, and the channel error model (bit inversion)

Residual Error Rate, Parity

Rer = Probability of occurrence of an undetected error in spite of an error detecting code as a function of the bit error probability

Example:

Hamming Distance 2

Bit error probability E_r

Residual error rate $R_{er} = 1 - (1 - E_r)^n - n \cdot E_r \cdot (1 - E_r)^{n-1}$

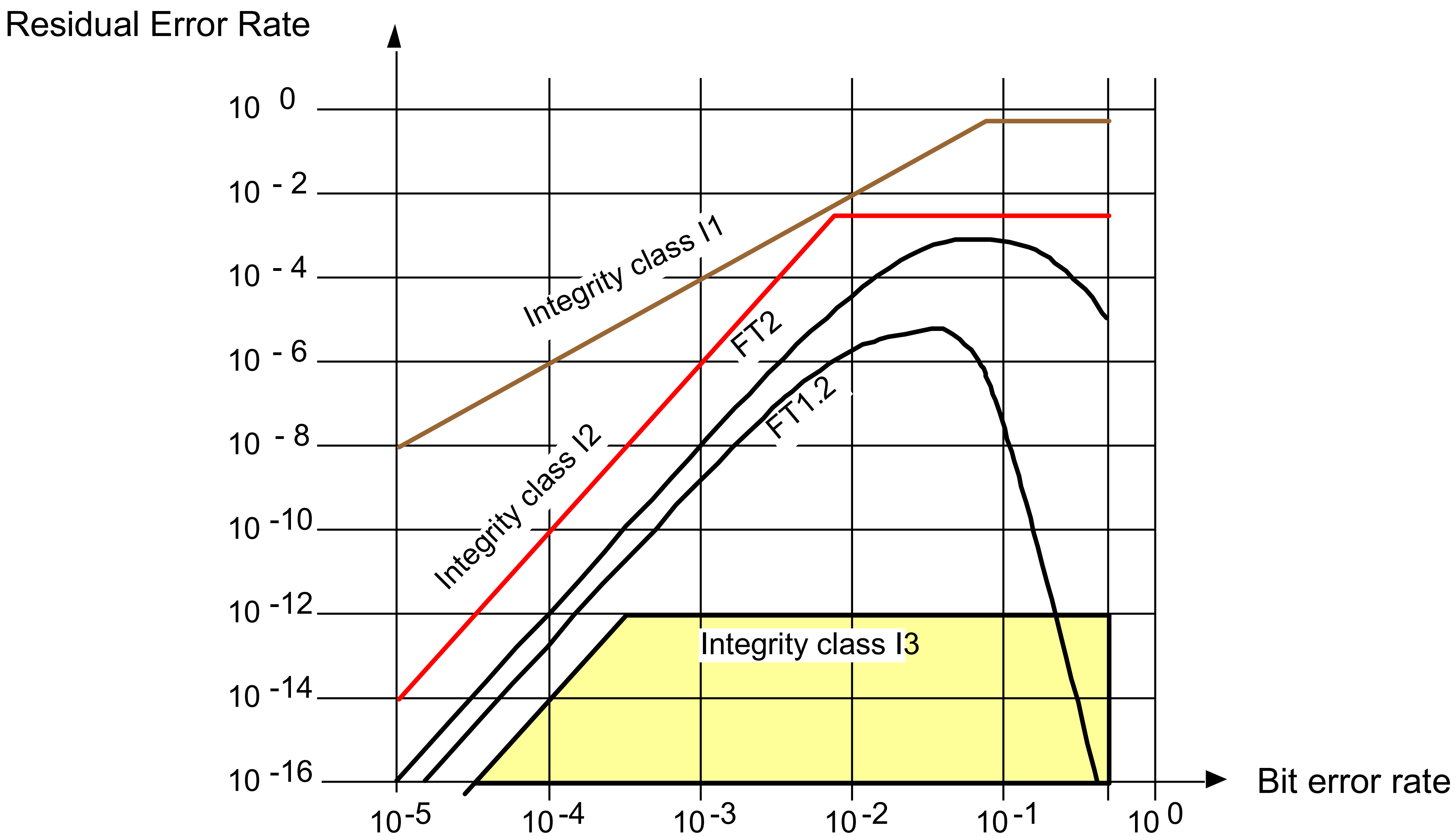
no error — exactly one error

Rer for two word length:

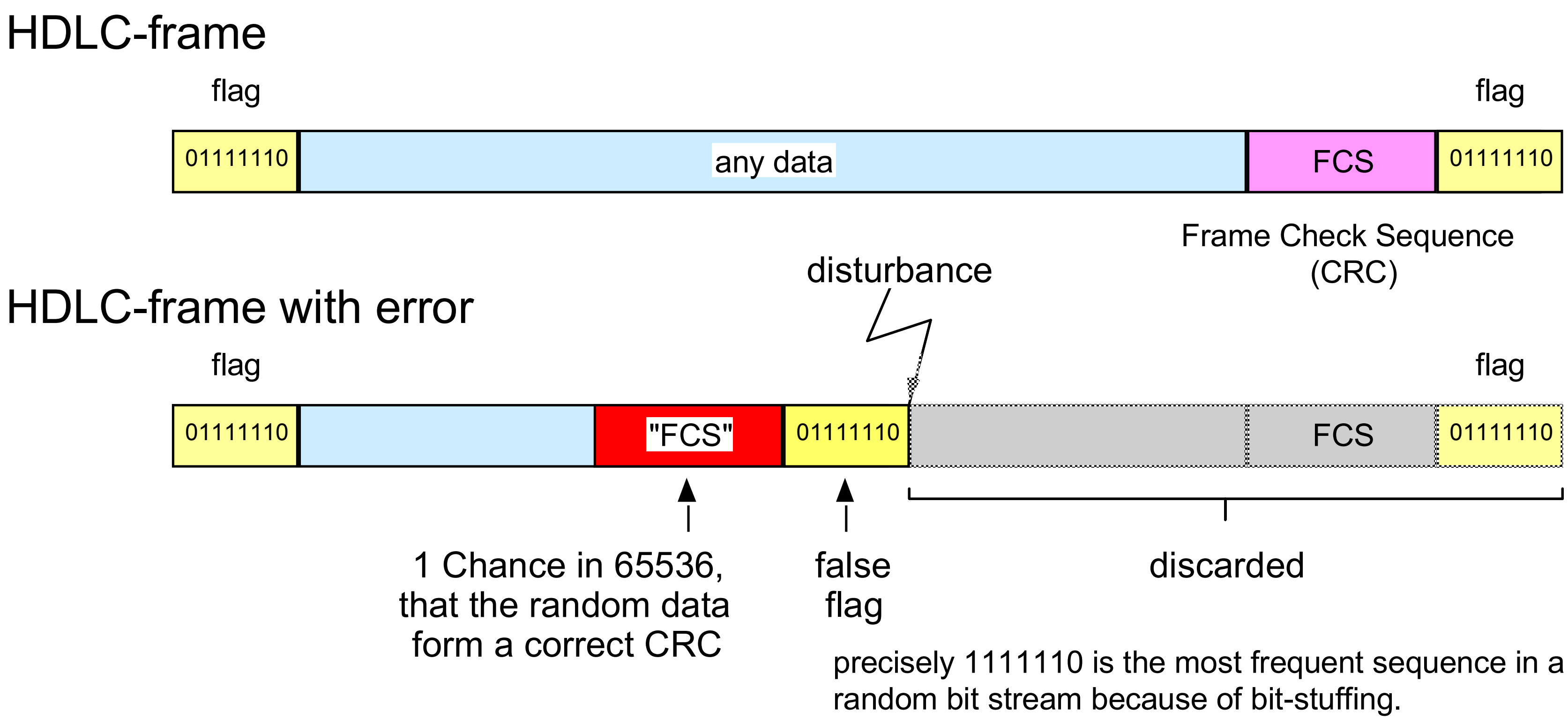
$E_r = 10^{-5}$	n = 9 bit	$R_{er} = 72 \cdot 10^{-10}$	quite efficient….
$E_r = 10^{-5}$	n = 513 bit	$R_{er} = 2.6 \cdot 10^{-5}$	quite useless ...

Integrity classes and bit error rate

The standard IEC 61870-5 (protocols for remote control of substations)
defines several classes of robustness in function of the bit error rate (bad/good bits)



Synchronization Errors



A single error can falsify a frame -> HD = 1

It is uninteresting how likely this case is, the fact is, that it can occur.

Because of this bug, HDLC when used in industry require additional error checks.

The synchronization should have a higher Hamming distance than the data itself.

Link Layer Outline

Link Layer in the OSI model
- Stacks
- HDLC as example

Frame sub-layer
- Error detection
- **Error correction**

Medium Access control
- Quality Criteria
- Single Master
- Rings
- Ethernet
- Collision with winner
- Token Passing
- Comparison

Logical Link Control
- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

Error Correcting Codes

Error correcting codes are used where the probability of disturbances is high (e.g. power line communication) and the time to retransmit is long (e.g. space probe near Jupiter).

In industry, error correcting codes are normally directly embedded in the physical layer, e.g. as part of a multitone transmission (ADSL) or of a radio protocol (Bluetooth).

Error correction necessarily decreases the data integrity, i.e. the probability of accepting wrong data, since the redundancy of correction is taken from the code redundancy.

It is much more important to reject erroneous data (low residual error rate) than to try to correct transmission.

However, when the medium is very bad (radio), error correction is necessary to transmit even short messages.

Assigning bits for error detection and correction is a tradeoff between integrity and availability.

Link Layer Outline

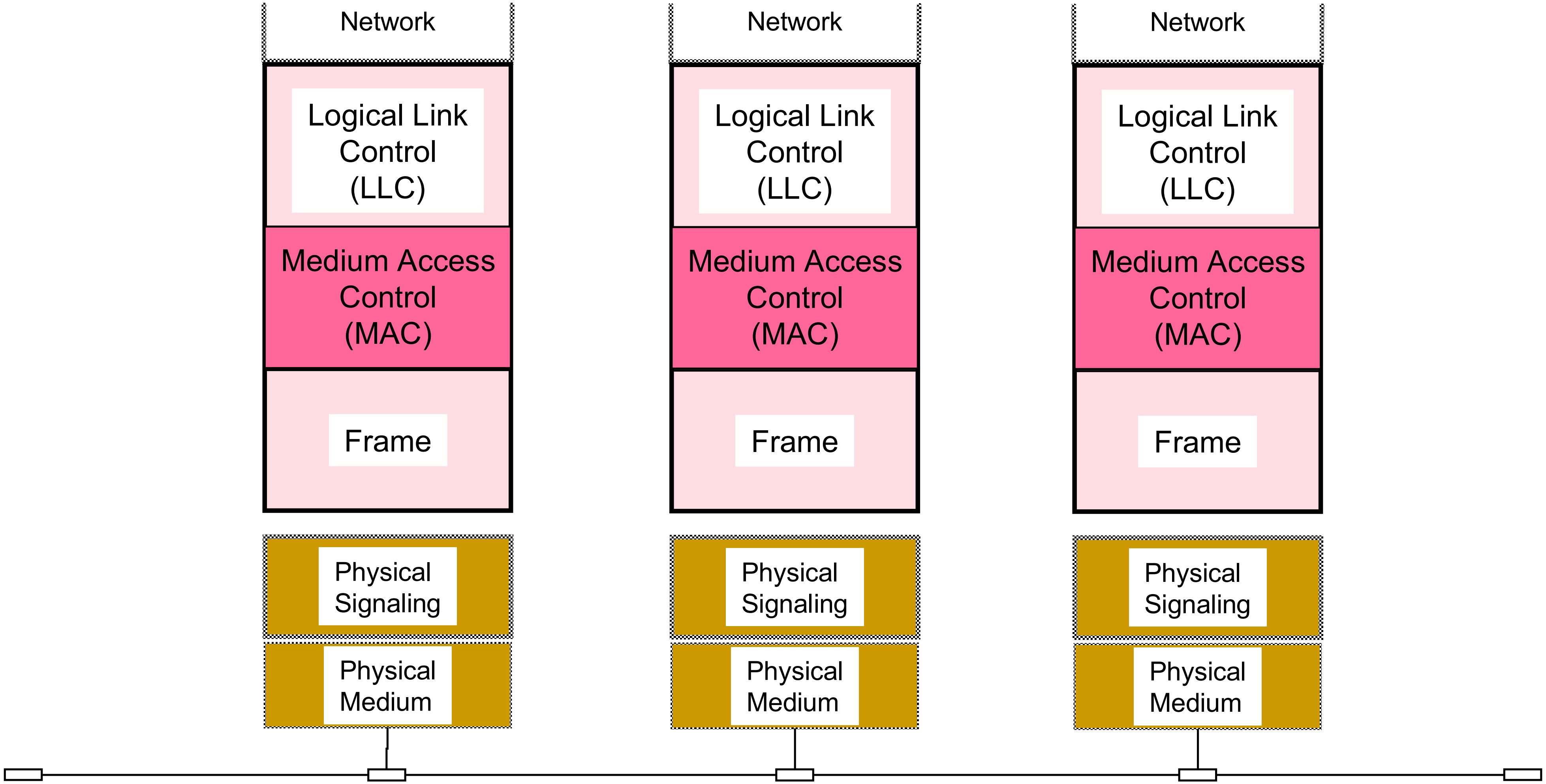
Link Layer in the OSI model
- Stacks
- HDLC as example

Frame sub-layer
- Error detection
- Error correction

Medium Access control
- Quality Criteria
- Single Master
- Rings
- Ethernet
- Collision with winner
- Token Passing
- Comparison

Logical Link Control
- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

Medium Access Control



Medium Access Control gives the right to send in a multi-master bus

Link Layer Outline

Link Layer in the OSI model

- Stacks
- HDLC as example

Frame sub-layer

- Error detection
- Error correction

Medium Access control

- **Quality Criteria**
- Single Master
- Rings
- Ethernet
- Collision with winner
- Token Passing
- Comparison

Logical Link Control

- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

Medium Access Control - quality criteria

Fairness	all requesters will eventually be allowed to transmit
Determinism	all requesters will be allowed to transmit within a finite time
Timelyness	all requesters will be allowed to transmit within a certain time, depending on their priority.
Robustness	communication errors or the permanent failure of one component does not prevent the others to access the medium.

Link Layer Outline

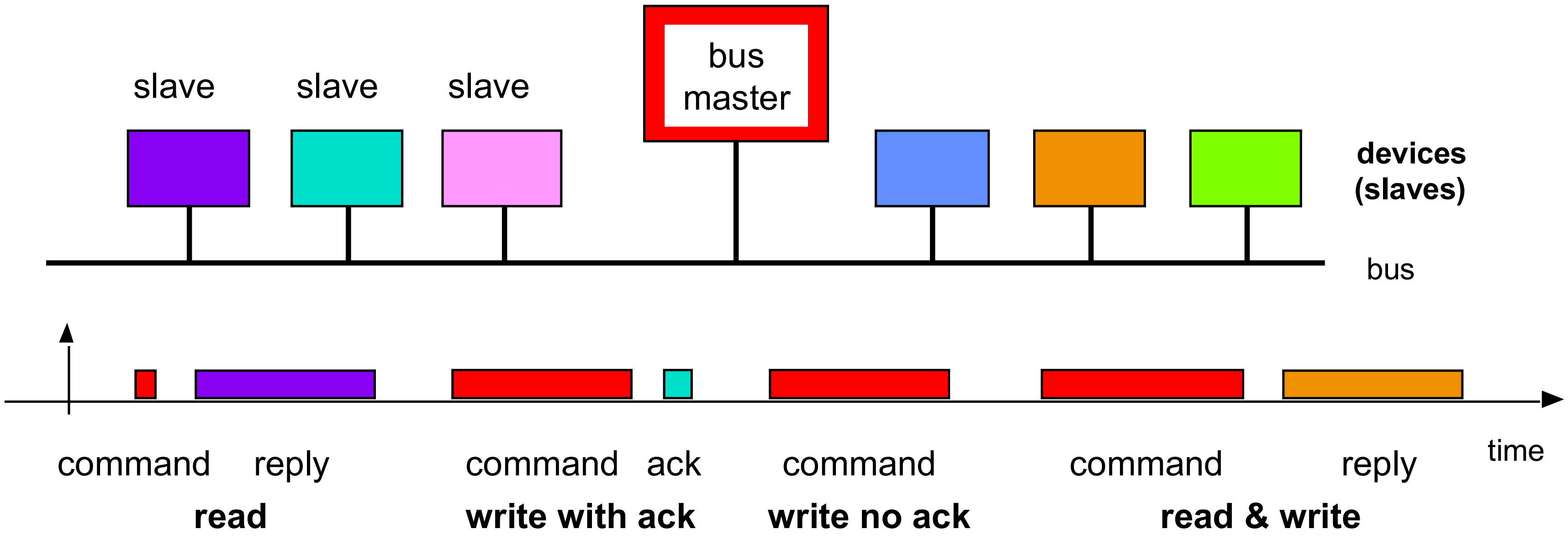
Link Layer in the OSI model
- Stacks
- HDLC as example

Frame sub-layer
- Error detection
- Error correction

Medium Access control
- Quality Criteria
- **Single Master**
- Rings
- Ethernet
- Collision with winner
- Token Passing
- Comparison

Logical Link Control
- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

MAC single master (e.g. Profibus DP)



the bus master allocates time slots to each slave
it may assign priorities (or no priority: round-robin, all are treated equally)
the master may be the source and the destination of data

+ strictly deterministic, complete and flexible control
- polling takes time, since devices which have nothing to transmit must be polled
- the master has little knowledge about data importance

improvement: “ look-at-me ” (short poll frame allowing slave to request poll of a longer frame)
 = "slave initiative" used in Profibus DP

Link Layer Outline

Link Layer in the OSI model

- Stacks
- HDLC as example

Frame sub-layer

- Error detection
- Error correction

Medium Access control

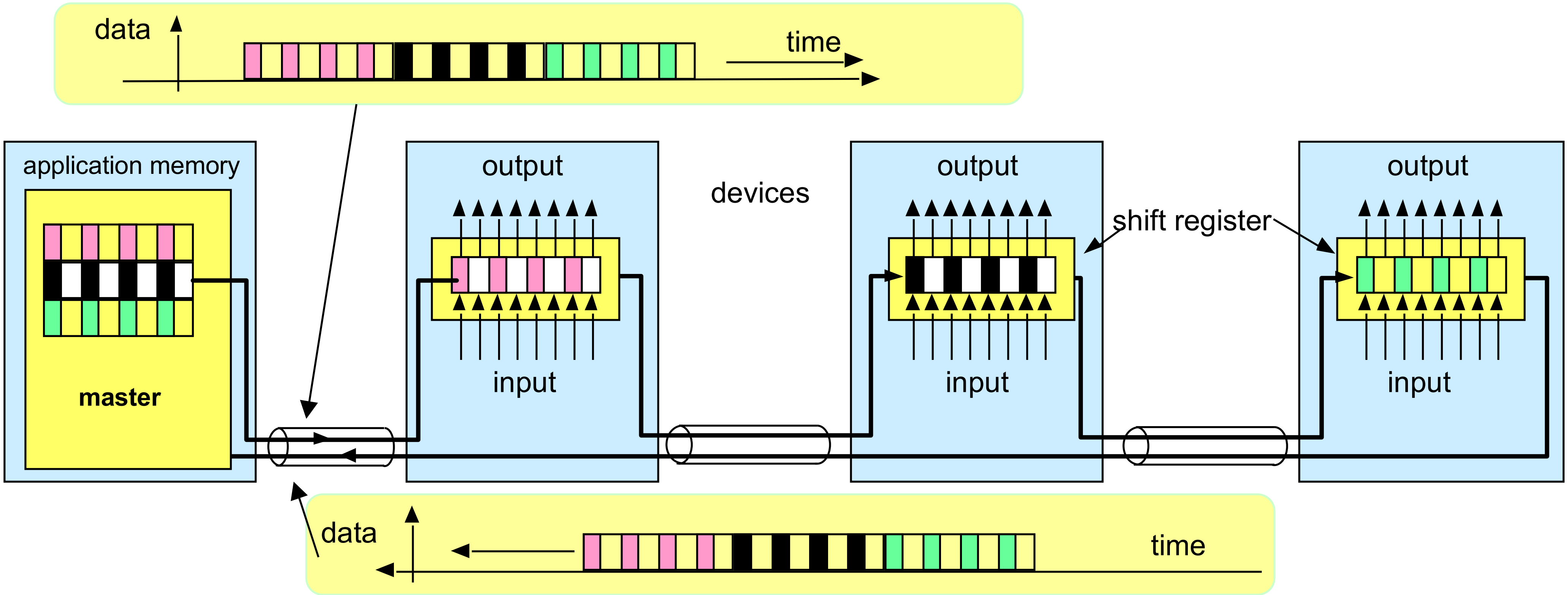
- Quality Criteria
- Single Master
- **Rings**
- Ethernet
- Collision with winner
- Token Passing
- Comparison

Logical Link Control

- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

MAC Rings (1): register insertion principle

Devices are connected by point-to-point links (no bus!), there is one sender per segment.
The operation is similar to a large shift register.
The master sends a long frame with the output data to the first device in the ring.
Each device reads the data intended for it, inserts its data instead and passes the frame to the next device.
The last device gives the frame back to the master.



MAC Rings (2): pros and cons

Two major field busses use the ring topology, Interbus-S and SERCOS
and the register-insertion principle described.

the position of the bit in the frame corresponds to the position of the device in the ring,
there are no device addresses - easy to use, but prone to misconfiguration.
each device introduces a delay at least equal to a few bits

+ deterministic access, good usage of capacity, addresses are given by device sequence on the ring, only point-to-point links

- long delays (some µs per device)

Link Layer Outline

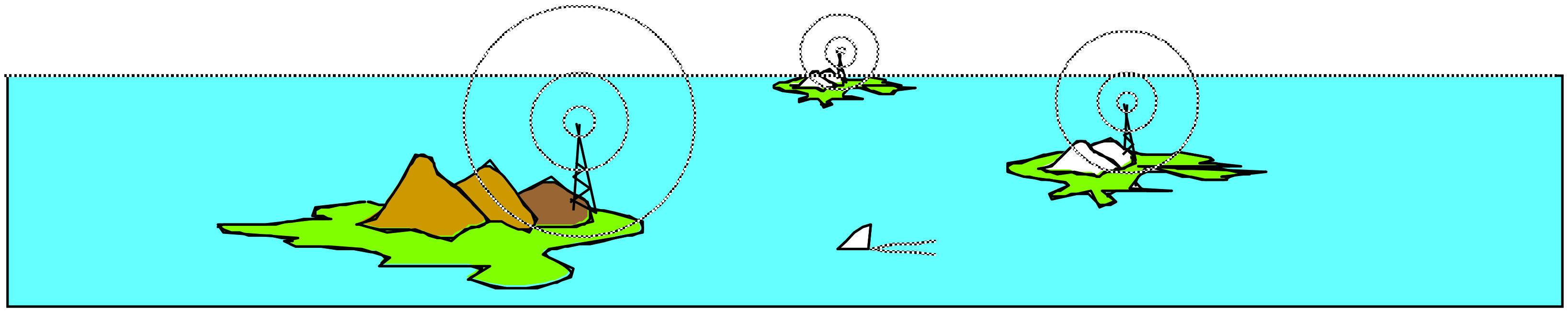
Link Layer in the OSI model
- Stacks
- HDLC as example

Frame sub-layer
- Error detection
- Error correction

Medium Access control
- Quality Criteria
- Single Master
- Rings
- **Ethernet**
- Collision with winner
- Token Passing
- Comparison

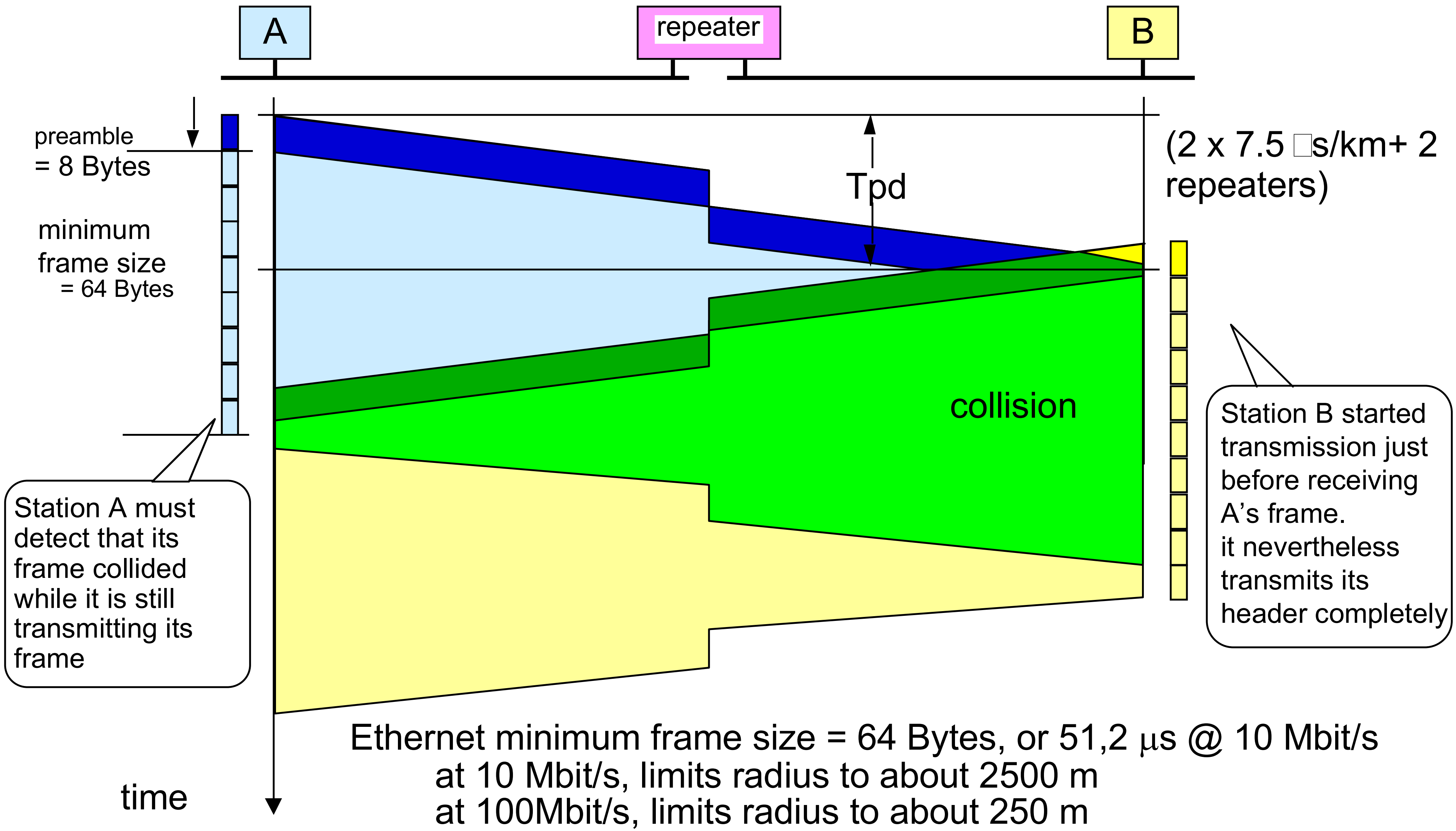
Logical Link Control
- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

MAC Ethernet (1): CSMA-CD principle (stochastic)



Principle	Every station sends as it pleases if no acknowledgement comes, it retransmit	(pure Aloha)
improvement 1:	do not send when the medium is occupied	(Carrier Sense)
improvement 2:	be aware that you are jammed	(Collision Detection)
improvement 3:	retry after a random time, doubled after each collision, max 15 times	(Binary Backoff)
Advantage:	Arbitration does not depend on number or on address of the stations	
Drawback:	No upper limit to the waiting time, mean waiting time depends on the arrival rate of frames and on their average length. The medium access is not deterministic, but for light traffic (<1%) there is no noticeable delay.	

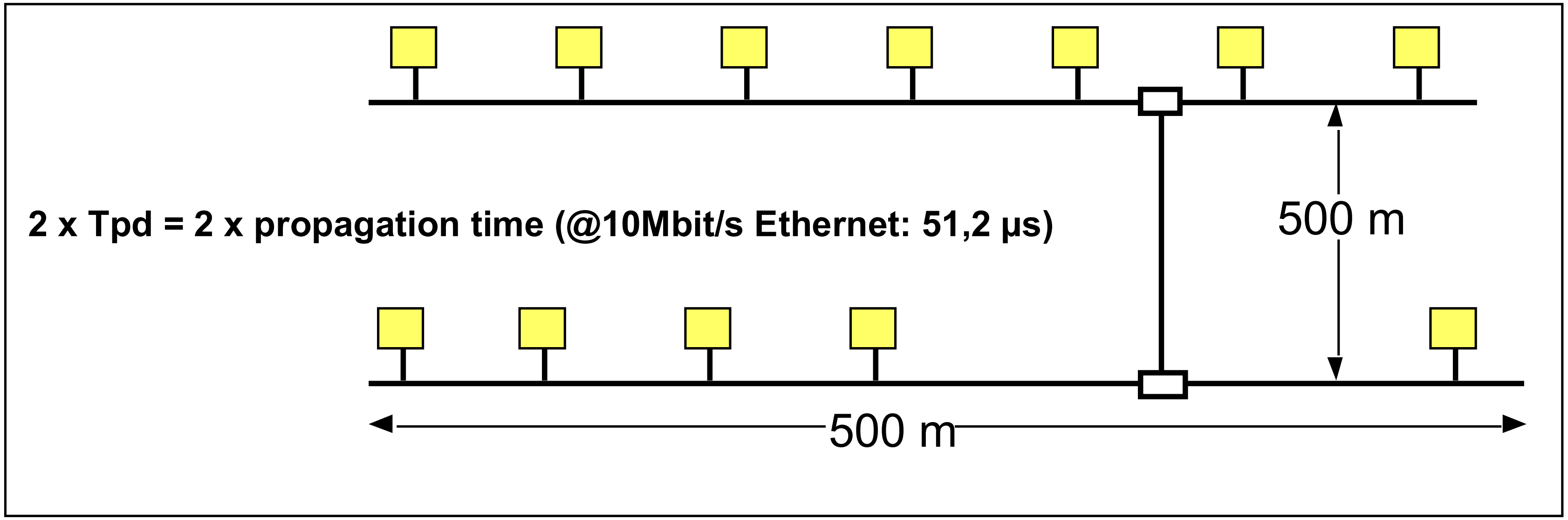
MAC Ethernet (2): collision conditions



Ethernet minimum frame size = 64 Bytes, or 51,2 µs @ 10 Mbit/s
at 10 Mbit/s, limits radius to about 2500 m
at 100Mbit/s, limits radius to about 250 m

MAC Ethernet (3): propagation conditions and bus diameter

The frame must be long enough to allow all stations to detect a collision while the frame is being transmitted.



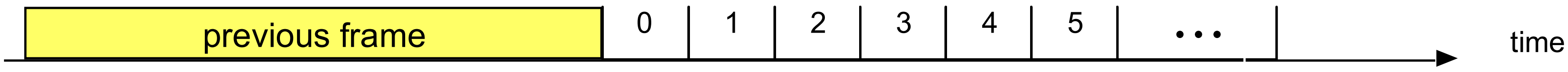
Collisions can only be detected reliably when the frame size is longer than the propagation delay -> padding to a minimum size (512 bits = 64 Bytes)

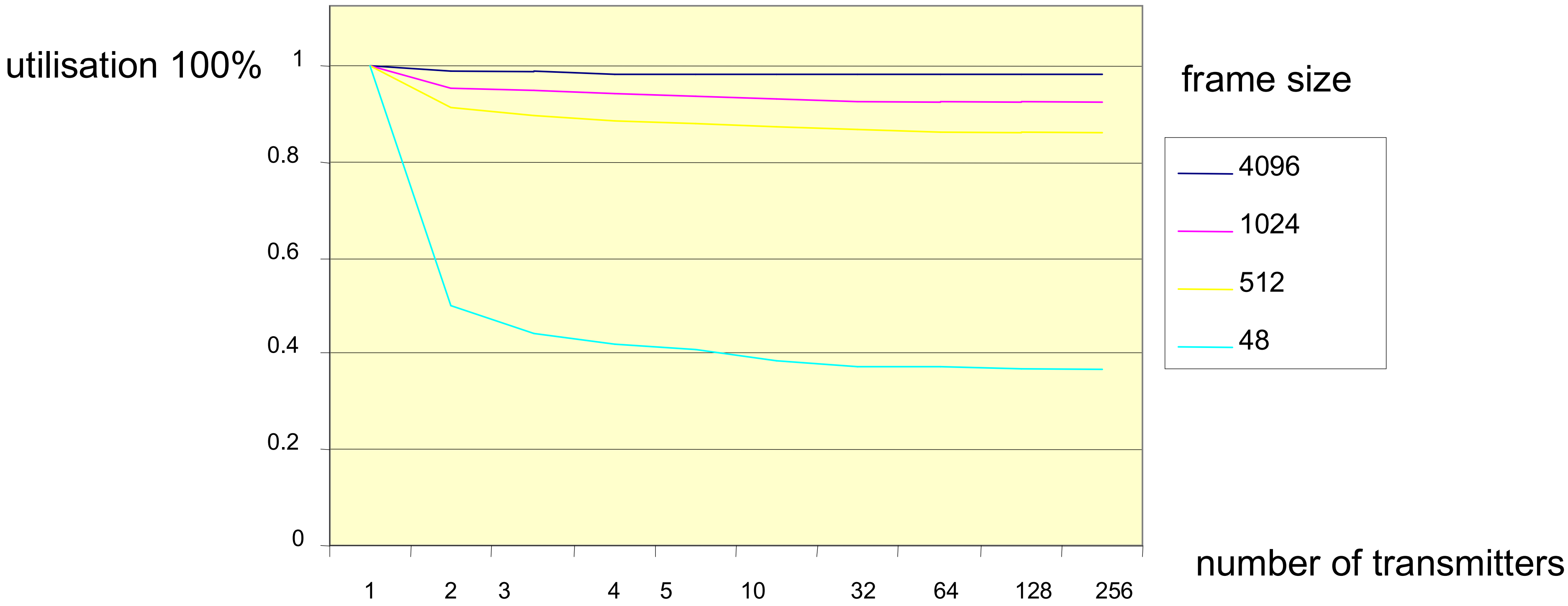
The "diameter" of the network is limited to 2,5 km

Since a station which expects a collision must wait one slot time before transmitting, the maximum frame throughput on Ethernet is limited by the slot time.

MAC Ethernet (4): collision probability and simultaneous transmitters

After the end of a frame, a transmitter chooses a slot at random from a fixed number of slots





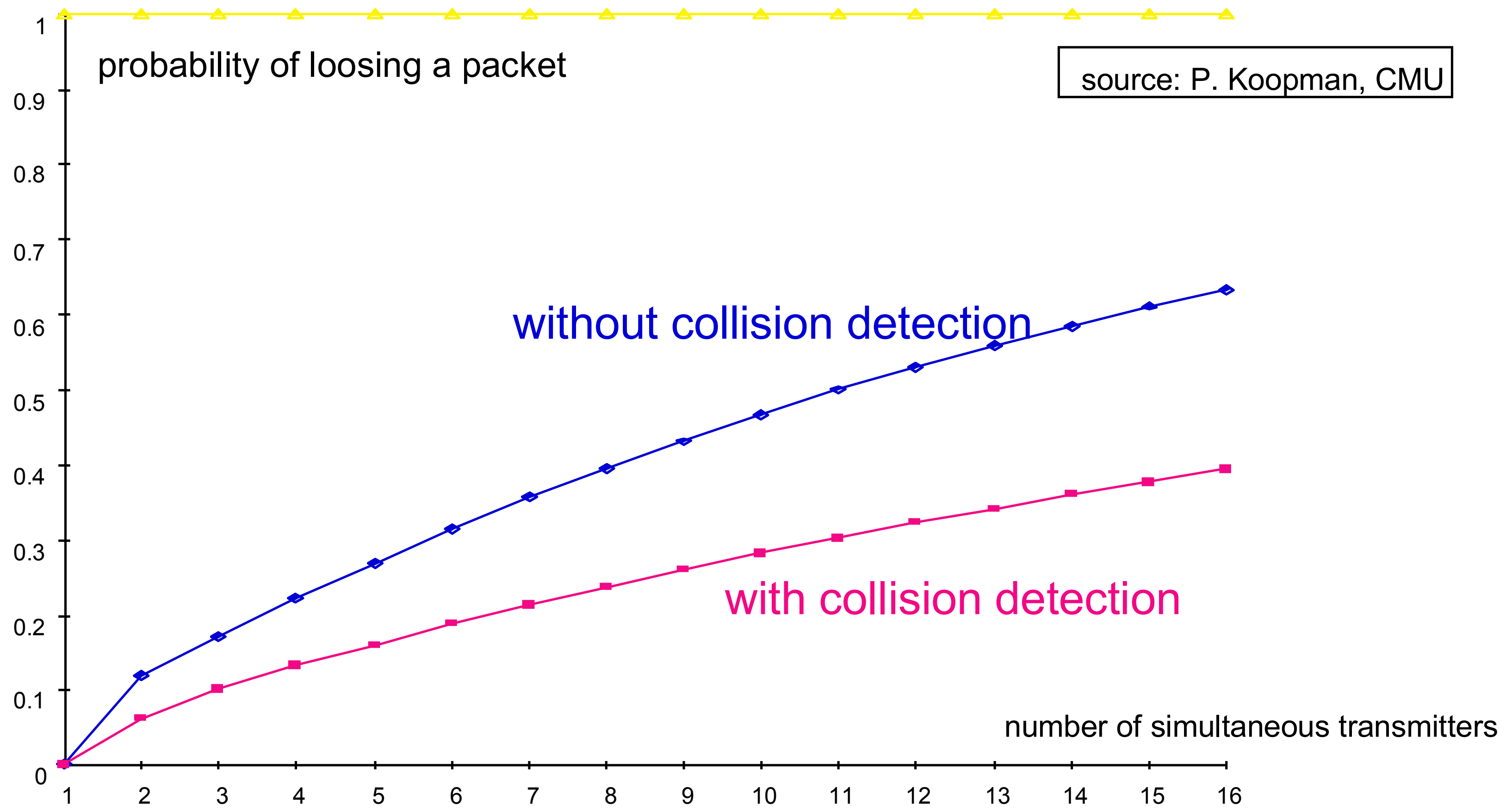
Ethernet is not efficient for small frame size and large number of simultaneous transmitters

Ethernet is considered to enter overload when reaching 10%-18% load

MAC Ethernet (5): when collisions can't be detected

It is not always possible to detect collisions.

LON uses a p-persistent MAC with 16-slot collision avoidance (p = 1/16)



⇨ A small number of simultaneous transmitters causes a high probability loss of a packet .
LON can retry up to 255 times: probability of lost message is low, but delay is long

Link Layer Outline

Link Layer in the OSI model

- Stacks
- HDLC as example

Frame sub-layer

- Error detection
- Error correction

Medium Access control

- Quality Criteria
- Single Master
- Rings
- Ethernet
- **Collision with winner**
- Token Passing
- Comparison

Logical Link Control

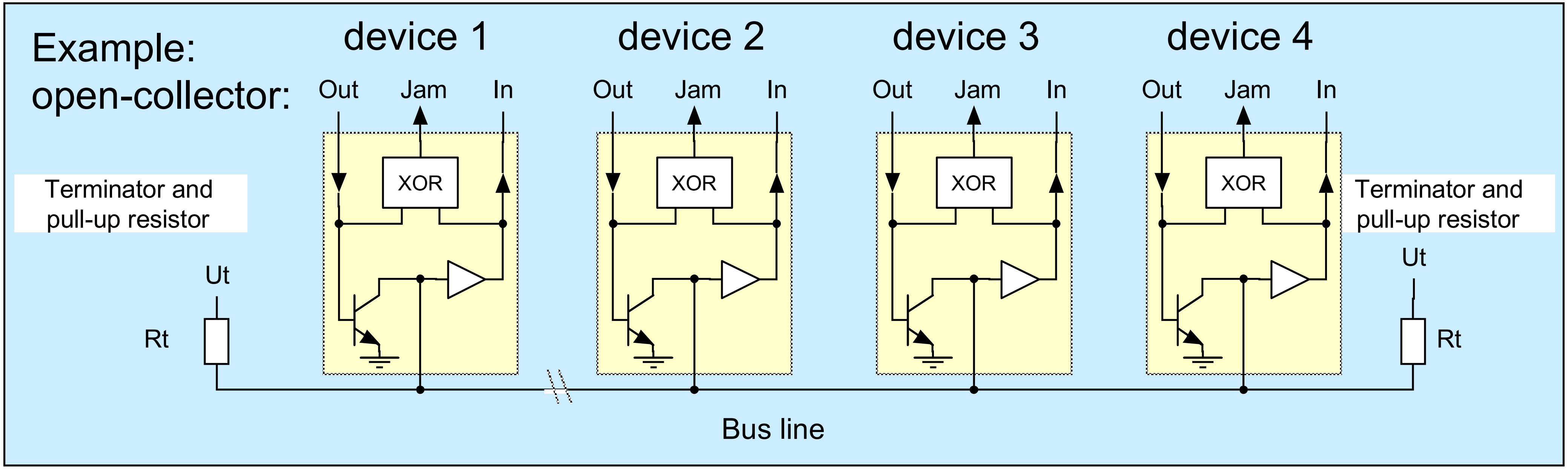
- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

MAC CAN (1): Deterministic Arbitration

The CAN fieldbus uses media with a dominant and a recessive state

When several transmitters are simultaneously active, the dominant state wins over the recessive state if there is at least one transmitter sending the dominant state (dominant is “Low” in open collector, "Bright" in an optical fiber, or a collision on a medium that allows collisions).

Such a medium allows a bit-wise "Wired-OR" operation

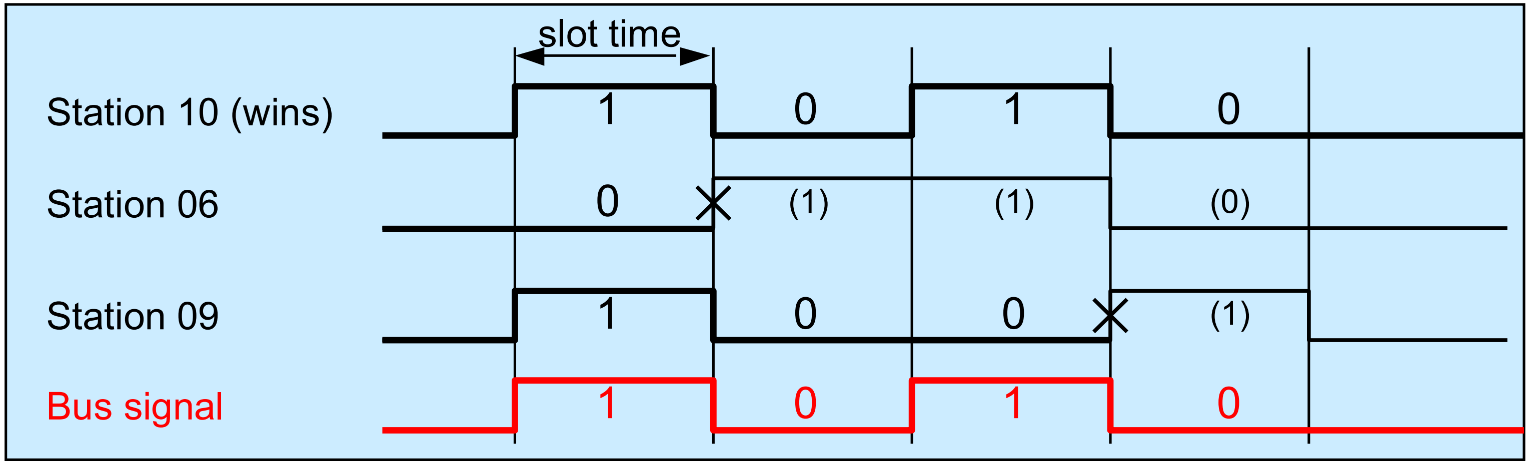


A device is capable to know if the signal it puts on the line is disturbed (XOR).

MAC CAN (2): Collision with Winner

Also known as "Urn" or "binary bisection" method

- Each station has a different identity (in this example: 4 bit ID)
- Each station listens before sending and can detect collisions
- Competing stations start transmitting at the same time (1st bit is a SYNC-sign)
- Each station begins its transmission with its identity, bit by bit
- In case of collision, "1" wins over "0" ("1" = Low, bright, dominant).
- A station, whose "0" bit was transformed into a “1" retires immediately
- The winning station has not been disturbed and transmits.
- Loosing stations await the end of the ongoing transmission to start again.



Advantage: deterministic arbitration (assumes fairness), good behavior under load

Drawback: the size of the unique ID defines arbitration time,
transmission delay defines slot time -> 40m @ 1 Mbit/s, 400m @ 100 kbit/s

MAC CAN (3): Deterministic Arbitration

Advantage: deterministic arbitration (assumes fairness, I.e. a device only transmits again when all losers could), good behavior under load.

Drawback: the slot time (one bit time) must be long enough so that every station can detect if it has been disturbed - I.e. twice as long as the propagation time from one end of the bus to the other (signal speed = 5 ns / m).

Therefore, the bit rate is dependent on the bus extension:

40m @ 1 Mbit/s, 400m @ 100 kbit/s

the size of the unique ID defines arbitration time.

Link Layer Outline

Link Layer in the OSI model
- Stacks
- HDLC as example

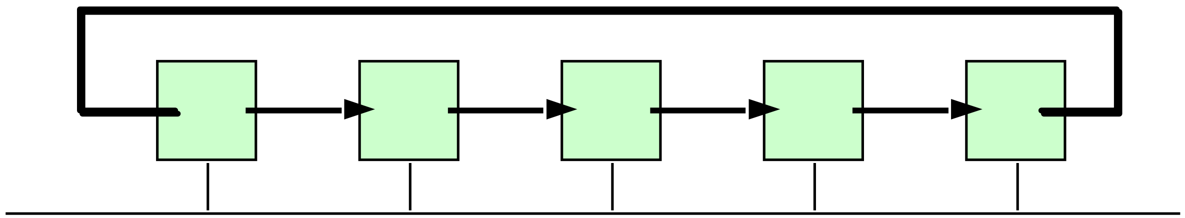
Frame sub-layer
- Error detection
- Error correction

Medium Access control
- Quality Criteria
- Single Master
- Rings
- Ethernet
- Collision with winner
- **Token Passing**
- Comparison

Logical Link Control
- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

MAC Profibus (1): Token principle

z.B.: Token Bus (IEEE 803.4), Profibus (IEC 61158)



All stations form a logical ring

Each station knows the next station in the ring (and the overnext)

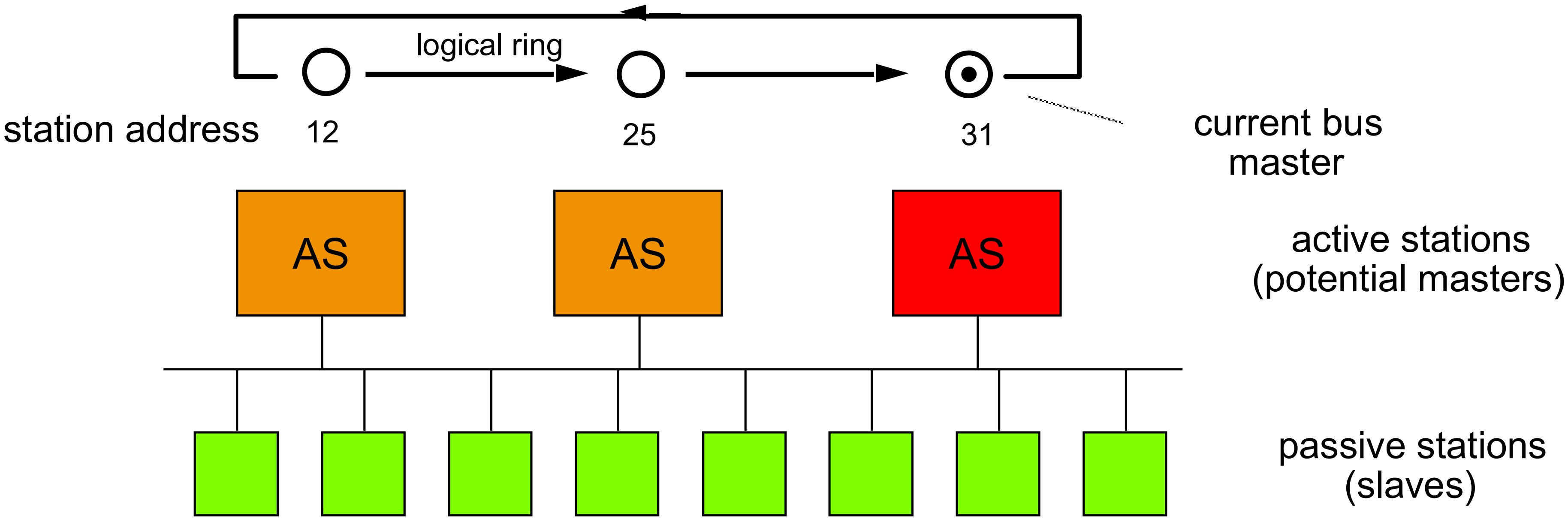
Each station delegates the right to send to the next station in the ring,
(in the form of a token frame or as an appendix to a data frame).

Variants Token with Priority (back to the station with the highest priority)

Problems: Loss or duplication of token, initialization

do not confuse with token ring !

MAC Profibus (2): Token passing



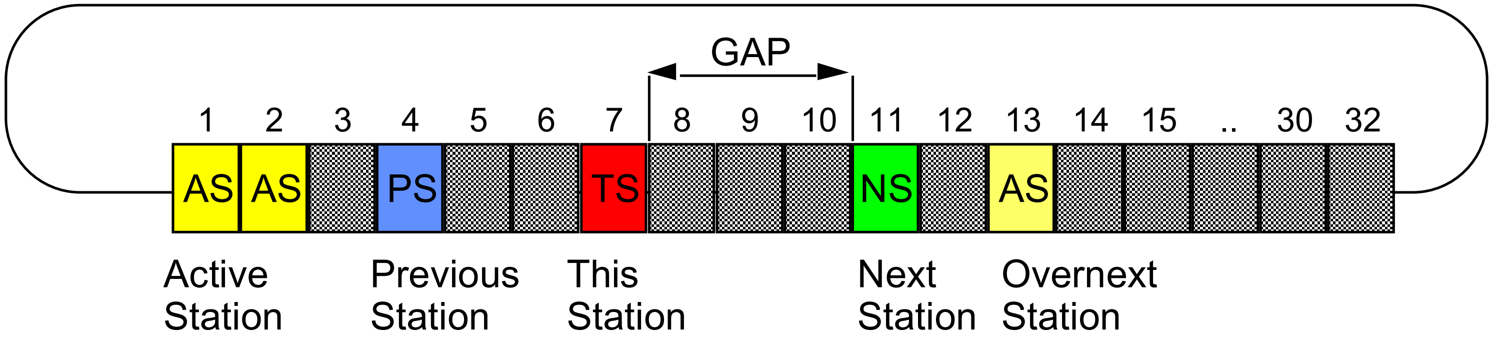
Active Stations (AS) can become master if they own the token, for a limited duration (one frame only).

After that time, the master passes the token to a station with a higher address or, if it has the highest address, to the station with the lowest address.

A station must send at least one frame (data or token) when it gets its turn.

MAC Profibus (3): Token passing algorithm

Each station holds a List of Active Stations (LAS)



When the current master has no more data to send, or when its turn expires, it sends a token frame to the Next Station (NS) in the ring.

NS acknowledges reception of the token.

If the master does not receive an acknowledgement for two consecutive trials, the master removes the station from the LAS and declares the overnext active station (OS) as NS.

This station accepts the token only if it receives it twice.

The master tests at regular intervals with a "Request FDL-Status" for the presence of further stations in the gap between itself and NS.

MAC Profibus (4): Token initialization

A starting station listens to the bus before trying to send.

If it senses traffic, a station records the token frames and builds a list of active stations (LAS).

In particular, it observes if a station with the same name as itself already exists.

When a master checks the gap, the station will receive a token offer.

If a station does not register any traffic during a certain time, it sends a token frame to itself.

It sends the first token frame to itself, to let other starting stations register it.

Only when it detects no other station does a station begin with a systematic
poll of all addresses, to build the LAS.

Link Layer Outline

Link Layer in the OSI model
- Stacks
- HDLC as example

Frame sub-layer
- Error detection
- Error correction

Medium Access control
- Quality Criteria
- Single Master
- Rings
- Ethernet
- Collision with winner
- Token Passing
- **Comparison**

Logical Link Control
- Connection-Oriented and connectionless
- Error recovery
- Flow control
- HDLC

Comparison of Medium Access Control Methods

stochastic	deterministic
optimistic	pessimistic
carrier sense	central master
collision detection	collision with winner
p-persistent collision	token passing

Link Layer Outline

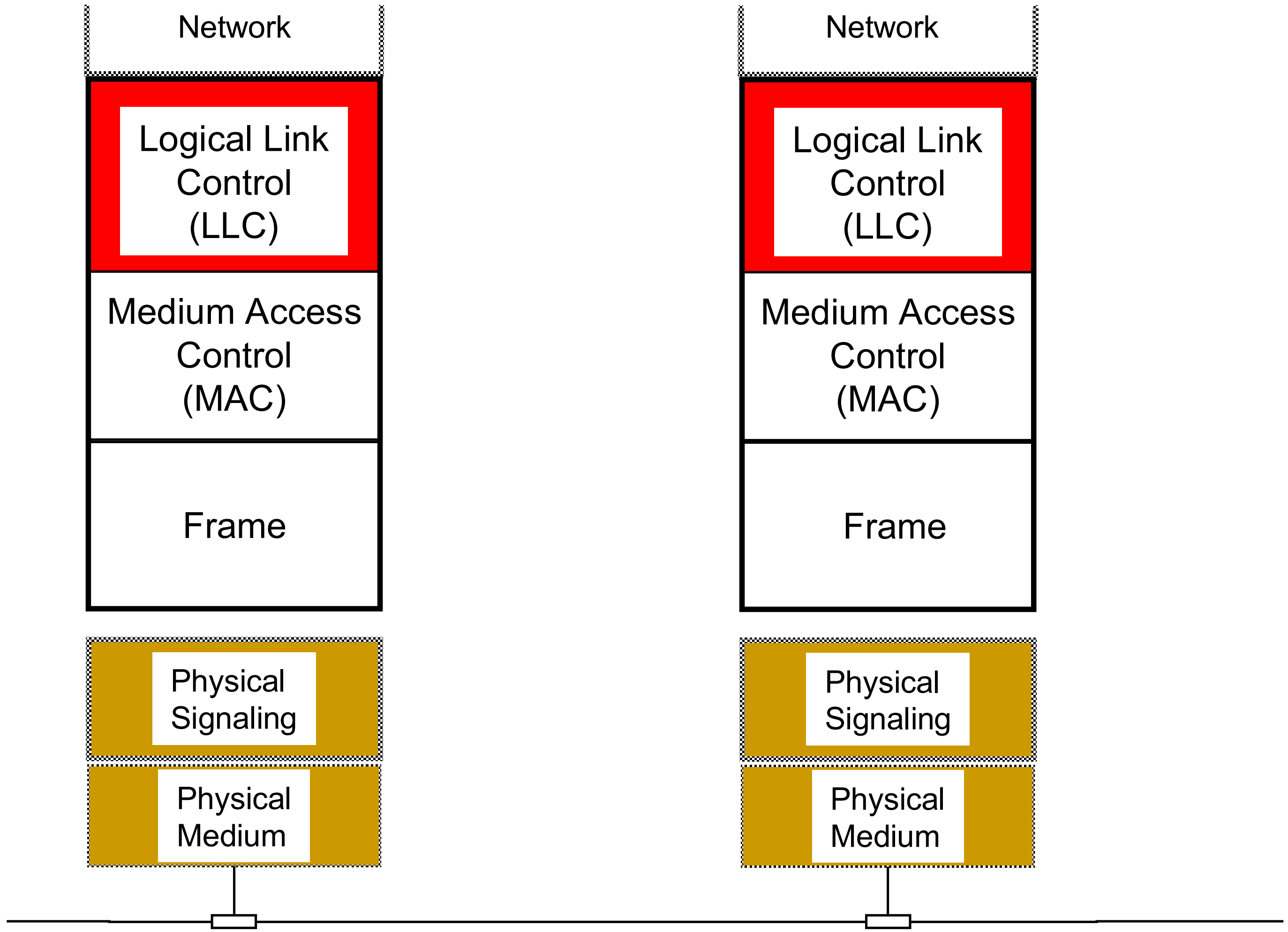
Link Layer in the OSI model
- Stacks
- HDLC as example

Frame sub-layer
- Error detection
- Error correction

Medium Access control
- Quality Criteria
- Single Master
- Rings
- Ethernet
- Collision with winner
- Token Passing
- Comparison

Logical Link Control
- **Connection-Oriented and connectionless**
- Error recovery
- Flow control
- HDLC

Logical Link Control Sublayer



Two Link services:

- unacknowledged connectionless service and
- connection oriented services

Connection-Oriented and Connection-Less communication

these considerations apply to all levels of the OSI model

Connectionless mode (Datagram ≈ letter)

Each packet (Datagram) contains all information needed to forward it to its final destination over the network, including the path back to the sender.

The network assumes no responsibility for the ordering of packets and does not try to recover damaged datagrams.
The burden of flow control and error recovery is shifted to the application.

Semantic of CL-transmission

Send_Packet (source, destination, Packet1);
Send_Packet (source, destination, Packet2);

Connection-Oriented mode (Virtual Circuit ≈ telephone)

A connection is first established between sender and receiver.
Information packets are identified by their connection identifier and by their sequence number within that connection.
The network cares for opening and closing connections and ensures that packets are received in same order as they are sent, recovering lost packets and controlling the flow.
Applications see the network as a reliable pipe.
Connection is closed after use (and reused)

Semantic of CO-transmission

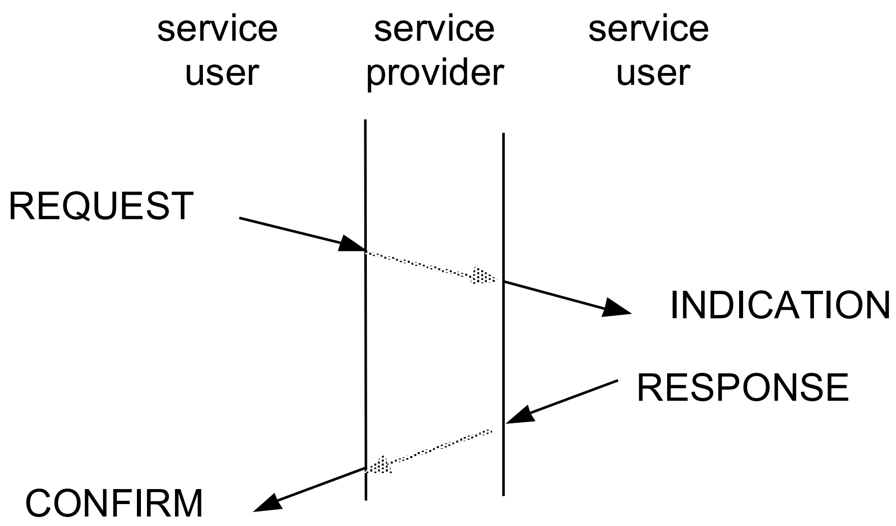
Open_Channel(Node, Task, Channel_Nr);
Send_Message (Channel_Nr, Msg1);
Send_Message (Channel_Nr, Msg2);
Close(Channel_Nr);

Msg1 will be received before Msg2, sequence is maintained.

Connection-Oriented Link Service

Task: Flow Control and Error Recovery

Connection establishment and disconnection
Send and receive frames
Flow Control (Buffer control)
Retry in case of error



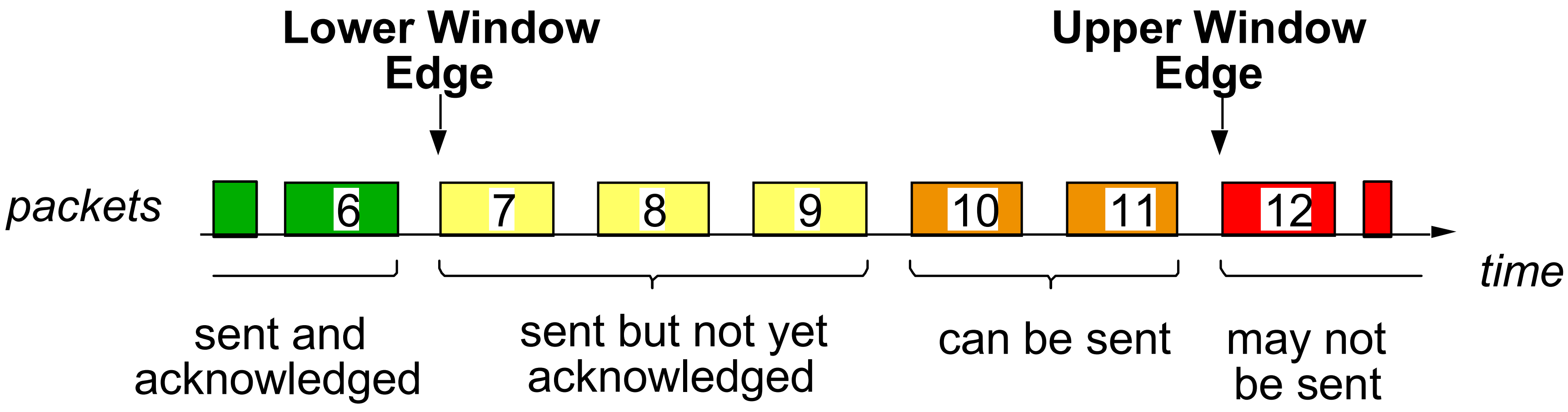
Flow Control

(= synchronization at the link layer)

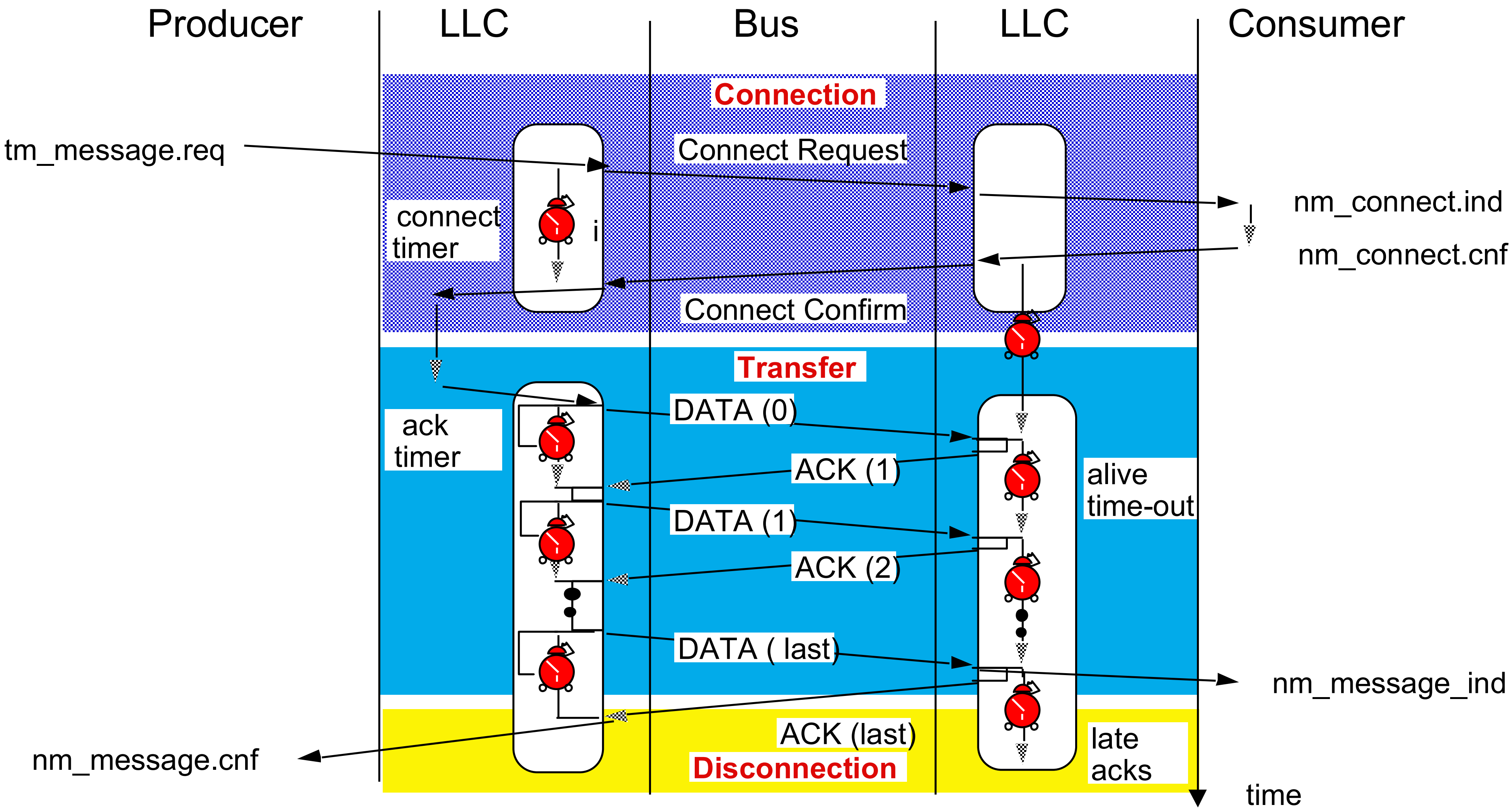
"Adapt the speed of the sender to that of the receiver"

Methods

- Use Acknowledgements: do not send until an acknowledgement is received (acknowledgements have two purposes: error recovery and flow control !)
- Credit: the receiver indicates how many frames it can accept (sliding Window protocol). Improvement: variable window size.
- Explicit braking (CTRL-S/ CRTL-Q)



Simple transfer with window size = 1



Every packet takes at least two propagation times

Error Recovery

General rule: Erroneous information frames are repeated
(error correcting codes belong to physical layer)

1) In cyclic transmission, information is not explicitly repeated in case of loss,
the receiver will receive a fresh information in the next cycle.
A freshness control supervises the age of the data in case communication ceases.

2) In event-driven transmission, no information may be lost, a repetition is explicit:

a) The sender of information frames expects acknowledgement of the receiver, indicating which information it received.
b) The sender repeats the missing information a number of times, until it receives an acknowledgement or until a time-out expires.
c) To distinguish repetitions from new information, each packet receives a sequence number (in the minimum odd/even).
d) The receiver acknowledges repetitions even if it already received the information correctly.

3) In broadcast transmission, it is relatively difficult to gather acknowledgements from all possible receivers, so in general unacknowledged broadcast is used. The receiver is expected to protest if it does not receive the information.

Link Layer Outline

Link Layer in the OSI model

- Stacks
- HDLC as example

Frame sub-layer

- Error detection
- Error correction

Medium Access control

- Quality Criteria
- Single Master
- Rings
- Ethernet
- Collision with winner
- Token Passing
- Comparison

Logical Link Control

- Connection-Oriented and connectionless
- Error recovery
- Flow control
- **HDLC**

Example: HDLC

HDLC (High-level Data Link Control is derived from IBM's SDLC (Synchronous Data Link Control)

These protocols were developed for connection of point-of-sale terminals to a one mainframe computer.

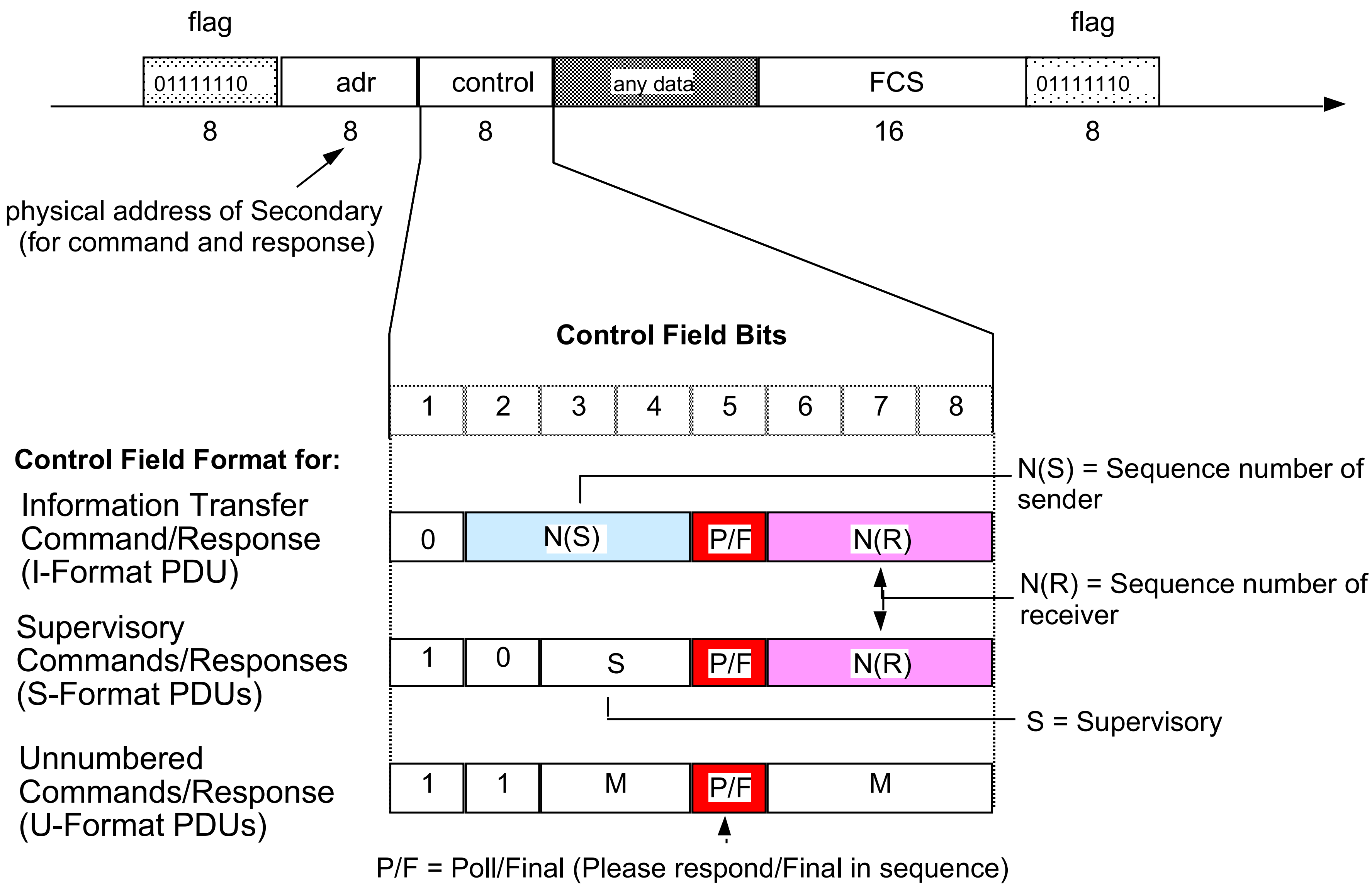
HDLC is the most frequently used link layer protocol.

It is the base for the CCITT-standard X25 (Telenet, Datex-P, Telepac) and used in Bitnet, Appletalk, etc...

The HDLC protocol is implemented in the hardware of numerous microcontrollers (e.g. Zilog 80C30, Intel, Siemens 82525,... and in some microprocessors (e.g. 68360).

HDLC is the base for the Local Area Network protocol IEEE 802.2

HDLC Control Field (ISO 4335)



HDLC Connection Types

Traffic is divided into *packets (= information frame)* each receiving a sequence number (Modulo 8).

The sender includes the sequence number in each packet.

The receiver indicates which packet it expects next, either through a special frame (Receiver Ready N(R)) or within its information frames (I-Frame, N(R))

At the same time, this sequence number acknowledges all previously received frames with number N(R) -1.

HDLC provides different connection types:

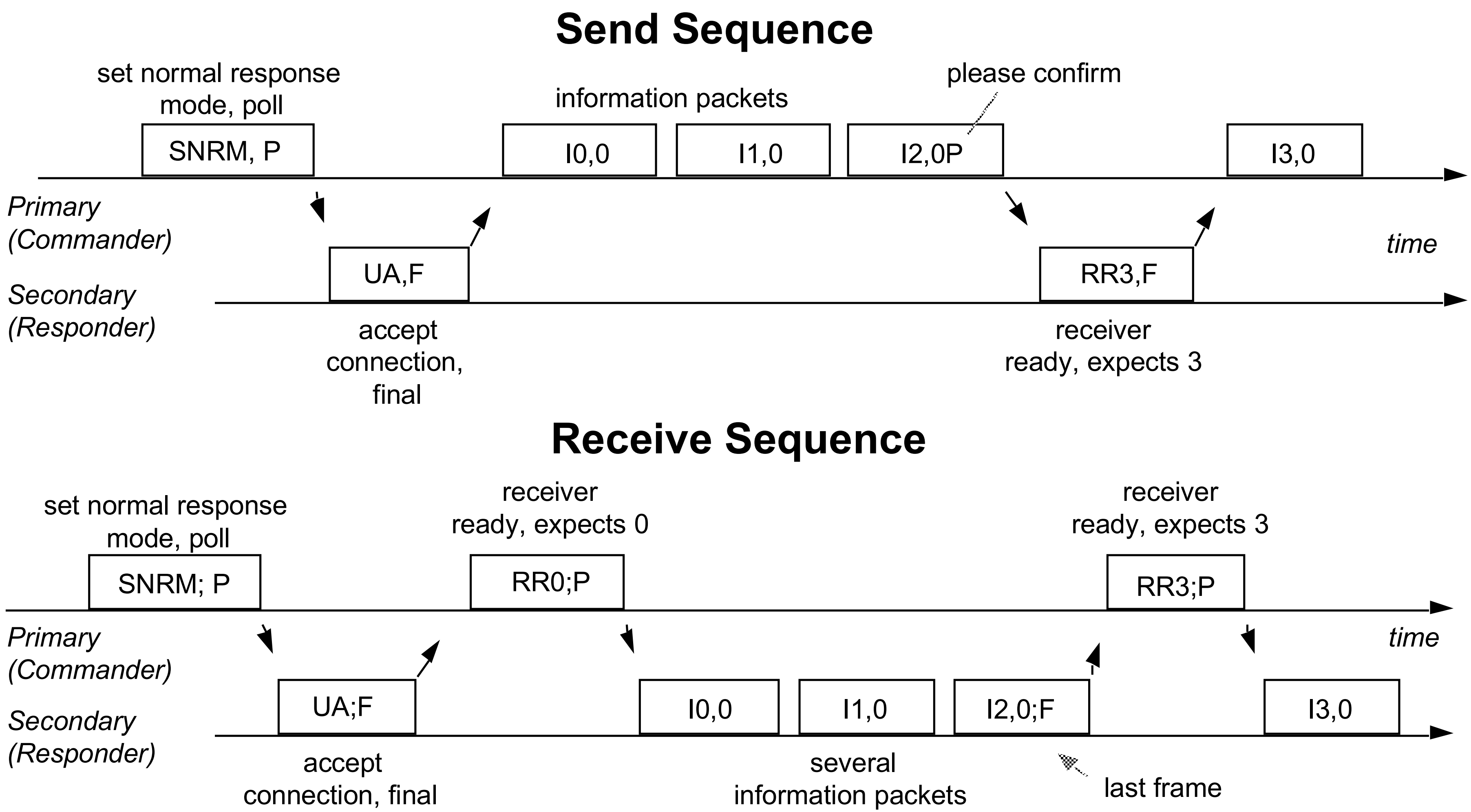
LAP (link access procedure): assymetric Primary/Secondary;

NRM (normal response mode): only one station as primary;

ARM (asynchronous response Mode): spontaneous transmission of secondary;

LAPB (LAP-balanced): every station can become primary and start transmitting (if medium access allows).

HDLC Exchange (NMR in ISO 4335)



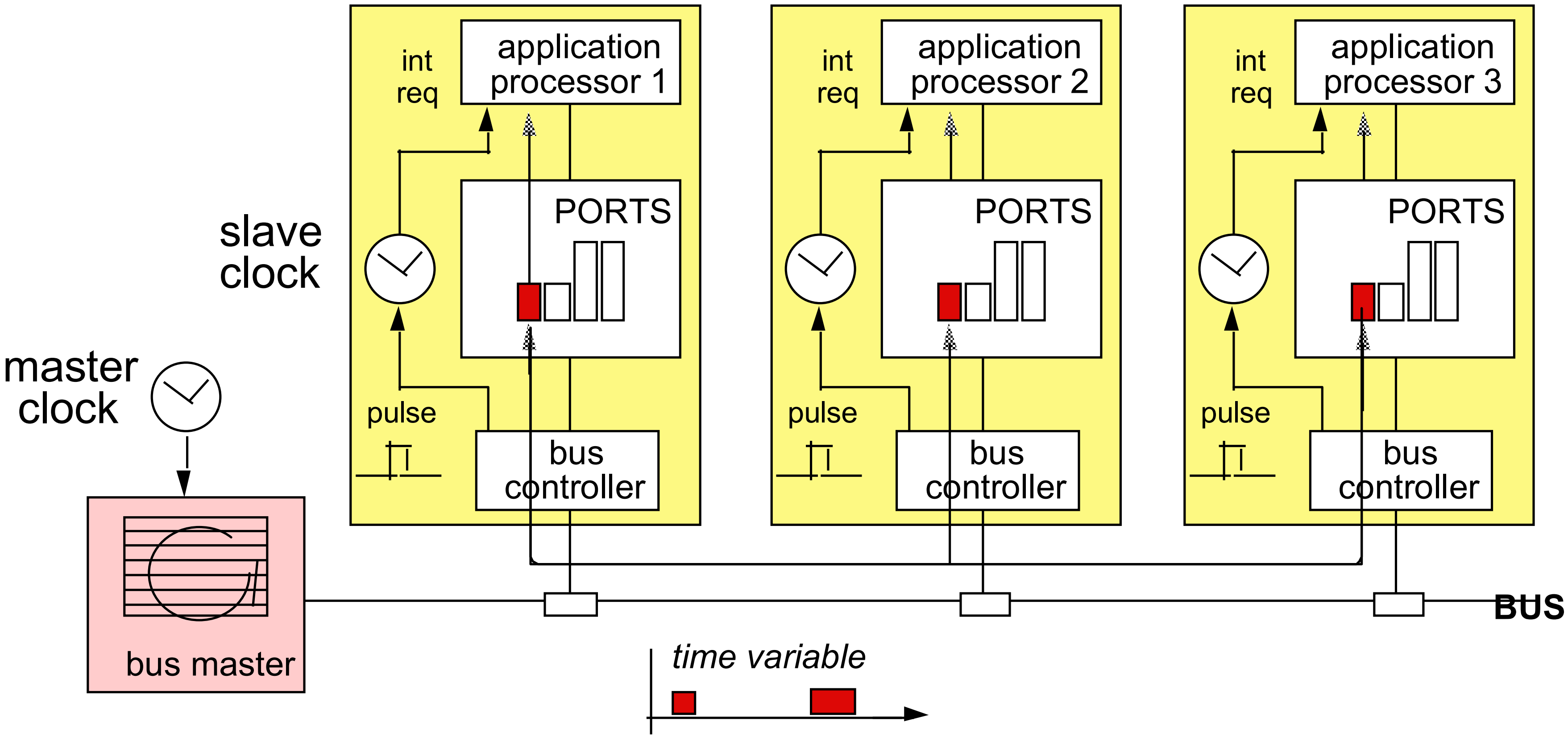
The data transmission takes place under control of the Primary. Therefore, both "Send Frame" and "Receive Frame" are supported

Clocks

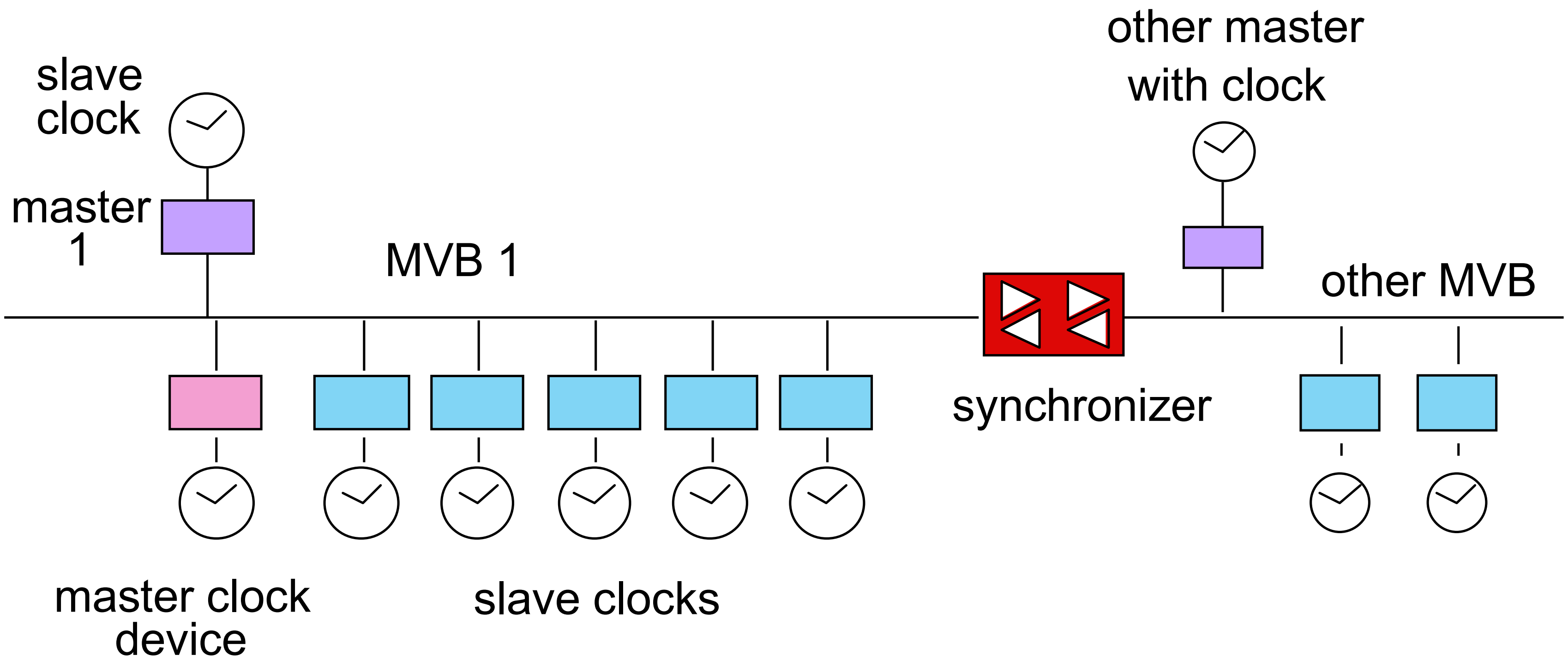
In a fieldbus, devices must be synchronized to a common clock to time-stamp their transmissions.

Time Distribution in a single master system

At fixed intervals, the Master broadcasts the exact time as a periodic variable. When receiving this variable, the bus controllers generate a pulse which can resynchronize a slave clock or generate an interrupt request.



Clock compensation for transmission delays



The clock does not need to be generated by the Master, but the master must poll the clock

The clock can synchronize sampling within a few μs across several bus segments.

IEEE 1588 PTP Clock Synchronization

IEEE 1588 defines the Precision Time Protocol, a clock synchronization that assumes that two consecutive frames have the same delay, but the moment of sending suffers jitter.

The clock device (possibly coupled to a radio signal) sends the first frame with an coarse time stamp, but registers in hardware the exact moment when the frame was sent.

It then sends a second frame with the exact time at which the first frame was sent.

Bridges and switches are responsible to compensate for their internal delays and send a corrected time frame.

High precision clock synchronization

In some application, even the PTP protocol is insufficient.

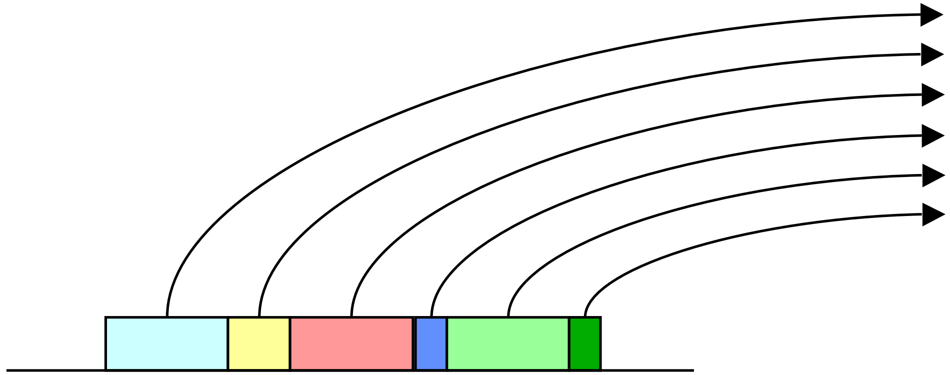
In this case, either the clock is distributed by a separate, dedicated medium
(as in railways signalling and electrical substations.

Alternatively, all devices receive a radio signal from GPS to recalibrate their internal clocks.

Assessment

What is the purpose of the link layer ?
Which is the role of the three sublayers in the link layer ?
What is the Hamming Distance ?
What is the Residual Error Rate ?
What is the code efficiency ?
Where are error-correcting codes used ?
What is the formal of an HDLC frame ?
What is the purpose of medium access control ?
Which medium access does not require an arbitration ?
Which kinds of arbitration exist ?
How does the CAN arbitration works and what is its assumption on the medium ?
How does the Ethernet arbitration works ?
What is the influence of collision detection in a LON arbitration ?
Which medium access are deterministic ?
What is the difference between connection oriented and connectionless transmission ?
How are error corrected by the logical link control in cyclic transmission ?
How are error corrected by the logical link control in event-driven transmission ?
How does a sliding window protocol works ?
How does a transmission in HDLC work ?
How are clocks synchronized ?

Industrial Automation
Automation Industrielle
Industrielle Automation



```
PersonnelRecord ::= [APPLICATION 0] IMPLICIT SET {
     name     Name,
     title    [0] VisibleString,
     number      EmployeeNumber,
     dateOfHire [1] Date,
     nameOfSpouse  [2] Name,
     children    [3] IMPLICIT
        SEQUENCE OF ChildInformation
     DEFAULT {}
```

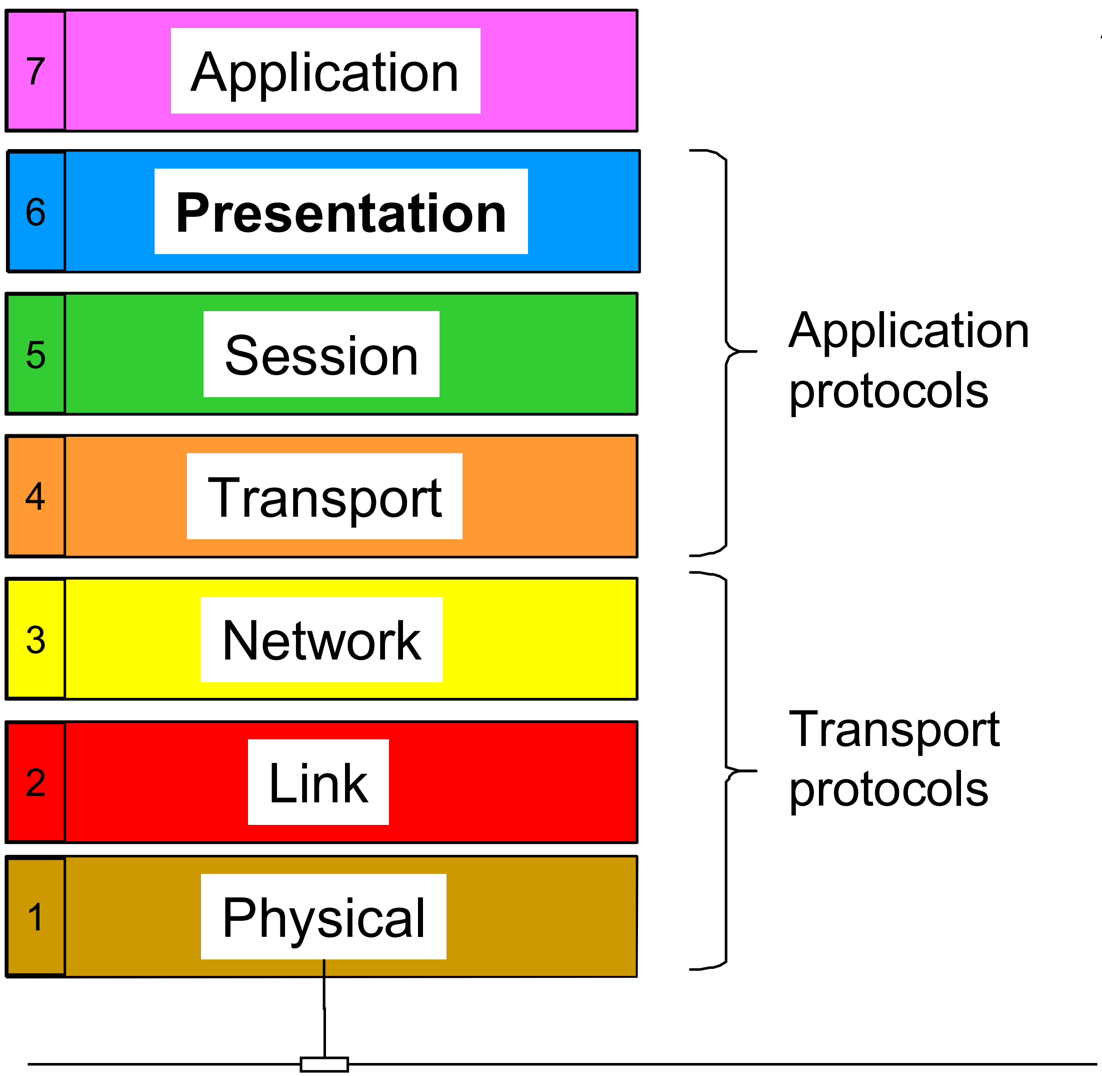
3.3.4

OSI Upper Layers - Presentation Layer, ASN.1 and data types
Niveaux supérieurs OSI - couche de présentation, ASN.1 et types de données
Obere OSI-Schichten – Darstellungsschicht, ASN.1 und Datentypen

Prof. Dr. Hubert Kirrmann
ABB Ltd, Baden, Switzerland

2005, May, HK

Presentation layer in the OSI-Model (ISO/IEC standard 7498)



All services directly called by the end user (Mail, File Transfer,...)

Definition and conversion of the data formats (e.g. ASN 1)

Management of connections (e.g. ISO 8326)

End-to-end flow control and error recovery (e.g. TP4, TCP)

Routing, possibly segmenting (e.g. IP, X25)

Error detection, Flow control and error recovery, medium access (e.g. 802.2, HDLC)

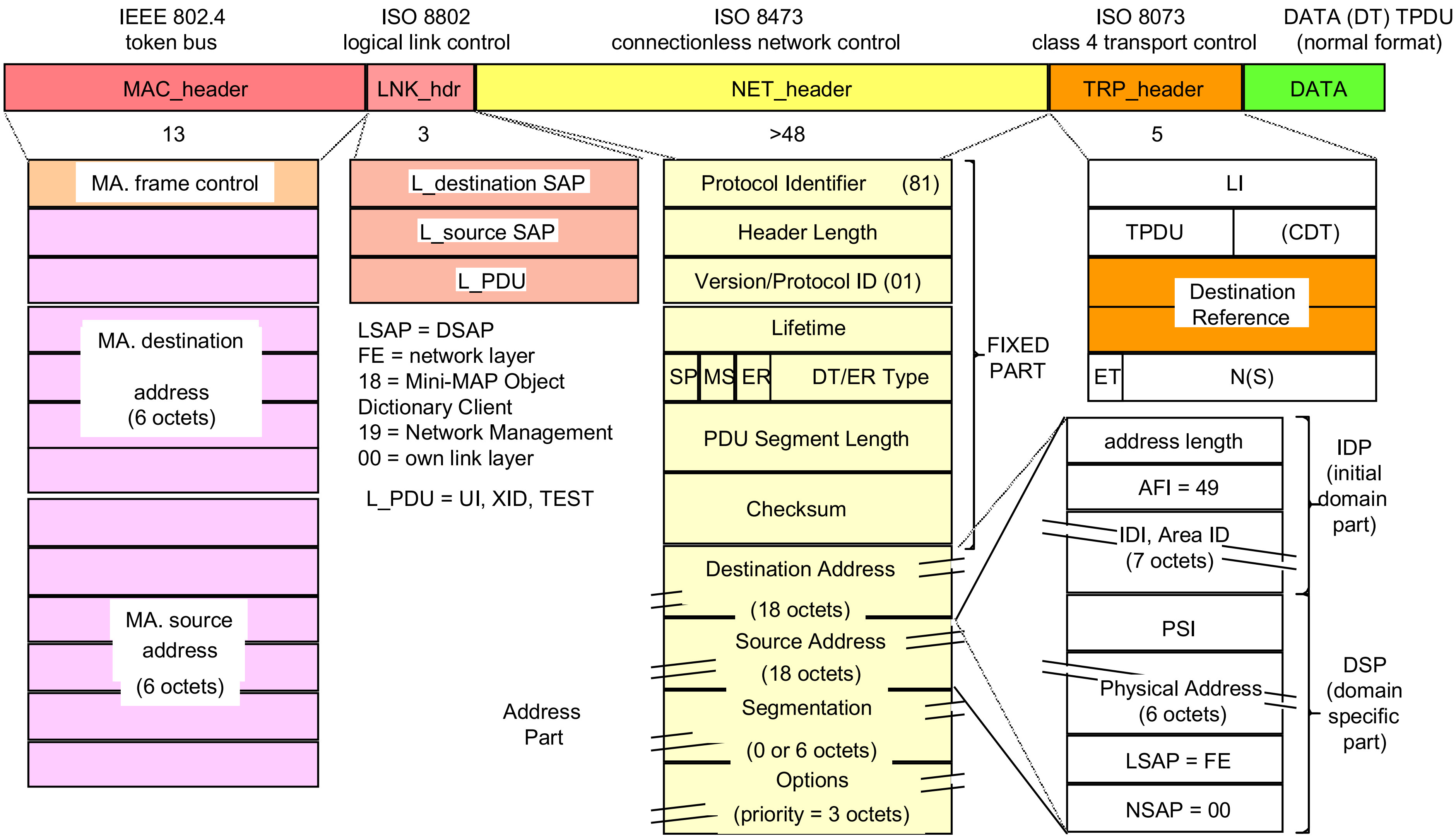
Coding, Modulation, Electrical and mechanical coupling (e.g. Ethernet)

Presentation Layer

The presentation layer is responsible that all communication partners agree on the format of the data

Transfer Syntax: describe what is in a frame ?

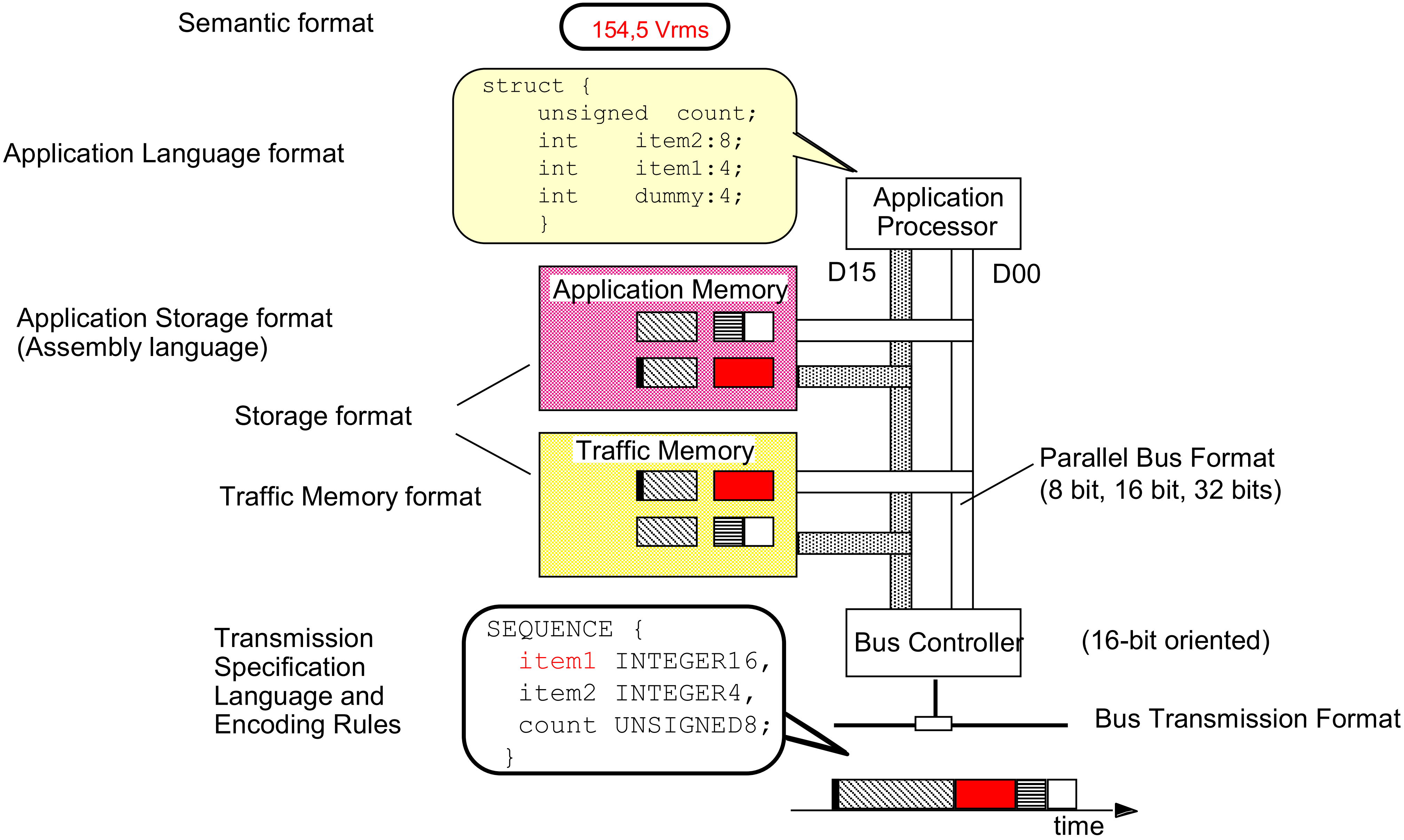
Role: how to define formally the format and meaning of the transmitted data



ASN.1 justification

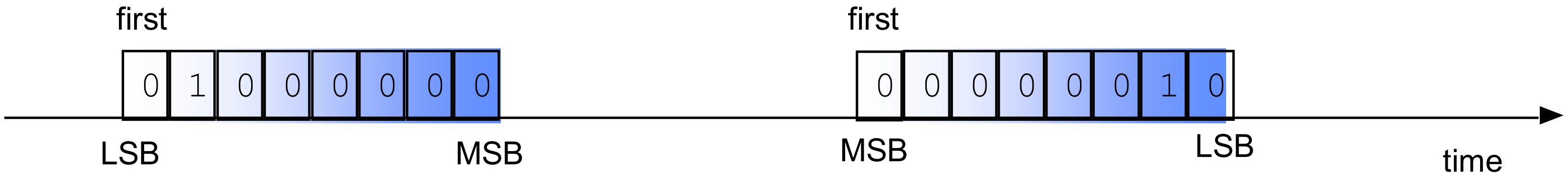
Why do we need ASN.1 (or a similar notation)

Semantic Levels of Data Representation



Bit Transmission Order

Most data links are byte-oriented (transmit 8 bit as an indivisible unit)
The order of bit transmission within a byte (octet) is dependent on the link.
It does not matter as long as all bus participants use the same scheme

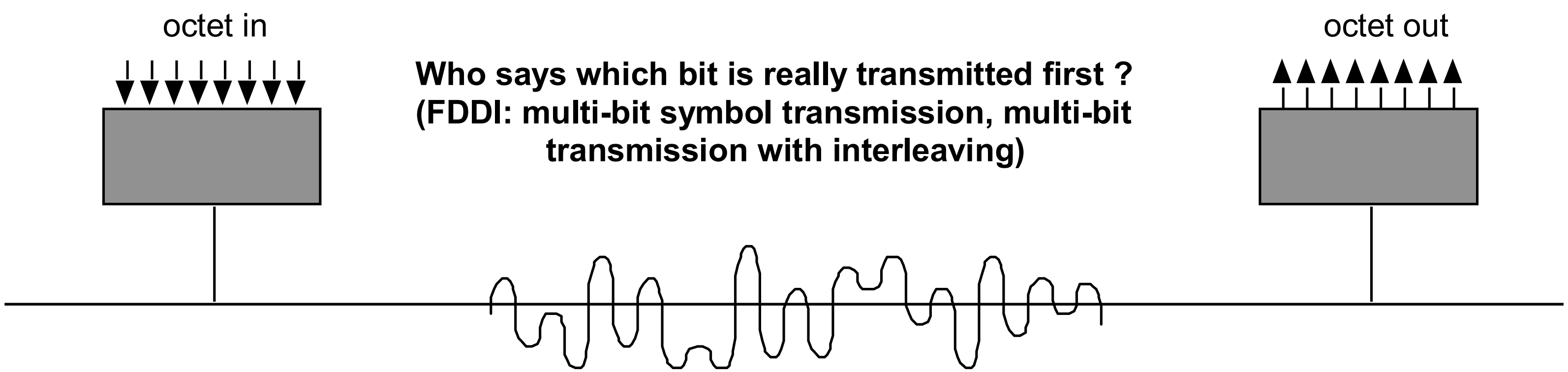


UART, HDLC, Ethernet

Legacy of the old telex

FDDI, FIP, CAN, MVB

There is no relation between the bit and the byte ordering scheme



Convention: only consider octet streams

Integer representation in memory: Big-Endian vs Little Endian

Memory contents

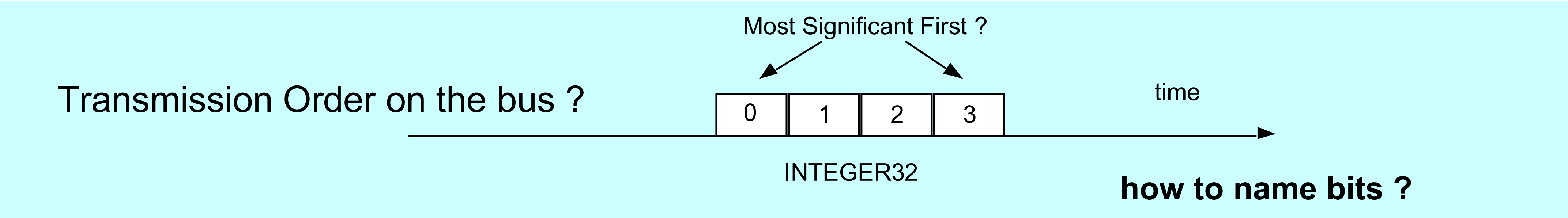
address	7 (MSB)	6	5	4	3	2	1	0 (LSB)
x0000	0	0	0	0	0	0	1	0
x0001	0	0	0	0	0	0	0	1
x0002	0	0	0	0	0	0	0	0
x0003	0	0	0	0	0	0	0	1
x0004	0	0	0	0	0	0	0	0
x0005	0	0	0	0	0	0	0	0
x0006	0	0	0	0	0	0	0	0

	LE	BE
	Intel DEC	Motorola IBM, TCP/IP, Unix, RISC
INTEGER8 (x0000)	= 2	= 2
INTEGER16 (x0001–x0002)	= 1	= 256
INTEGER32 (x0003–x0006)	= 1	= 16777216

Processor	B7 B6 B5 B4 B3 B2 B1 B0	(bit name)
TCP/IP	0 1 2 3 4 5 6 7	(bit offset)
Profibus	1 2 3 4 5 6 7 8	(bit position)

three different naming schemes !

How are sequence of octets transmitted ?



The standard in network protocols is always:
Most Significant Octet first (Big-Endian)

So, how to respect network byte ordering ?

Since network protocols require a “most significant octet first” transmission (NBO =Network Byte Ordering), all little-endian processors must convert their data before putting them into the transmission buffer, pipe or socket, by calling the Unix functions:

Function Name	Description
htons	Host order to network order, short (2 bytes, 16 bits)
htonl	Host order to network order, long (4 bytes, 32 bits)
ntohs	Network order to host order, short (2 bytes, 16 bits)
ntohl	Network order to host order, long (4 bytes, 32 bits)

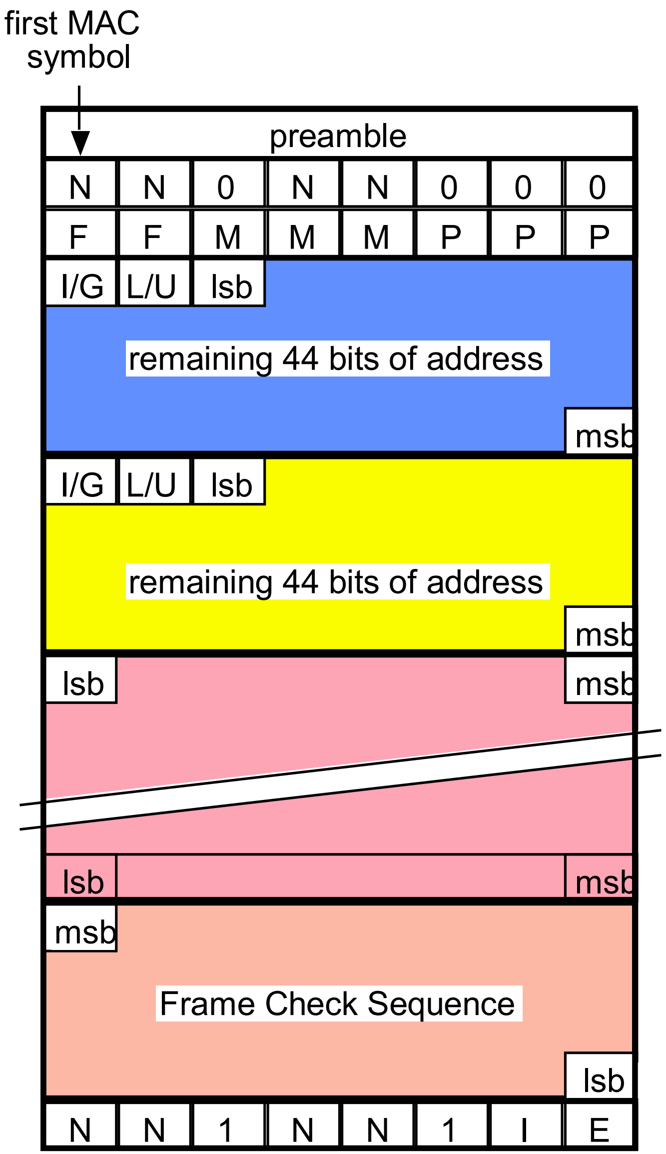
All data "seen" by the sockets layer and passed on to the network must be in network order

```
struct sockaddr_in s;

/* WRONG */

s.sin_port = 23;

/* RIGHT */

s.sin_port = htons(23);
```

- Extremely error prone

Inconsistency: Token-bus frame (IEEE 802.4)

read this picture from left to right,
then top to bottom



Destination Address

Addresses are transmitted **least** significant bit first

Source Address

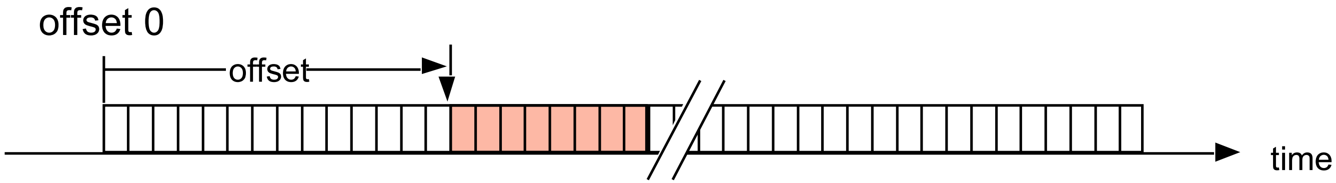
first octet

Data are transmitted **least** significant bit first within an octet, but **most** significant octet first within a word (imposed by higher layers)

last data octet

Checksum is transmitted **most** significant bit first

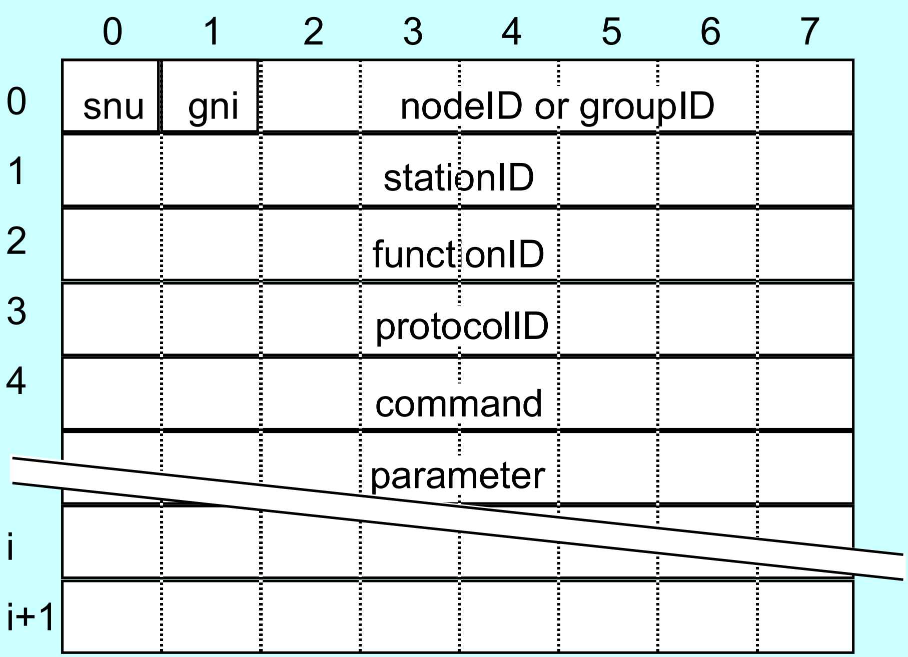
How to specify transmitted data



first: agree on a bit and byte ordering scheme: {0,0} scheme recommended.

second: describe the data stream formally (machine-readable)

intuitive



formal

```
CommandPDU :== SEQUENCE
 {
  BOOLEAN1     snu -- system
  BOOLEAN1     gni -- group
  CHOICE (gni)
   {
    Group      : ENUM6 groupID;
    Individual: ENUM6 nodeID;
   }
  ENUM8        stationID;
  ...
```

Can a “C”-declaration serve as an encoding syntax ?

```
typedef struct {
  char location[ LOCATION_LEN ];
  unsigned long object_id;
  alarm_type_t alarm_type;
  priority_level_t priority_level;
  unsigned long index_to_SNVT;
  unsigned value[ 4 ];
  unsigned long year;
  unsigned short month;
  unsigned short day;
  unsigned short hour;
  unsigned short minute;
  unsigned short second;
  unsigned long millisecond;
  unsigned alarm_limit[ 4 ];
  } SNVT_alarm;
```

how is size given?

allowed values in enum ?

to which structure does it point?

is "short" a byte ?

is “unsigned” 16 bits or 32 bits ?

Such a machine-dependent syntax is only valid if all applications use the same syntax on the same machine with the same compiler, it is **not** suited to describe the bus traffic.

Abstract Syntax Notation Number 1 (ASN.1)

- The IEC/ISO define a standard notation in IEC 8824 (ASN.1), allowing to define simple types (primitive) and constructed types.
- Data structures can take forms not usually found in programming languages.
- Each data structure is identified during transmission by a *tag*.
- In principle, ASN.1 only defines data structures to be transmitted, but not how they are encoded for transmission.
- One possible coding of the primitive and constructed data types is defined in ISO/IEC 8825 as "Basic Encoding Rules“ (BER) defined in ISO 8825.
- More efficient encodings (PER,…) also exist.
- ASN.1 can be used for defining memory contents, file contents or communication data, and in general any exchanged information.
- ASN.1 has the same role as XML, but it is far more efficient.

ASN.1 Syntax Example

Informal

```
Name:                John P Smith
Title:               Director
Employee Number:     51
Date of Hire:        17 September 1971
Name of Spouse:      Mary T Smith
Number of            2
Children:

Child Information    Ralph T Smith
  Name:              11 November 1957
  Date of Birth

Child Information    Susan B Jones
  Name:              17 July 1959
  Date of Birth
```

ASN.1

```
PersonRecord ::= [APPLICATION 0] SEQUENCE {
  name               Name
  title              [0] VisibleString,
  number             EmployeeNumber,
  dateOfHire         [1] Date,
  nameOfSpouse       [2] Name,
  children           [3] SEQUENCE OF
                     Childinformation   DEFAULT {} }

ChildInformation ::= SEQUENCE {
   name                  Name,
   dateOfBirth      [0]  Date}


Name ::= [APPLICATION 1] SEQUENCE {
  givenName  VisibleString,
  initial    VisibleString,
  familyName VisibleString}

EmployeeNumber ::= [APPLICATION 2] INTEGER

Date ::= [APPLICATION 3] VisibleString -- YYYYMMDD
```

Abstract syntax and transfer syntax

An abstract syntax describes the elements of information without considering their encoding (i.e. how they are represented in memory or on a bus)

E.g. A transfer syntax defines the name of the structures and elements, their value range [e.g. 0..15], ….

ASN.1 is defined in the standard ISO 8824-1.

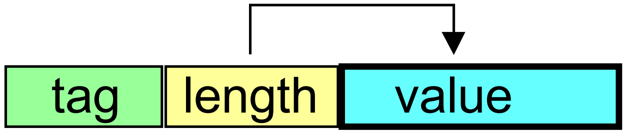
A transfer syntax describes how the structures and elements are effectively encoded for storing and transmission, so that the receiver can fully decode the transmitted
information. At transmission time, only the transfer syntax is visible, it cannot be interpreted
without knowing the abstract syntax.

E.g. A transfer syntax defines that an array of 33 Unicode characters is transmitted. The receiver knows from the abstract syntax that this is a person's name.

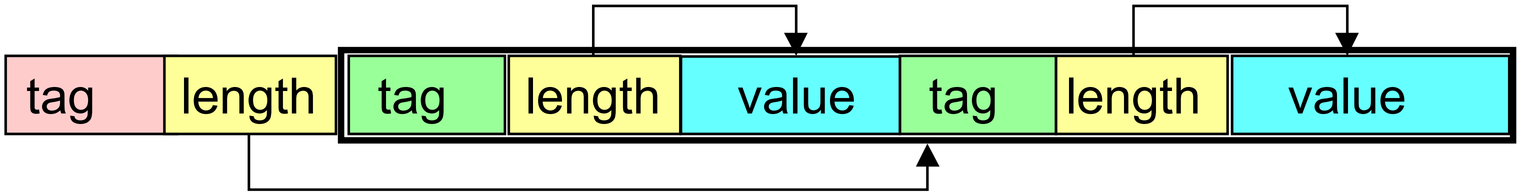
The basic encoding rules for ASN.1 are defined in the standard ISO 8825-1.

ASN.1 encoding: TLV

ASN.1 does not say how data are stored nor transmitted, but it assumes that the transmission format consists for each item of: a *tag*, a *length* and a *value* (TLV)



The value may be itself a structured object:



(recursive)

the tag specifies the data type of the value that follows, implicitly or explicitly.

ASN.1 Data Types

four Tag Types

UNIVERSAL
APPLICATION
CONTEXT_SPECIFIC
PRIVATE

Basic Types

BOOLEAN
INTEGER
BITSTRING
OCTETSTRING
NULL
OBJECT_ID
OBJECT_DESC
EXTERNAL
REAL
ENUMERATED
ANY

Constructed Types

SEQUENCE	ordered sequence of types (**record**)
SEQUENCE OF	ordered sequence of same type (**array**)
CHOICE	one of an unordered, fixed sequence of different types.
SET	unused
SET OF	unused

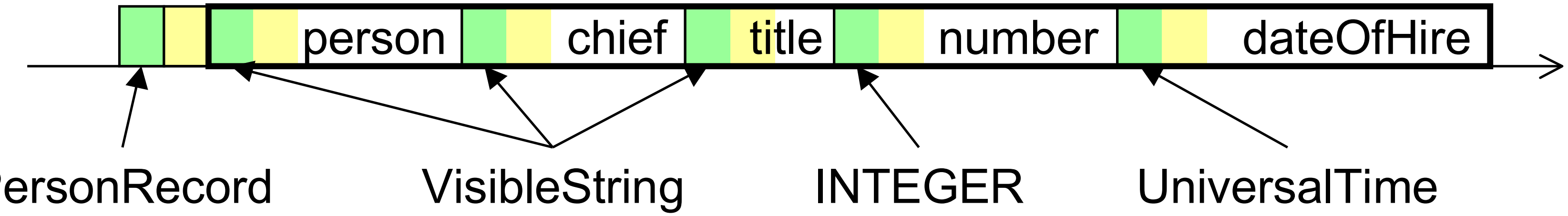
Open arrays (dynamic size) and pointers are not allowed

ASN.1: The Type SEQUENCE

An ASN.1 SEQUENCE is similar to a “C”-struct:

Example:
```
PersonRecord ::= SEQUENCE {
          person      VisibleString
          chief       VisibleString
          title       VisibleString,
          number      INTEGER,
          dateOfHire  UniversalTime }
```

This notation assumes that all elements in the sequence are present and are transmitted in the specified order.



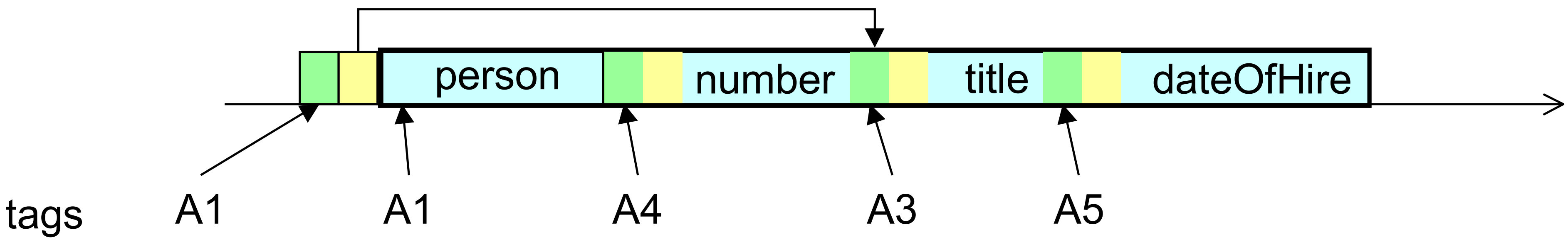
Tagging

When elements of a sequence may be missing, it is necessary to *tag* the items, i.e. identify the items by an integer.

Of course, if all types would be different, it would be sufficient to specify the type, but this practice (called EXPLICIT tagging) should not be followed.

Structured types need in any case a tag, otherwise they cannot be distinguished from another structured type (note how “PersonRecord” is identified)

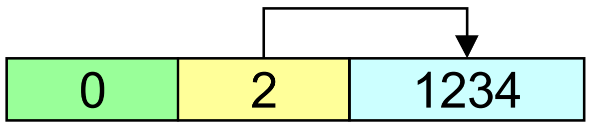
```
PersonRecord ::= [APPLICATION 1] SEQUENCE {
                    person        [1] VisibleString,
                    chief         [2] VisibleString,
                    title         [3] VisibleString,
                    number        [4] INTEGER,
                    dateOfHire    [5] UniversalTime }
```



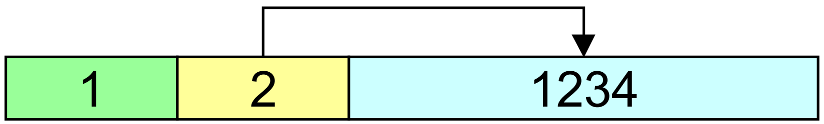
ASN.1: The CHOICE type

A choice selects exactly one alternative of several.
There is normally a distinct tag for each choice, but the type can also be used as tag, as long as all types are distinct:

```
Quantity ::= CHOICE {
                  [0] units                INTEGER16,
                  [1] millimetres          INTEGER32,
                  [2] kilograms  FLOAT
                  }
```



quantity in units



quantity in millimeters

kilograms, IEEE format

Difference between ASN.1 and "C"* or XML/XDR

ASN.1 is a format for data exchange, "C" is a compiler-dependent format for storage in RAM.

The basic data types are defined differently.

ASN.1 types may have a variable size (INTEGER may be 8, 16, 32, 64 bits).

ASN.1 SEQUENCE differs from a "C" struct since not all elements must be transmitted, nor is their order to be maintained (if tagging is used).

ASN.1 CHOICE differs from a "C" union since the length depends on the chosen item, and the contents differ.

ASN.1 has the same role as XML. XML is however both an abstract and a transfer syntax, it is sent in clear text.

Unix systems use XDR (Sun's external data representation) that is 32-bit oriented, not efficient for small data items, also a mixture of abstract and transfer syntax.

Explicit and implicit tagging

ASN.1 can assign automatically a tag to elements of a sequence.

This practice is dangerous, because another encoding (PER) could use other tag numbers.

It is preferable to make tags explicit everywhere that is needed.

Transfer Syntax: Basic Encoding Rules (BER)

A companion to ASN1, BER (ISO 8825-1) defines encoding rules for ASN.1 data types

BER tags all data, either implicitly or explicitly

Type, Tag and size are transmitted before every value or structured data

Example: type = UNIVERSAL Integer (tag included in basic types)

length = 2 octets

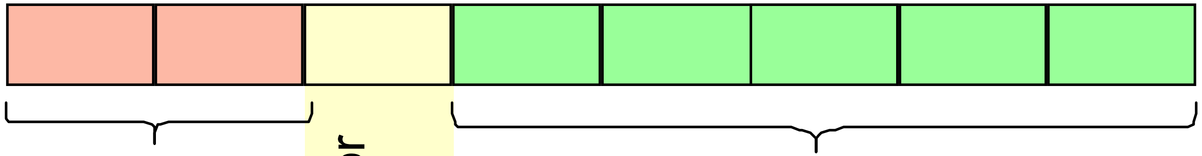
value = 1234

		0 2	02	12	34		

BER supports all ASN.1 structures.

Exception: if size is 0, there is no value field (== NULL)

BER – Tag/Type field



00 = UNIVERSAL
01 = APPLICATION
10 = CONTEXT_SPECIFIC
11 = PRIVATE

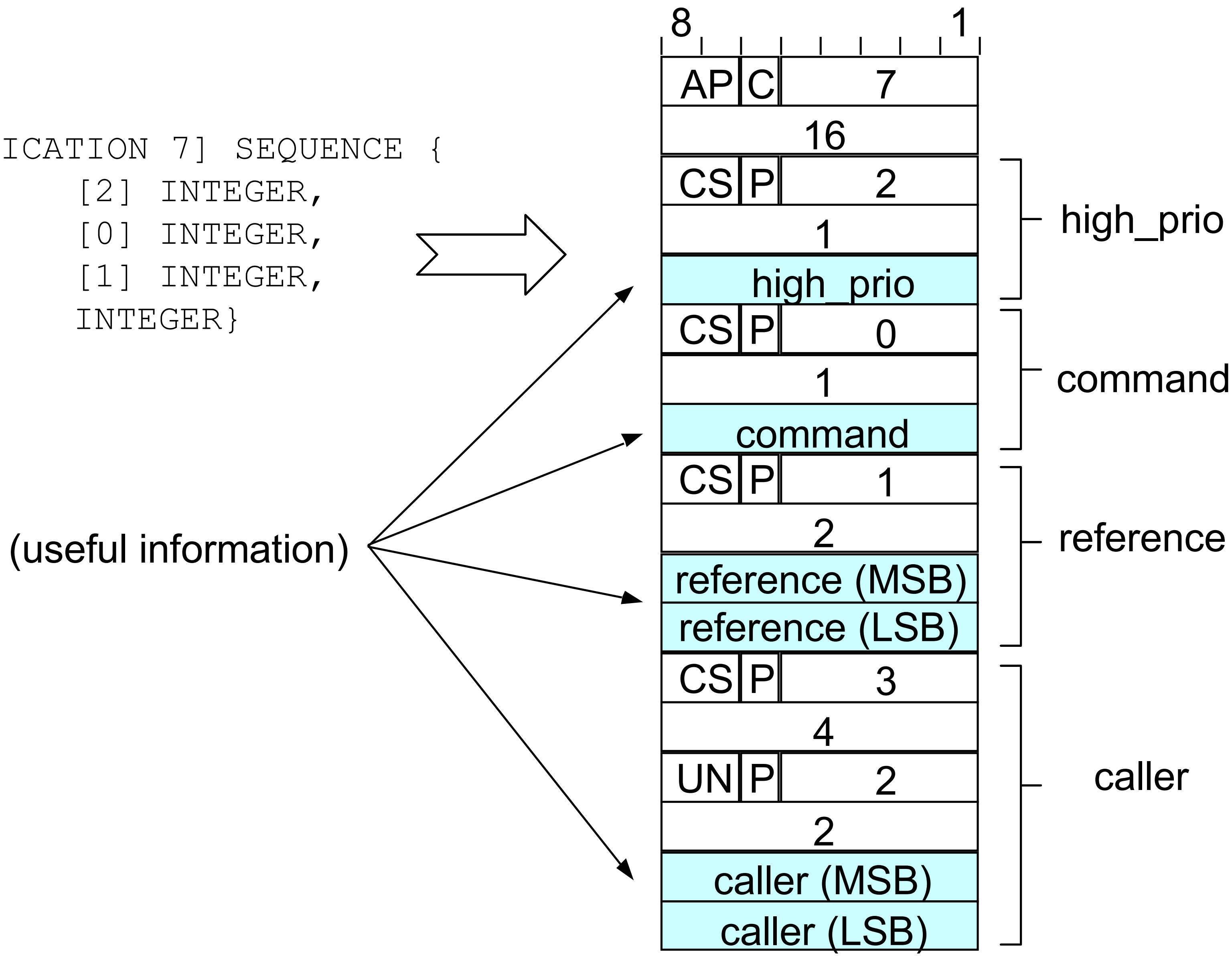
P: after the size comes a value -
C: after the size comes a tag

(for universal class only):
00 = null (size = 0, no value)
01 = Boolean type
02 = Integer type
03 = Bitstring type
04 = Octetstring type
05 = Null type
06 = Object Identifier type
07 = Object Descriptor type
16 = Sequence and Sequence_Of types
17 = Set and Set_Of types
18-22 = Character strings (numeric, printable, …)
23-24 = Time types
25 = Graphic string
26 = VisibleString (ISO646)
> 28 = reserve and escape: use a second octet.

Example: 00000010 = UNIVERSAL INTEGER
10100001 = [CONTEXT_SPECIFIC 1] SEQUENCE

Example ASN.1 and BER

```
CallerRef :== [APPLICATION 7] SEQUENCE {
        priority        [2] INTEGER,
        command         [0] INTEGER,
        reference       [1] INTEGER,
        caller          INTEGER}
```



Examples

Tag/Type:

•	1000’0000	Context Specific, not constructed, implicit, tag = [0]
•	0110’0001	Application Specific, constructed, implicit, tag = [1]
AB	1010’1101	Context Specific, constructed, implicit, tag = [11]
01	0000’0001	Basic Type, not constructed, boolean

decoding the first digit:

0,1:	Universal, not constructed	
2,3:	Universal, constructed	
4,5:	Application Specific, not constructed	[APPLICATION 1]
6,7:	Application Specific, constructed	
8,9:	Context Specific, not constructed	[6]
A,B:	Context Specific, constructed	

constructed means: the next octet is not a value, but a type/tag !

Examples

A0 0E 02 01 0A A1 09 A0 03 80 01 00 A1 02 80 00

null

green: tag / type
red: size
black: value

A1 67 02 01 0A A1 62 A0 5D 1A 0B 54 65 6D 70 65
72 61 74 75 72 65 1A 0C 54 65 6D 70 65 72 61 74
75 72 65 31 1A 07 61 72 72 61 79 5F 35 1A 04 62
6F 6F 6C 1A 0F 66 65 65 64 65 72 31 5F 33 5F 70
68 61 73 65 1A 05 66 6C 6F 61 74 1A 0F 68 65 72
62 73 5F 74 65 73 74 5F 74 79 70 65 1A 08 75 6E
73 69 67 6E 65 64 81 01 00
--
A0 18 02 01 0B A6 13 A0 11 80 0F 66 65 65 64 65
72 31 5F 33 5F 70 68 61 73 65
A1 34 02 01 0B A6 2F 80 01 00 A1 16 81 14 66 65
65 64 65 72 31 5F 33 5F 70 68 61 73 65 24 41 64
64 72 A2 12 A2 10 A1 0E 30 05 A1 03 85 01 10 30
05 A1 03 85 01 10
--
A0 1E 02 01 0C A4 19 A1 17

Beyond ASN.1 and BER

ASN.1 / BER could not impose themselves in field busses because of the high overhead involved (32 bits for a single boolean !)

ISO / UIT developed more efficient encodings, such as

ISO/IEC 8825-2: Packed Encoding Rules (PER), that
exists in two versions: aligned (on an 8-bit boundary) or not aligned (bit stream)

In low speed busses such as fieldbus, this is still too much overhead.

IEC 61158-6 (Fieldbus) offers 3 encodings:

Traditional Encoding Rules (Profibus)
Compact Encoding Rules (for FAL)
Buffer Encoding Rules (FIP)

100 MBit/s Ethernet has sufficient bandwidth, but the burden is shifted to the processors

(Data compression gives variables length messages, costs a lot in compression & decompression)

ROSIN Compact – Retrofit Encoding

The railways operators needed to define formally the exchange rules for data of already existing devices, of different manufacturers and vintage.

Therefore, a notation was developed (ROSIN notation) to describe any data transfer, on a bit rather than a byte-orientation. It also allows to cope with alignment (data should be transmitted at an offset that is a multiple of their size to reduce processor load)

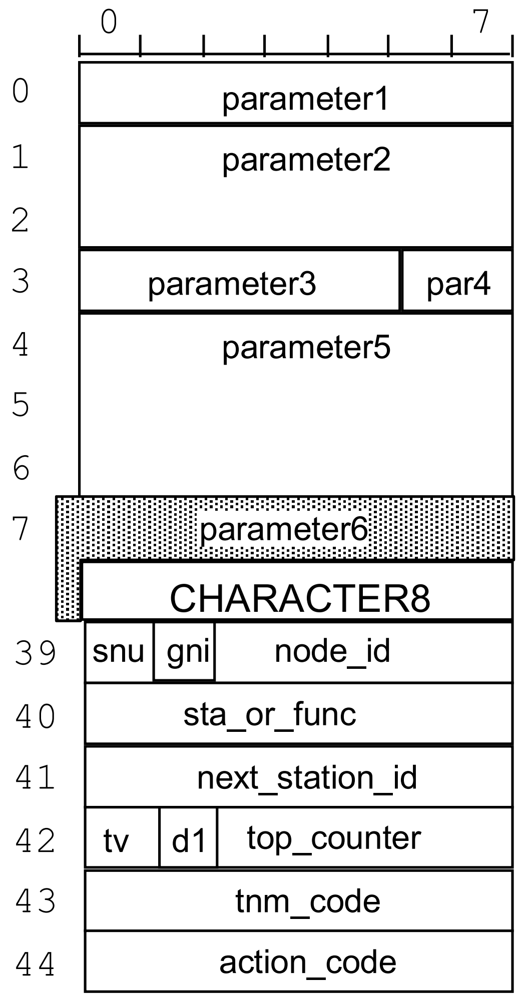
Indeed, since these devices already communicate using a proprietary protocol, transmission must already be unambiguous.

Each data type is specified, e.g. Integer32 differs from Integer32_LE (Little Endian)

The ROSIN notation uses the ASN.1 meta-syntax, but it does not imply a TLV scheme.
- the length can be deduced from the type or position,
- typing information is inserted explicitly when:
  - a choice exists among several alternative types (e.g. depends on success/failure)
  - types have a (large) variable size (e.g. text strings, files)
  - sequences have optional fields and out-of-sequence fields (occurs too often).

ROSIN - Retrofit encoding rules example

```
Type_InfoMessage ::= RECORD {
  parameter1         INTEGER8       -- octet.
  parameter2         INTEGER16,     -- 16-bit word, MSB first
  parameter3         UNSIGNED6,     -- 6- bit value
  par4               ANTIVALENT2,   -- 2 bits for par4, e.g. check variable.
  parameter5         Parameter5,    -- parameter5 has a structured type
  parameter6         STRING32,      -- an array of up to 32 8-bit characters.
                                    -- trailed if shorter with "0" characters.
  snu                ENUM1 {
    USER    (0),                    -- 0 = user
    SYSTEM (1)                      -- 1 = system
    },
  gni                ENUM1 {        -- could also be expressed as BOOLEAN1
    INDIV (0),                      -- individual function addressing
    GROUP (1),                      -- group addressing.
    },
  node_id            UNSIGNED6,     -- 6-bit unsigned integer
  sta_or_func        ONE_OF [snu] { -- meaning depends on 'snu'
    function [USER] UNSIGNED8,       -- if snu = user, function identifier
    station  [SYSTEM] UNSIGNED8      -- if snu = system, station identifier
    },
  next_station_id    UNSIGNED8,      -- next station or 'FF'H if unknown
  tv                 BOOLEAN1,       -- TRUE if 1
  res1               BOOLEAN1 (=0),  -- 0 (place holder)
  topo_counter       UNSIGNED6,      -- 6-bit unsigned integer
  tnm_code           ENUM8 {         -- has only two defined values
    FIRSTCASE  ('1E'H)               -- one of two defined values
    SECONDCASE ('84'H)               -- the other defined value
    },
  action_code        Action_Code,    -- this type is used several times
}
```



Comparing Coding Efficiency

	(ISO)	(SUN)	(UIT)		(Profibus)	(FIP)	
	BER	XDR	A-PER	U-PER	FER	BuER	ROSIN
Boolean	24	16	5	5	16	8	1
Integer8	24	16	16	12	16	16	8
Unsigned8	24	16	11	11	16	16	8
Integer16	32	24	24	20	24	24	16
Unsigned 16	32	24	24	19	24	24	16
Integer32	48	40	24..48	36	40	40	32
Unsigned32	48	40	24..48	35	40	40	32
String [32]	272	272	272	272	272	272	256

encoding/decoding highly packed data may cost more than is won by shorter transmission

Engineering Units

Many process variables represent analog, physical values of the plant.

Data presentation (e.g. integer) is insufficient to express the meaning of the variable.

Therefore, it is necessary to allocate to each variable a data type in engineering units.

> **"A unit of measure for use by operating/maintenance personnel usually provided by scaling the input quantity for display (meter, stripchart or CRT)"**
>
> IEEE

Scaling (determined the possible range of a variable) is necessary for analog displays.

It requires the definition of the possible range of values that the variable may take.

SI Units

All physical variables should be restricted to SI Units (NIST 330-1991, IEEE 268A-1974) or refered directly to them

For instance:

- speeds shall be expressed in meter/second, not in km/h or in miles/hour
- angles shall be represented in radian rather than degree or grad.

variable	unit	variable	unit
position, distance	m	power	W
mass	kg	energy	J
time	s	pressure	Pa
current	A	volume	m^3
angle	rad	flow	m^3/s
force	N	mass flow	kg/s
torque	Nm	tension	V
frequency	s-1	reactive power	var
angular velocity	rad/s	impedance	Ω
angular acceleration	rad s^{-2}	temperature	K

Why floating point ?

Floating point format (IEEE Std 254)

- require twice the place (32 bits vs 16 bits),
- adds about 50% to traffic (analog values are only 10% of total),
- cost more to process (floating point unit)

but

- removes all ambiguities, rounding errors, overflow and underflow.

Floating point format is the only safe representation of a physical variable.

Internally, devices can use other formats.

Therefore, devices shall indicate their exported and imported variables as REAL32.

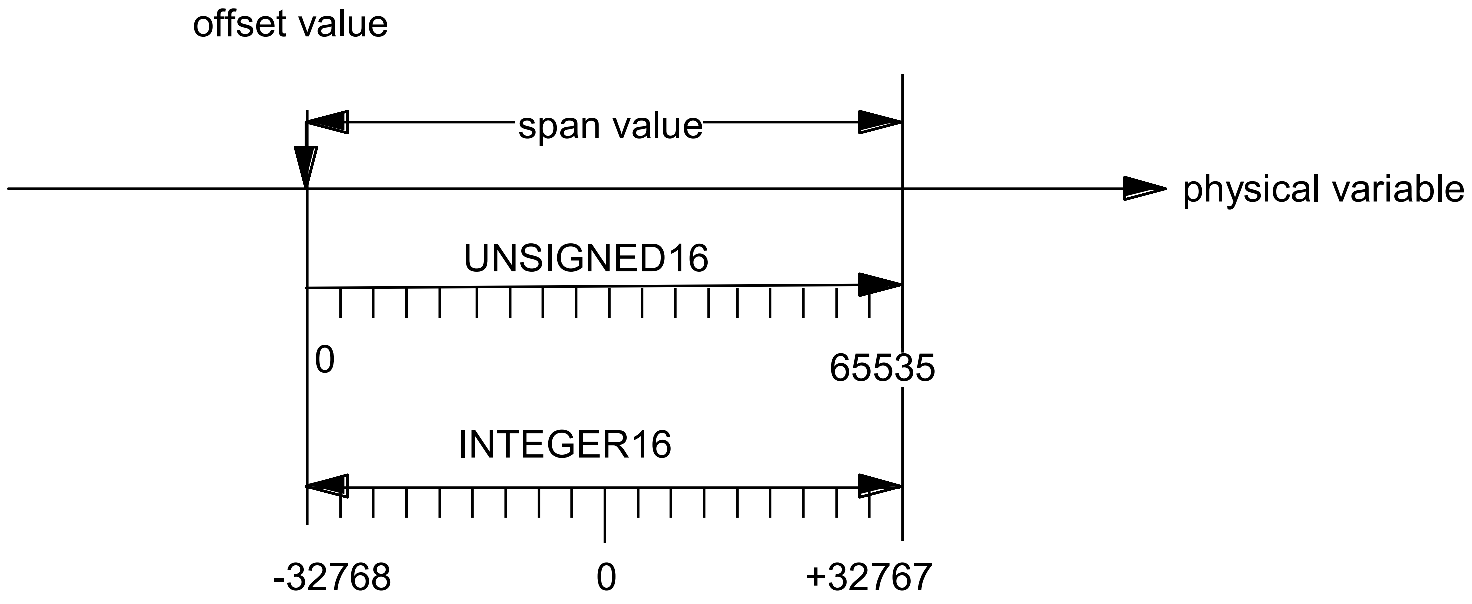
Exception:

Variables, whose precision do not depend on their absolute value:
time, elapsed distance (odometer), energy, money, countable objects

In special applications (e.g. GPS data), an ASCII representation may be more appropriate, albeit not efficient. In this case, data can also be processed as BCD (as in pocket calculators)

Fractionals

A device can indicate the format of a fractional analog variable by specifying (as a REAL32) the span and the offset to be applied to the base unit:



e.g. offset = 0,0 m/s, span = 100,0 m/s, base unit = UNSIGNED16
means: 0 = 0,0 m/s, 65536 == 100,0 m/s

e.g. offset = - 32,768 V, span = 65,536 V, base unit = INTEGER16
means: 0 = -32,768 m/s, 65536 == 32,767 V

Scaled Variables

Process Variables are often transmitted and processed as fractionals:

Fractional format expresses analog values as integer multiples of the resolution

e.g.:

distance = 0..65535 x 0.1 m (resolution)	--> 0 .. 65535	(UNSIGNED16)
distance = -32768..+32767 x 0.1 m	--> -32768 .. 32768	(INTEGER16)
speed = 0.. 6553,5 m/s, resolution = 0.1 m/s	--> 0 .. 65535	(UNSIGNED16)
speed = 0.. 6.5535 m/s, resolution = 0,0001 m/s	--> 0 .. 65535	(UNSIGNED16)

The resolution is often a decimal fraction of a unit !

Some standards provides a bipolar or unipolar analog data format:

e.g. :

0..200% of physical variable	== 0..65536	(UNSIGNED16)
-200%..+200%$^{-\varepsilon}$ of 10 kV	== -32768 .. 32768	(INTEGER16)

fractionals require producer and consumer to agree on range and resolution:
e.g. resolution 0.5 V, range 6553.5 V

fractionals are easy to process but error prone (overflow, underflow)

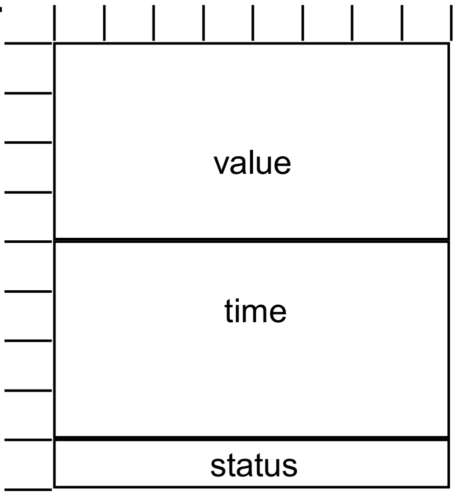
A conversion from fractional to floating point did cost over 5 Mia € (Ariane 501 accident)

Constructed Application Data Types

Circuit Breaker command:

Code	DoubleCommand	Persistent Regulating Command	RegulatingStep Command	Double-Point Information
00	not permitted	not permitted	not permitted	indeterminate
01	OFF	Lower	Next Step Lower	determined OFF
10	ON	Higher	Next Step Higher	determined OFF
11	not permitted	not permitted	not permitted	not permitted

Time-Stamped Variable:



Application Data Types

Structured Text:

RTF - Microsoft Word Native Format

HTTP - Hypermedia data presentation

Some standards for video:

QuickTime -- an Apple Computer specification for video and audio.

Motion Picture Experts Group (MPEG) -- video compression and coding.

Some graphic image formats:

Graphics Interchange Format (GIF) -- compression and coding of graphic images.

Joint Photographic Experts Group (JPEG) -- compression and coding for graphic images.

Tagged Image File Format (TIFF) -- coding format for graphic images.

Presentation Layer in the application

Coding and conversion functions to application layer data cannot be done in the presentation layer due to the lack of established rules

These functions ensure that information sent from the application layer of one system will be readable by the application layer of another system.

Examples of presentation layer coding and conversion schemes in the application:

Common data representation formats -- The use of standard image, sound, and video formats allow the interchange of application data between different types of computer

Conversion of character representation formats -- Conversion schemes are used to exchange information with systems using different text and data representations (such as ASCII and Unicode).

Common data compression schemes -- The use of standard data compression schemes allows data that is compressed at the source device to be properly decompressed at the destination (compression can take place at different levels)

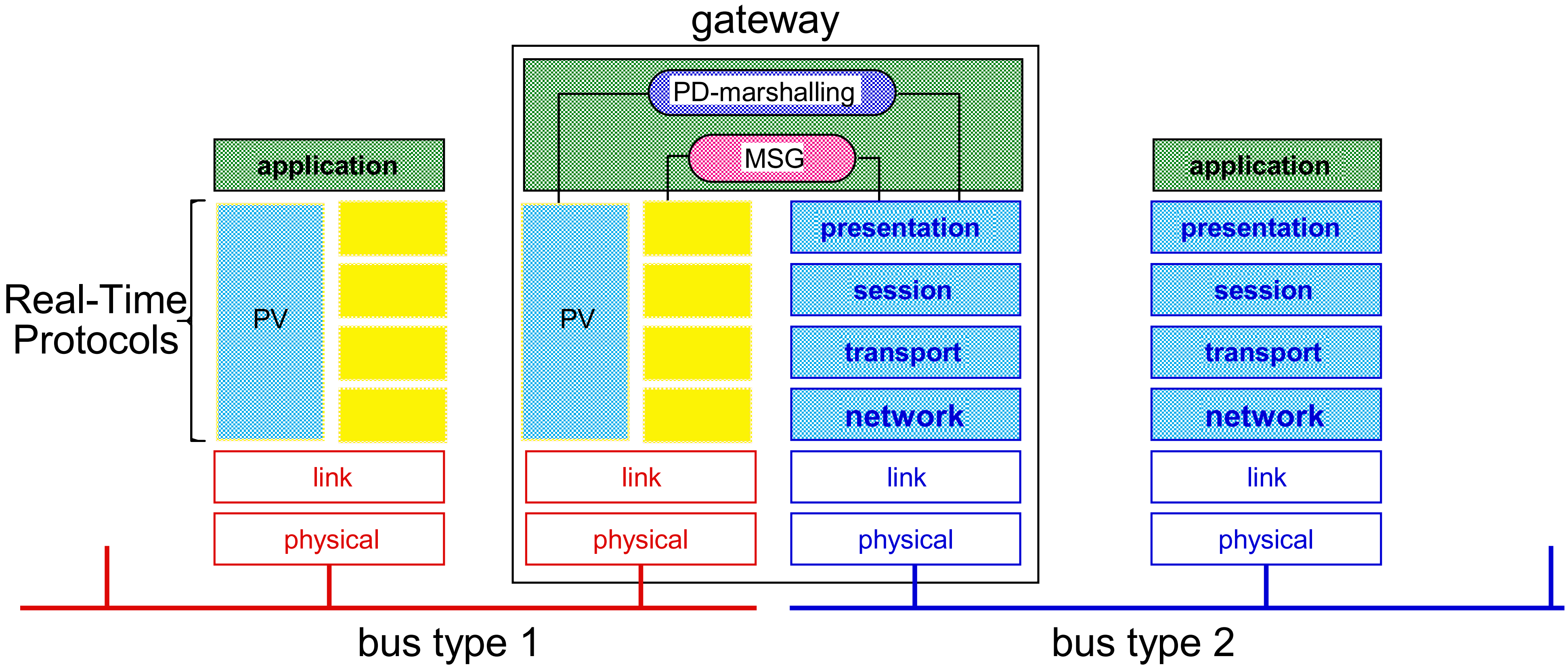
Common data encryption schemes -- The use of standard data encryption schemes allows data encrypted at the source device to be properly unencrypted at the destination.

Gateway

When devices share no common transport layer protocol, gateways act as protocol converters and application layer protocols must ensure end-to-end control.

Protocol Conversion is costly in development and real time, since protocols are in general insufficiently specified, custom-designed, and modified without notice.

Protocol Conversion requires at least an semantical equivalent of the objects on both sides of the gateway, so that one command can be converted into another - if possible.



To probe further

http://www.isi.salford.ac.uk//books/osi/all.html - An overview of OSI

http://www.oss.com/ - Vendor of ASN.1 tools

http://www-sop.inria.fr/rodeo/personnel/hoschka/347.txt - List of ASN.1 tools

http://lamspeople.epfl.ch/kirrmann/mms/OSI/osi_ASN1products.htm - ASN.1 products

Assessment

which are the “upper layers”?
which is the function of the network level ?
what is the difference between repeater, bridge, router and gateway ?
what is the difference between hierarchical and logical addressing ?
what is the role of the transport layer ?
when is it necessary to have a flow control at both the link and the transport layer ?
what is the role of the session layer in an industrial bus ?
which service of the session layer is often used in industrial networks ?
what is the role of the presentation layer ?
what is ASN.1 ?
what is the difference between an abstract syntax and a transfer syntax ?
why is ASN.1 not often used in industrial networks and what is used instead ?
what is the ROSIN notation for and how is a data structure represented ?
how should physical (analog) variables be represented ?

EPFL

3.4 MVB: a fieldbus case study

Prof. Dr. H. Kirrmann
ABB Research Center, Baden, Switzerland

MVB Outline

1. Applications in rail vehicles

2. Physical layer
    1. Electrical RS 485
    2. Middle-Distance
    3. Fibre Optics

3. Device Classes

4. Frames and Telegrams

5. Medium Allocation

6. Clock Synchronization

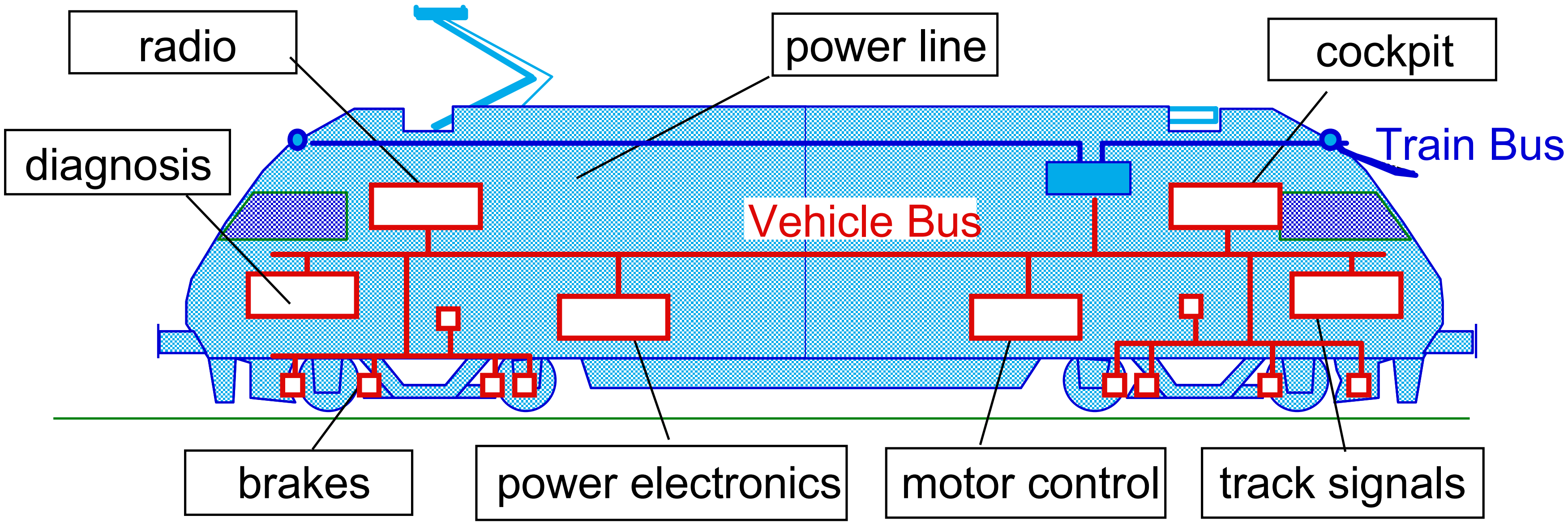
7. Fault-tolerance concept

8. Integrity Concept

9. Summary

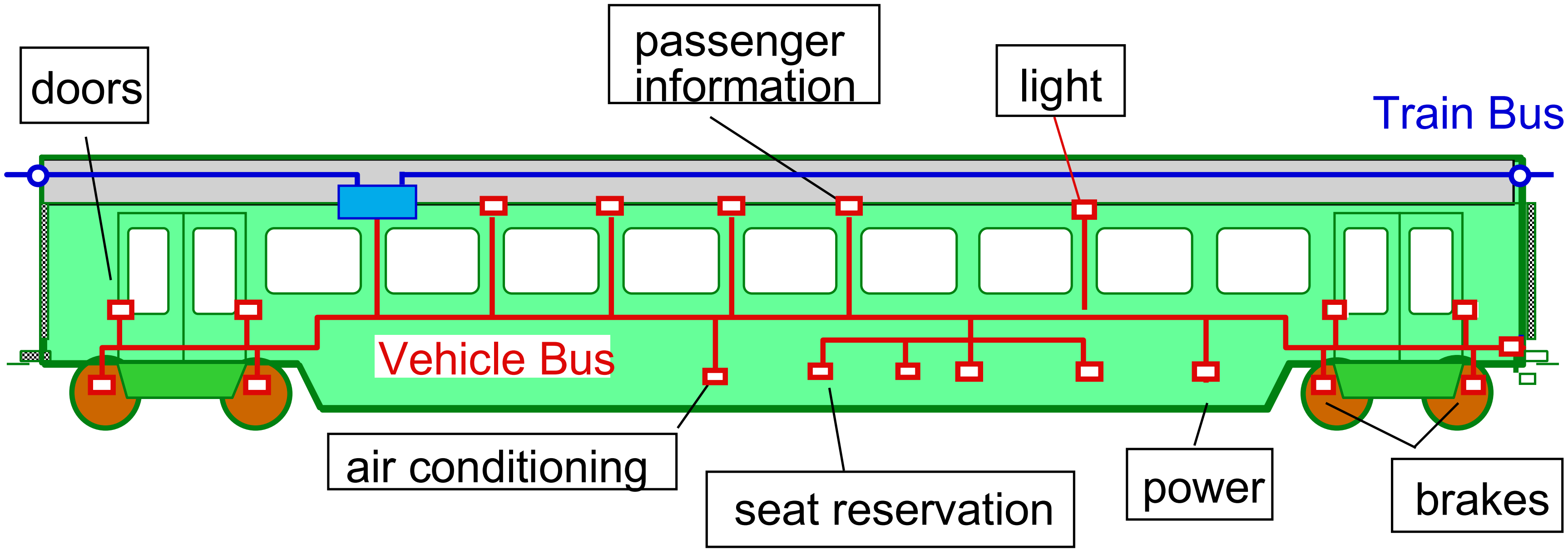
Multifunction Vehicle Bus in Locomotives

standard communication interface for all kind of on-board equipment



data rate	1'500'000 bits/second
delay	0,001 second
medium	twisted wire pair, optical fibres
number of stations	up to 255 programmable stations up to 4095 simple sensors/actuators
status	> 600 vehicles in service in 1998

Multifunction Vehicle Bus in Coaches



covered distance: > 50 m for a 26 m long vehicle
< 200 m for a train set

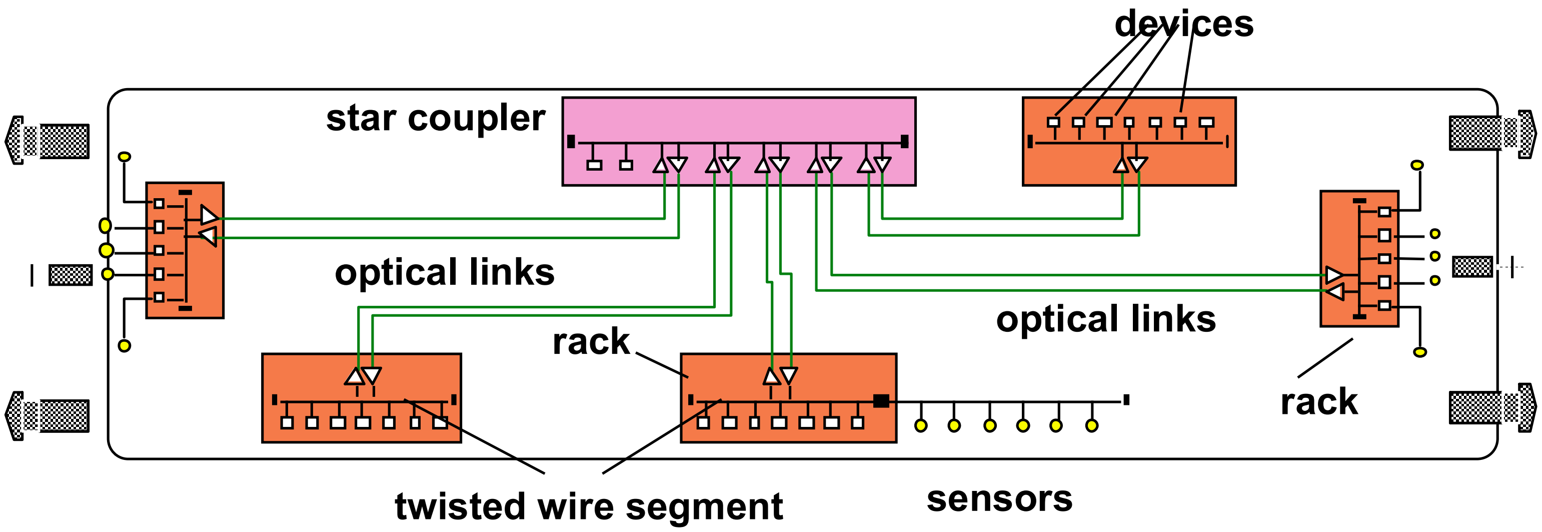
diagnostics and passenger information require relatively long, but infrequent messages

MVB Physical Media

• OGF	optical fibres	(2000 m)
• EMD	shielded, twisted wires with transformer coupling	(200 m)
• ESD	wires or backplane with or without galvanic isolation	(20 m)

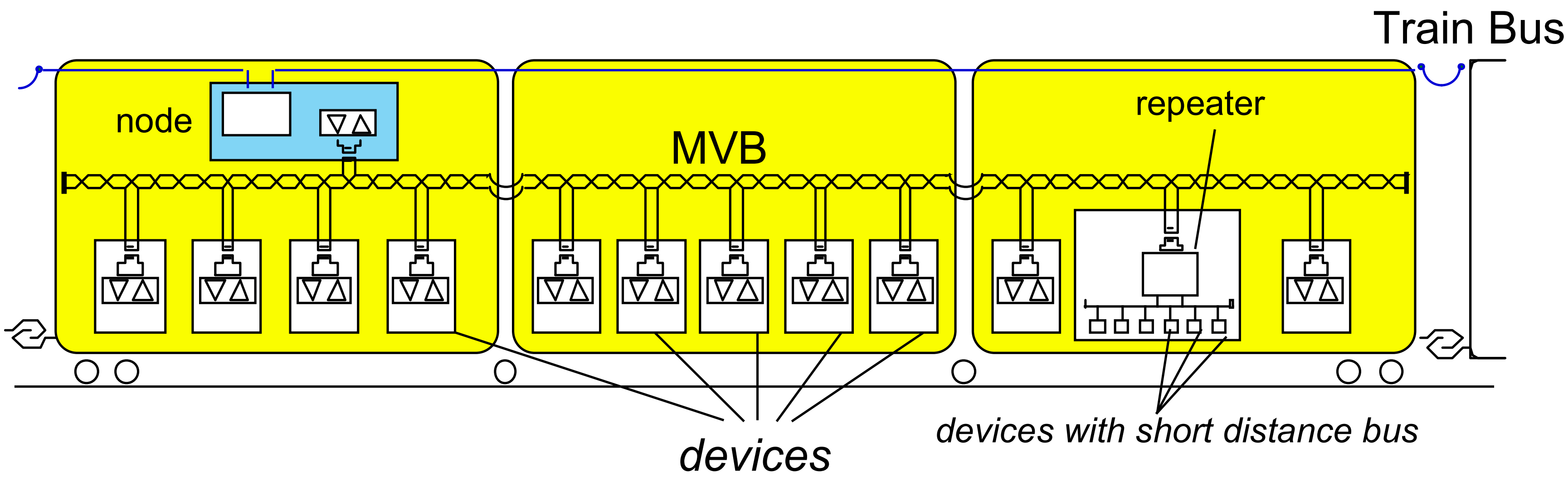
Media are directly connected by repeaters (signal regenerators)

All media operate at the same speed of 1,5 Mbit/s.



MVB Covered Distance

The MVB can span several vehicles in a multiple unit train configuration:

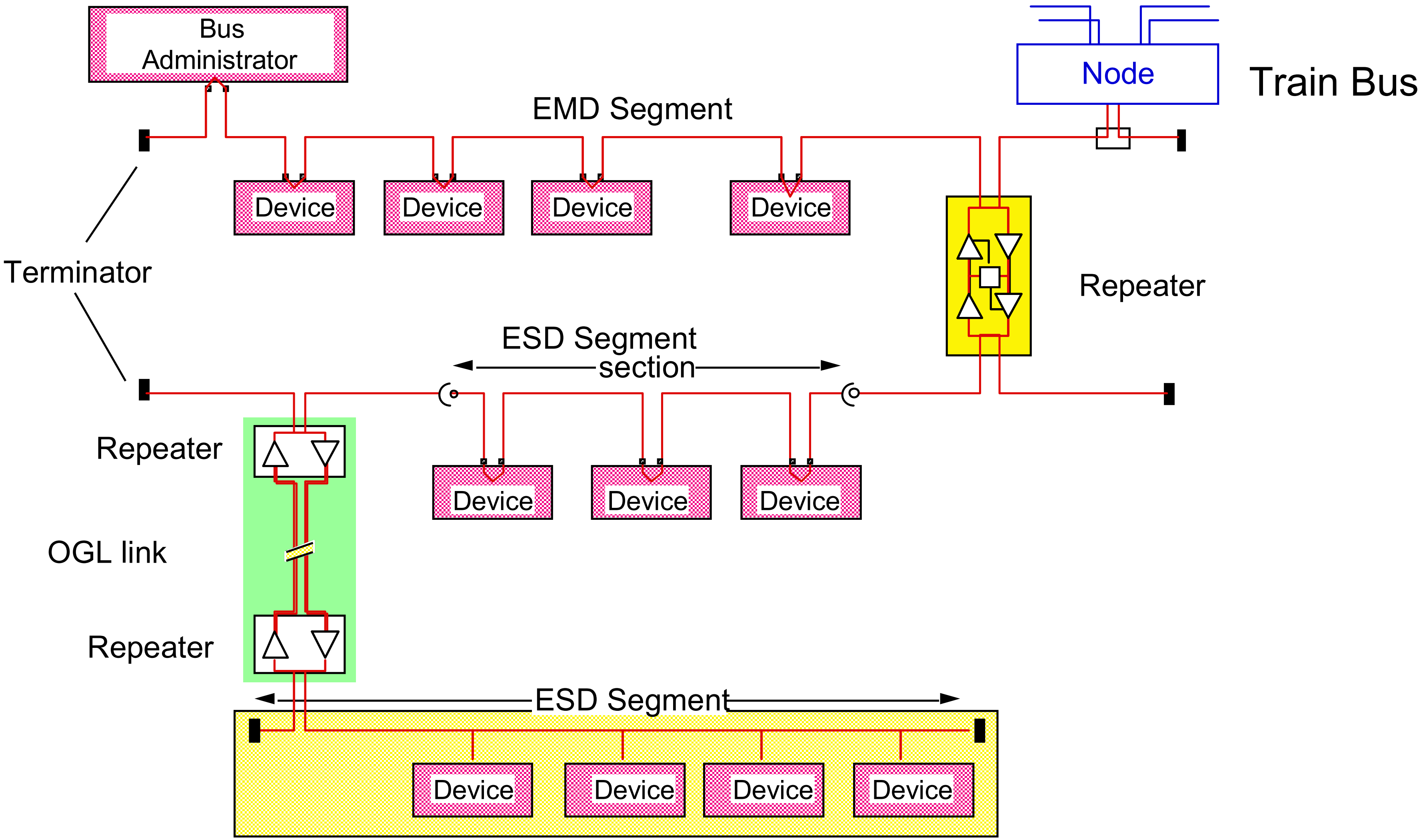


The number of devices under this configuration amounts to 4095.

MVB can serve as a train bus in trains with fixed configuration, up to a distance of:

> 200 m (EMD medium or ESD with galvanic isolation) or
> 2000 m (OGF medium).

MVB Topography



all MVB media operate at same speed, segments are connected by repeaters.

MVB Outline

1. Applications in vehicles

2. Physical layer
    1. ESD (Electrical, RS 485)
    2. EMD (Transformer-coupled)
    3. OGF (Optical Glass Fibres)

3. Device Classes

4. Frames and Telegrams

5. Medium Allocation

6. Clock Synchronization

7. Fault-tolerance concept

8. Integrity Concept

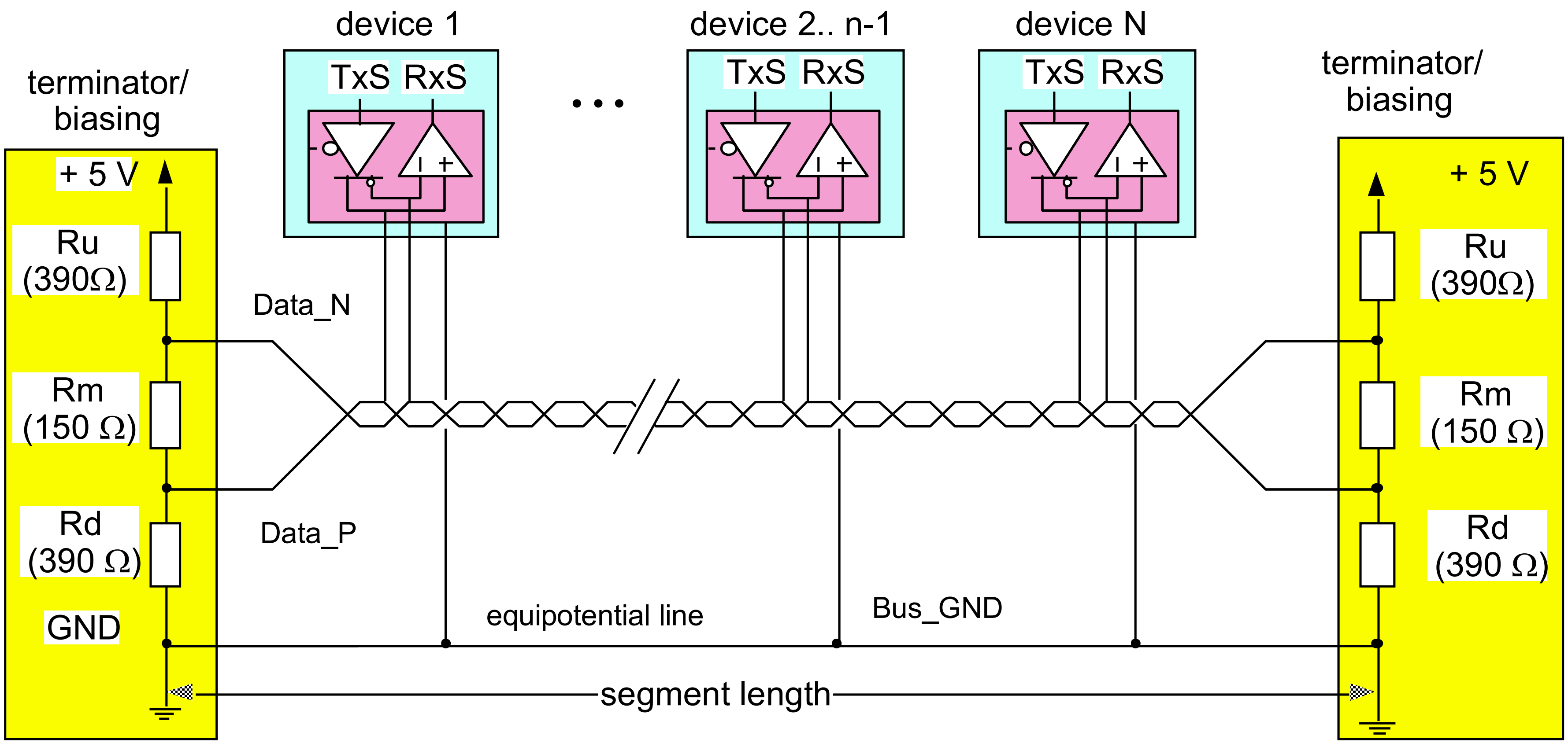
9. Summary

ESD (Electrical Short Distance) RS485

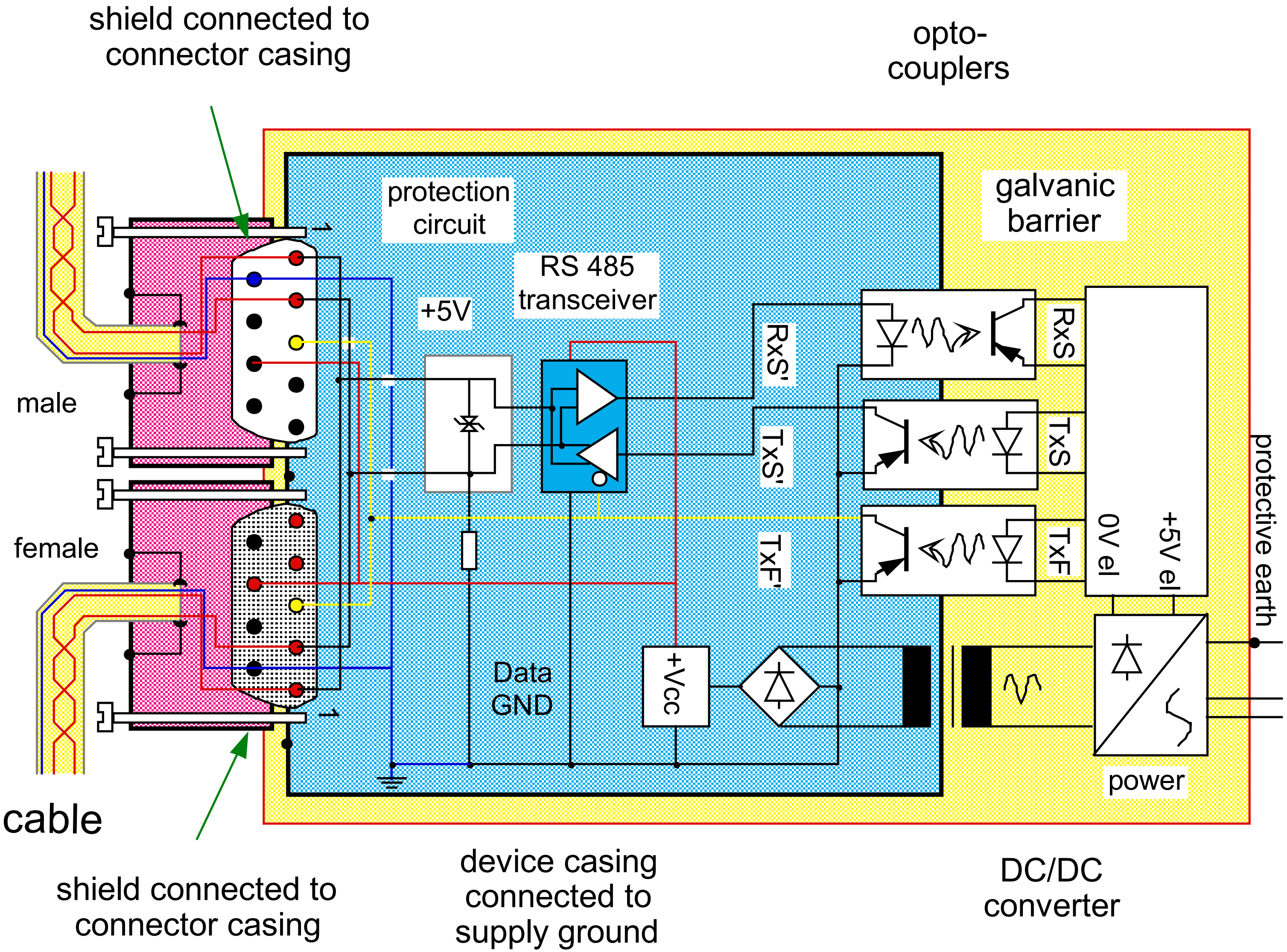
Interconnects devices over short distances (- 20m) without galvanic separation

Based on proven RS-485 technology (Profibus)

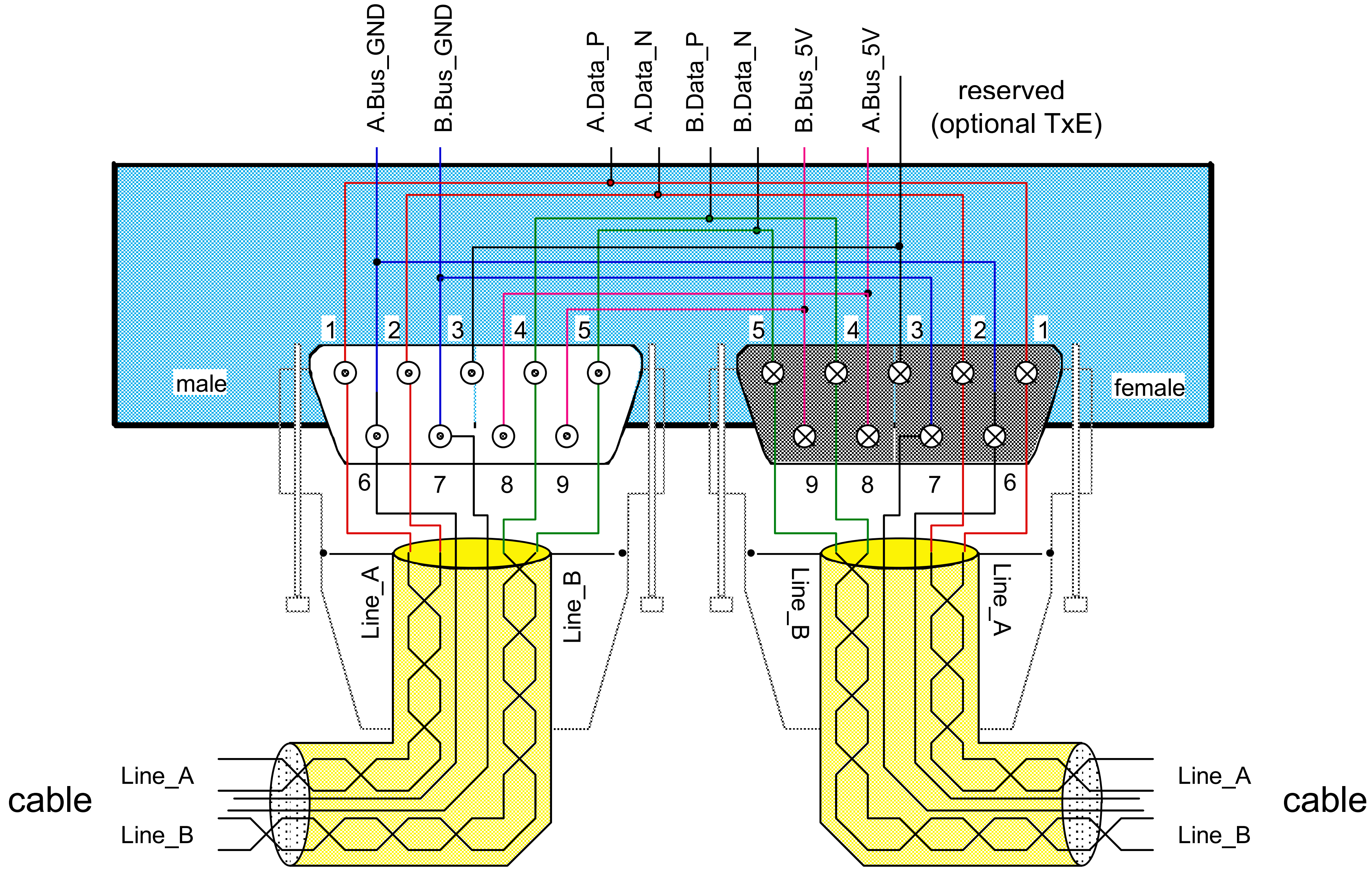
Main application: connect devices within the same cabinet.



ESD Device with Galvanic Isolation

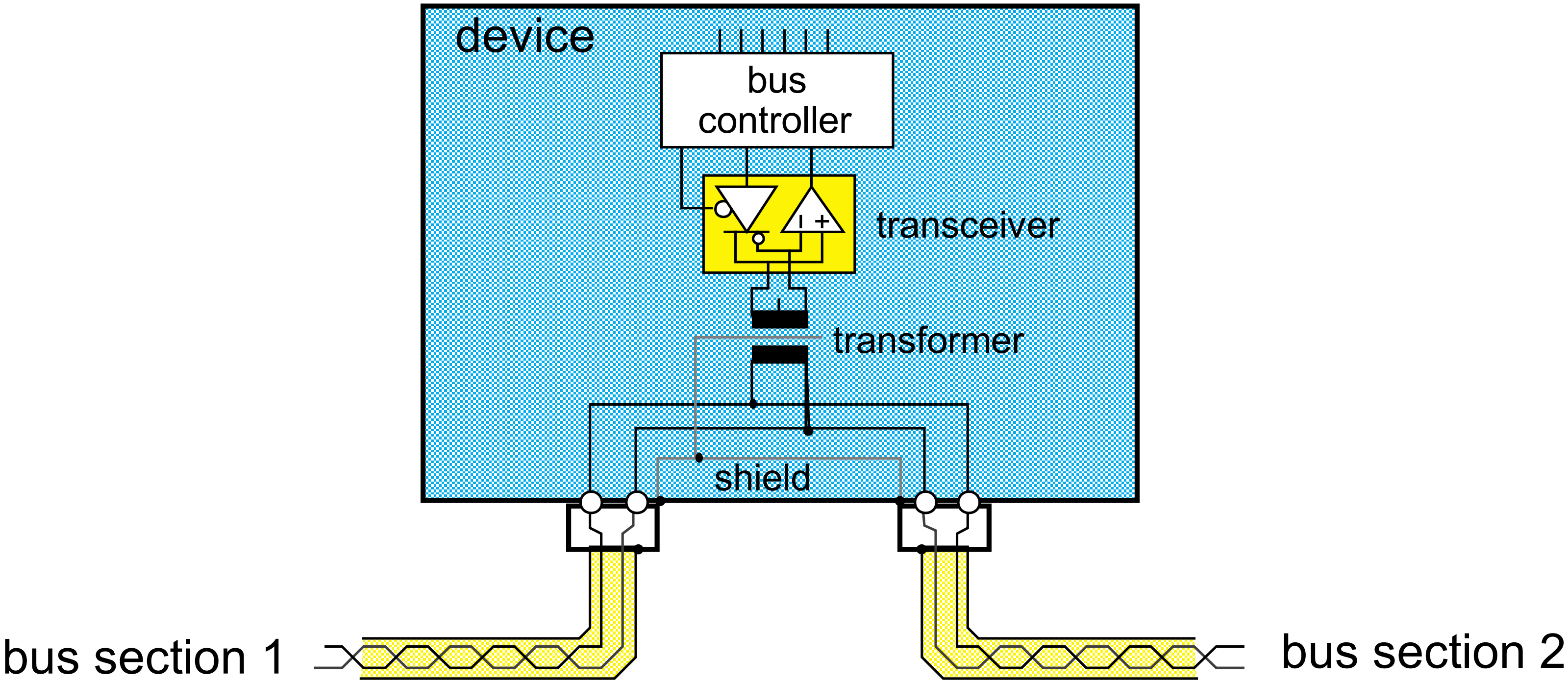


ESD Connector for Double-Line Attachment

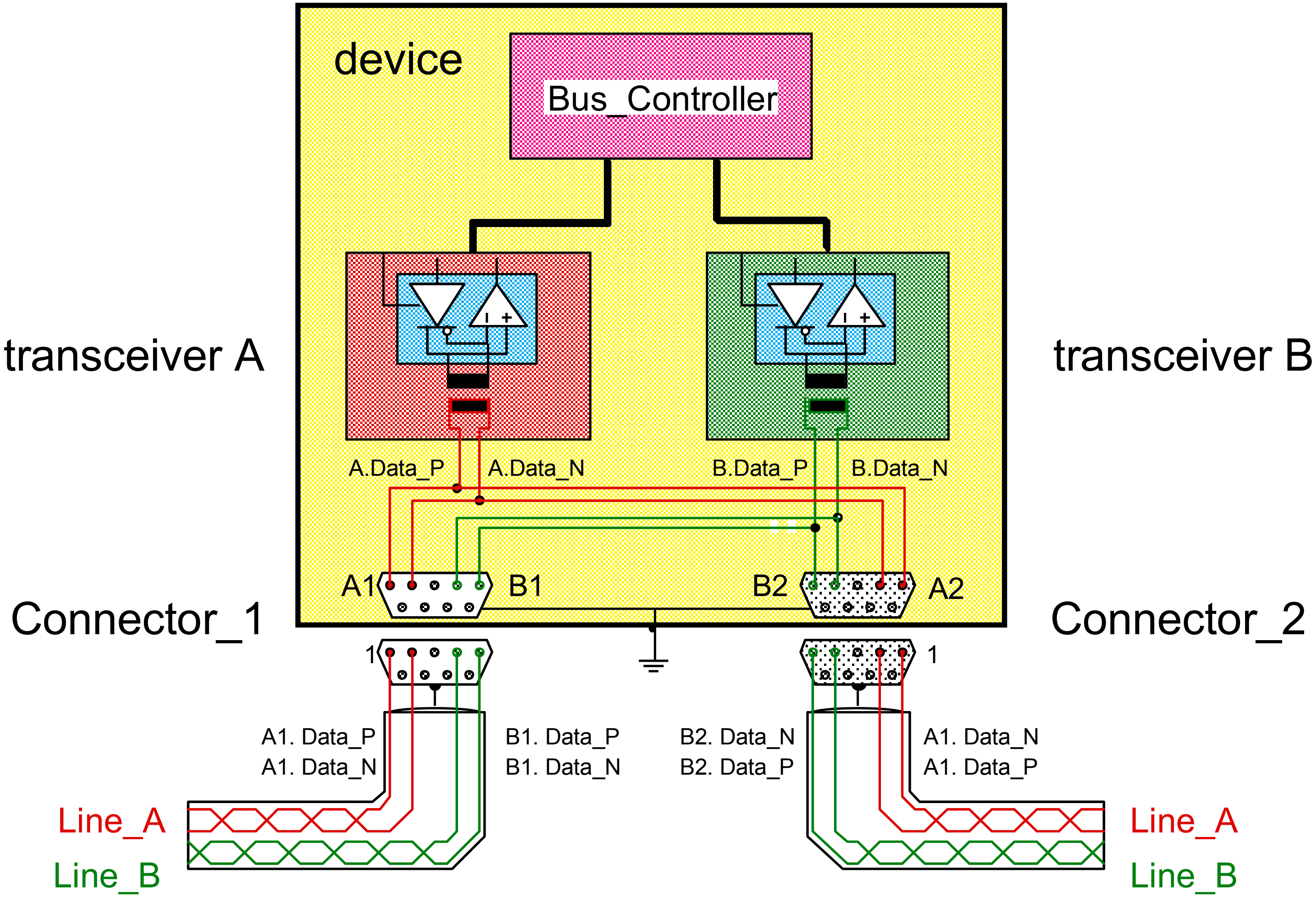


EMD (Electrical Medium Distance) - Single Line Attachment

- Connects up to 32 devices over distances of 200 m.
- Transformer coupling to provide a low cost, high immunity galvanic isolation.
- Standard 120 Ohm cable, IEC 1158-2 line transceivers can be used.
- 2 x 9-pin Sub-D connector
- Main application: street-car and mass transit



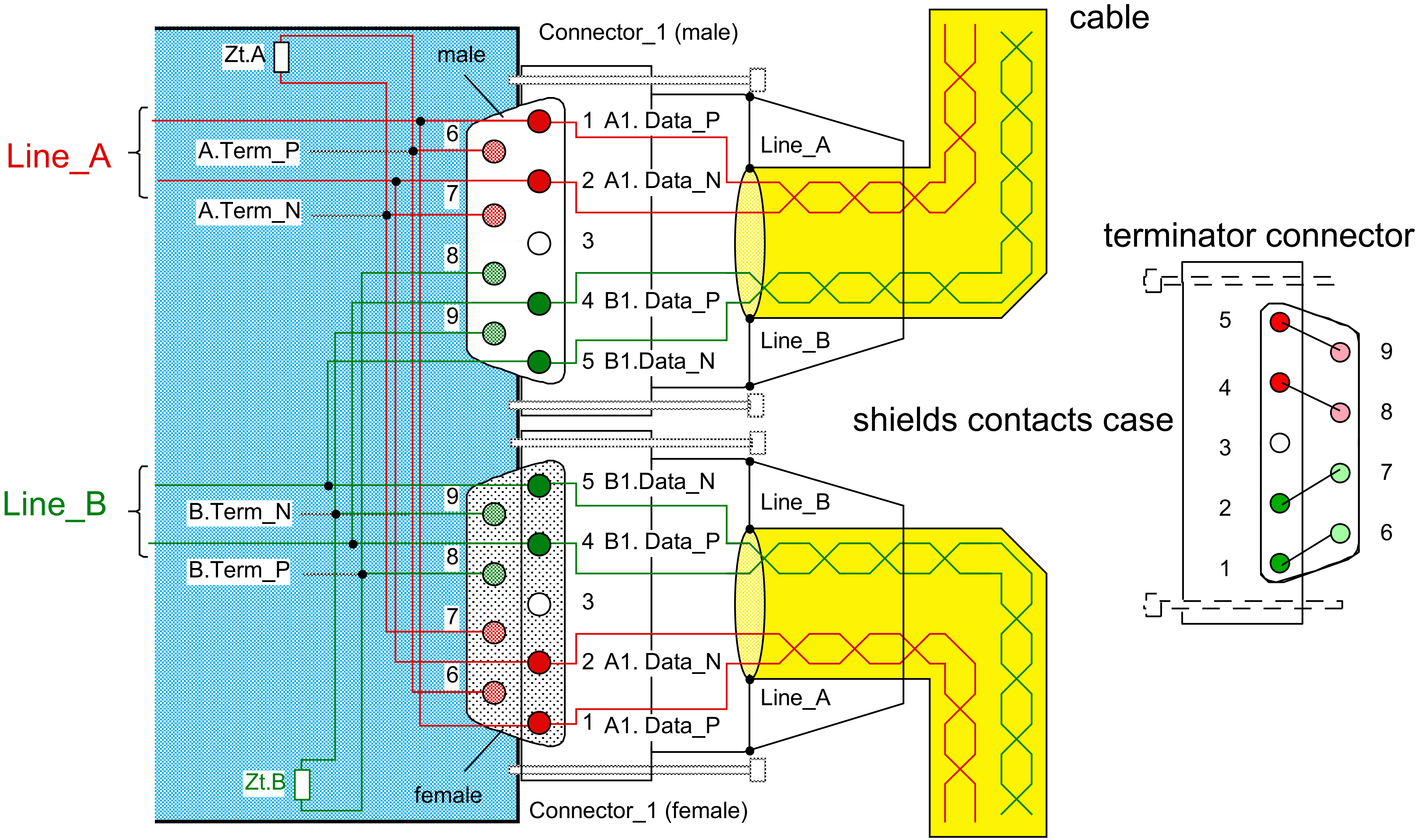
EMD Device with Double Line Attachment



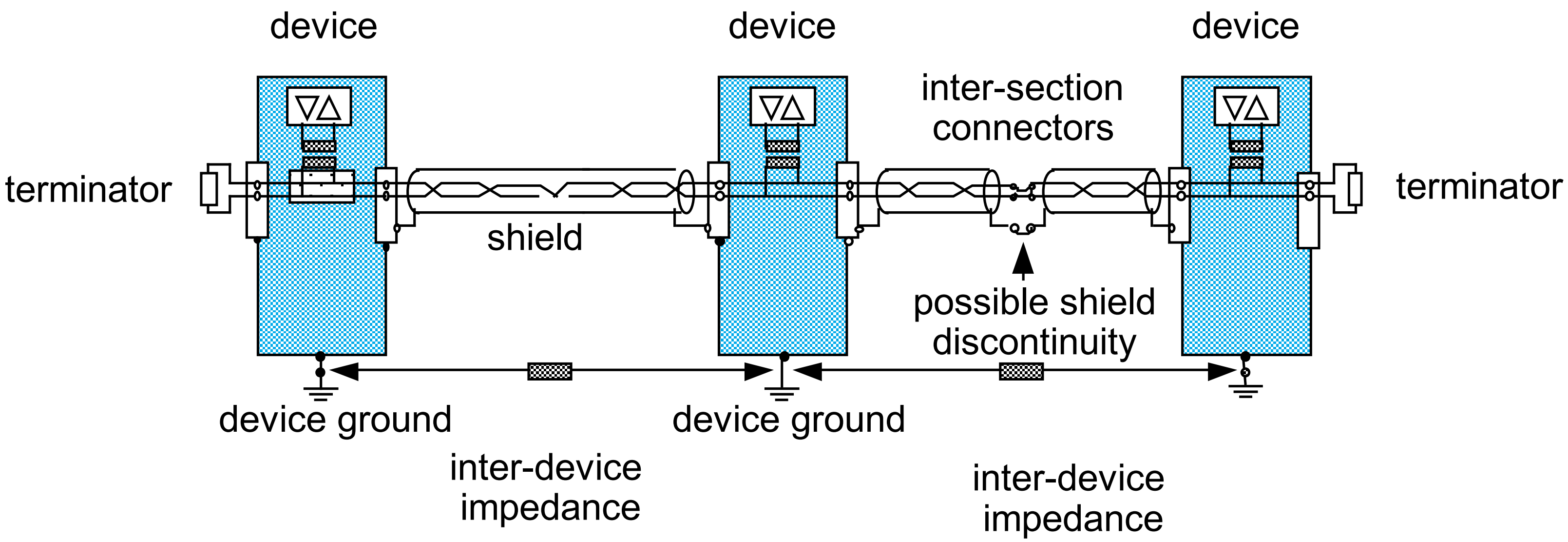
Carrying both redundant lines in the same cable eases installation

it does not cause unconsidered common mode failures in the locomotive environment
(most probable faults are driver damage and bad contact)

EMD Connectors for Double-Line Attachment



EMD Shield Grounding Concept



Shields are connected directly to the device case

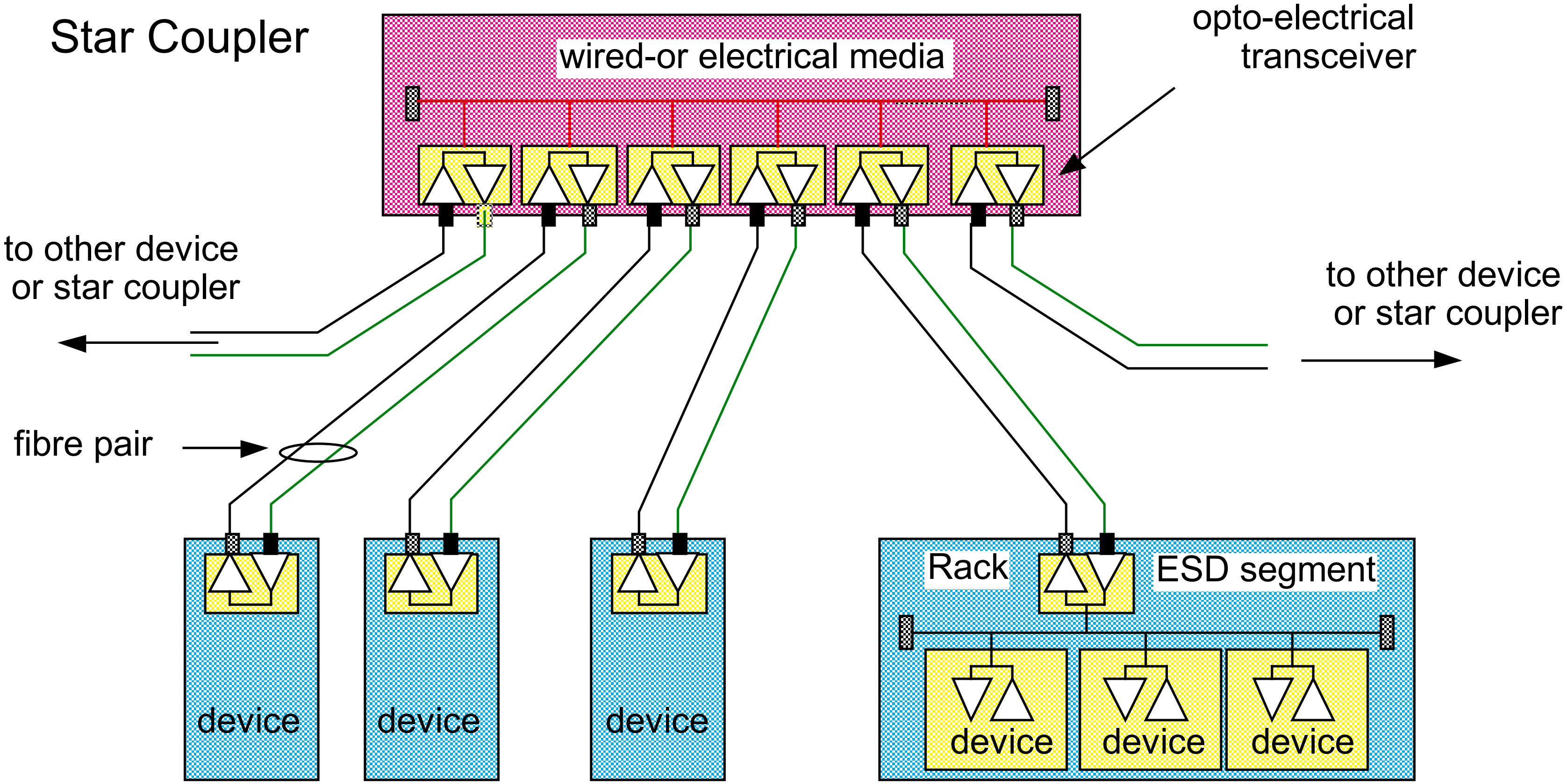
Device cases should be connected to ground whenever feasible

OGF (Optical Glass Fibre)

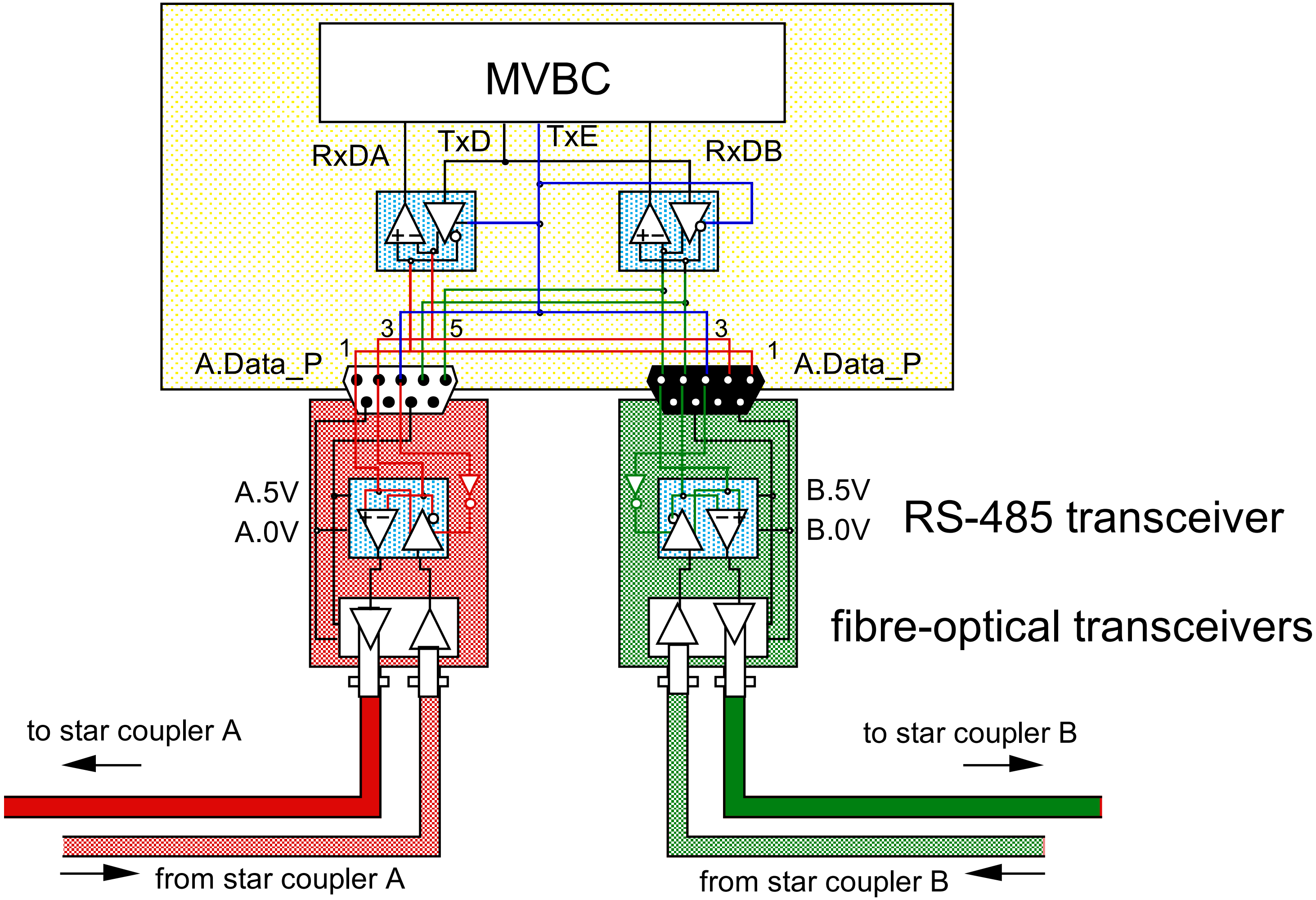
Covers up to 2000 m

Proven 240µm silica clad fibre

Main application: locomotive and critical EMC environment



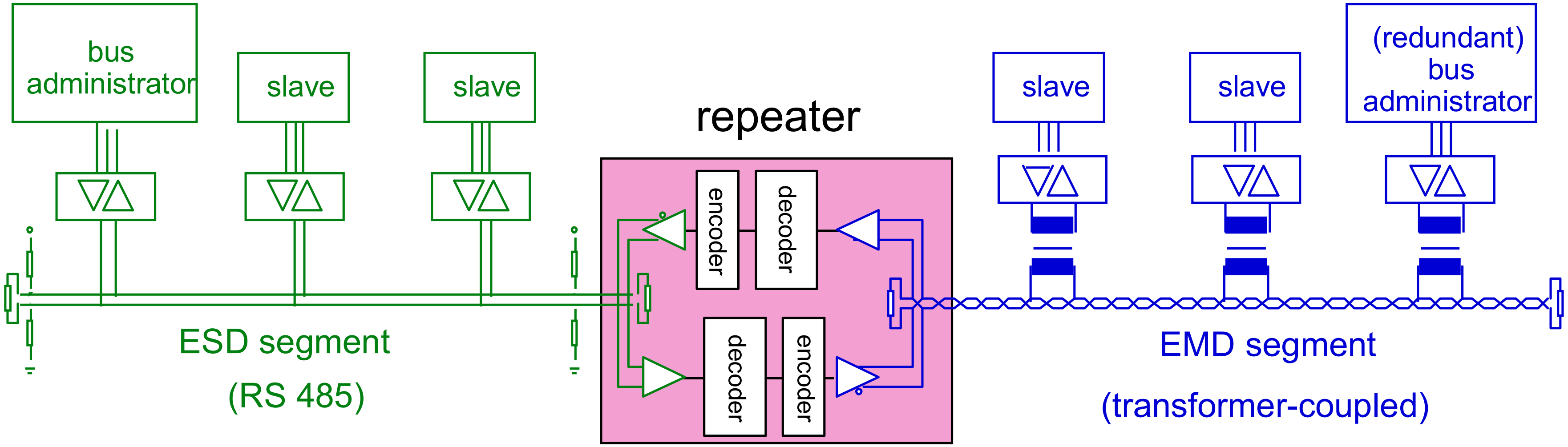
OGF to ESD adapter



Double-line ESD devices can be connected to fibre-optical links by adapters

MVB Repeater: the Key Element

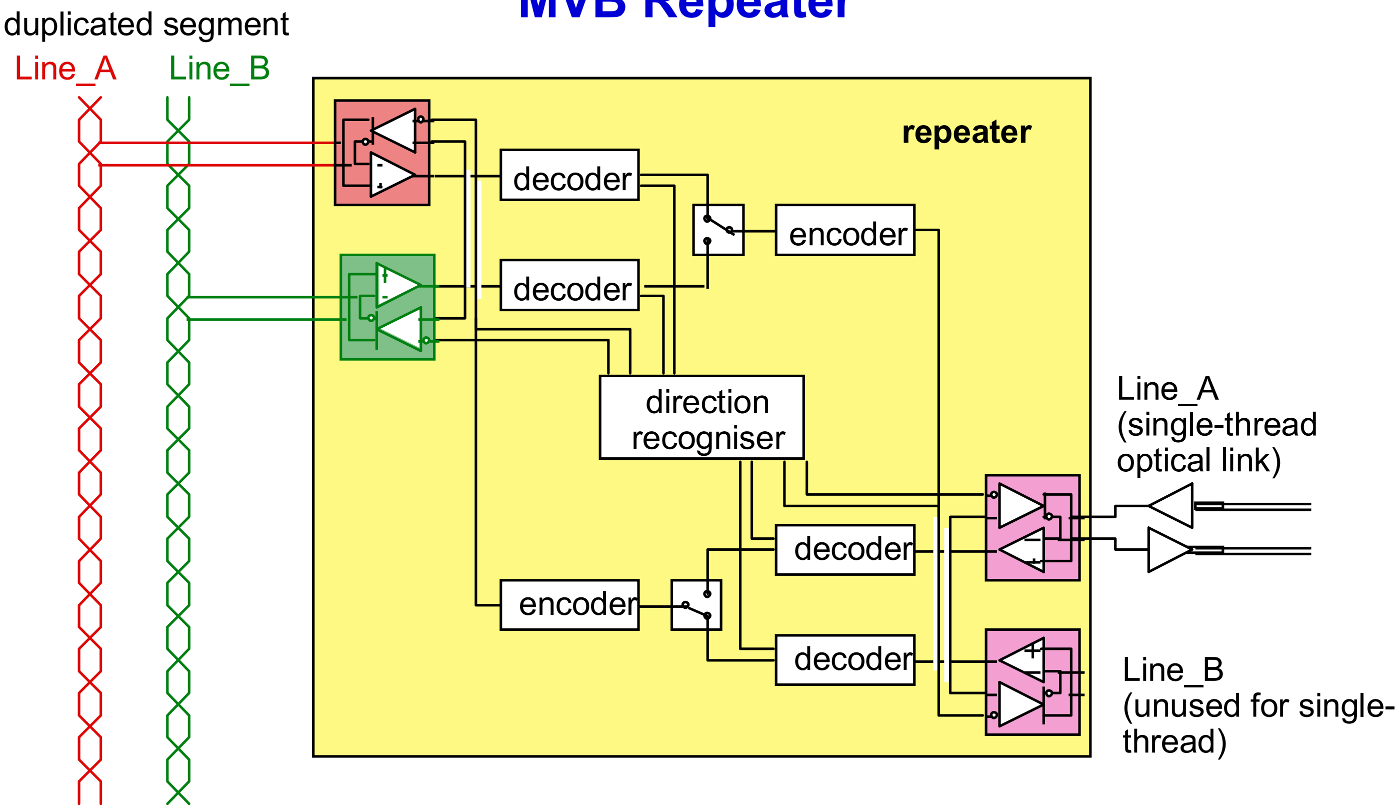
A repeater is used at a transition from one medium to another.



The repeater:

- decodes and reshapes the signal (knowing its shape)
- recognizes the transmission direction and forward the frame
- detects and propagates collisions

MVB Repeater

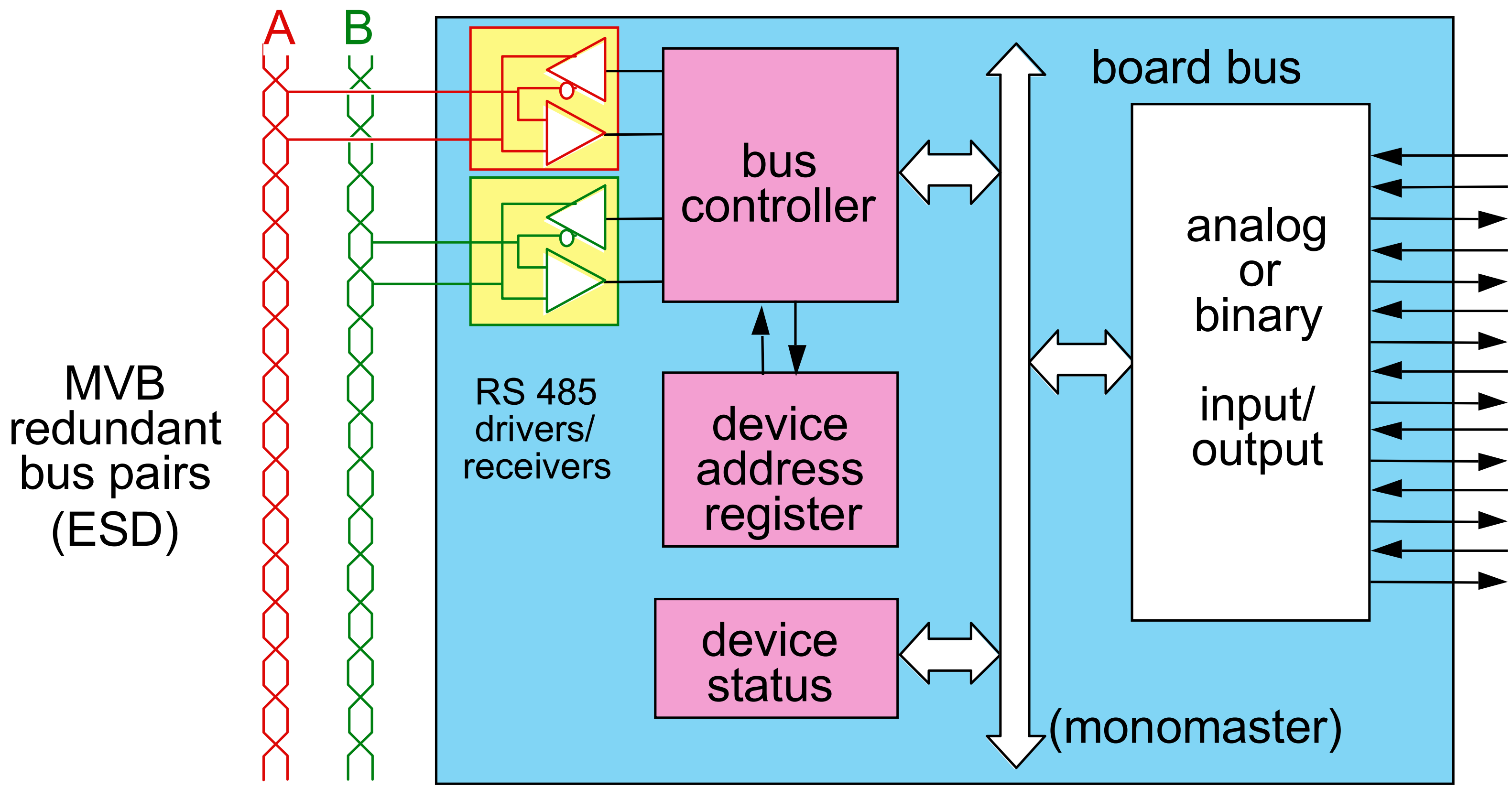


recognize the transmission direction and forward the frame
decode and reshape the signal (using a priori knowledge about ist shape)
jabber-halt circuit to isolate faulty segments
detect and propagate collisions
increase the inter-frame spacing to avoid overlap
can be used with all three media
appends the end delimiter in the direction fibre to transformer, remove it the opposite way
handles redundancy (transition between single-thread and double-thread)

MVB Outline

1. Applications in vehicles
2. Physical layer
    1. Electrical RS 485
    2. Middle-Distance
    3. Fibre Optics
3. Device Classes
4. Frames and Telegrams
5. Medium Allocation
6. Clock Synchronization
7. Fault-tolerance concept
8. Integrity Concept
9. Summary

MVB Class 1 Device



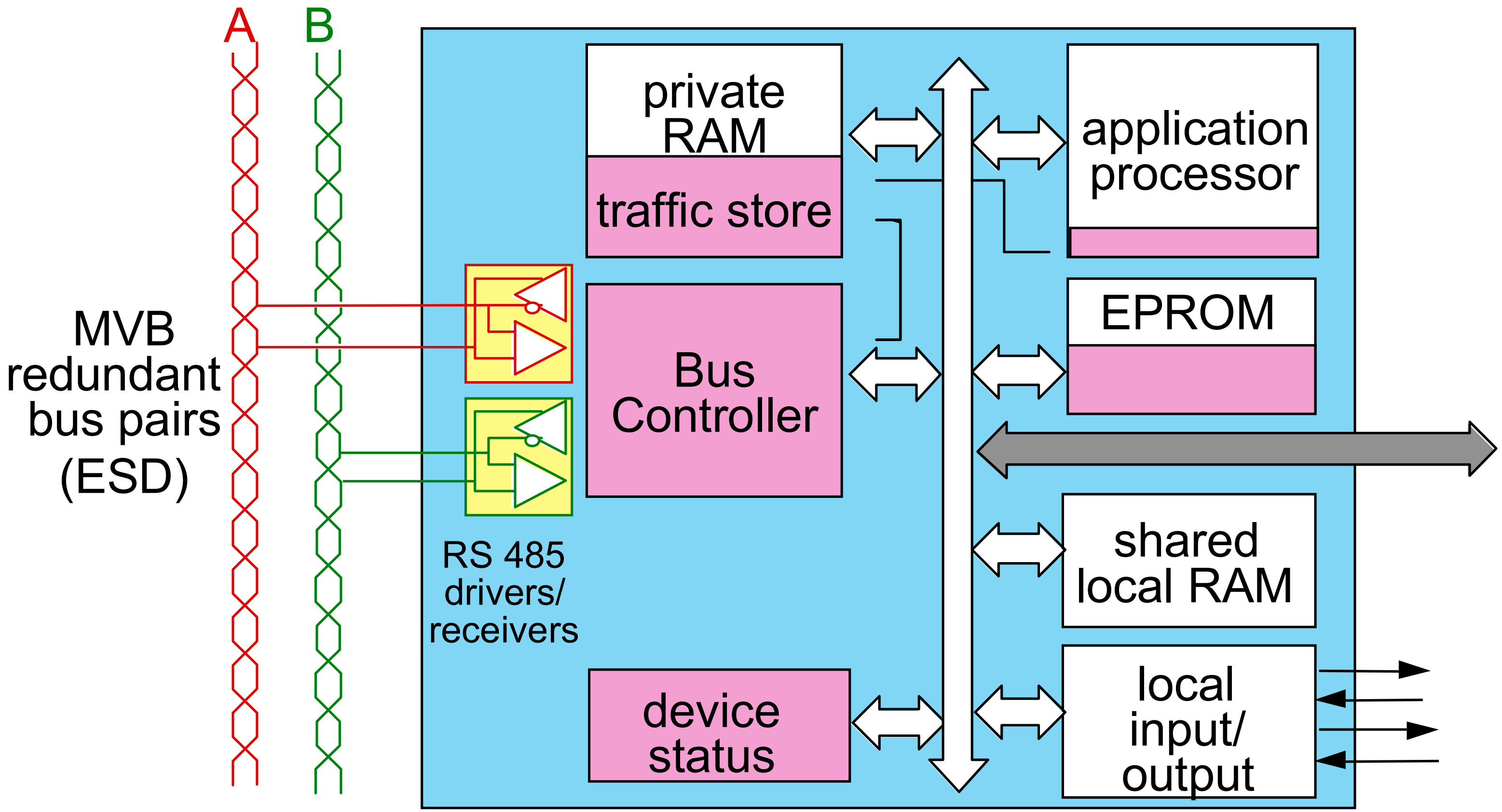
Class 1 or field devices are simple connections to sensors or actuators.

They do not require a micro-controller.

They do not participate in message data communication.

The Bus Controller manages both the input/output and the bus.

MVB Class 2-3 Device



- Class 2 and higher devices have a processor and may exchange messages.
- Class 2 devices are configurable I/O devices (but not programmable)
- The Bus Controller communicates with the Application Processor through a shared memory, the traffic store, which holds typically 256 ports.

MVB Class 4-5 Device

Class 4 devices present the functionality of a Programming and Test station

Class 4 devices are capable of becoming Bus Administrators.

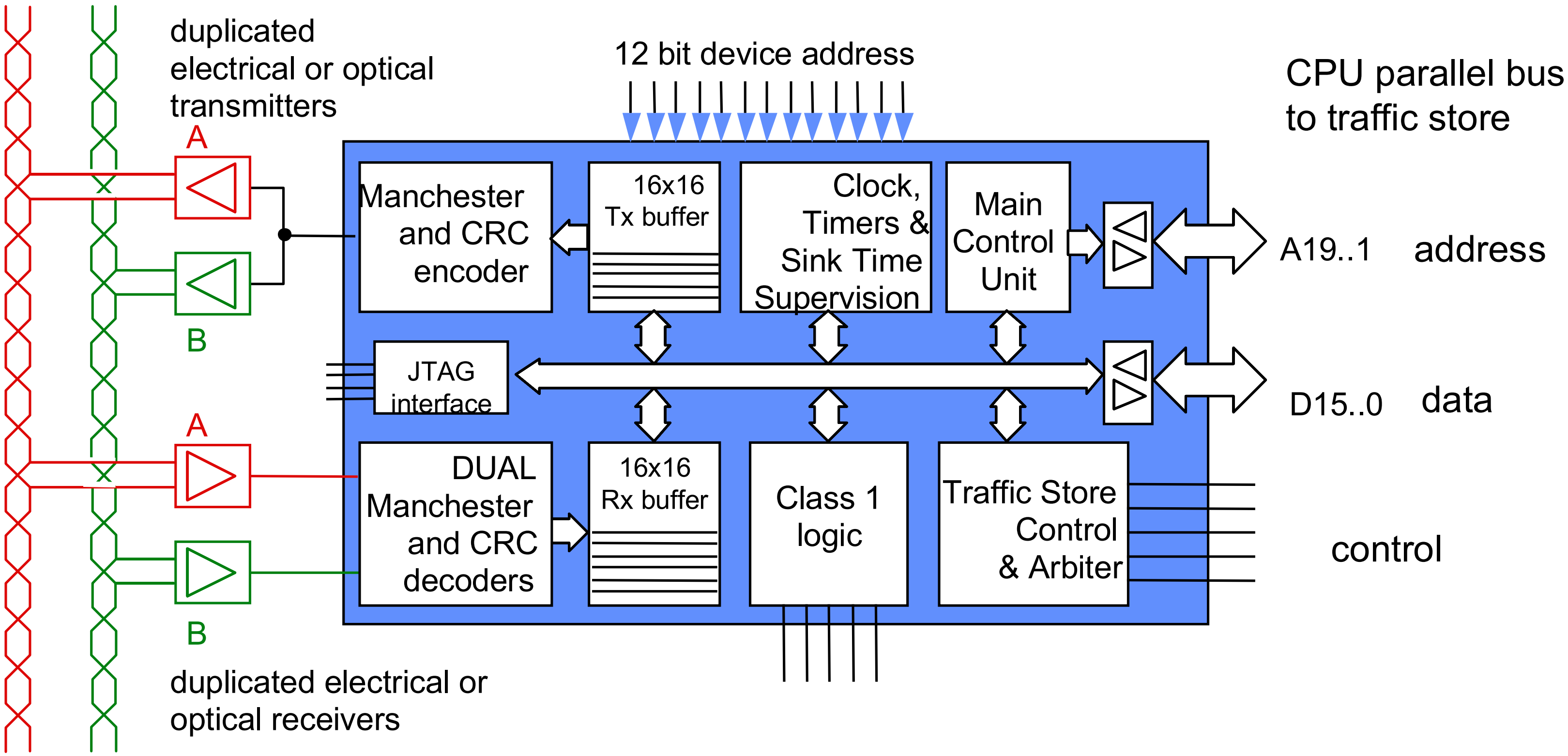
To this effect, they hold additional hardware to read the device status of the other devices and to supervise the configuration.

They also have a large number of ports, so they can supervise the process data transmission of any other device.

Class 5 devices are gateways with several link layers (one or more MVB, WTB).

The device classes are distinguished by their hardware structure.

MVBC - bus controller ASIC

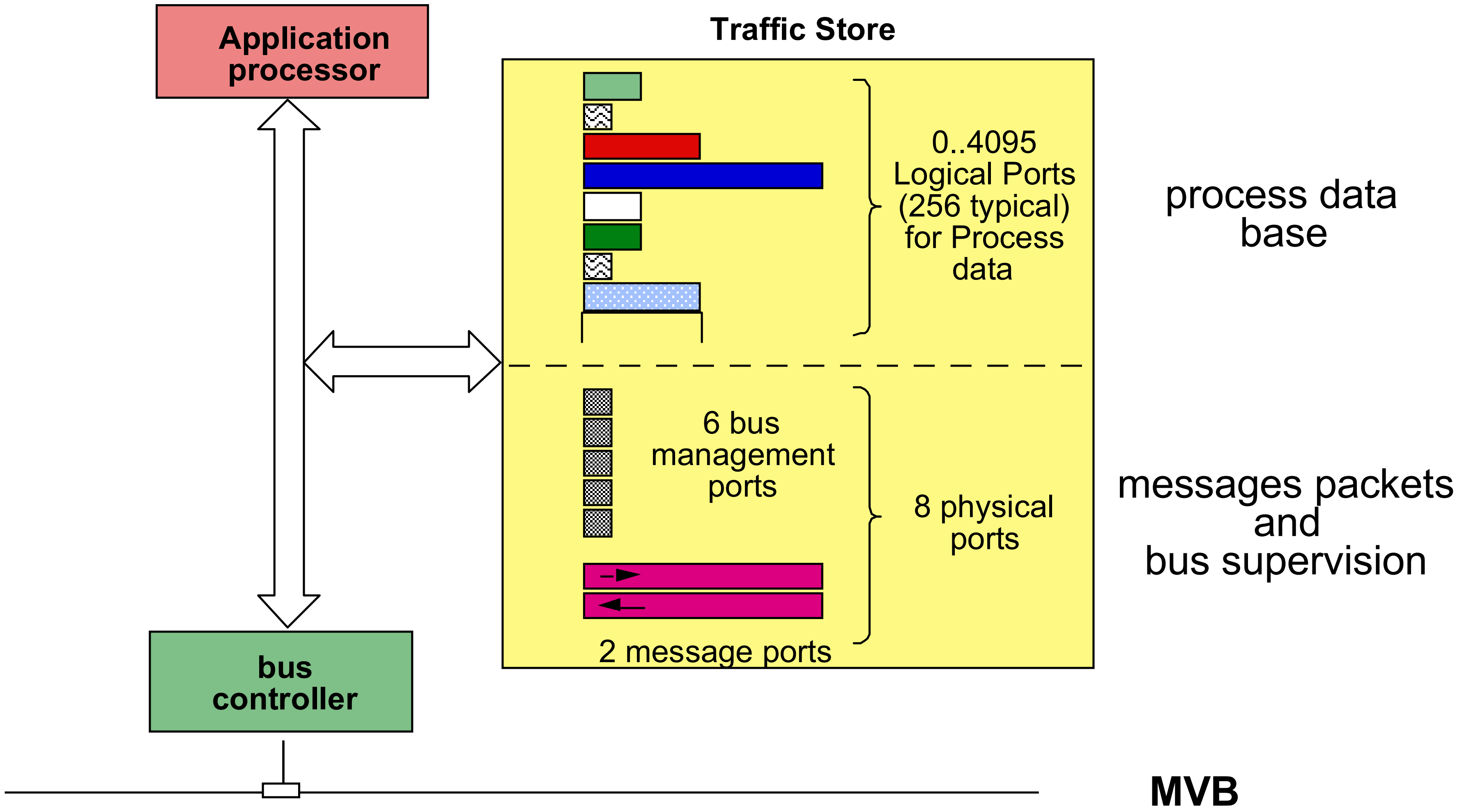


- Automatic frame generation and analysis
- Adjustable reply time-out
- Up to 4096 ports for process data
- 16KByte.. 1MByte traffic store
- Freshness supervision for process data
- In Class 1 mode: up to 16 ports
- Bit-wise forcing
- Time and synchronization port

- Bus administrator functions
- Bookkeeping of communication errors
- Hardware queueing for message data
- Supports 8 and 16-bit processors
- Supports big and lirttle endians
- 24 MHz clock rate
- HCMOS 0.8 μm technology
- 100 pin QFP

MVB Bus Interface

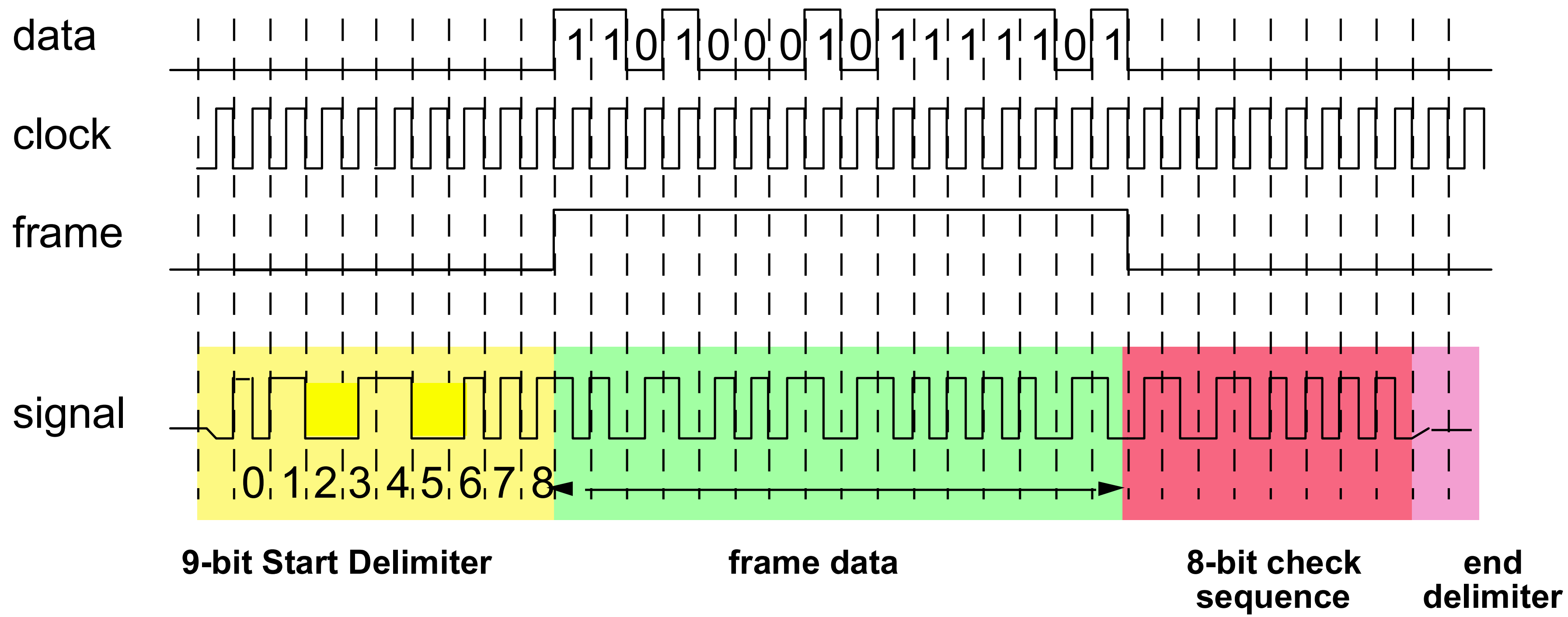
The interface between the bus and the application is a shared memory, the *Traffic Memory* , where Process Data are directly accessible to the application.



MVB Outline

1. Applications in vehicles
2. Physical layer
    1. Electrical RS 485
    2. Middle-Distance
    3. Fibre Optics
3. Device Classes
4. Frames and Telegrams
5. Medium Allocation
6. Clock Synchronization
7. Fault-tolerance concept
8. Integrity Concept
9. Summary

MVB Manchester Encoding

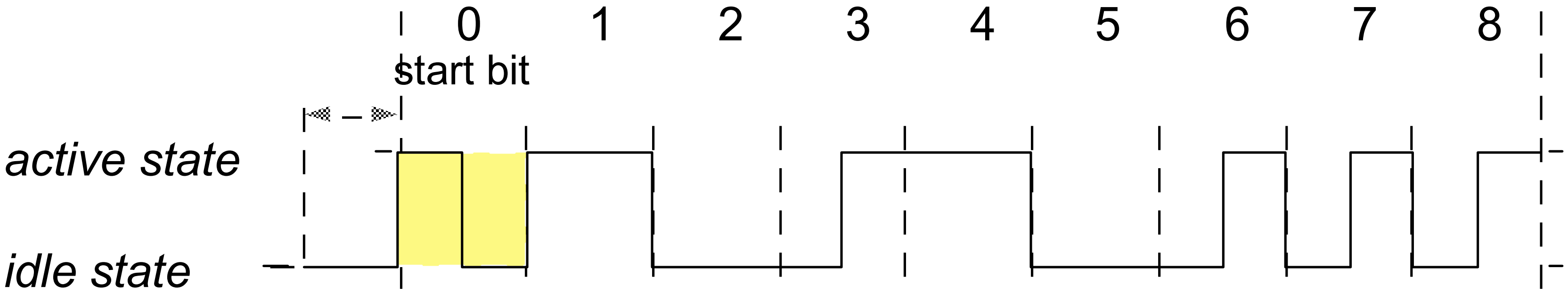


The Manchester-coded frame is preceded by a Start Delimiter containing non-Manchester signals to provide transparent synchronization.

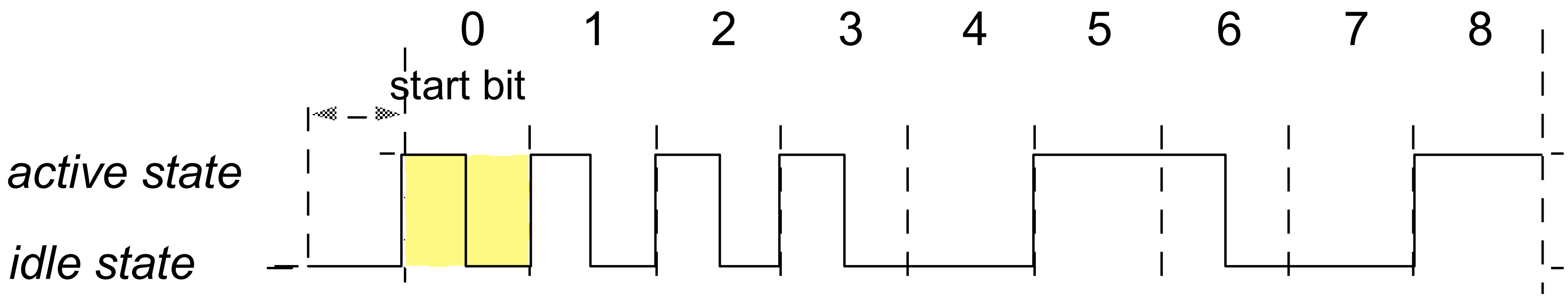
MVB Frame Delimiters

Different delimiters identify master and slave frames:

Master Frame Delimiter



Slave Frame Delimiter

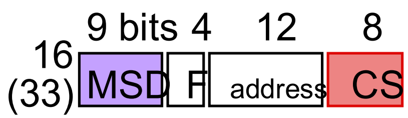


This prevents mistaking the next master frame when a slave frame is lost.

MVB Frames Formats

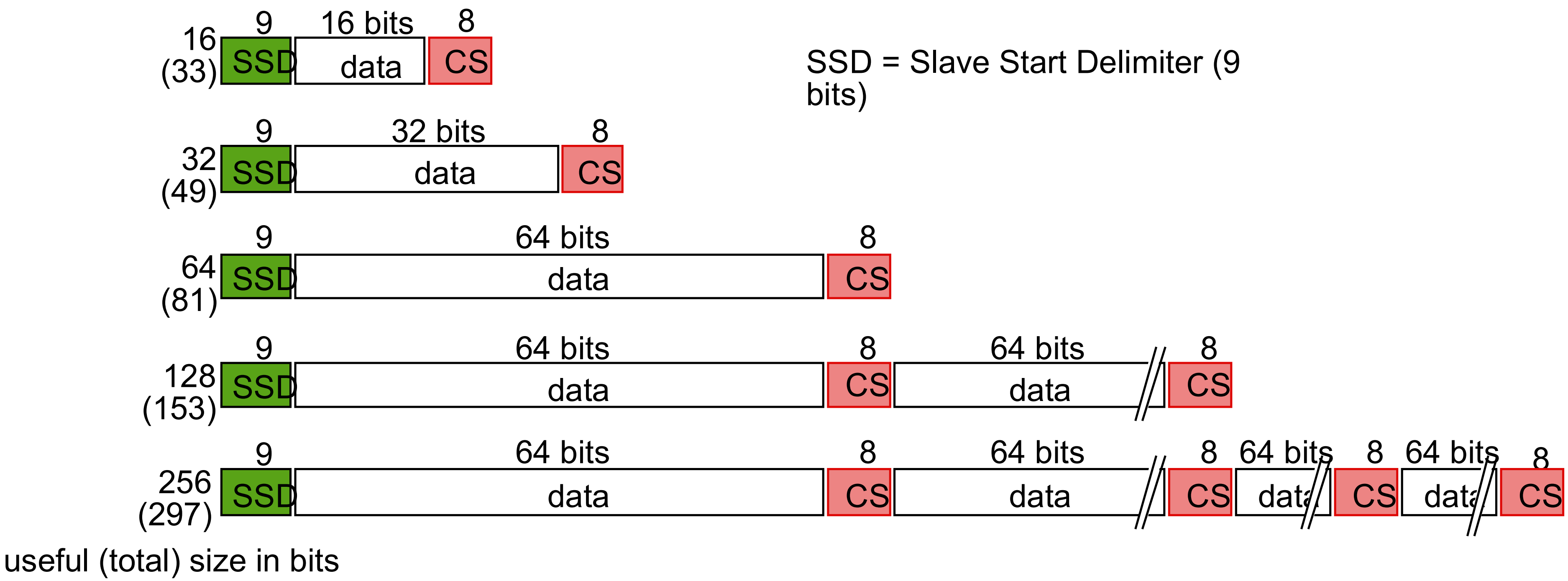
The MVB distinguishes two kinds of frames:

master frames issued by the master



MSD = Master Start Delimiter (9 bits)
CS = Check Sequence (8 bits)
F = F_code (4 bits)

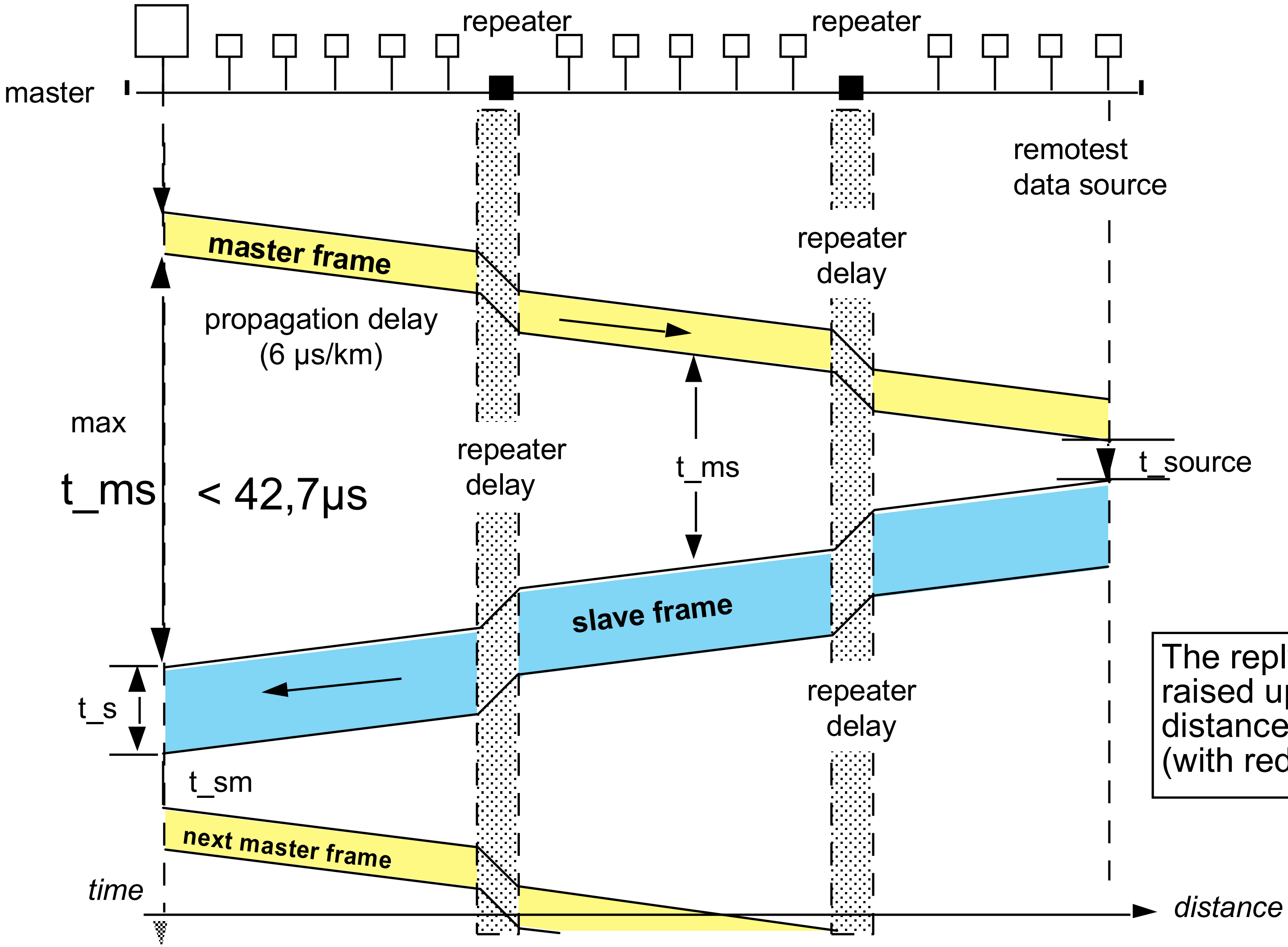
slave frames sent in response to master frames



SSD = Slave Start Delimiter (9 bits)

MVB Distance Limits

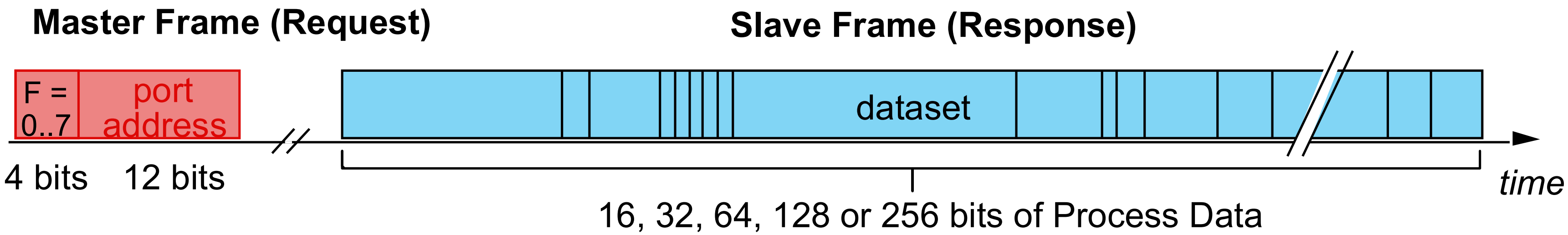
The distance is limited by the maximum allowed *reply delay* of 42,7 µs between a master frame and a slave frame.



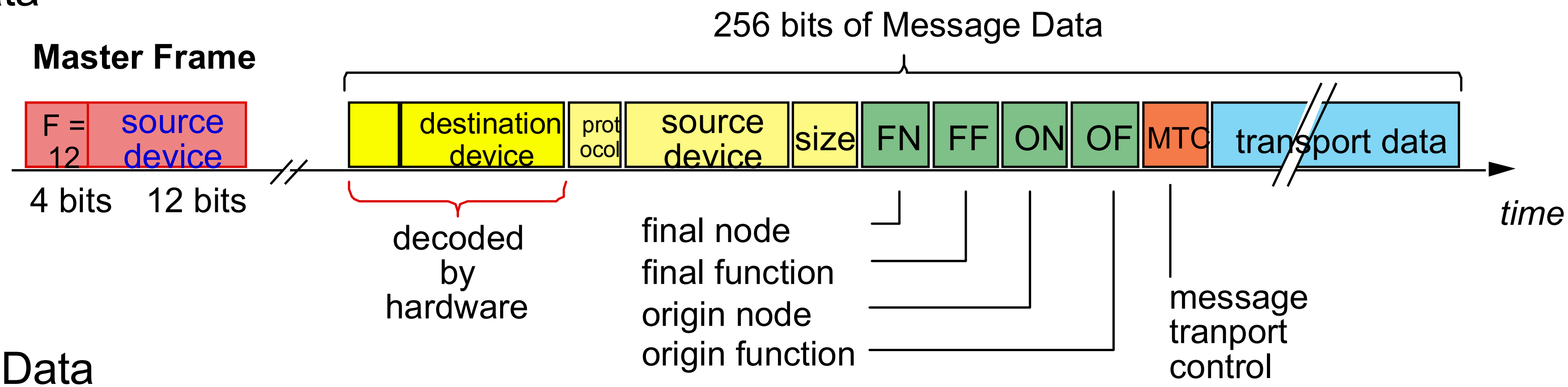
The reply delay time-out can be raised up to 83,4 µs for longer distances
(with reduced troughput).

MVB Telegrams

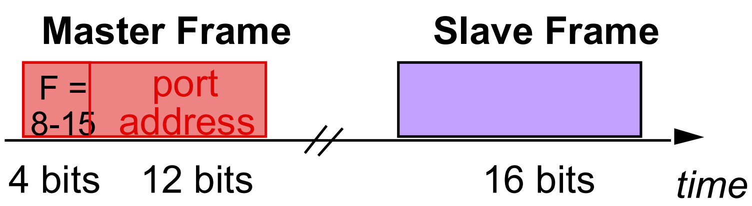
Process Data



Message Data



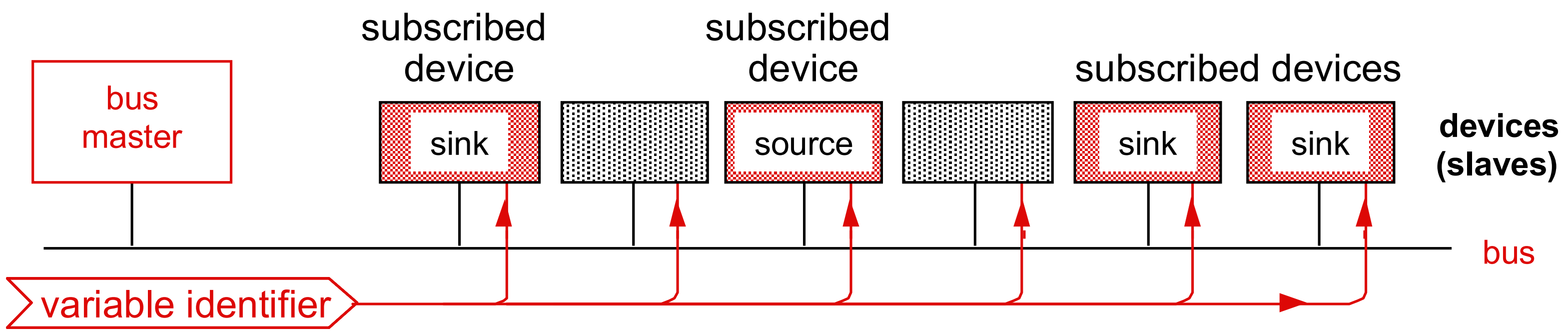
Supervisory Data



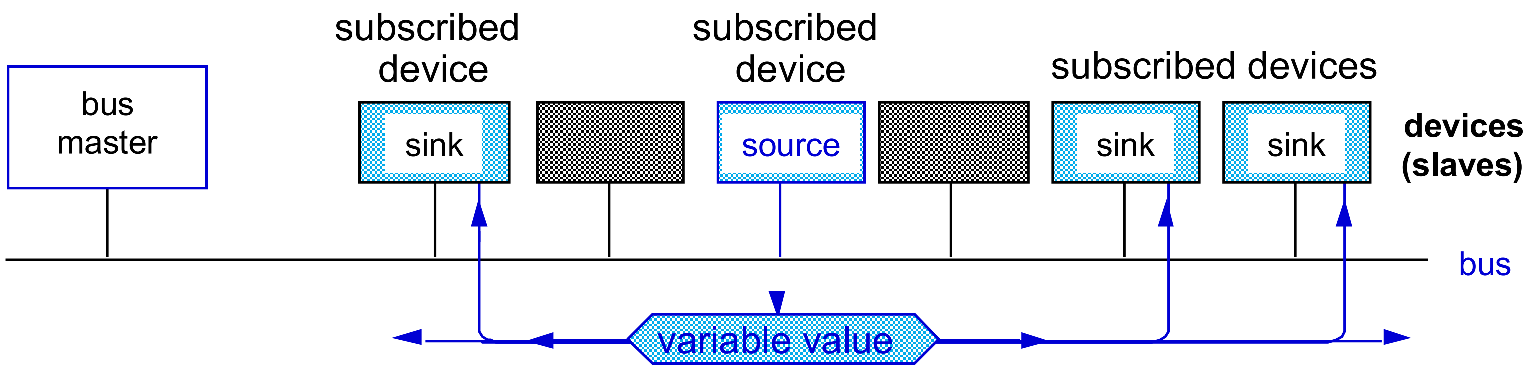
Telegrams are distinguished by the F_code in the Master Frame

Source-addressed broadcast

Phase1: The bus master broadcasts the identifier of a variable to be transmitted:

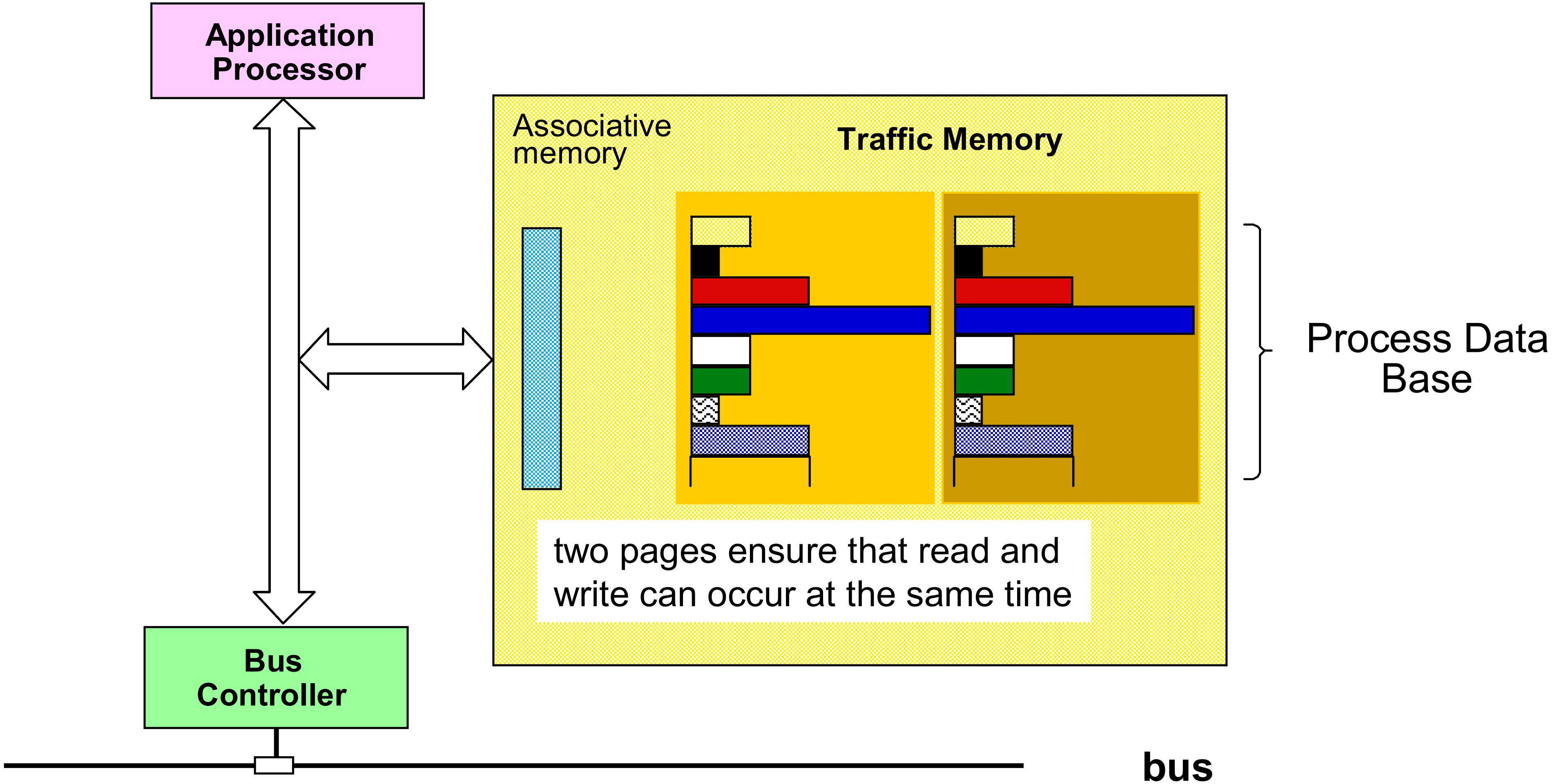


Phase 2: The device which sources that variable responds with a slave frame containing the value, all devices subscribed as sink receive that frame.

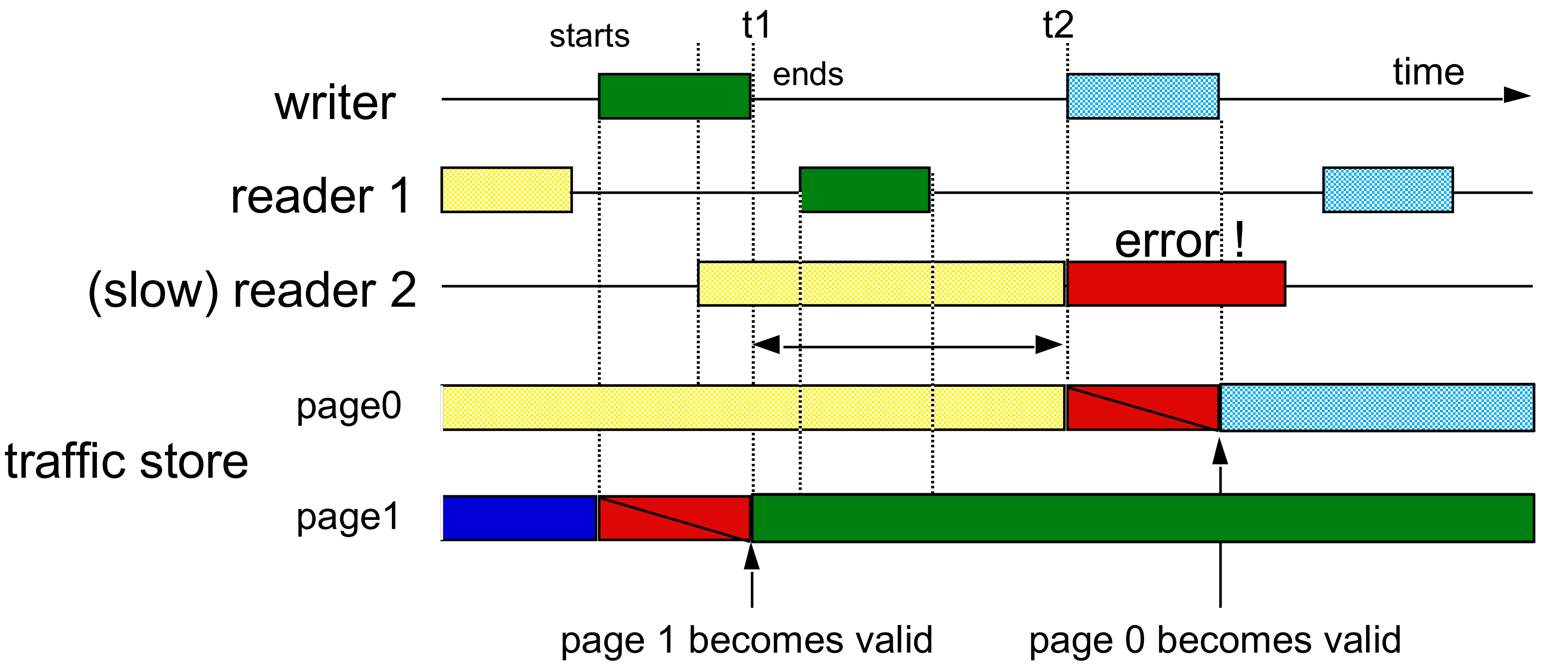


Traffic Memory

The bus and the application are (de)coupled by a shared memory, the *Traffic Memory,* where process variables are directly accessible to the application.



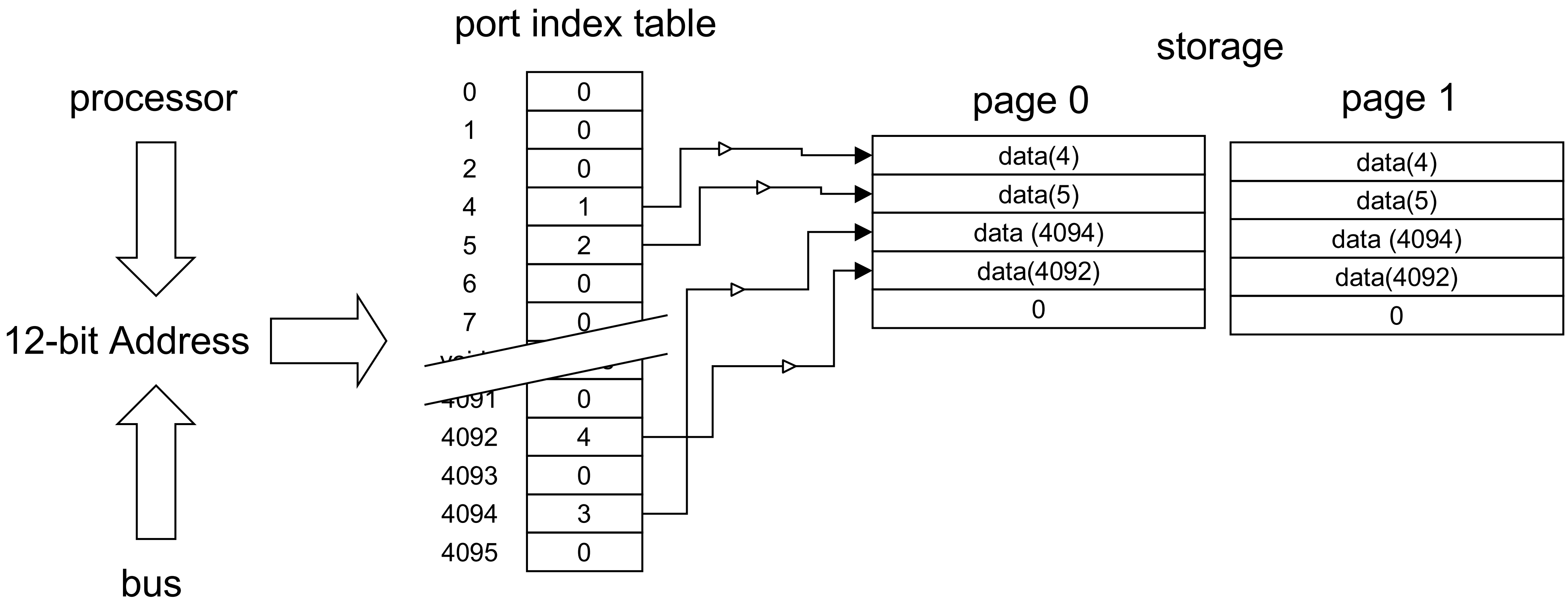
Restriction in simultaneous access



- there may be no semaphores to guard access to a traffic store (real-time)
- there may be only one writer for a port, but several readers
- a reader must read the whole port before the writer overwrites it again
- therefore, the processor must read ports with interrupt off.

Operation of the traffic memory

In content-addressed ("source-addressed") communication, messages are broadcast, the receiver select the data based on a look-up table of relevant messages.
For this, an associative memory is required.
Since address size is small (12 bits), the decoder is implemented by a memory block:



MVB F_code Summary

Master Frame			Slave Frame			
F_code	address	request	source	size	response	destination
0				16		
1			single	32		all
2	logical	Process_Data	device	64	Process_Data	devices
3			subscribed	128	(application	subscribed
4			as	256	-dependent)	as
5		reserved	source	-		sink
6		reserved		-		
7		reserved		-		
8	all devices	Master_Transfer	Master	16	Master_Transfer	Master
9	device	General_Event	>= 1devices	16	Event_Identifier	Master
10	device	reserved	-	-		
11	device	reserved	-	-		
12	device	Message_Data	single device	256	Message_Data	selected device
13	group	Group_Event	>= 1devices	16	Event_Identifier	Master
14	device	Single_Event	single device	16	Event_Identifier	Master
15	device	Device_Status	single device	16	Device_Status	Master or monitor

MVB Outline

1. Applications in vehicles
2. Physical layer
    1. Electrical RS 485
    2. Middle-Distance
    3. Fibre Optics
3. Device Classes
4. Frames and Telegrams
5. Medium Allocation
6. Clock Synchronization
7. Fault-tolerance concept
8. Integrity Concept
9. Summary

Master Operation

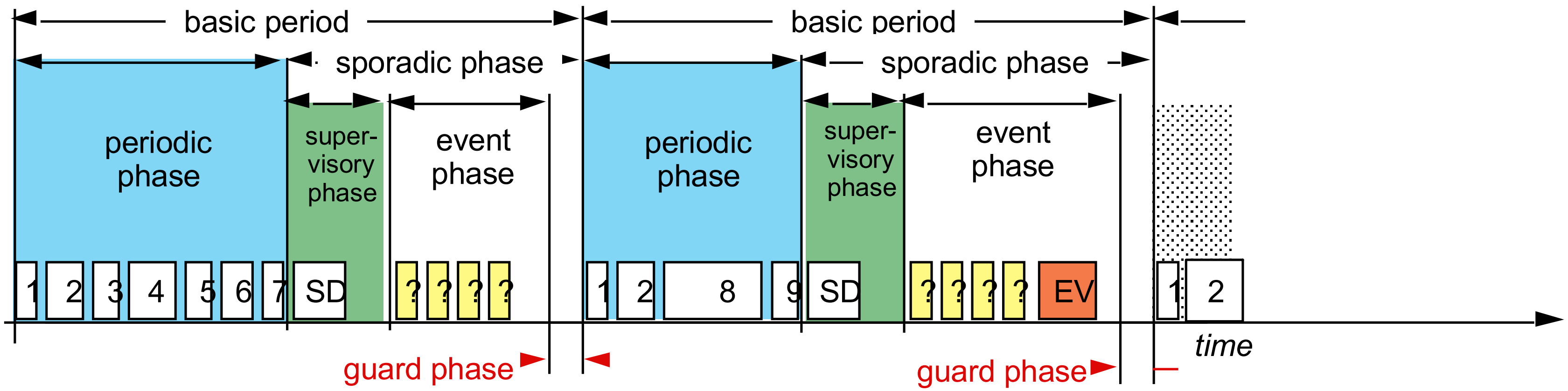
The Master performs four tasks:

1) Periodic Polling of the port addresses according to its Poll List

2) Attend Aperiodic Event Requests

3) Scan Devices to supervise configuration

4) Pass Mastership orderly (last period in turn)



The Administrator is loaded with a configuration file before becoming Master

Bus Traffic

State Variable

State of the Plant
Response in 1..200 ms

... commands, position, speed

Periodic Transmission

Spurious data losses will be compensated at the next cycle

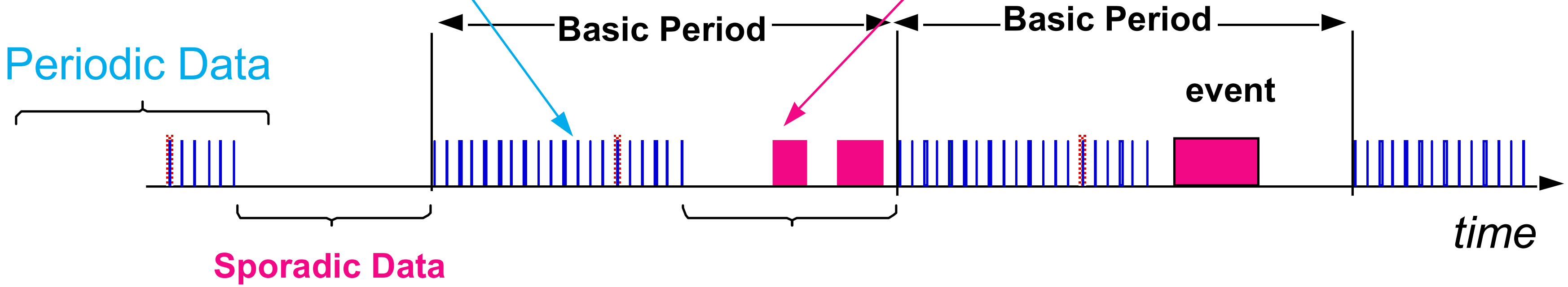
Messages

Events of the Plant
Response at human speed: > 0.5 s

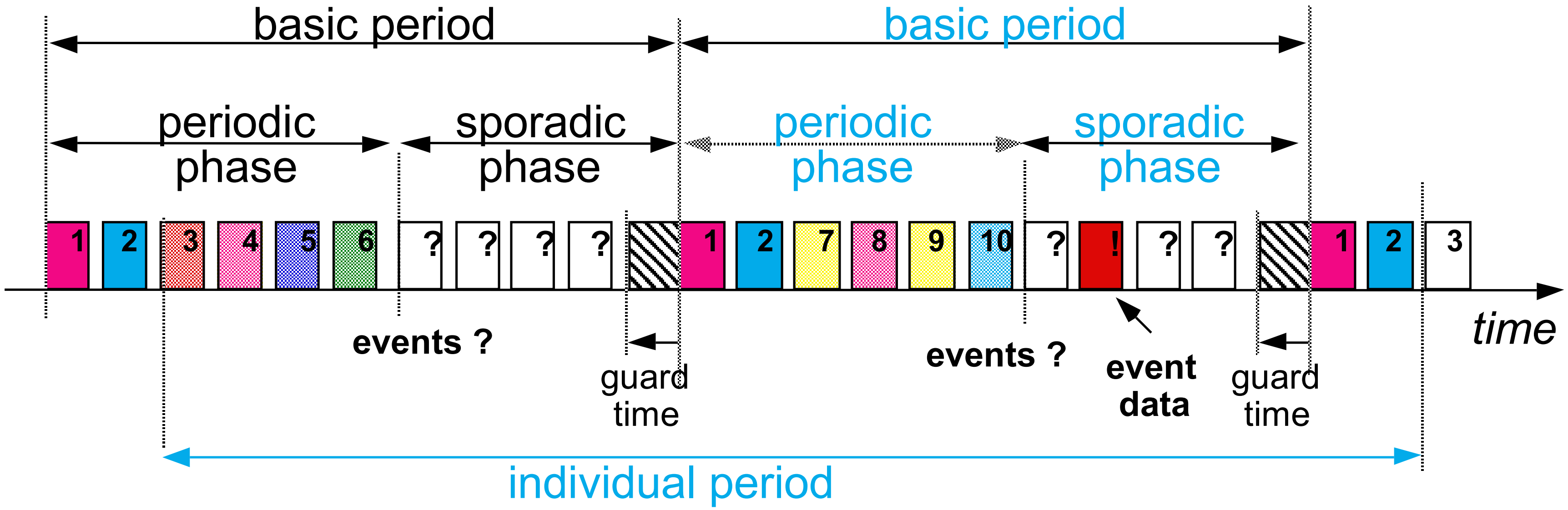
- Diagnostics, event recorder
- Initialisation, calibration

On-Demand Transmission

Flow control & error recovery protocol for catching all events



MVB Medium Access



A basic period is divided into a periodic and a sporadic phase.

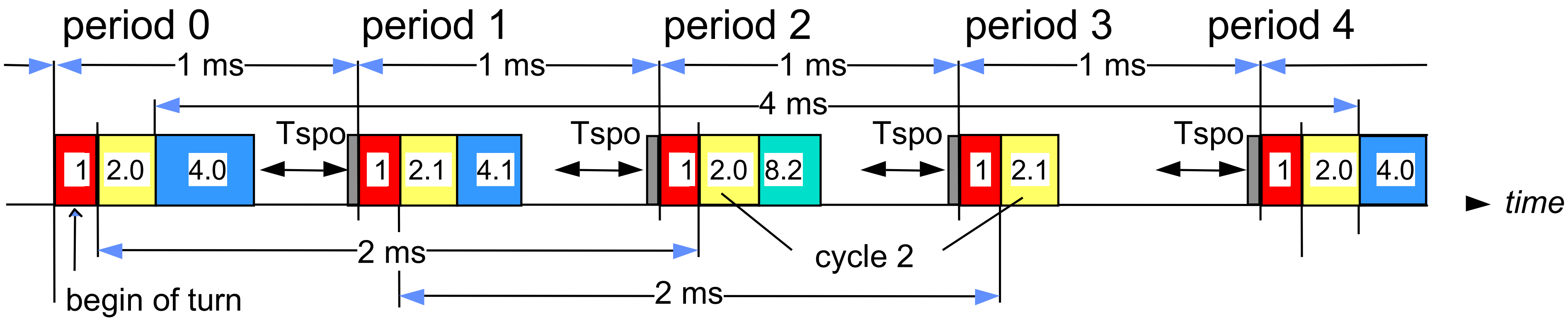
During the periodic phase, the master polls the periodic data in sequence.

Periodic data are polled at their individual period (a multiple of the basic period).

Between periodic phases, the Master continuously polls the devices for events.

Since more than one device can respond to an event poll, a resolution procedure selects exactly one event.

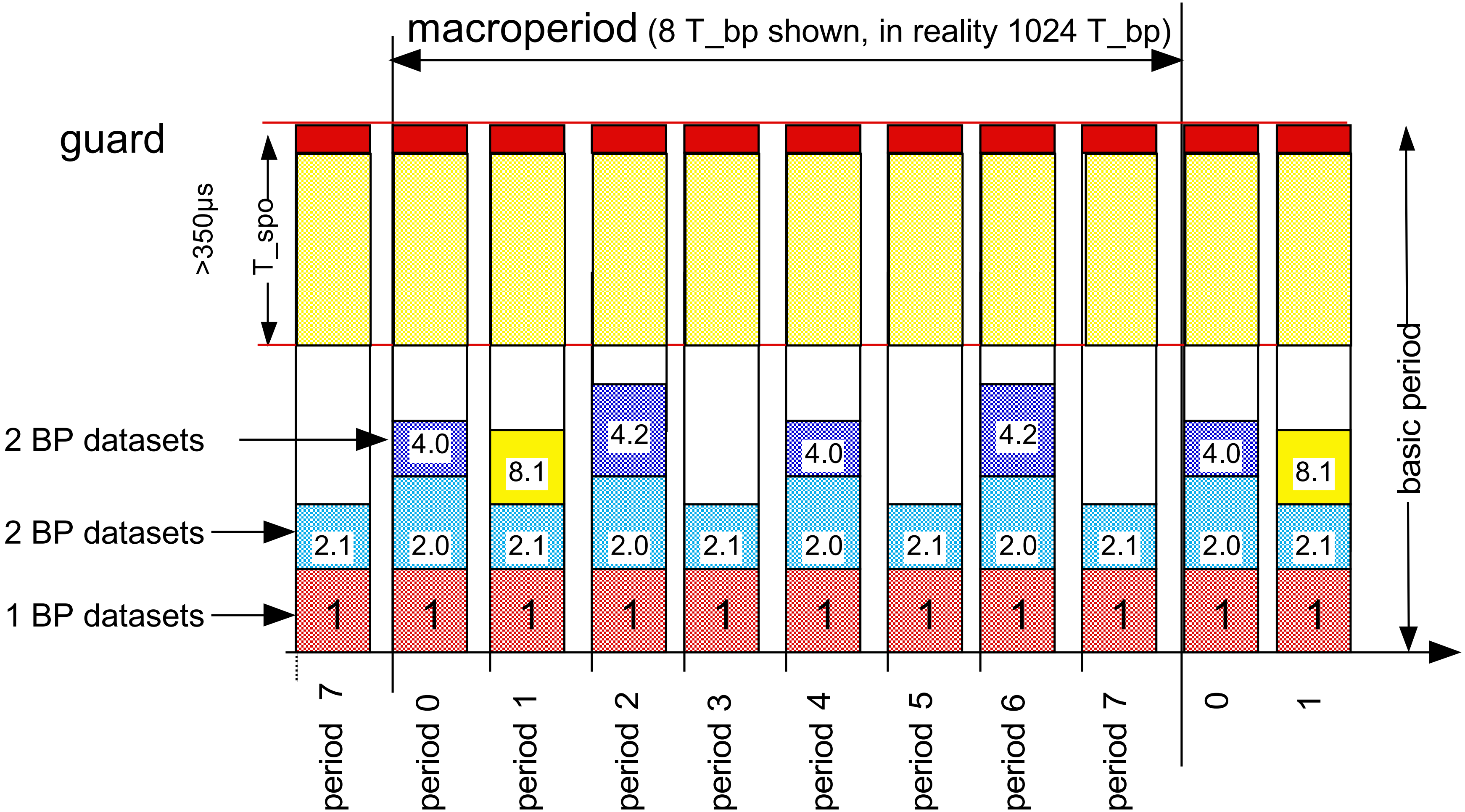
MVB Bus Administrator Configuration



The Poll List is built knowing:

- the list of the port addresses, size and individual period
- the reply delay of the bus
- the list of known devices (for the device scan
- the list of the bus administrators (for mastership transfer)

MVB Poll List Configuration



The algorithm which builds the poll table spreads the cycles evenly over the macroperiod

MVB Event Resolution (1)

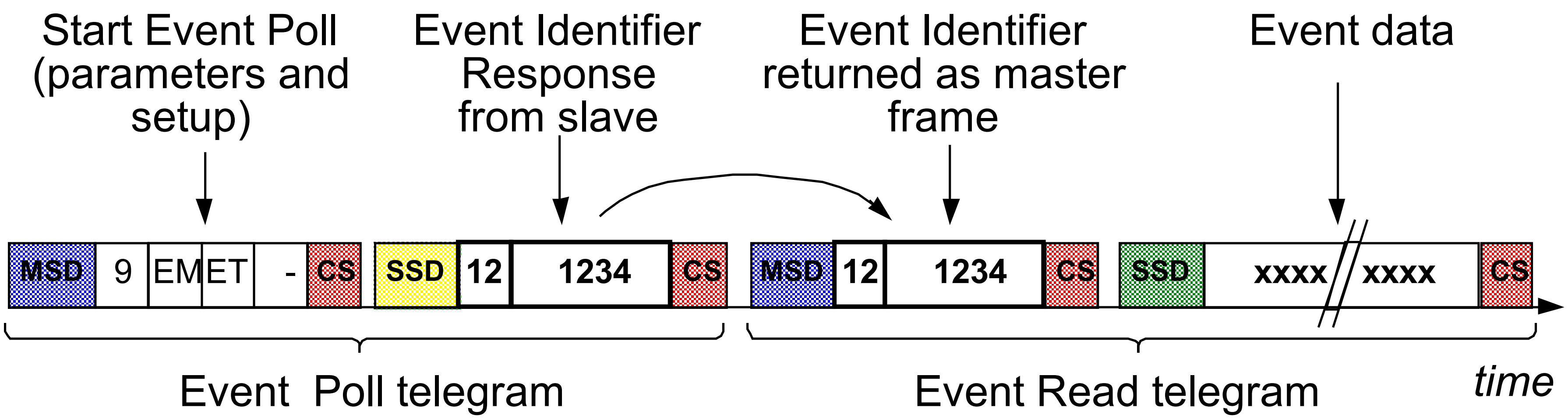
To scan events, the Master issues a General Event Poll (Start Poll) frame.

If no device responds, the Master keeps on sending Event Polls until a device responds or until the guard time before the next periodic phase begins.

A device with a pending event returns an Event Identifier Response.

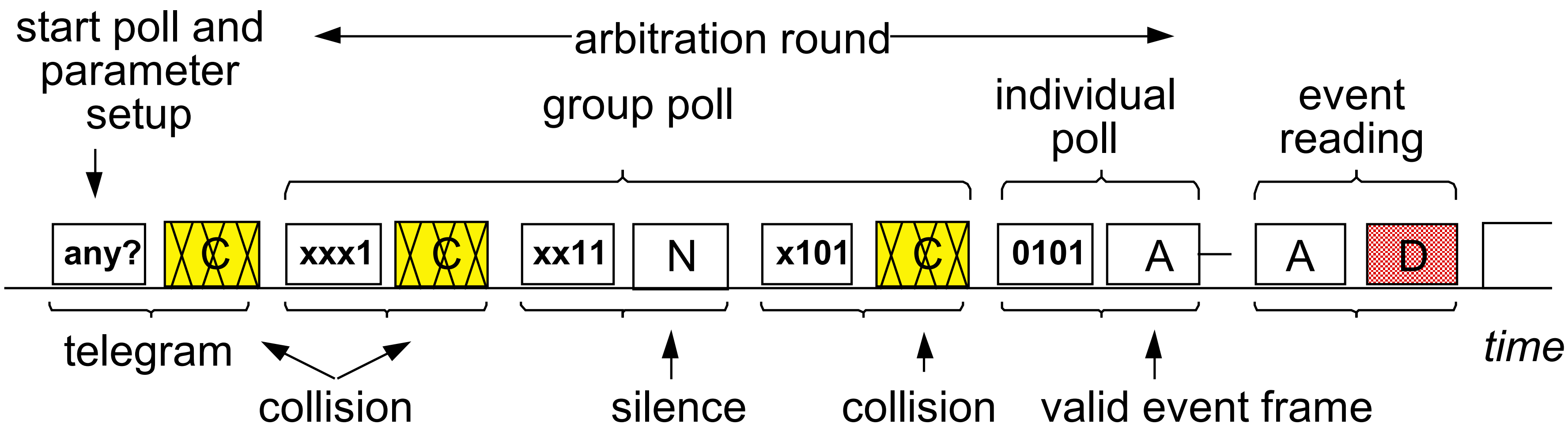
If only one device responds, the Master reads the Event Identifier (no collision).

The Master returns that frame as an Event Read frame to read the event data



MVB Event Resolution (2)

If several devices respond to an event poll, the Master detects the collision and starts event resolution

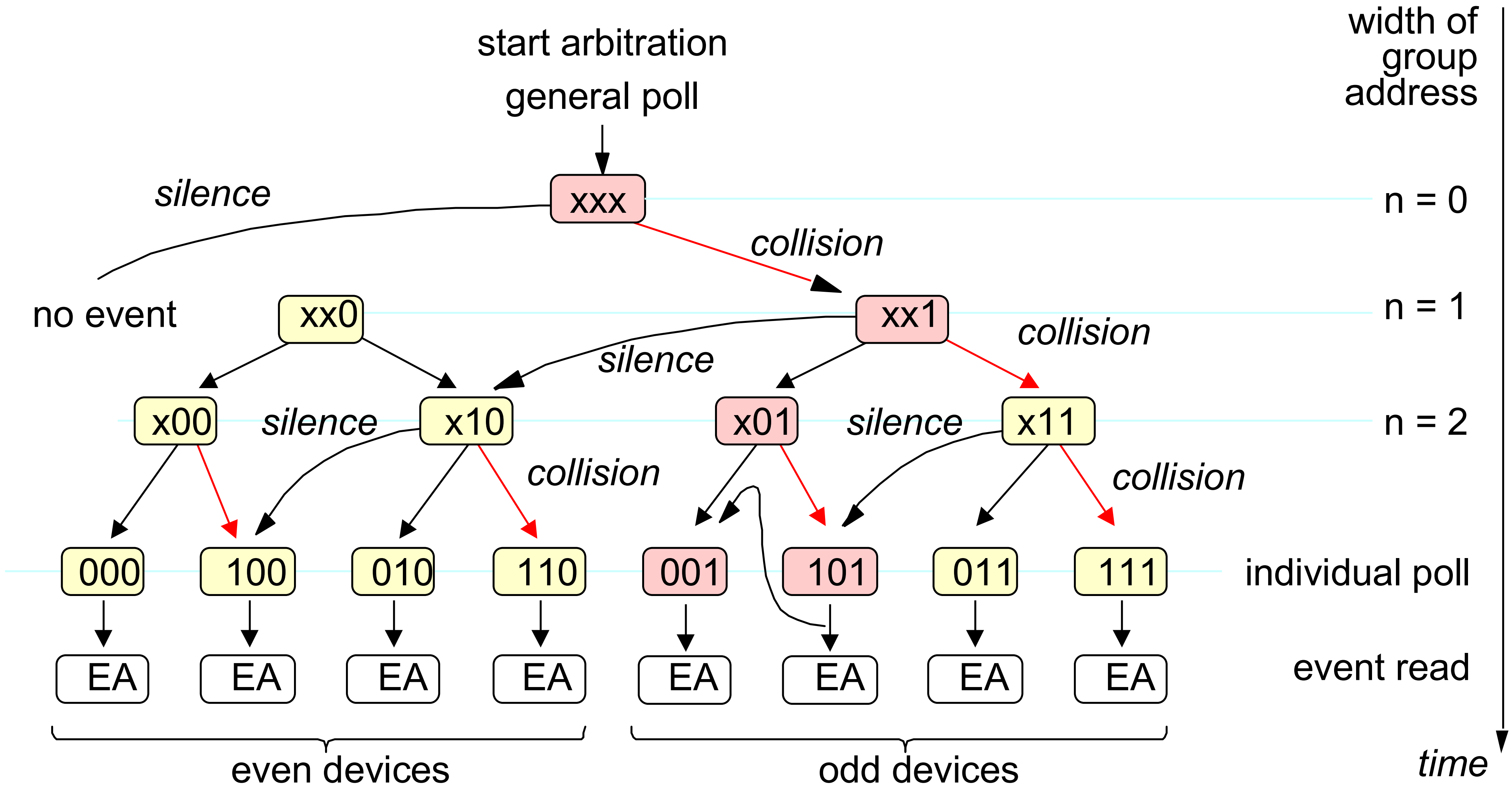


The devices are divided into groups on the base of their physical addresses. The Master first asks the devices with an odd address if they request an event.

- *If only one response comes, the master returns that frame to poll the event.*
- *If there is no response, the master asks devices with an even address.*
- *If collision keeps on, the master considers the 2nd bit of the device address.*

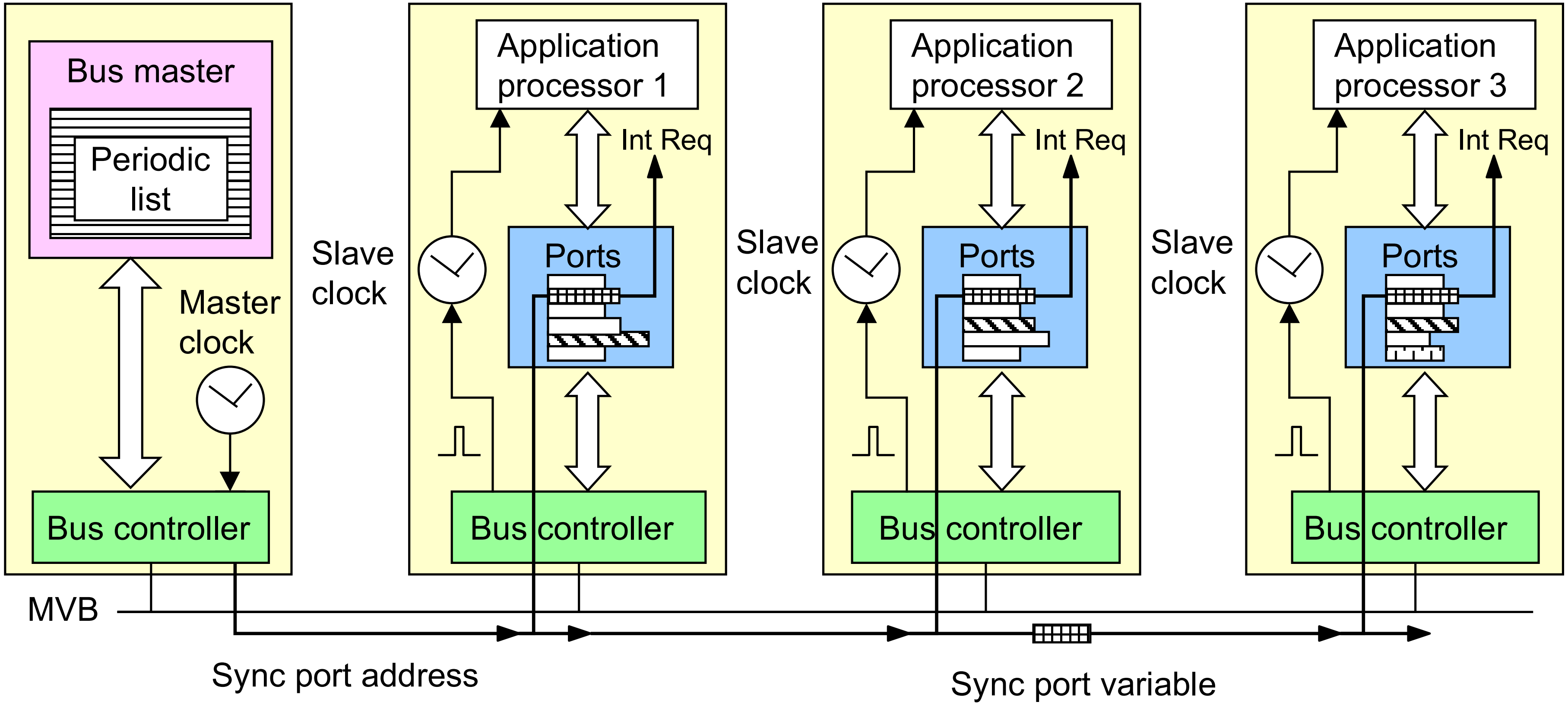
MVB Event Resolution (3)

Example with a 3-bit device address: 001 and 101 compete

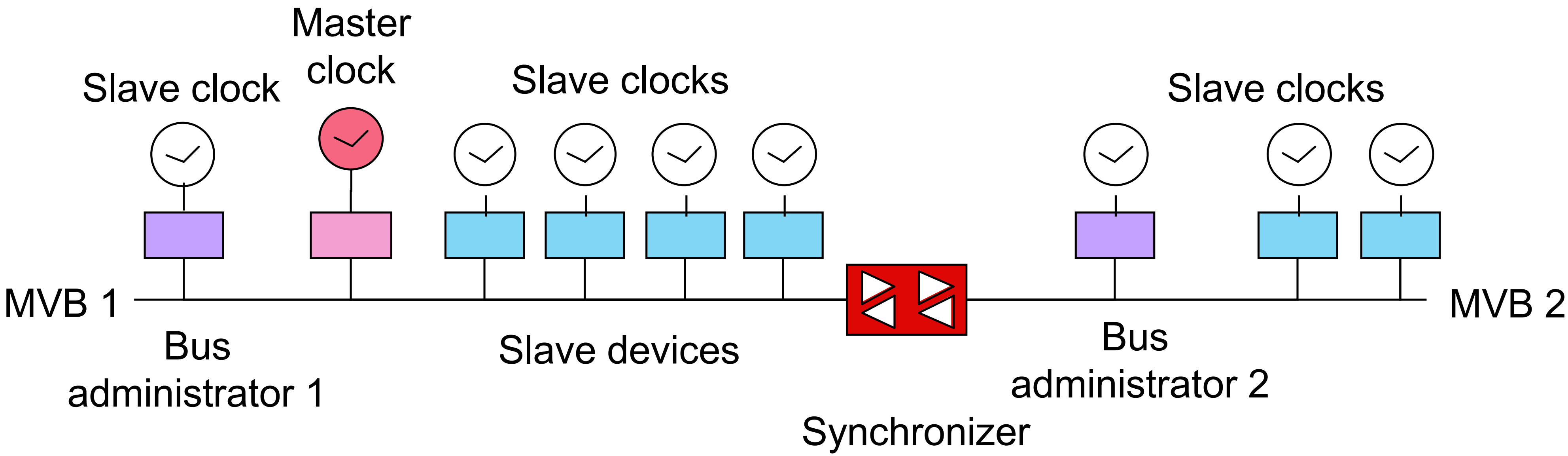


MVB Time Distribution

At fixed intervals, the Master broadcasts the exact time as a periodic variable. When receiving this variable, the bus controllers generate a pulse which can resynchronize a slave clock or generate an interrupt request.



MVB Slave Clock Synchronization



The clock does not need to be generated by the Master.

The clock can synchronize sampling within 100 µs across several bus segments.

MVB Outline

1. Applications in vehicles
2. Physical layer
   1. Electrical RS 485
   2. Middle-Distance
   3. Fibre Optics
3. Device Classes
4. Frames and Telegrams
5. Medium Allocation
6. Clock Synchronization
7. Fault-tolerance concept
8. Integrity Concept
9. Summary

MVB Fault-tolerance Concept

Transmission Integrity

MVB rather stops than provides false data.
The probability for an undetected transmission error (residual error rate) is low enough to transmit most safety-critical data.

This is achieved through an extensive error detection scheme

Transmission Availability

MVB continues operation is spite of any single device error. In particular, configurations without single point of failure are possible.

This is achieved through a complete duplication of the physical layer.

Graceful Degradation

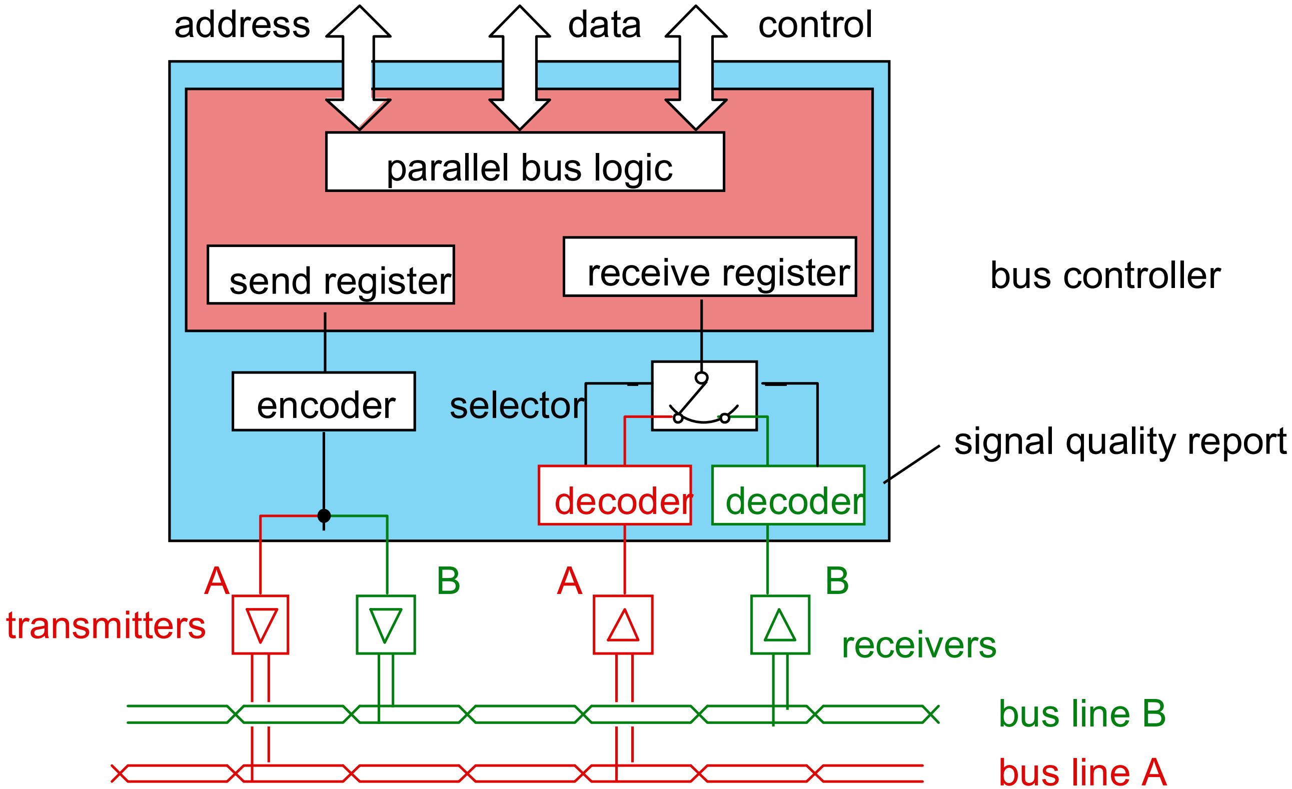
The failure of a device affects only that device, but not devices which do not depend on its data (retro-action free).

Configurability

Complete replication of the physical layer is not mandatory.
When requirements are slackened, single-thread connections may be used and mixed with dual-thread ones.

MVB Basic Medium Redundancy

The bus is duplicated for availability (not for integrity)

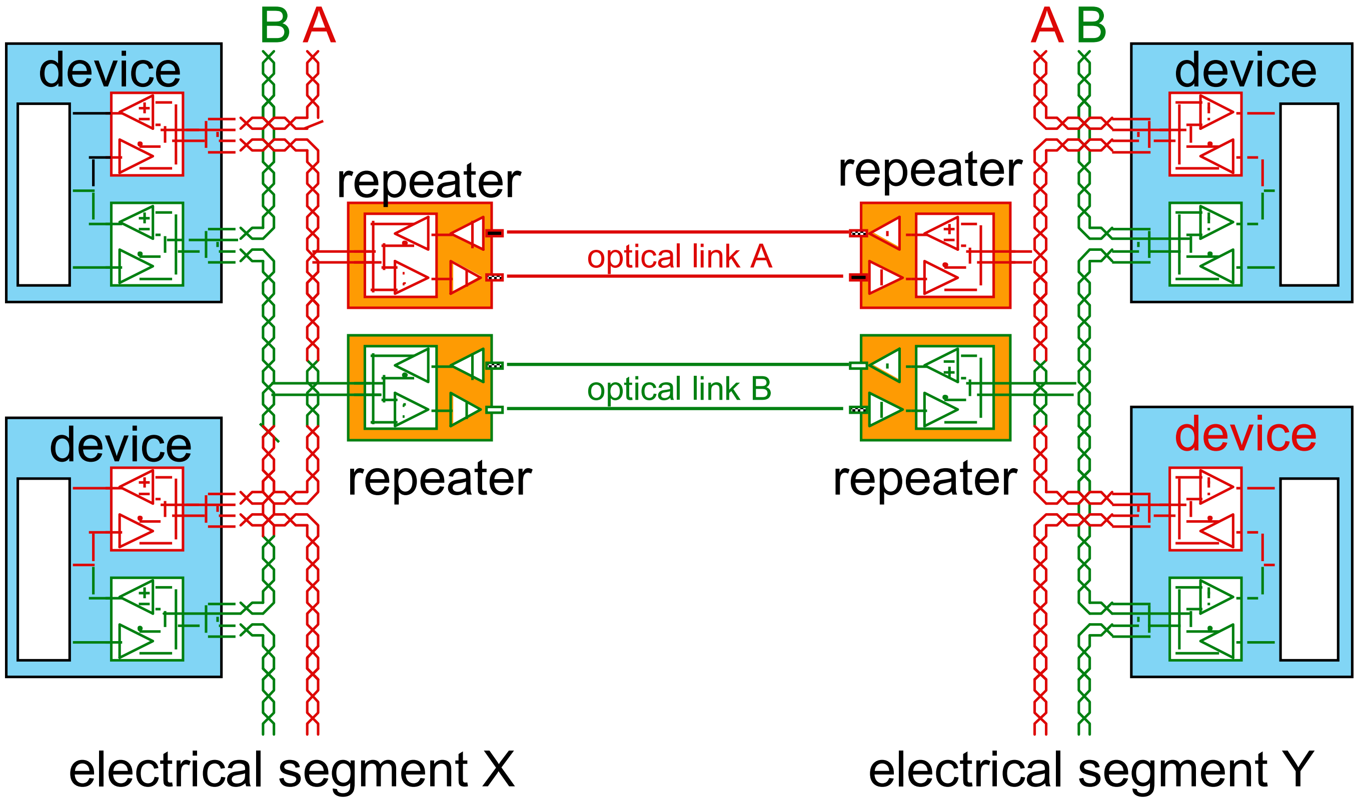


A frame is transmitted over both channels simultaneously.
The receiver receives from one channel and monitors the other.
Switchover is controlled by signal quality and frame overlap.
One frame may go lost during switchover

MVB Medium Redundancy

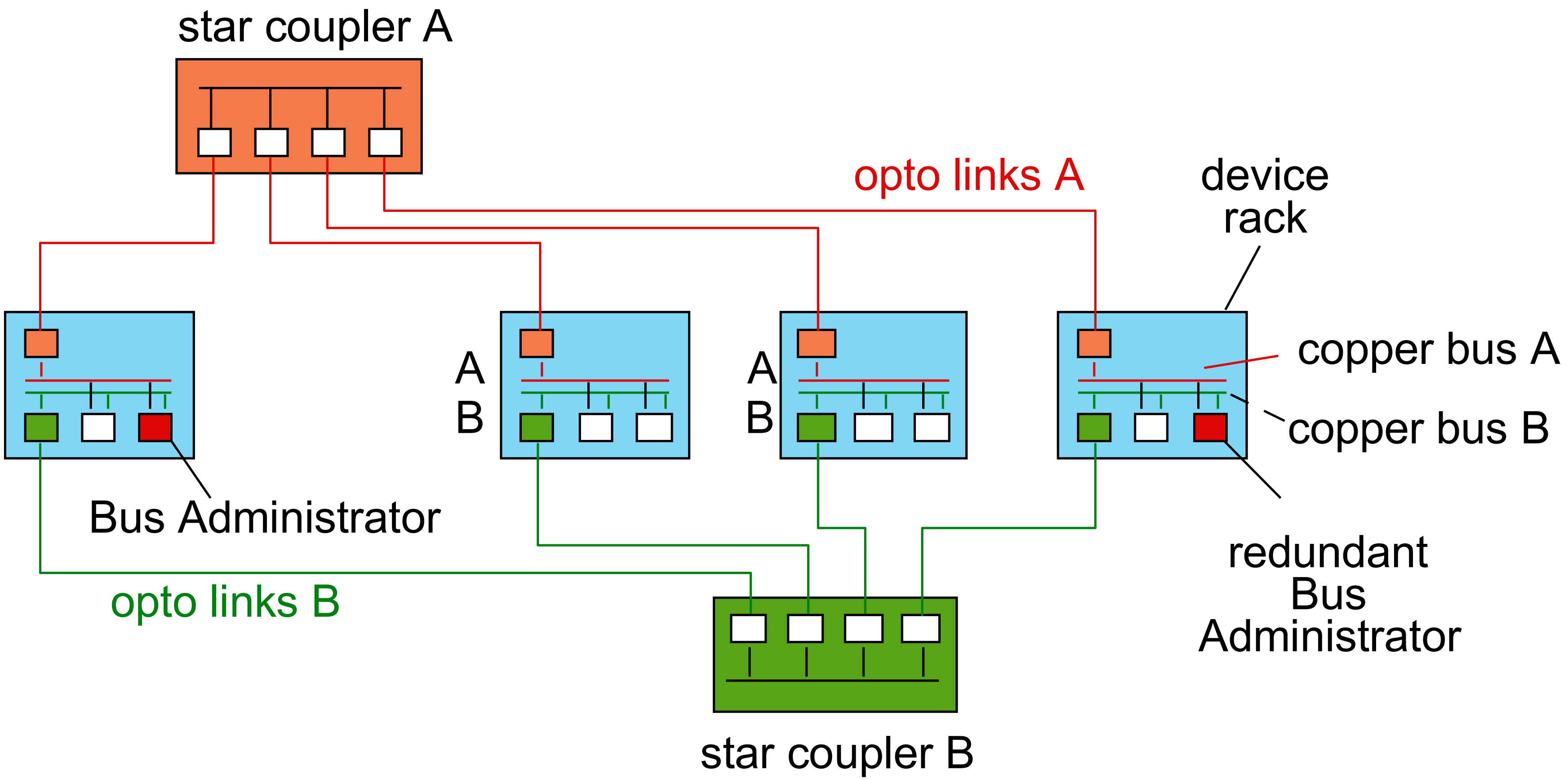
The physical medium may be fully duplicated to increase availability.

Principle: send on both, receive on one, supervise the other



Duplicated and non-duplicated segments may be connected

MVB Double-Line Fibre Layout



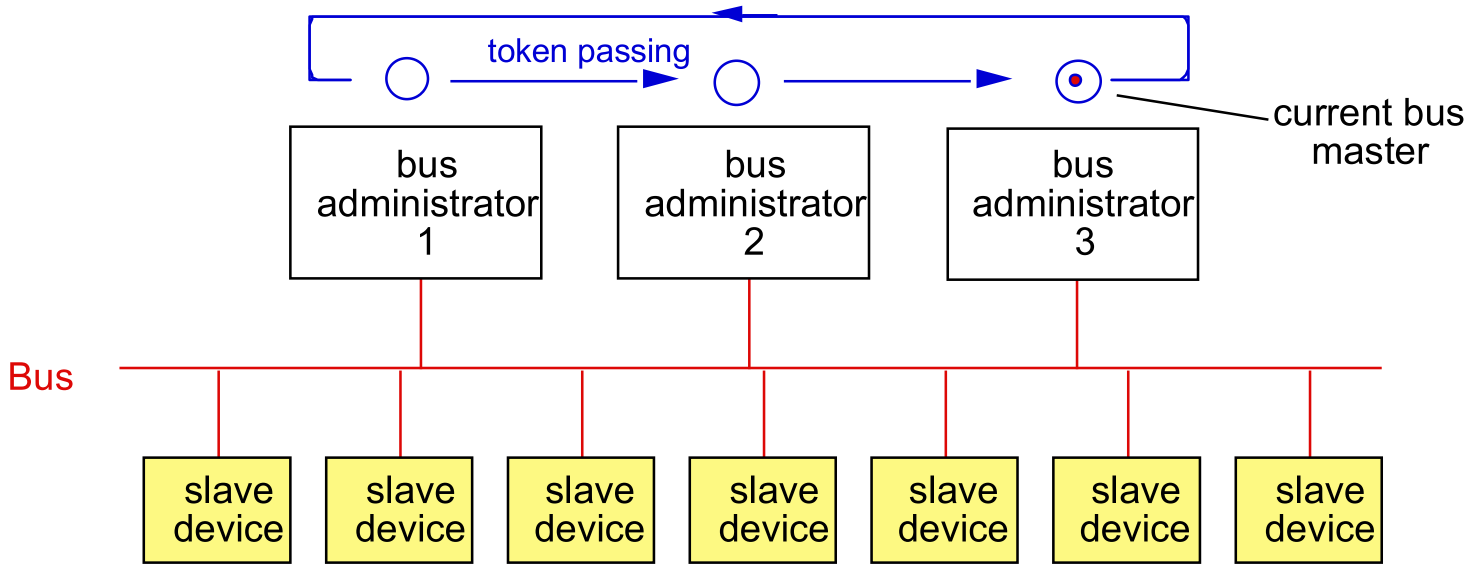
The failure of one device cannot prevent other devices from communicating.
Optical Fibres do not retro-act.

MVB Master Redundancy

A centralized bus master is a single point of failure.

To increase availability, the task of the bus master may be assumed by one of several *Bus Administrators*

The current master is selected by token passing:



If a bus administrator detects no activity, it enters an arbitration procedure. If it wins, it takes over the master's role and creates a token.

To check the good function of all administrators, the current master offers mastership to the next administrator in the list every 4 seconds.

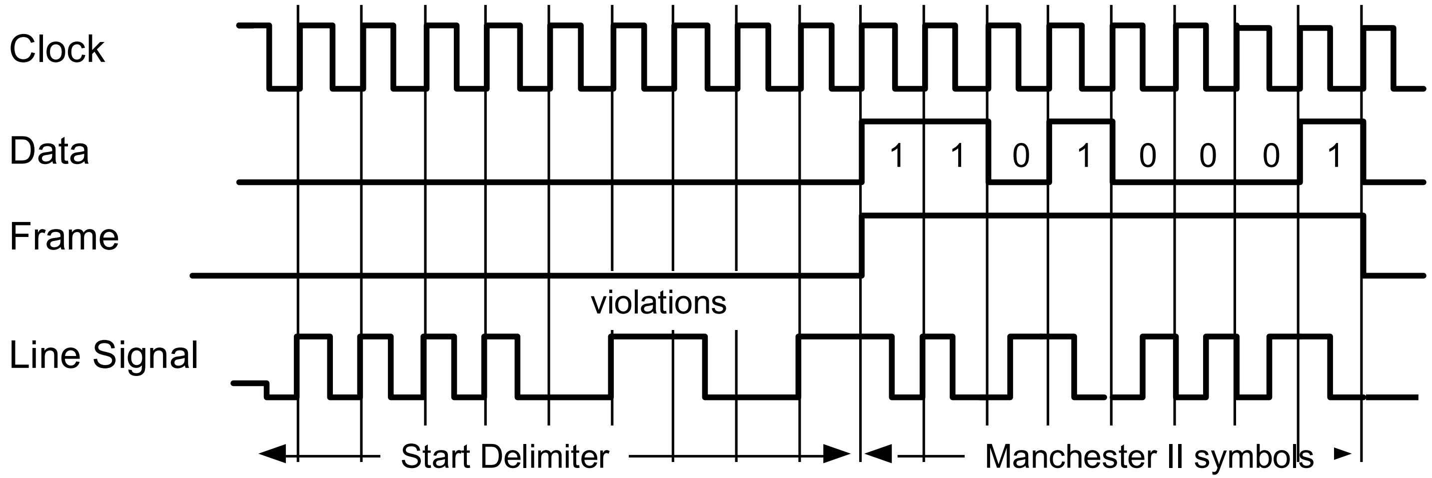
MVB Outline

1. Applications in vehicles
2. Physical layer
   1. Electrical RS 485
   2. Middle-Distance
   3. Fibre Optics
3. Device Classes
4. Frames and Telegrams
5. Medium Allocation
6. Clock Synchronization
7. Fault-tolerance concept
8. Integrity Concept
9. Summary

MVB Transmission Integrity (1)

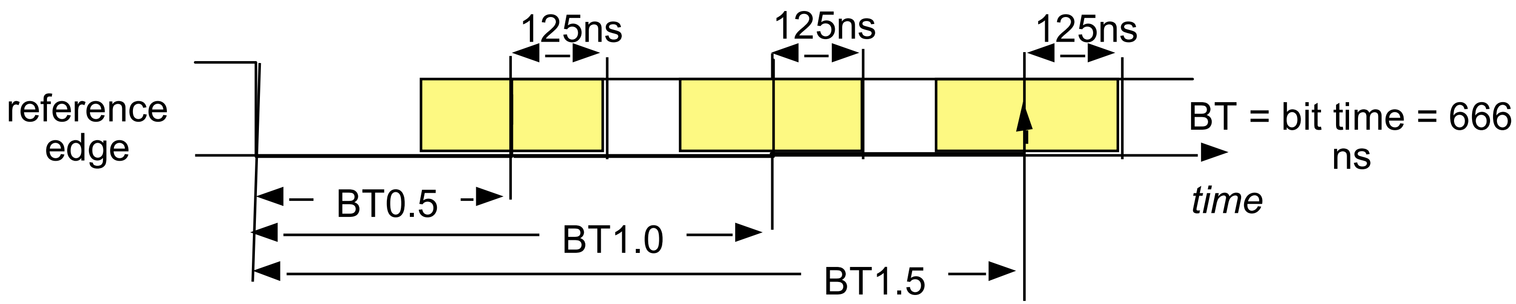
1) Manchester II encoding

Double signal inversion necessary to cause an undetected error, memoryless code



2) Signal quality supervision

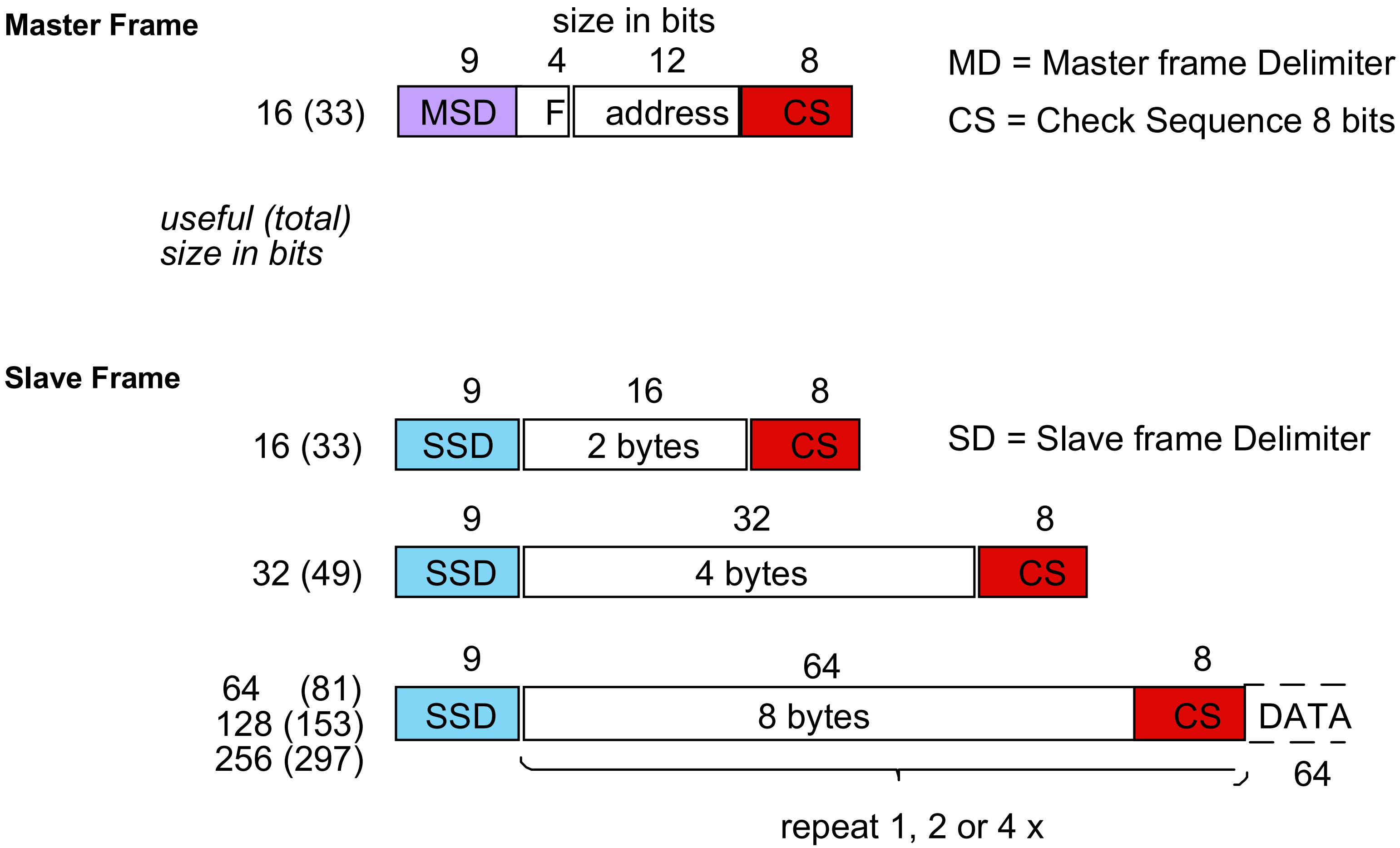
Adding to the high signal-to-noise ratio of the transmission, signal quality supervision rejects suspect frames.



MVB Transmission Integrity (2)

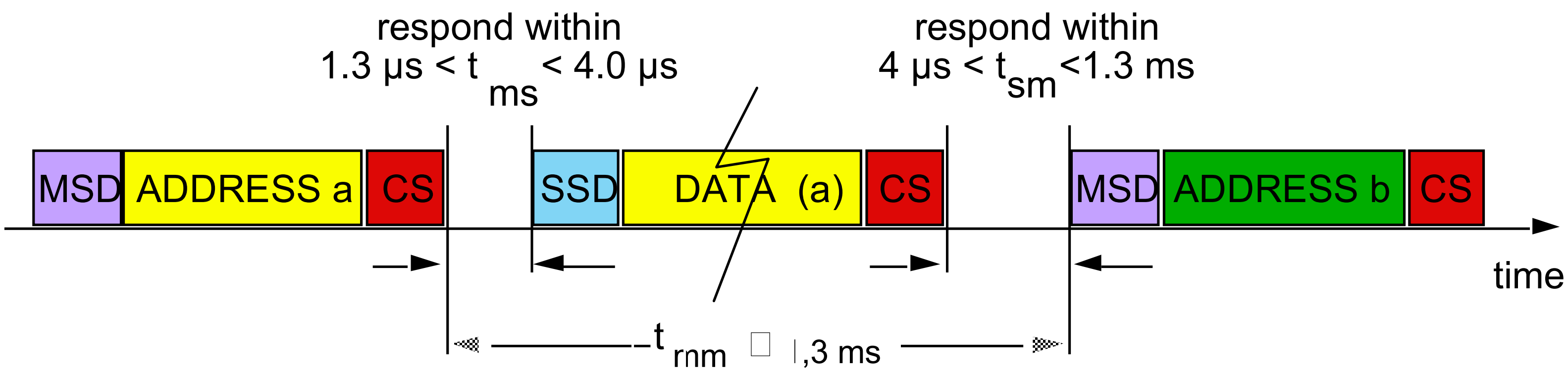
3) A check octet according to TC57 class FT2 for each group of up to 64 bits, provides a Hamming Distance of 4 (8 if Manchester coding is considered):

(Residual Error Rate < 10^{-15} under standard disturbances)

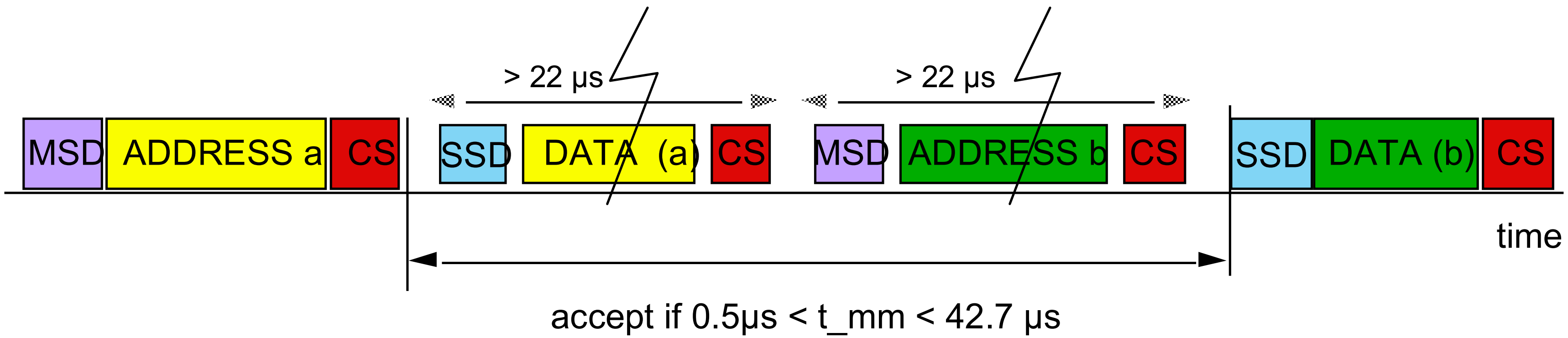


MVB Transmission Integrity (3)

4) Different delimiters for address and data against single frame loss:



5) Response time supervision against double frame loss:



6) Configuration check: size at source and sink ports must be same as frame size.

MVB Safety Concept

Data Integrity

Very high data integrity, but nevertheless insufficient for safety applications (signalling)
Increasing the Hamming Distance further is of no use since data falsification becomes more likely in a device than on the bus.

Data Transfer

- critical data transmitted periodically to guarantee timely delivery.
- obsolete data are discarded by sink time supervision.
- error in the poll scan list do not affect safety.

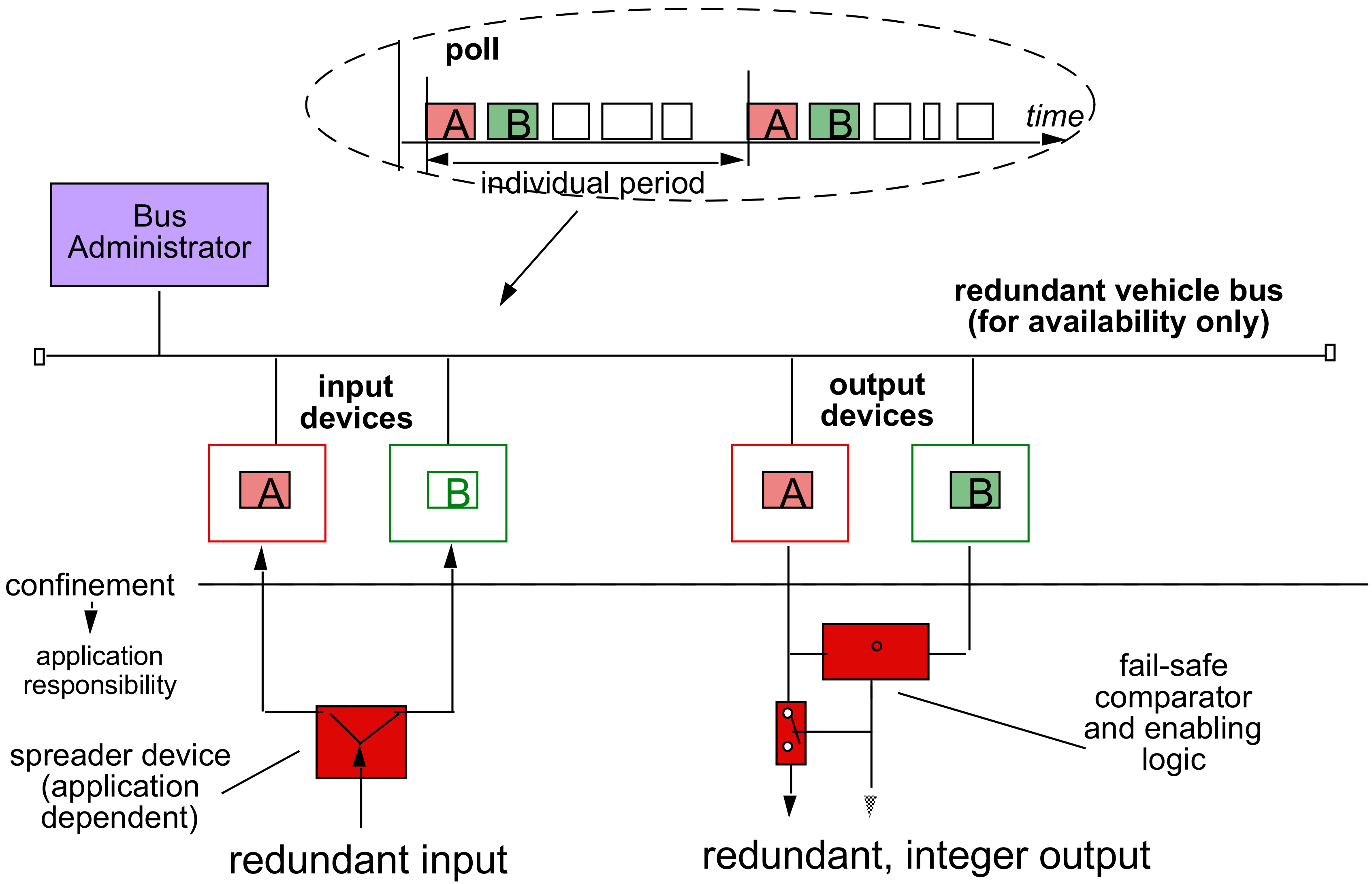
Device Redundancy

Redundant plant inputs A and B transmitted by two independent devices.
Diverse A and B data received by two independent devices and compared.
The output is disabled if A and B do not agree within a specified time.

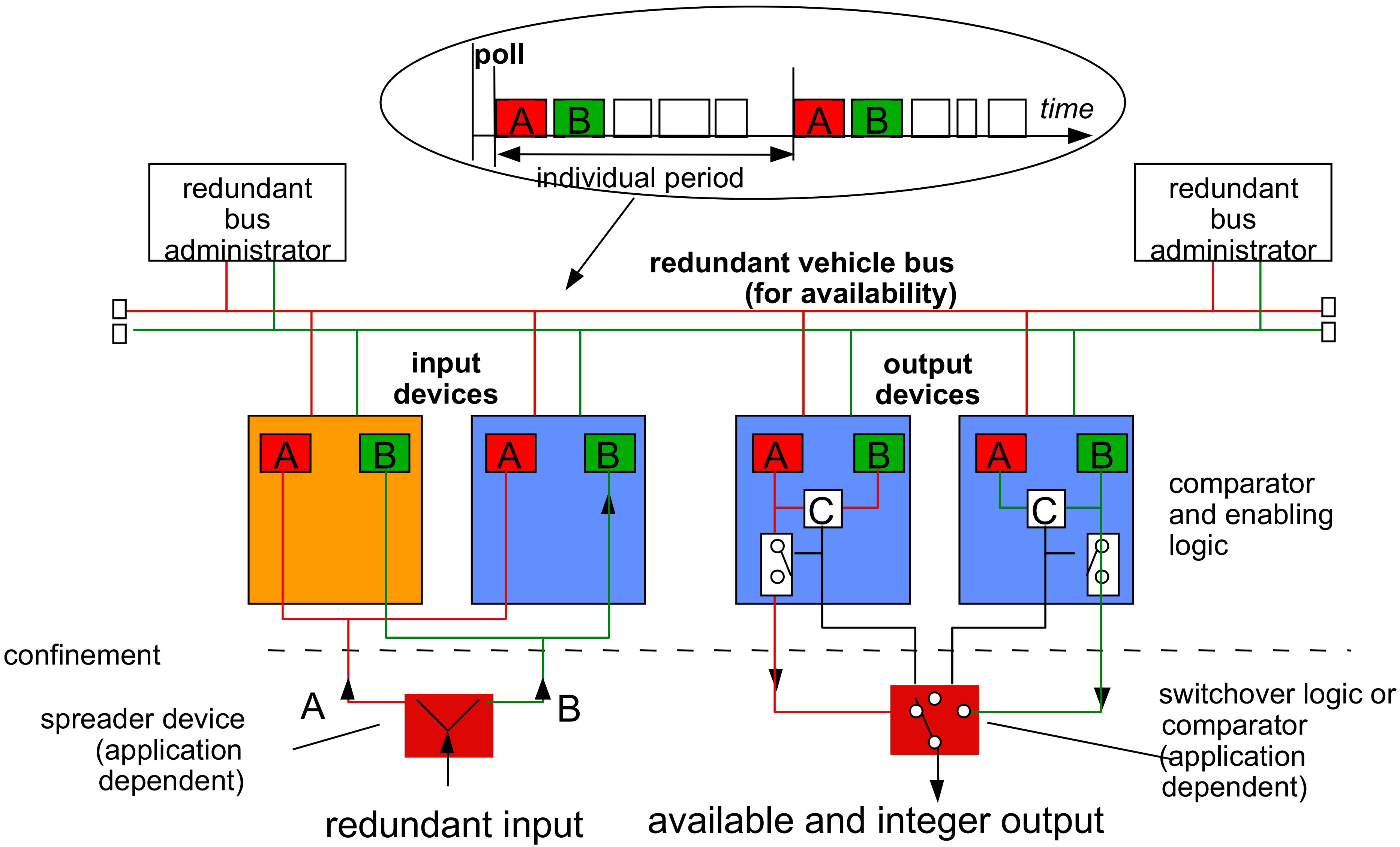
Availability

Availability is increased by letting the receiving devices receive both A and B. The application is responsible to process the results and switchover to the healthy device in case of discrepancy.

MVB Integer Set-up



MVB Integer and Available Set-up



MVB Outline

1. Applications in vehicles

2. Physical layer

    1. Electrical RS 485
    2. Middle-Distance
    3. Fibre Optics

3. Device Classes

4. Frames and Telegrams

5. Medium Allocation

6. Clock Synchronization

7. Fault-tolerance concept

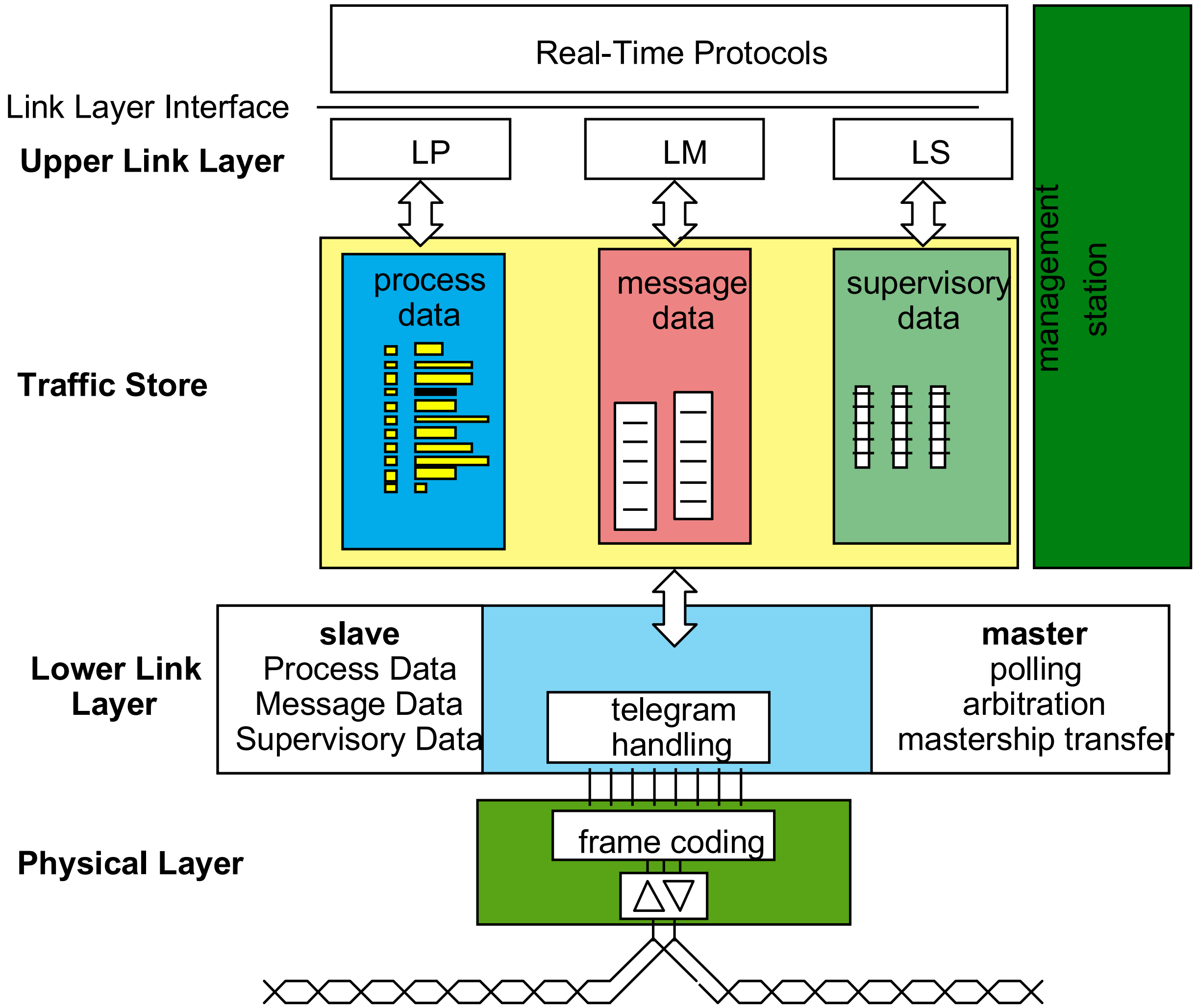
8. Integrity Concept

9. Summary

MVB Summary

Topography:	bus (copper), active star (optical fibre)
Medium:	copper: twisted wire pair optical: fibres and active star coupler
Covered distance:	OGF: 2000 m, total 4096 devices EMD: 200 m copper with transformer-coupling ESD: 20 m copper (RS485)
Communication chip	dedicated IC available
Processor participation	none (class 1), class 2 uses minor processor capacity
Interface area on board	20 cm2 (class 1), 50 cm2 (class 2)
Additional logic	RAM, EPROM , drivers.
Medium redundancy:	fully duplicated for availability
Signalling:	Manchester II + delimiters
Gross data rate	1,5 Mb/s
Response Time	typical 10 µs (<43 µs)
Address space	4096 physical devices, 4096 logical ports per bus
Frame size (useful data)	16, 32, 64, 128, 256 bits
Integrity	CRC8 per 64 bits, HD = 8, protected against sync slip

MVB Link Layer Interface



MVB Components

Bus Controllers:

BAP 15 (Texas Instruments, obsolete)

MVBC01 (VLSI, in production, includes master logic

MVBC02 (E2S, in production, includes transformer coupling)

Repeaters:

REGA (in production)

MVBD (in production, includes transformer coupling)

Medium Attachment Unit:

OGF: fully operational and field tested (8 years experience)

ESD: fully operational and field tested (with DC/DC/opto galvanic separation)

EMD: lab tested, first vehicles equipped

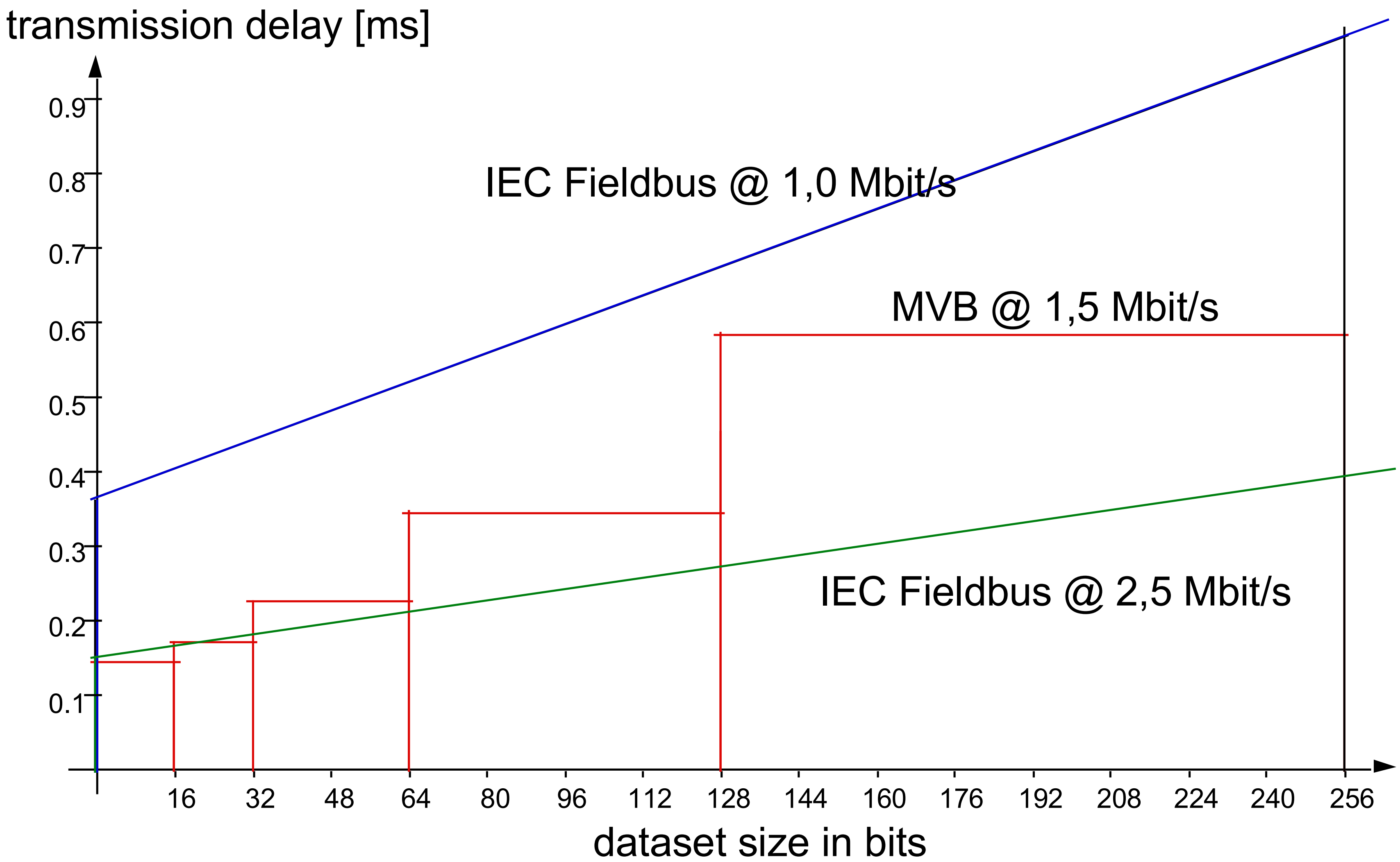
Stack:

Link Layer stack for Intel 186, i196, i960, 166, 167, Motorola 68332, under DOS, Windows, VRTX,...

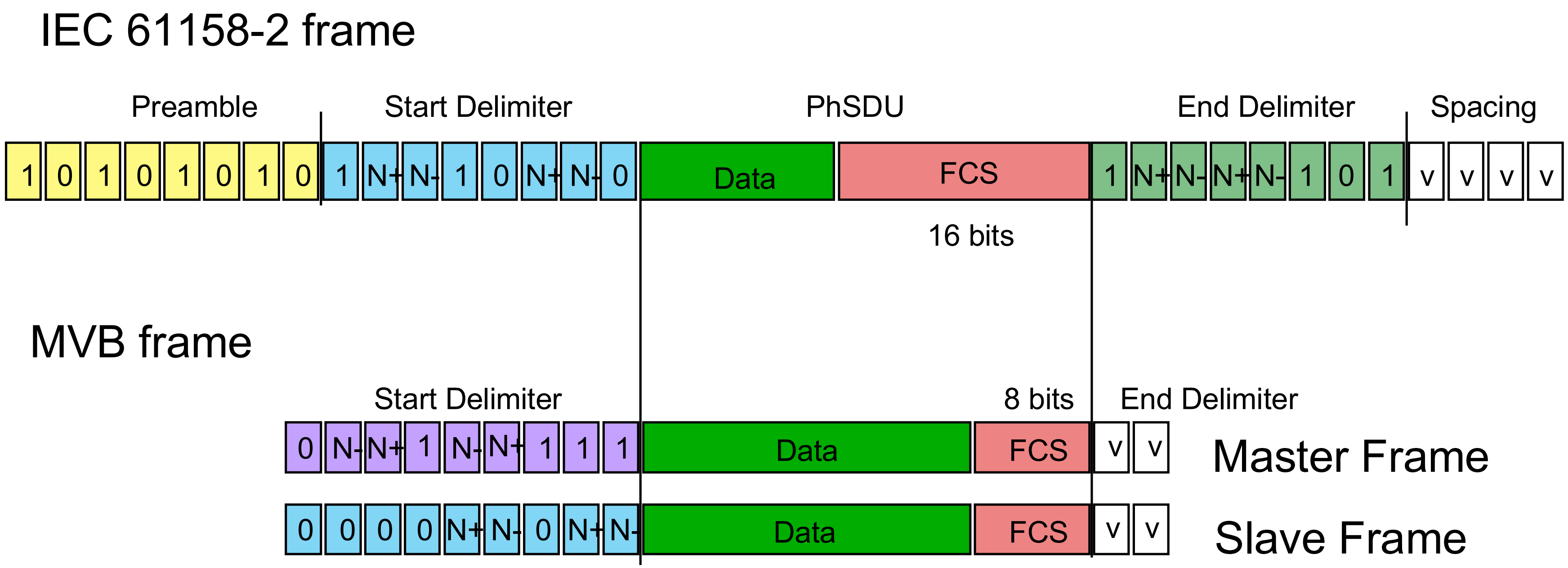
Tools:

Bus Administrator configurator
Bus Monitor, Download, Upload, remote settings

MVB Throughput (raw data)



MVB & IEC 61158-2 Frames



IEC65 frames have a lesser efficiency (-48%) then MVB frames

To compensate it, a higher speed (2,5 Mbit/s) would be needed.