II Jornadas sobre Tecnologías y soluciones para la Automatización Industrial

DOCUMENTACIÓN ANEXA A LA CONFERENCIA

Transprent Ready :

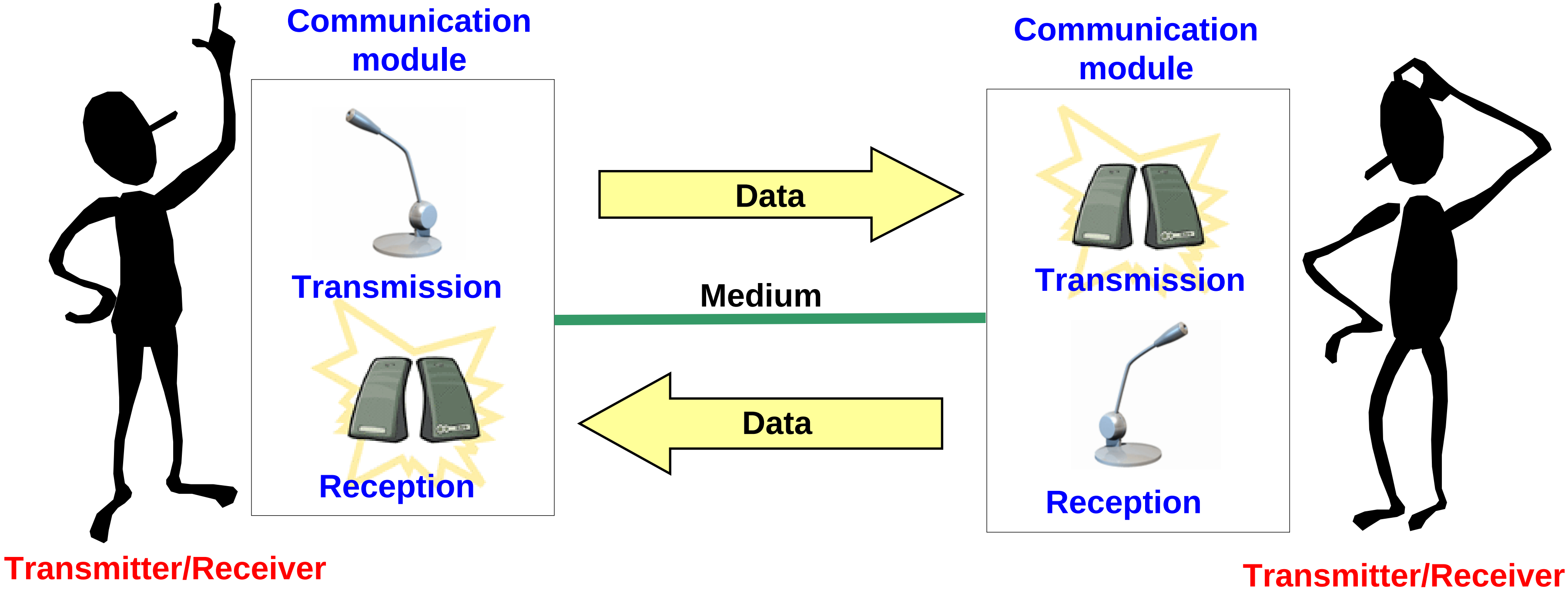
COMUNICACIONES INDUSTRIALES PARA LA INTEGRACIÓN DE DISPOSITIVOS Y SISTEMAS ABIERTOS.

Introduction to industrial communication networks

Introduction to industrial communication networks

Introduction to industrial communication networks

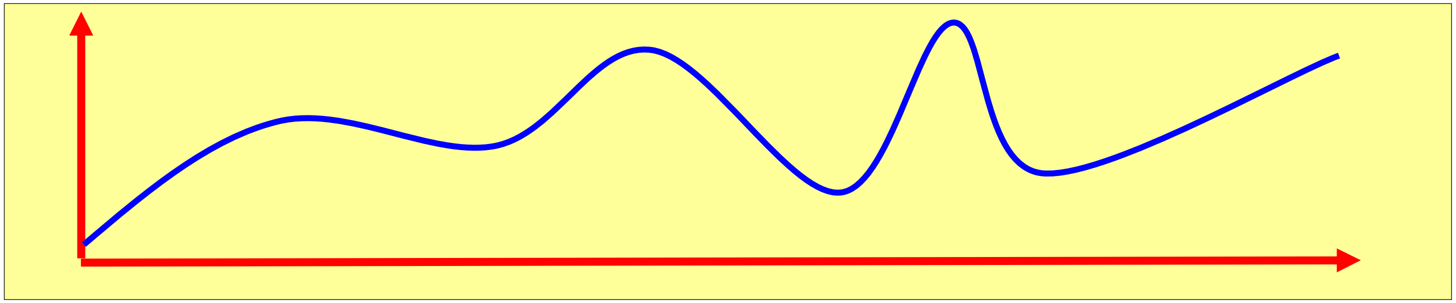
Elements used during communication



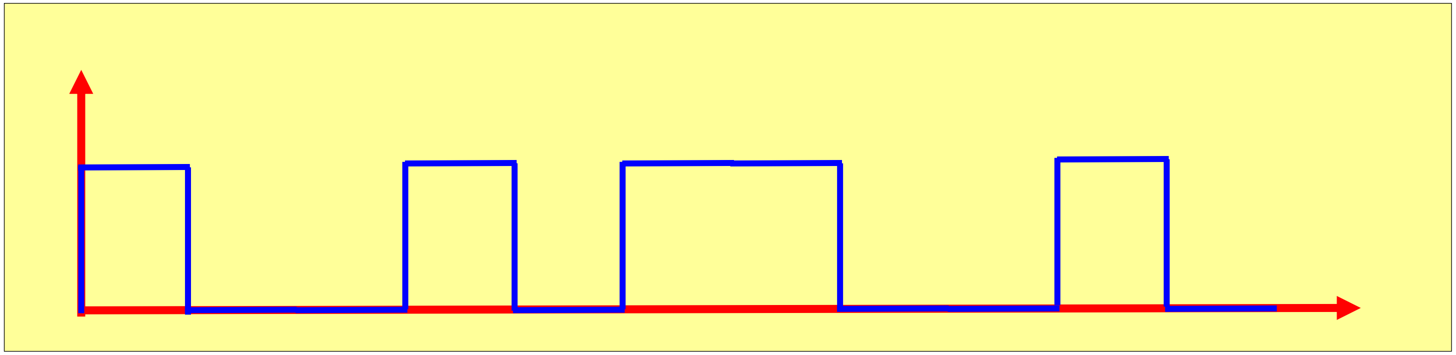
The data comprises physical elements (light, sound, images, electrical voltage, etc.) to which a direction has been attributed.

Transmission methods

Data can be transmitted in **analog** format:
Continuous progression of value



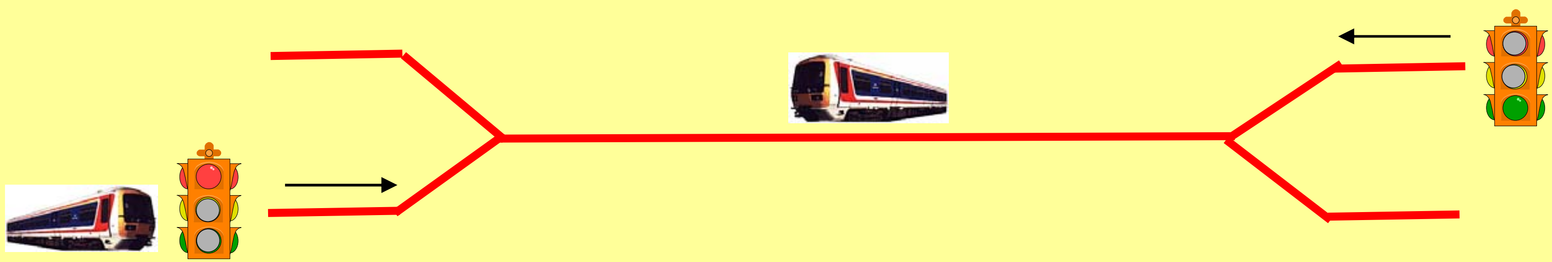
Or in **digital** format:
Discontinuous progression of value (sampling)



Transmission types

Simplex transmission: Unidirectional

Half duplex transmission: Alternate bidirectional



Full duplex transmission: Simultaneous bidirectional

Transmission types

Serial transmission:

The link usually requires 3 wires: send, receive and earth.

The bits in a byte are transmitted one after the other.

Parallel transmission:

The bits in a byte are transmitted simultaneously.

Used for short distances. As each channel tends to cause interference on neighbouring channels, the quality of the signal deteriorates rapidly.

Serial transmission types

Synchronous serial transmission:

Data is transmitted continuously.

A synchronization signal is transmitted in parallel with the data signals.

Asynchronous serial transmission:

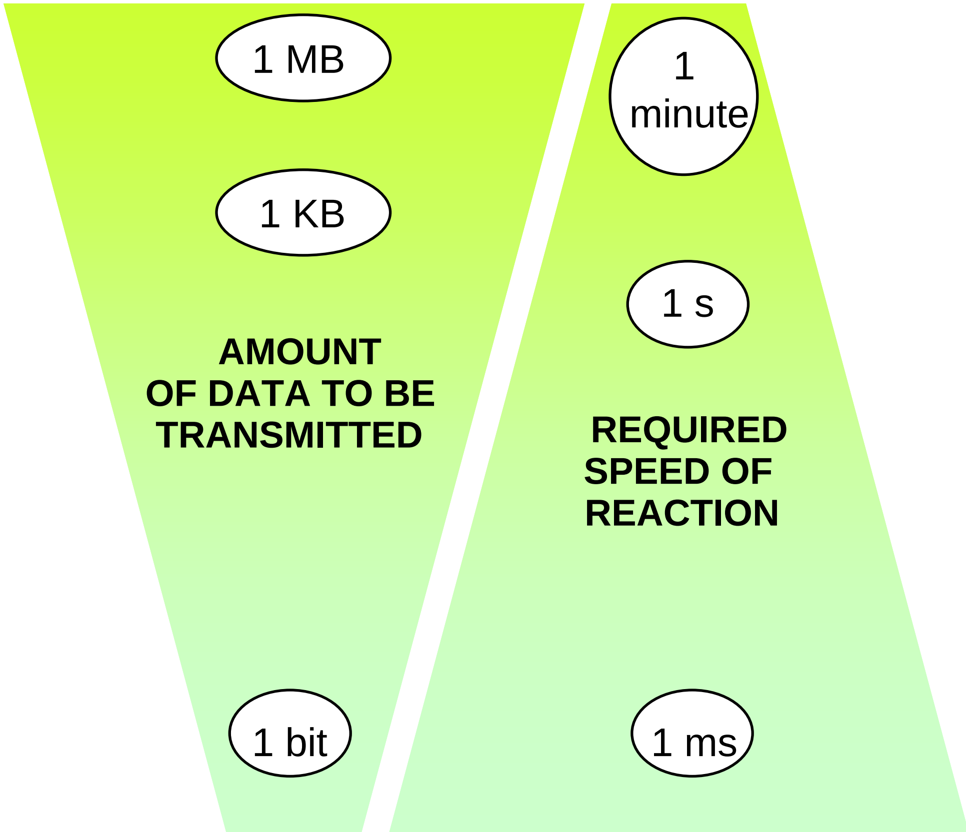
Data can be transmitted in an irregular fashion, although the interval between 2 bits is fixed.

Synchronization bits (START, STOP) encapsulate the data.

Industrial communication networks

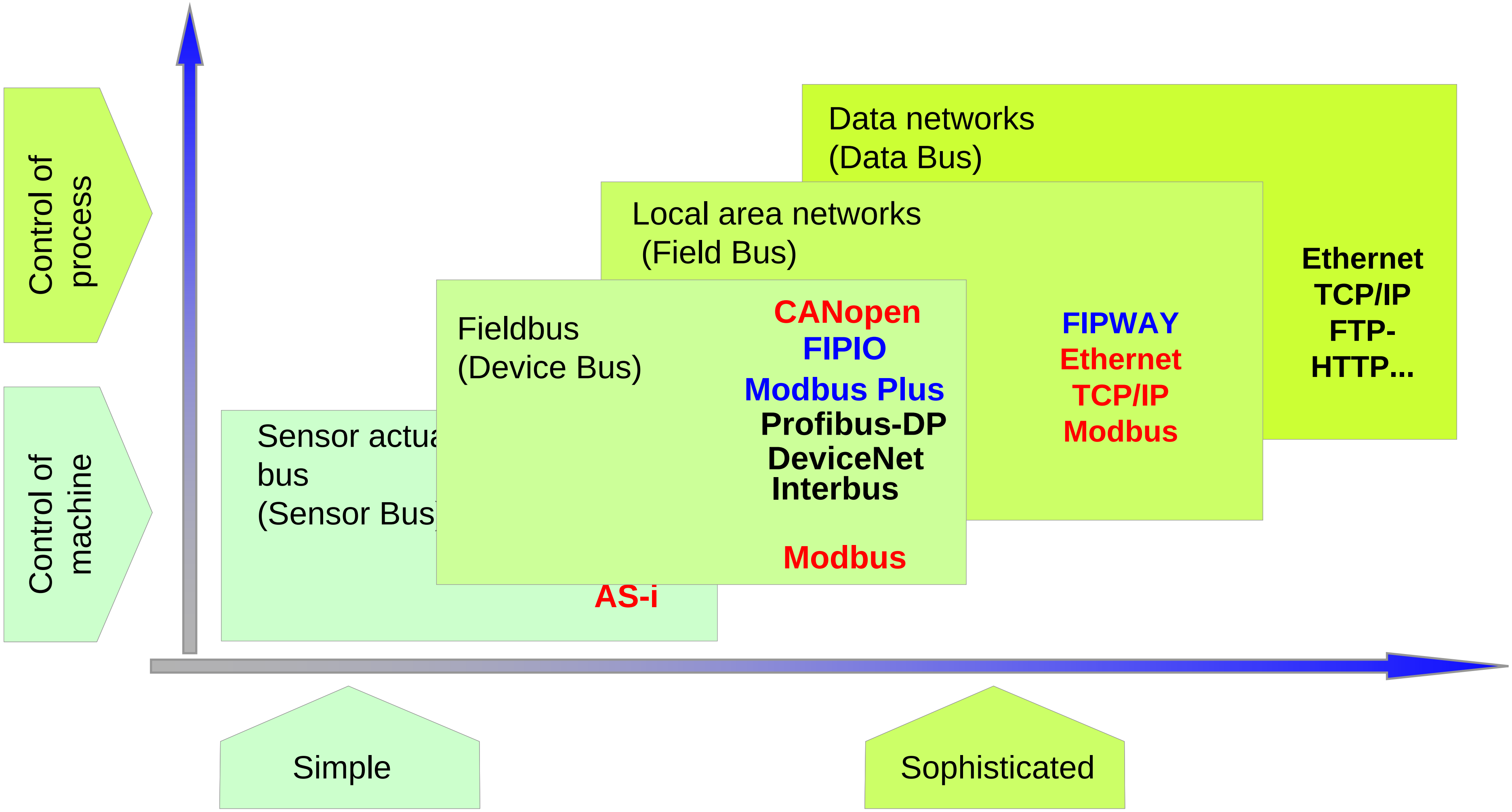
For reasons of cost and durability, most communication networks use **half duplex asynchronous serial digital transmission.**

Communication requirements



Level 3 Company	Information system
Level 2 Workshop	Production management Supervision
Level 1 Machines	Control system
Level 0 Sensors Actuators	Components

Main networks and buses



Network strategy of the Schneider industrial sector

■ Core networks:

Ethernet TCP/IP & Modbus

Levels 2 and 3: Information and control system (inter-PLC)

to be extended to fieldbus level (level 1)

CANopen

Like an internal device and panel bus (e.g.: Automation Island)

ASi

For the connection of sensors/actuators (level 0)

Modbus RS 485

When Ethernet is not suitable (price, topology, etc.)

Network strategy of the Schneider industrial sector

Legacy networks

FIPIO, Modbus Plus, Uni-Telway, Seriplex

Connectivity networks

A pragmatic approach when the market imposes a solution

DeviceNet (Allen-Bradley) **- Profibus** (Siemens) **- Interbus** (Phoenix) **etc.**

ISO model

ISO = International Organization for Standardization

STATION

Modbus or Unite...	APPLICATION LAYER	7	Network administration (starting and stopping the network, message handling)
	PRESENTATION LAYER	6	Entity used for PC/MAC dialogue
	SESSION LAYER	5	Organize and synchronize the exchanges between users of the network
Network concept Example: TCP/IP	TRANSPORT LAYER	4	End-to-end checking: restart on errors which have been signalled or otherwise by the network layer
Network concept Example: TCP/IP	NETWORK LAYER	3	Switching in a mesh network: establishment of route
Bus concept	LINK LAYER	2	Sub-layer: error correction, acknowledgement Sub-layer: management of access to physical medium
Bus concept	PHYSICAL LAYER	1	Twisted pair, shielded twisted pair, coaxial cable, optical fibre...

TCP: Transmission Control Protocol (Layer 4)

IP: Internet Protocol (Layer 3)

Examples of frames in relation to the ISO model

Modbus RTU frame

Request to read words W5 and W6 at slave address 7

Bytes	1	1	2	2	2
	Slave address = 7	**Function code = 3**	**No. of first word = 5**	**No. of words to be read = 2**	**CRC 16**

Ethernet TCP-IP frame

Bytes	8	6	6	2	20	20	46 to 1500	4
	Preamble	**Destin. addr.**	**Source addr.**	**LLC**	**IP**	**TCP**	**Application layers** **FTP, HTTP , SMTP Modbus etc.**	**FCS**

Physical media

Most popular transmission media

A few electrical standards for twisted pairs

The various topologies

Most popular transmission media

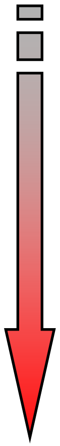
The **MEDIA** establish the transmission quality:

- speed
- distance
- electromagnetic immunity

Most commonly used media:

Cost of medium

Low



High

Pair of twisted wires

The simplest to install, and the **least expensive**.

Coaxial cable

This consists of a copper conductor, surrounded by grounding shielding. There is a plastic insulating layer between the conductor and the shielding. The coaxial cable has **excellent electrical properties** and is suitable for **high speed** transmission.

Optical fibre

Electrical signals are not carried by a copper cable, but an optical fibre transmits light signals.
This is suitable for use in **harsh industrial environments**. Transmission is reliable over **long distances.**

A few electrical standards for twisted pairs

RS232:

Point-to-point link via 25-pin SUB-D connector.

Distance < 15 meters, speed < 20 Kbps.

RS422A:

Full duplex (simultaneous bidirectional) multi-drop bus on 4 wires.

2 transmission wires, 2 reception wires.

Good immunity to interference. Max distance 1200 meters at 100 Kbps.

RS485:

Same characteristics as RS422A but on 2 wires.

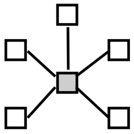
Half duplex (alternate bidirectional) multi-drop bus on 2 wires.

The various topologies

POINT-TO-POINT TOPOLOGY

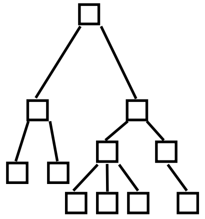
(Between 2 units in communication)

STAR TOPOLOGY



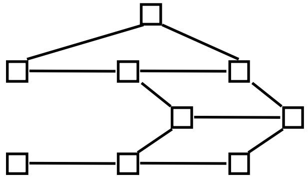
(Several units communicating via their own line line with a Central unit)

TREE TOPOLOGY



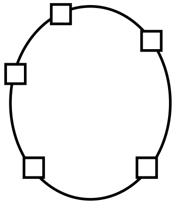
(This is a variant of the star topology)

GRID TOPOLOGY



(Devices are linked to one another, forming a "spider's web".
There are a number of possible paths for reaching a node)

RING TOPOLOGY



(All the units are connected in series in a closed loop.
⇒ Communications must pass via all the units to arrive at the receiver)

BUS TOPOLOGY

(The network consists of a main line to which all the units are connected)

The main medium access methods

Master - Slave

Token ring

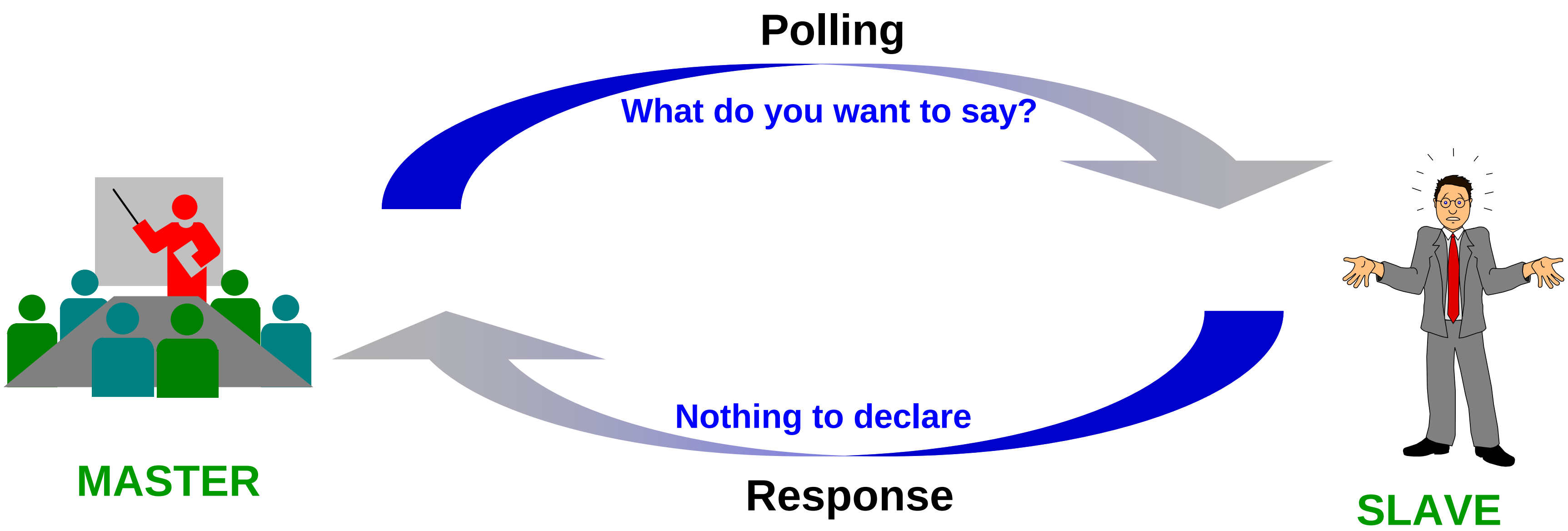
Random access

Master - Slave

Located at the link layer level

The **MASTER** **is the entity which grants access to the medium.**

The **SLAVE** **is the entity which accesses the medium after requesting it from the master.**



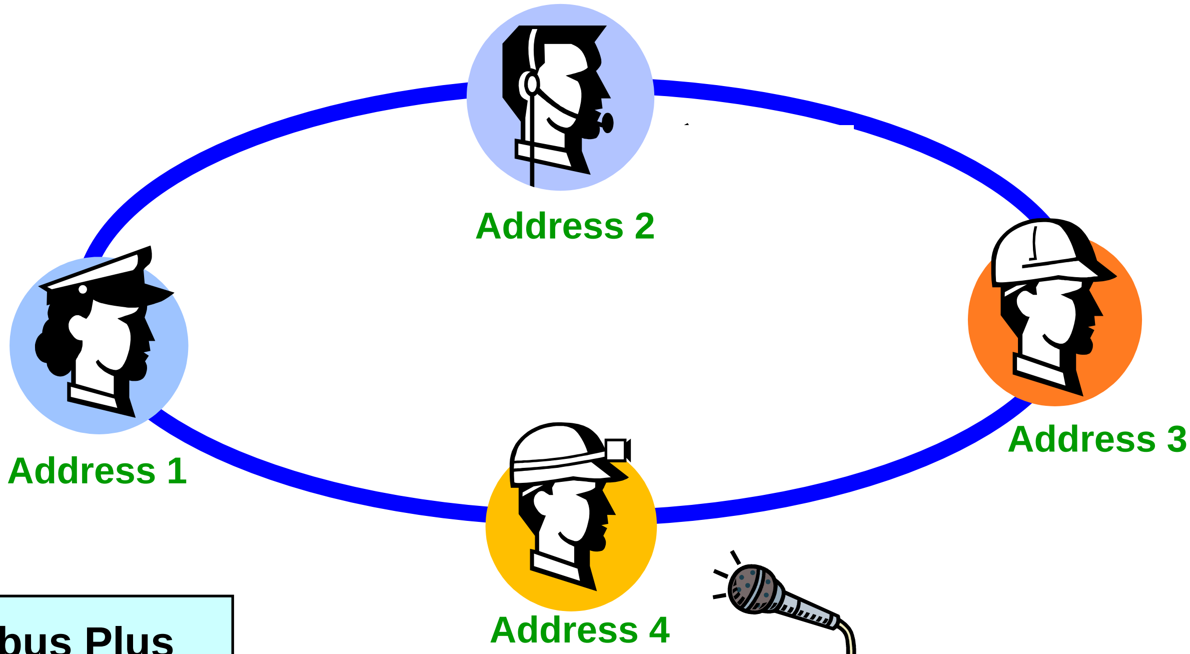
Eg: Profibus-DP

Token ring

Located at the link layer level

The members of a logical **RING** gain access to the network upon receipt of a token.

The **TOKEN** is a group of bits that is passed in a rotating address sequence from one node to another.



Eg: Modbus Plus

Random access

Located at the link layer level

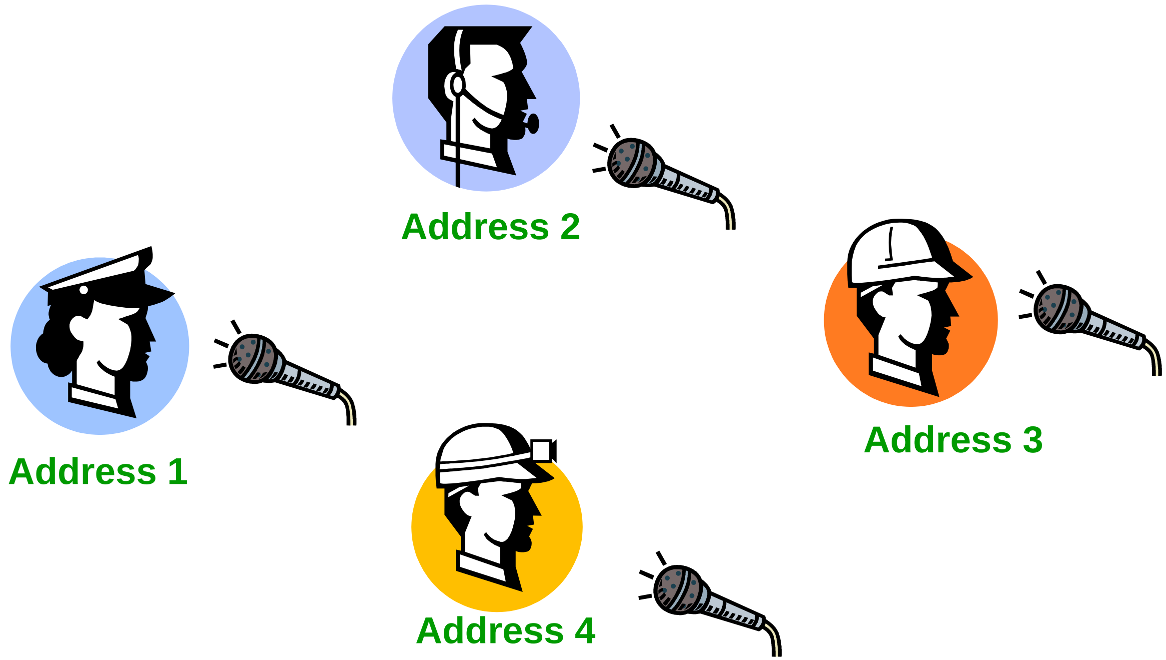
***C**arrier **S**ense **M**ultiple **A**ccess*

A set of rules determining how network devices respond when two devices attempt to use the medium simultaneously (called a *collision*).

CSMA/CD is a type of contention protocol: competition for resources

Informal discussion between undisciplined individuals:

As soon as there's a silence, the one who wants to talk begins to speak.



CSMA/CD CSMA/CA

CSMA/CD = ***Carrier Sense Multiple Access Collision Detect***: Destructive collision

1 - Collision detection

2 - Stop of the emitted frame

3 - Scrambling frame emission

4 - Wait a random time

5 - Frame re-emission

Eg: Ethernet

CSMA/CD = ***Carrier Sense Multiple Access Collision Avoidance:*** Non destructive collision

1 - Non destructive collision detection

2 - The device with the lower priority stops its transmission

3 - End of the high priority frame transmission

4 - The device with lower priority can send its frame

Eg: CAN

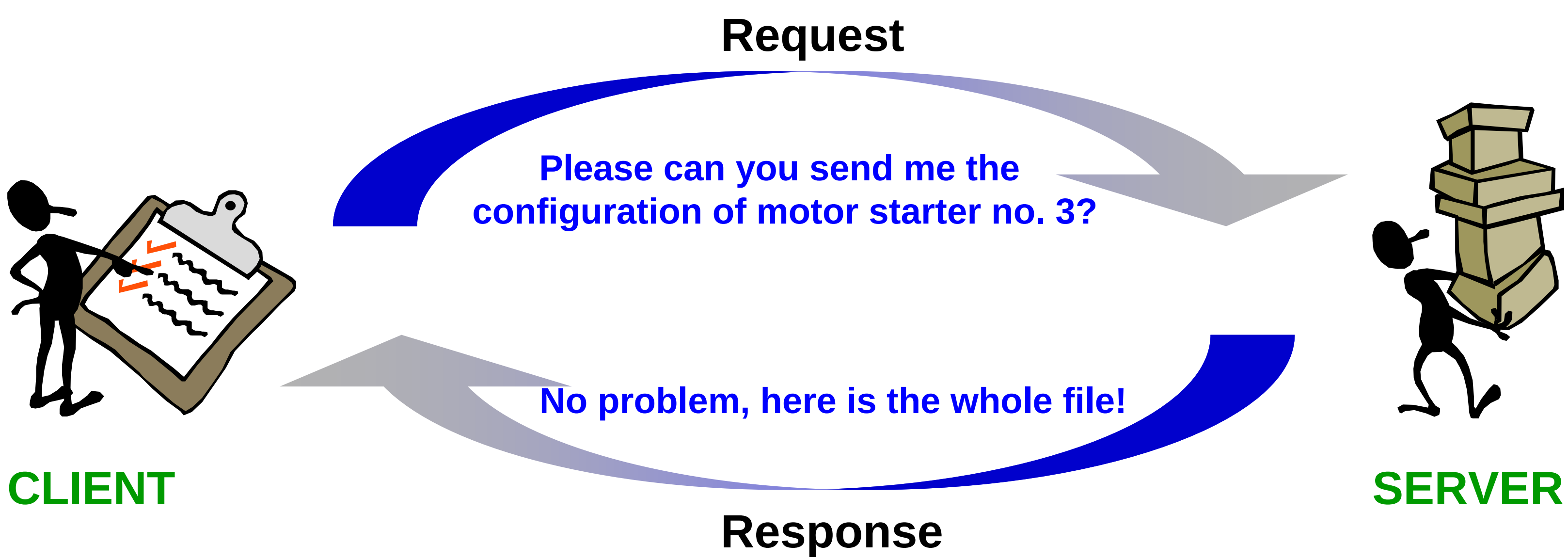
Concepts used at application level

- Client - Server
- Producer - Consumer
- Traffic types
- The concept of a profile

Client - Server

The **CLIENT is an entity requesting a service on the network**

The **SERVER is the entity which responds to a request from a client**

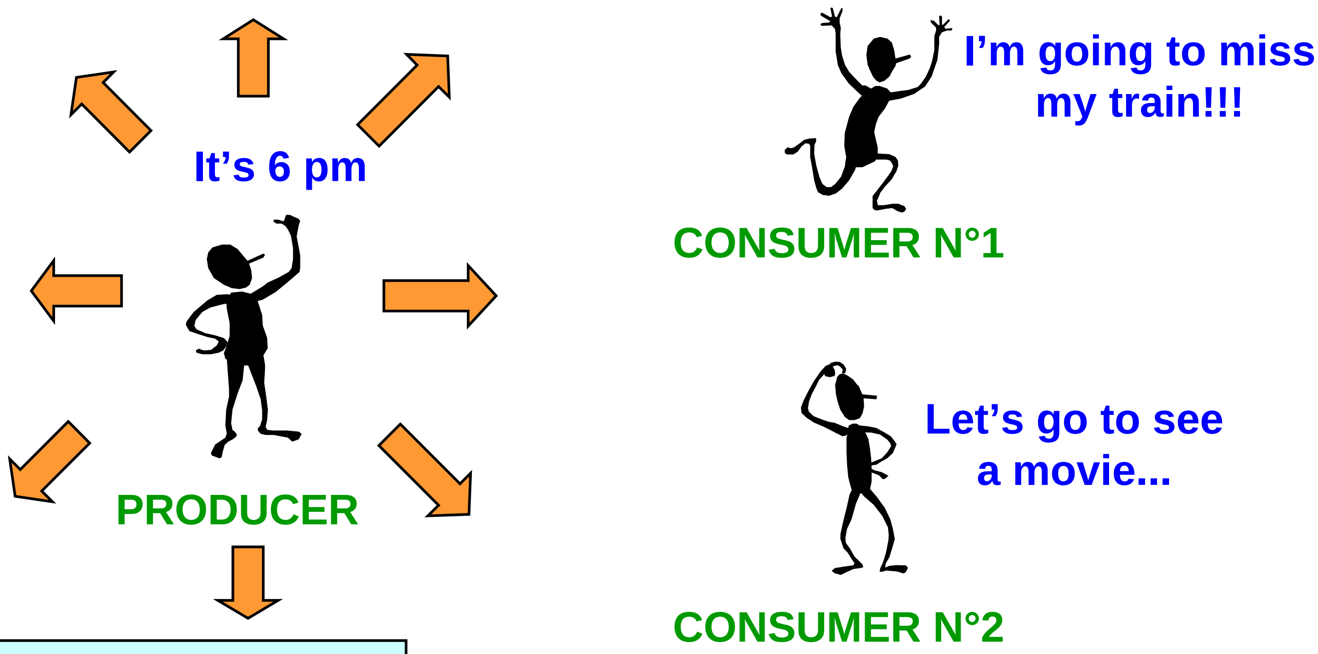


Eg: Modbus

Producer - Consumer

The **PRODUCER** is a single entity which produces information.

The **CONSUMER** is an entity which uses it (several entities can use the same information).



Eg: CANopen DeviceNet

Traffic types

Cyclical data:

Data that is refreshed periodically according to a pre-determined time.

This is process data.

A small amount of information refreshed frequently.

Acyclic data:

Data that is refreshed according to a request or to an event.

This is used at start-up for configuration and setup, or for diagnostics in the event of a fault.

A lot of information without time constraints.

Open system

An open system comprises interoperable and interchangeable components.

Interoperability is the ability to communicate intelligibly with other devices.

It is achieved by means of strict adherence to protocol specifications.

Interchangeability is the ability to replace one device with another (possibly supplied by a different manufacturer).

It is achieved by means of adherence to profile specifications.

All manufacturers reserve the right to define whether or not they wish to offer manufacturer-specific functions in addition to those which are part of the minimum profile or core.

The concept of a profile

A profile is a standardized way of describing functions which ensure components can be interchanged.

This description adheres to a strict syntax.

Data is grouped by function:

- Identification: product name, reference, version, family, manufacturer
- Characteristics relating to communication: Speeds supported, type and size of messages exchanged, etc.
- Characteristics relating to the application: Variables which can be accessed in write mode, in read mode, when stopped, when running, etc.

Most profiles are provided in electronic file format: EDS file, GSD file, etc. supplied on floppy disk or CD-ROM with the product.

This file provides details of the characteristics of the device "offline".

Extract from TEGO Power Quickfit CANopen EDS file

```
[FileInfo]
CreatedBy=Martin Rostan
ModifiedBy=Martin Rostan
Description=EDS for Tego Power CANopen
CreationTime=10:05PM
CreationDate=01-17-2001
ModificationTime=10:35PM
ModificationDate=01-17-2001
FileName=F:\Produkte\Tego Power\APP1CCO0
FileVersion=1
FileRevision=1
EDSVersion=4
[DeviceInfo]
VendorName=Schneider Electric SA (France)
VendorNumber=90
ProductName=APP-1CCO0
ProductNumber=1
RevisionNumber=1
OrderCode=APP-1CCO0
BaudRate_10=0
BaudRate_20=0
BaudRate_50=0
BaudRate_125=1
BaudRate_250=1
BaudRate_500=1
BaudRate_800=0
BaudRate_1000=1
```

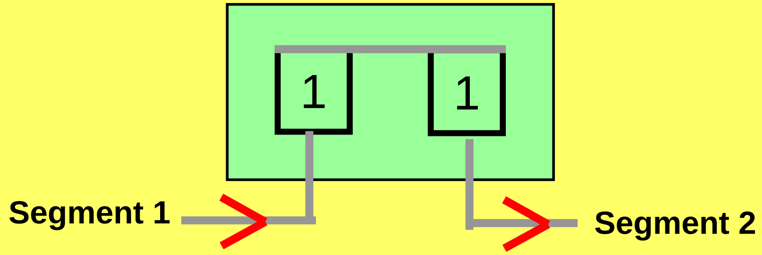
```
[MandatoryObjects]
SupportedObjects=2
1=0x1000
2=0x1001

[1000]
ParameterName=Device Type
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=0x30191
PDOMapping=0
```

Interconnection products

- Repeater
- Hub
- Switch
- Transceiver
- Bridge
- Router
- Gateway

Repeater - Hub - Switch

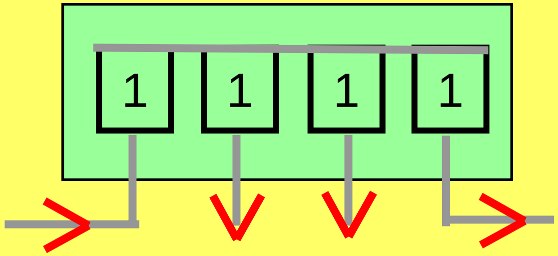
Repeater



Can be used to add segments to a network.
It amplifies and restores the same type of signal.

Example = RS485 repeater

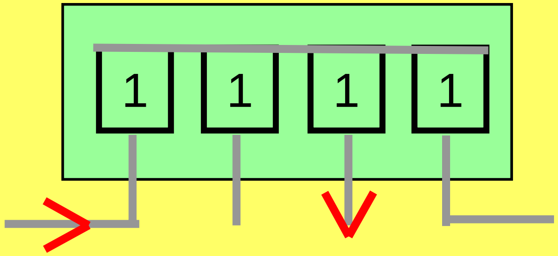
Hub



Can be used to extend a star network.
It amplifies and restores the same type of signal on all ports.

Example = Ethernet hub
(does not reduce the number of collisions)

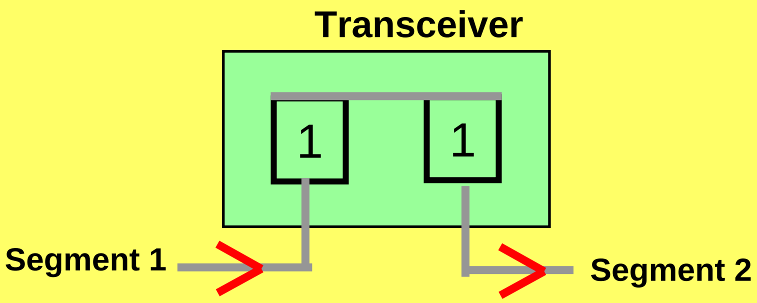
Switch



Can be used to extend a star network.
It amplifies and restores the same type of signal on a single port.

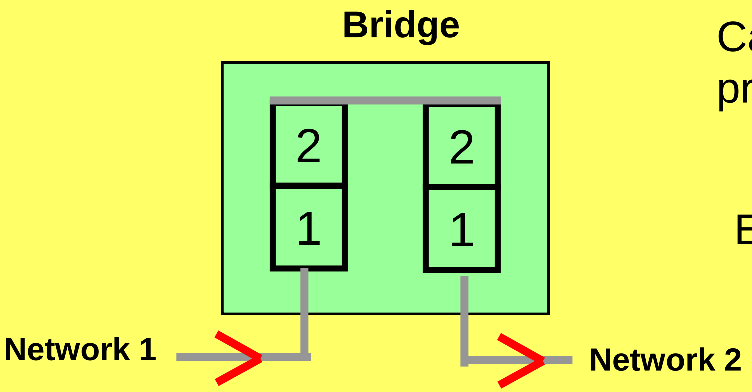
Example = Ethernet switch
(can be used to reduce the number of collisions)

Transceiver - Bridge



Can be used to add different types of segment to a network

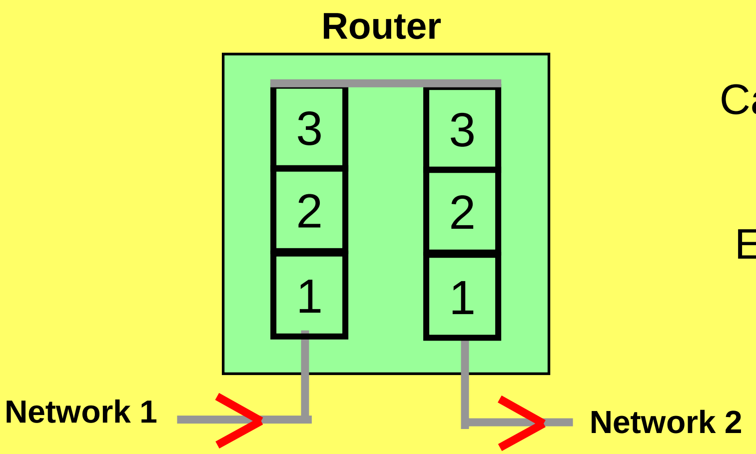
Example = RS232/RS485 transceiver



Can be used to connect 2 networks using the same protocol but different lower layers

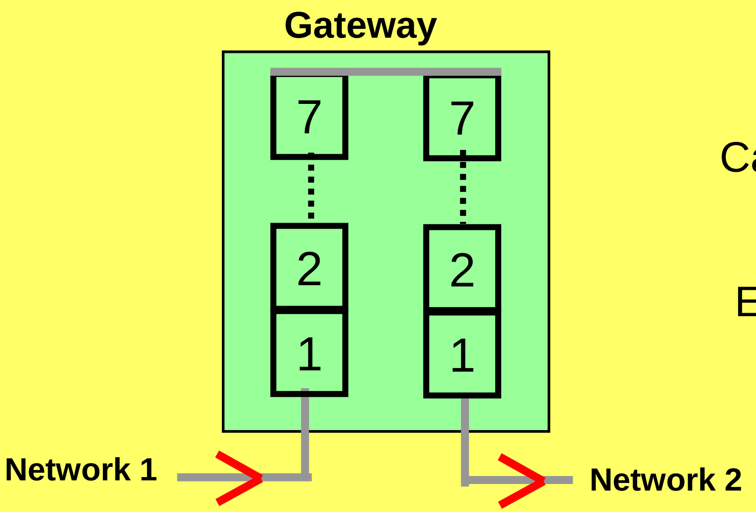
Example = Modbus RS485/Ethernet TCP-IP bridge

Router - Gateway



Can be used to connect 2 networks of the same type

Example = Ethernet TCP-IP router



Can be used to connect 2 networks of different types

Example = FIPIO/Modbus gateway

ASi

- History
- ASi and the ISO model
- Physical layer
- Link layer
- Application layer
- Profiles
- Strengths - Weaknesses

History

- **1990:**

2 universities and 11 companies (mainly German) create the ASi consortium in order to define a "low-cost" interface for connecting sensors and actuators.

- **1992:**

The first chips become available.
Creation of the international ASi association: **http://www.as-interface.net/** based in Germany. Schneider joins the association.

- **1995:**

Creation of national promotional associations (France, The Netherlands, UK)

- **2001:**

ASi V2 specifications: 62 slaves, support for analog products, improved diagnostics.

Integration of safety products: "Safety at work"

ASi and the ISO model

3 layers used + profiles

Generic discrete I/O interfaces	Discrete sensors	Motor starters	Analog I/O, etc.

7	APPLICATION	Client/Server via requests
6	PRESENTATION	EMPTY
5	SESSION	EMPTY
4	TRANSPORT	EMPTY
3	NETWORK	EMPTY
2	LINK = LLC + MAC	Master/Slave
1	PHYSICAL	Power supply and communication on the same media

Physical layer

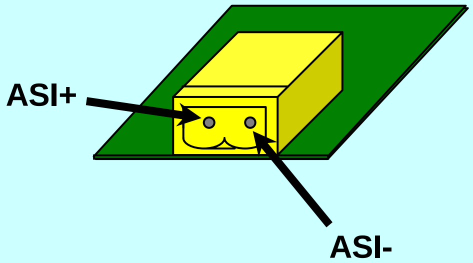
Medium:	**2-wire yellow flat ribbon cable with polarization** An unshielded round cable can also be used.
Topology:	**Free** **No line terminators**
Maximum distance:	**100 m without repeaters** **300 m with repeaters**
Speed:	**167 Kbps** 1 transaction (data exchange) lasts 150 ms. Cycle time = 5 ms for 31 slaves 10 ms for 62 slaves
Max. no. of devices:	**ASi V1: 1 master + 31 slaves** **ASi V2: 1 master + 62 A/B slaves**

Types of connection

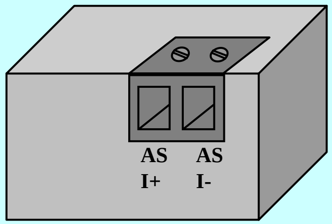
4 types of connection defined in the Schneider ASi specification

IP20

Yellow 2-pin removable connector

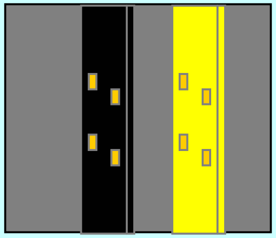


Screw or spring terminals

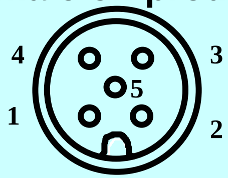


IP65

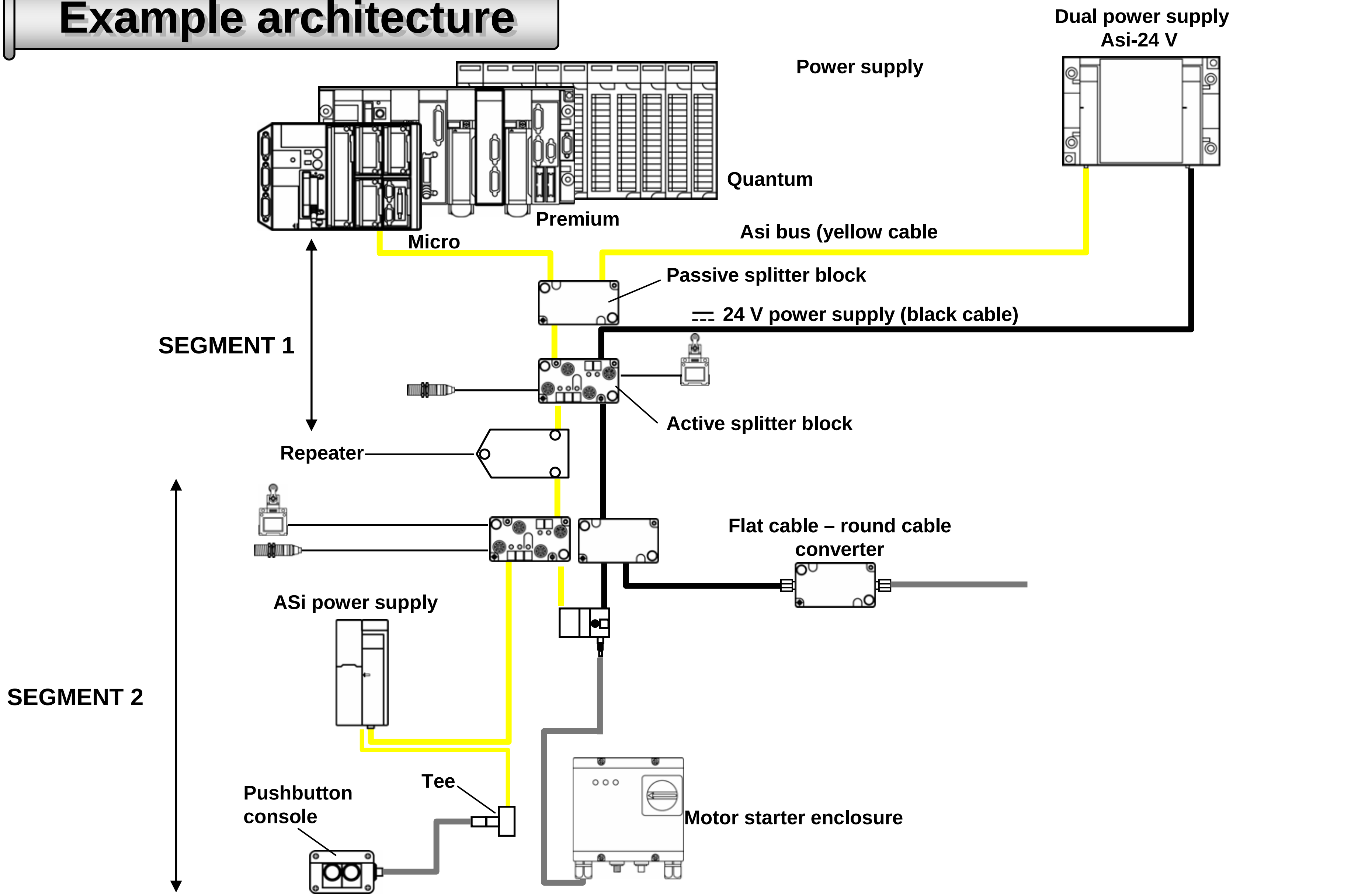
IDC connector



M12 connector (male on product)



Example architecture



Link layer

Medium access method: **Master/Slave**

Max. size of useful data: **4 output bits for a request**

(3 bits in ASi V2 for A/B slaves)

4 input bits for a response

Transmission security: **Numerous checks at bit and frame level**

Start bit delimiter, half-wave pulses,
length of pause between 2 bits,
end-of-frame parity, end bit delimiter,
length of frame

Application layer

A dozen standardized requests for:

1 . Network administration: Addressing, identification, parameter settings, reset.

2 . **Cyclic I/O exchange: Data exchange**

Max. 4 output bits for standard slaves, 3 for A/B slaves

Max. 4 input bits for all slaves

Cycle time: 5 ms max. for 31 slaves, 10 ms for 62

3 . Cyclic network monitoring: Read Status

Feedback of I/O errors for ASi V2 slaves

Cycle time: 155 ms for 31 slaves, 310 ms for 62 slaves

4 . Parameter data transmission: Write Parameter

Via programming of Write Parameter request

Max. 4 output bits for standard slaves, 3 for A/B slaves

155 ms maximum for 31 slaves, 310 ms for 62

Profiles

To ensure interchangeability between products, every ASi slave is identified by a fixed profile which is engraved in the silicon (read-only).

The profile for ASi V1 slaves is defined using 2 hexadecimal digits.

The profile for ASi V2 slaves is defined using 4 hexadecimal digits.

Profiles

ASi V1: 2 digits

Profile = IO$_{_code}$. ID$_{_code}$

IO$_{_code}$ = Indicates the number of inputs and outputs on the device (0 to F)

ID$_{_code}$ = Indicates the type of device (0 to F)

ASi V2: 4 digits

Profile = IO$_{_code}$. ID$_{_code}$. ID1$_{_code}$. ID2$_{_code}$

IO$_{_code}$ = Indicates the number of inputs and outputs on the device (0 to F)

ID$_{_code}$ = Indicates the type of device (0 to F)

ID1$_{_code}$ = Used for customizing the product (0 to F)

ID2$_{_code}$ = Indicates the product sub-type (0 to F)

CANopen

- History
- CANopen and the ISO model
- Physical layer
- Link layer
- Application layer
- Profiles
- Strengths - Weaknesses

History

1980-1983:

Creation of **CAN** as an initiative by the German equipment manufacturer **BOSCH** to meet a requirement in the **automotive industry**.

CAN only defines one part of layers 1 and 2 of the ISO model.

1983-1987:

The prices of drivers and micro-controllers featuring CAN become very attractive as they are used in high volume in the automotive industry.

1991:

CIA = CAN in Automation is born: **http://www.can-cia.de/** to promote industrial applications.

History

1993:

CAL = CAN Application Layer specifications published by **CiA** describing transmission mechanisms but not when and how to use them.

1995:

CiA publishes the **DS-301 communication profile: CANopen**

2001:

CiA publishes DS-304 which can be used to integrate **level 4 safety components** on a standard CANopen bus (**CANsafe**).

CANopen and the ISO model

CANopen is based on CAL

Device Profile CiA DSP-401 I/O modules	Device Profile CiA DSP-402 Drives	Device Profile CiA DSP-404 Measuring devices	Device Profile CiA DSP-4xx

7	APPLICATION	CiA DS-301 = Communication profile CAL= CAN Application Layer
6	PRESENTATION	EMPTY
5	SESSION	EMPTY
4	TRANSPORT	EMPTY
3	NETWORK	EMPTY
2	LINK = LLC + MAC	CAN 2.0 A and B + ISO 11898
1	PHYSICAL	CAN 2.0 A and B = ISO 11898-1 and 2 ISO 11898 + DS-102

Physical layer

Medium: **Shielded twisted pair**

2 or 4-wire (if power supply)

Topology: **Bus type**

With short tap links and 120 ohm line termination resistor

Maximum distance: **1000 m**

Speed: **9 possible speeds from 1 Mbps to 10 Kbps**

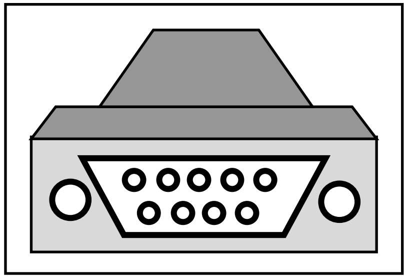
Depends on bus length and cable type: 25 m at 1 Mbps, 1000 m at 10Kbps

Max. no. of devices: **128**

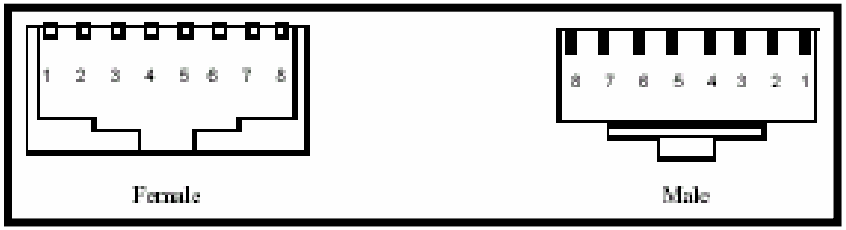
1 master and 127 slaves

Connectors

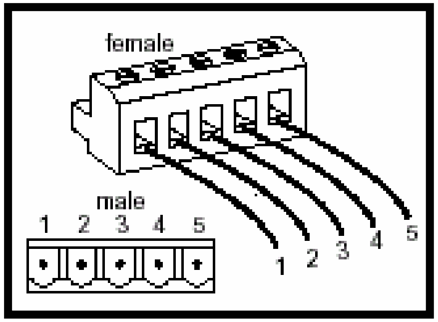
CiA recommendation DR-303-1 includes a list of suitable connectors divided into 3 categories with a description of their pin configuration.



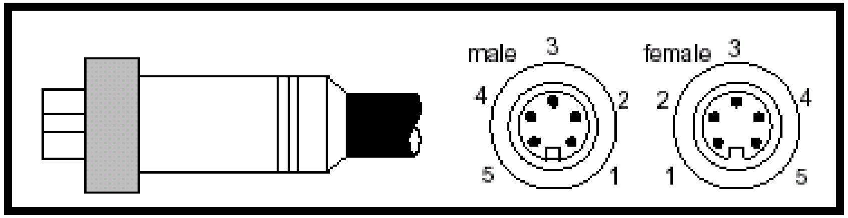
9-pin SUB D
DIN 41652



RJ45



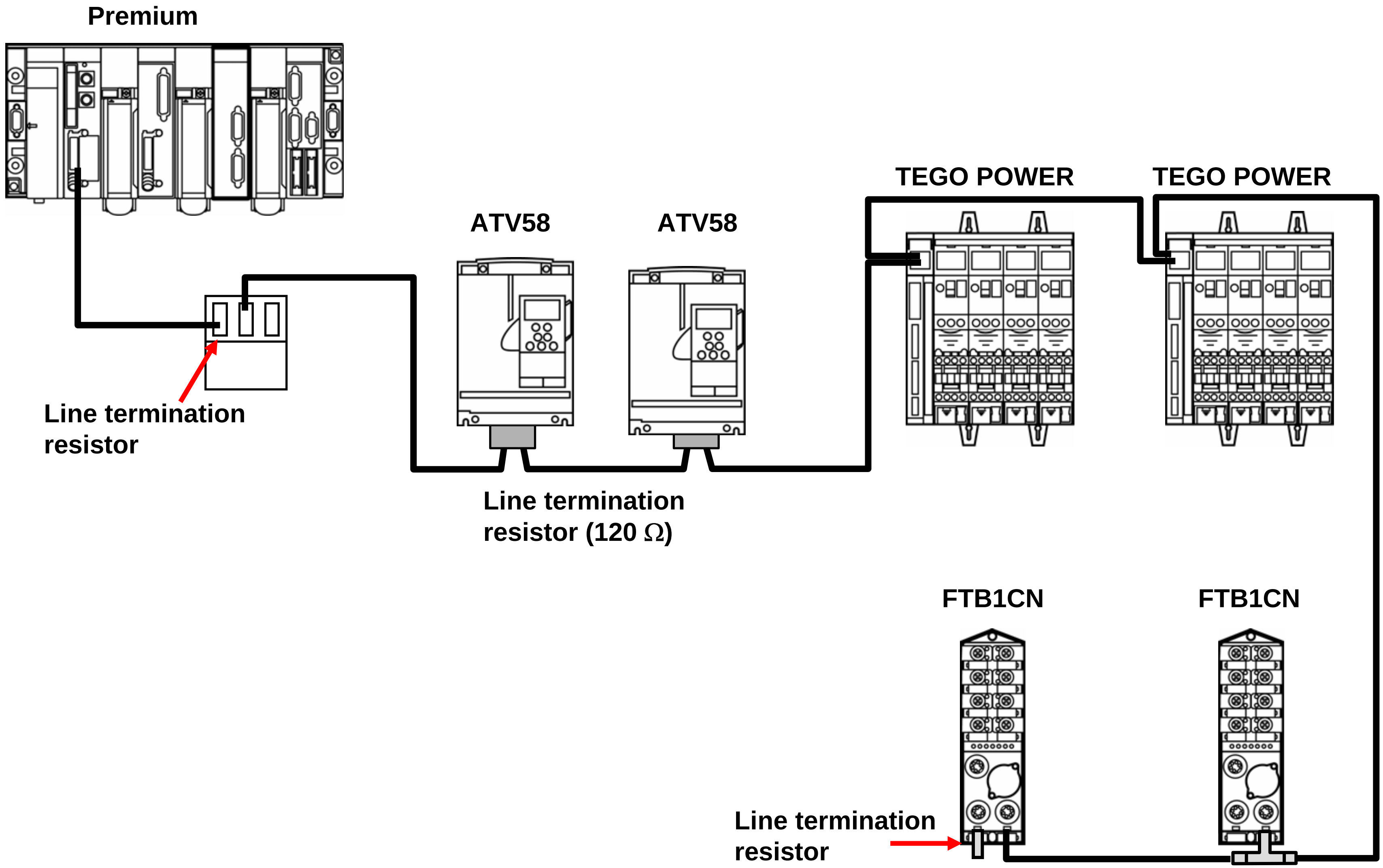
Open style



5-pin Micro Style = M12
ANSI/B93.55M-1981

Male, product side

Example architecture



Link layer

Medium access method: CSMA/CA

Every device may send data as soon as the bus is free.

The principle of dominant and recessive bits enables non-destructive bit-by-bit arbitration in the event of a collision.

The priority of a message is indicated by the value of the identifier: **The identifier with the lowest value has priority.**

Communication model: **Producer/Consumer**

An identifier coded on 11 bits and located at the start of the message informs the receivers about the type of data contained in each message. Each receiver decides whether or not to accept the data.

This concept permits **multiple communication models**:

Transmission on change of state, cyclic, SYNC signal, Master_Slave system.

Link layer

Max. size of useful data: 8 bytes per frame

Transmission security:

One of the best local industrial networks

Numerous signalling and error detection devices ensure high transmission security.

Application layer

4 types of standardized service:

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

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

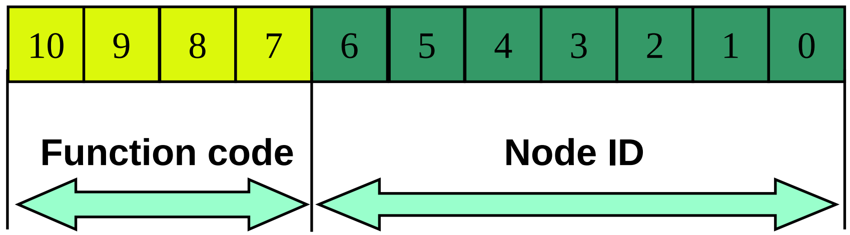
PDOs can be transmitted on changes of state, cyclically, on receipt of the SYNC message or at the request of the master.

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

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

Application layer

The allocation of identifiers on CANopen is based on the division of the identifier into 2 parts:



Function code is used to code 2 receiving PDOs, 2 sending PDOs, 1 SDO, 1 EMCY object, 1 Node Guarding identifier, 1 SYNC object, 1 time stamp object and 1 Node Guard.

Node ID corresponds to the address of the product coded for example using DIP switches.

Application layer

Broadcast objects			
Object	Function code	Node ID	CMS priority group
NMT	0000	0x000	0
SYNC	0001	0x080	0
TIME STAMP	0010	0x100	1

Peer-to-peer objects			
Object	Function code	Node ID	CMS priority group
Emergency	0001	0x081-0x0FF	0, 1
Transmit PDO 1	0011	0x181-0x1FF	0, 1
Receive PDO 1	0100	0x201-0x27F	2
Transmit PDO 2	0101	0x281-0x2FF	2, 3
Receive PDO 2	0110	0x301-0x37F	3, 4
Server SDO	1011	0x581-0x5FF	6
Client SDO	1100	0x601-0x67F	6, 7
NODE GUARD	1110	0x701-0x77F	1

Profiles

CANopen profiles are based on the object dictionary concept:

Device Object Dictionary (OD).

The CANopen object dictionary is an **ordered group of objects** which can be accessed via an index of 16 bits and, if required, a sub-index of 8 bits.

Each network node has an OD in an ASCII format **EDS (Electronic Data Sheet)** file (DSP 306 specification).

This dictionary contains all the elements describing the node along with its network characteristics.

Profiles

Object dictionary structure

Index (hex)	Object
0000	Reserved
0001 – 009F	Data types area
00A0 – 0FFF	Reserved
1000 – 1FFF	**Communication profile area**
2000 – 5FFF	**Manufacturer-specific profile area**
6000 – 9FFF	**Standardized device profile area**
A000 – FFFF	Reserved

Profiles

CANopen defines 2 types of profile:

DS-301 communication profile:

Describes the general structure of the OD and the objects in the "communication profile area" zone. It is valid for all CANopen products.

DSP-4xx device profiles:

Describes the various standard objects associated with the different types of product (discrete I/O modules, drives, measuring devices).

Some objects are mandatory, others are optional, some are read only, others are read/write.

DeviceNet

- History
- DeviceNet and the ISO model
- Physical layer
- Link layer
- Application layer
- Profiles
- Strengths - Weaknesses

History

1980-1983:

Creation of **CAN** as an initiative by the German equipment manufacturer **BOSCH** to meet a requirement in the **automotive industry**.

CAN only defines one part of layers 1 and 2 of the ISO model.

1983-1987:

The prices of drivers and micro-controllers featuring CAN become **very attractive** as they are used in high volume in the automotive industry.

1993-1994:

Allen Bradley (Rockwell Automation Group) develops and launches **DeviceNet** products.

History

1995:

Creation of the ODVA = Open DeviceNet Vendor Association: http://odva.org/ to **promote and provide technical support** for the DeviceNet specification.

1997:

The association includes approximately **200 member companies** and offers **a hundred different products**.

2002:

The ODVA starts to develop specifications for integrating **safety components**.

DeviceNet and the ISO model

AC Drives | HMI | Communication adapter | Etc.

7	APPLICATION	DeviceNet Specifications Volume 1
6	PRESENTATION	EMPTY
5	SESSION	EMPTY
4	TRANSPORT	EMPTY
3	NETWORK	EMPTY
2	LINK = LLC + MAC	CAN 2.0 A and B + ISO 11898
1	PHYSICAL	CAN 2.0 A and B = ISO 11898-1 and 2 DeviceNet Specifications Volume 1

Physical layer

Medium: **2 shielded twisted pairs**

2 wires for communication and 2 wires for power

Topology: **Bus type**

With short tap links and 120 ohm line termination resistor

Maximum distance: **1000 m**

Speed: **3 possible speeds: 125, 250 or 500 Kbps**

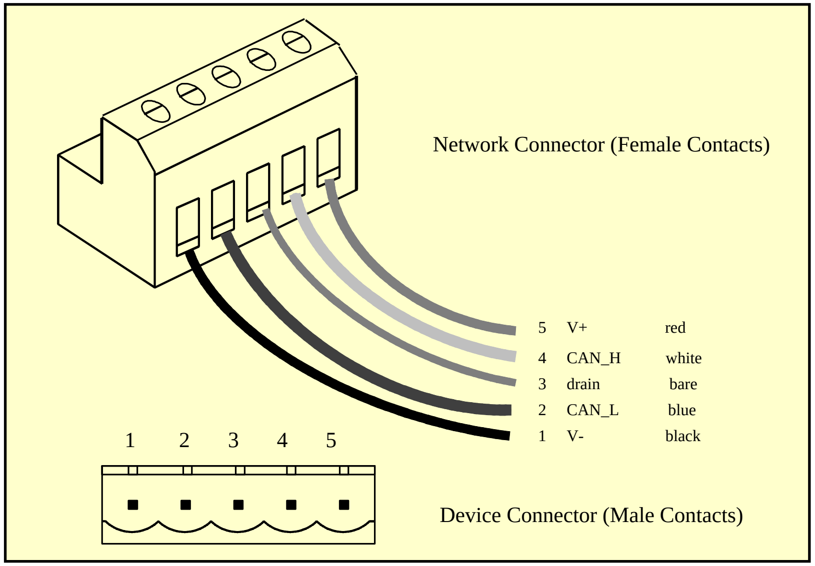
Depends on bus length and cable type as well as product consumption

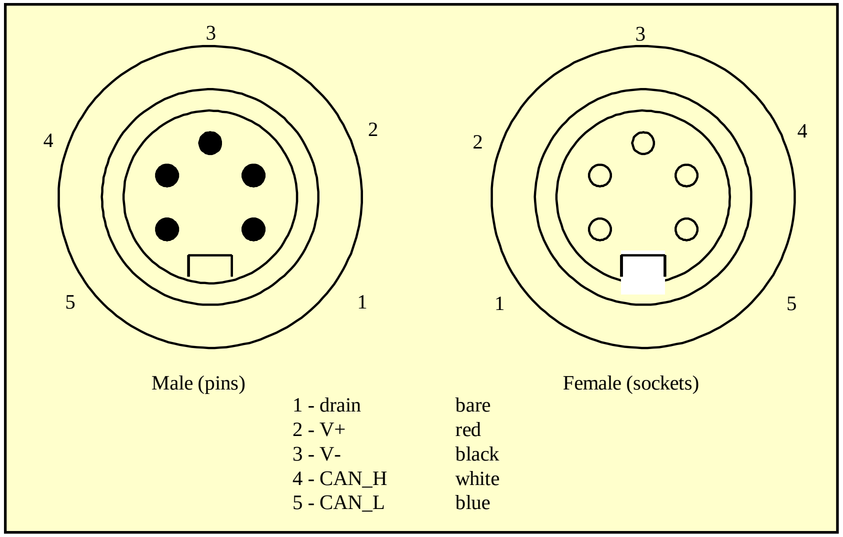
Max. no. of devices: **64 master modes (scanner) included**

Connectors

All connectors must have 5 pins.

The following connectors are recommended:





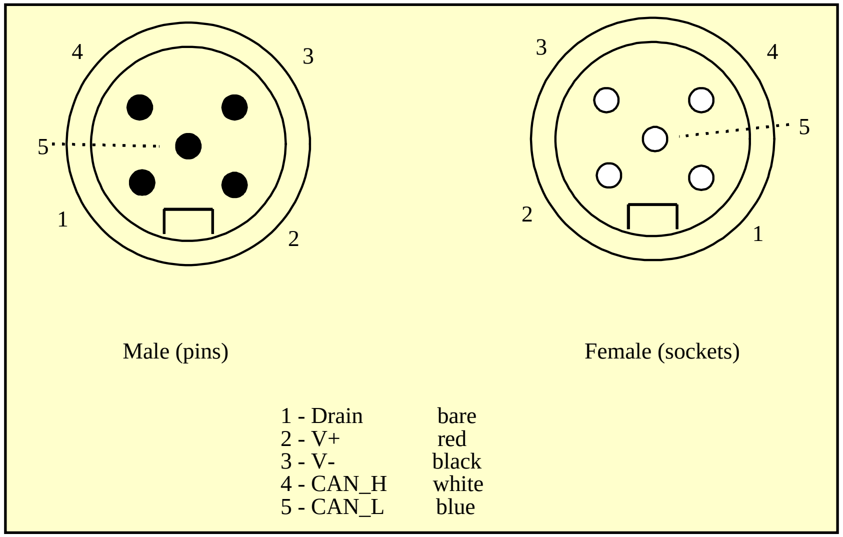
Phoenix Combicon

MSTB 2.5/5-ST-5.08-AU: Network cable side
MSTBA 2.5/5-G-5.08-AU: Product side, horizontal pins
MSTBVA 2.5/5-G-5.08-AU: Product side, vertical pins

Mini Style connector

ANSI/B93.55M-1981

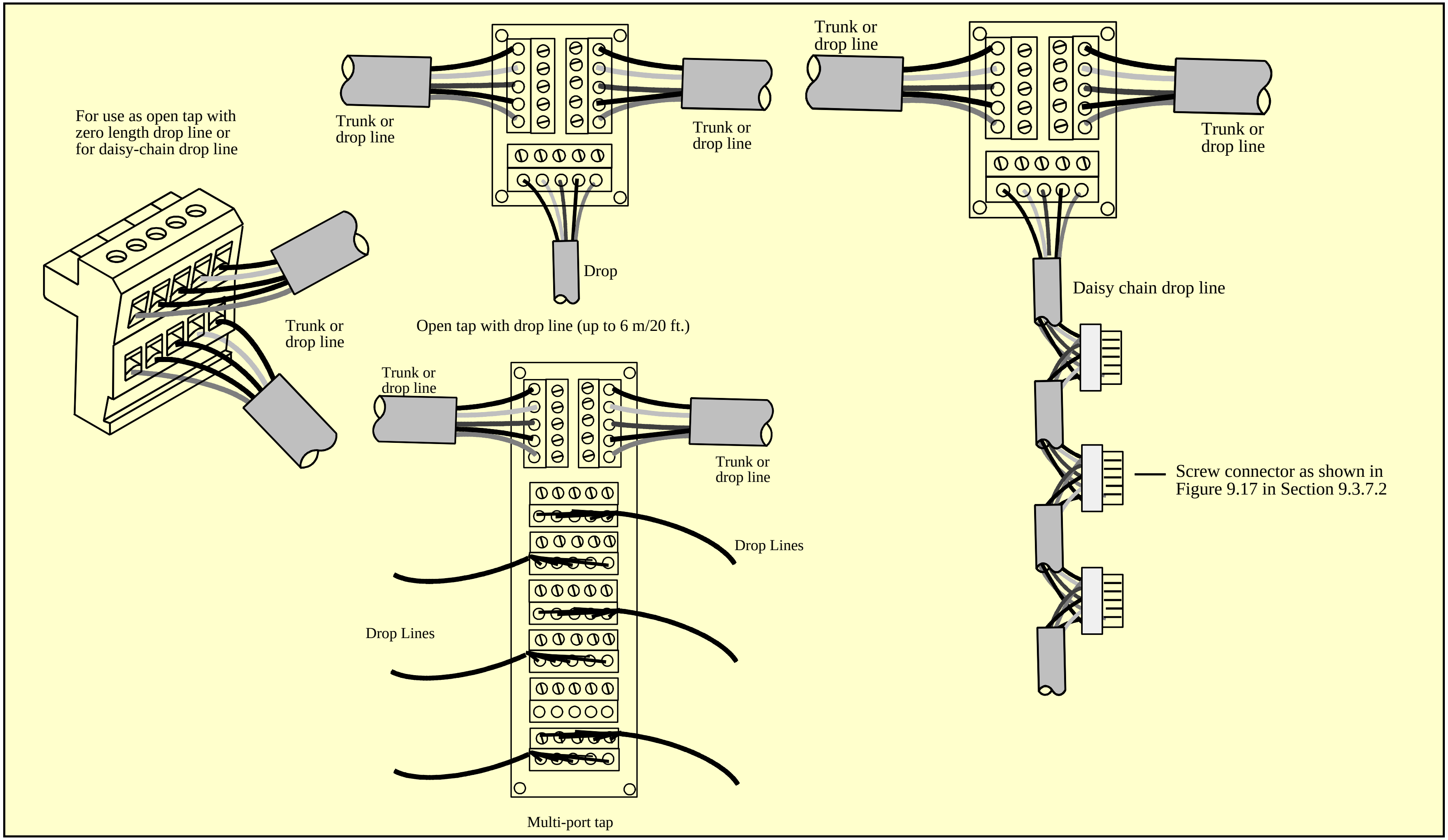
Connectors



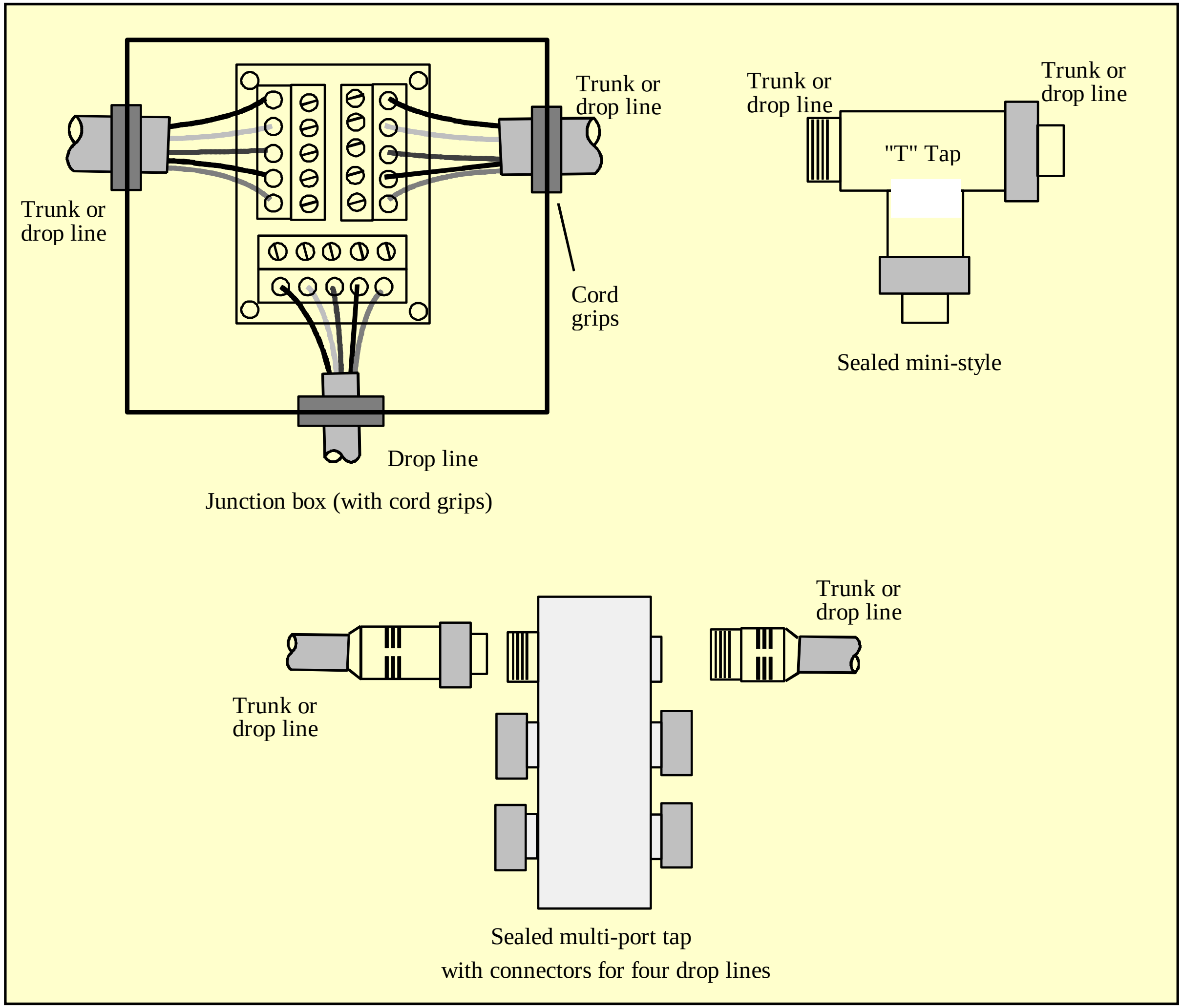
Micro Style connector (M12)

Lumberg RST 5-56/xm or equivalent

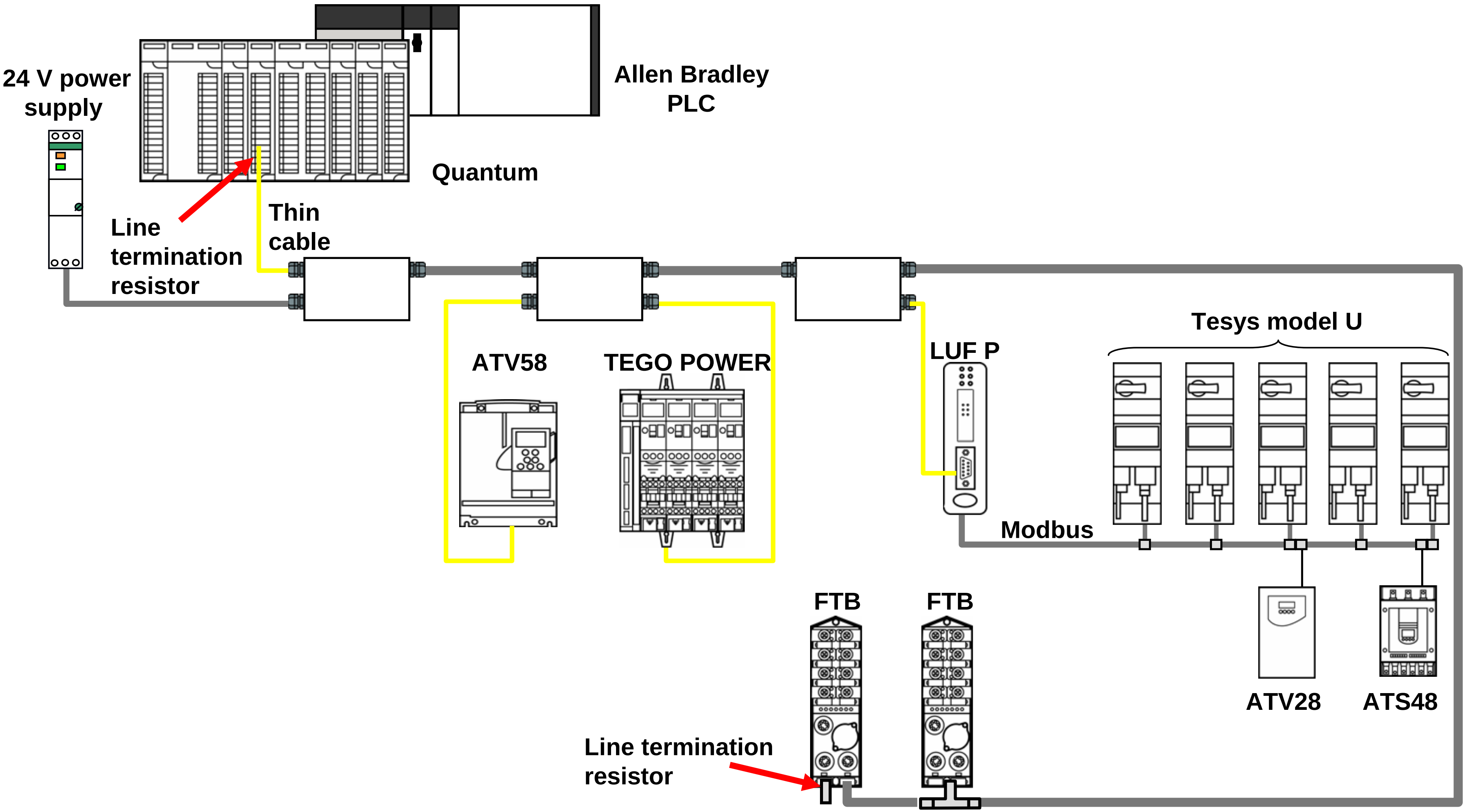
IP20 taps


For use as open tap with
zero length drop line or
for daisy-chain drop line
Trunk or
drop line
Trunk or
drop line
Trunk or
drop line
Drop
Open tap with drop line (up to 6 m/20 ft.)
Trunk or
drop line
Trunk or
drop line
Daisy chain drop line
Trunk or
drop line
Trunk or
drop line
Drop Lines
Drop Lines
Multi-port tap
Screw connector as shown in
Figure 9.17 in Section 9.3.7.2

IP65 taps


Trunk or
drop line
Trunk or
drop line
Cord
grips
Drop line
Junction box (with cord grips)
Trunk or
drop line
Trunk or
drop line
"T" Tap
Sealed mini-style
Trunk or
drop line
Trunk or
drop line
Sealed multi-port tap
with connectors for four drop lines

Example architecture



Link layer

Medium access method: **CSMA/CA**

Every device may send data as soon as the bus is free.

The principle of dominant and recessive bits enables non-destructive bit-by-bit arbitration in the event of a collision.

The priority of a message is indicated by the value of the identifier: **The identifier with the lowest value has priority.**

Communication model: **Producer/Consumer**

An identifier coded on 11 bits and located at the start of the message informs the receivers about the type of data contained in each message. Each receiver decides whether or not to accept the data.

This concept permits **multiple communication models**:

Transmission on change of state, cyclically, on Strobe signal, via Master_Slave system.

Link layer

Max. size of useful data: 8 bytes per frame

Fragmentation possible if more than 8 bytes

Transmission security:

One of the best local industrial networks.

Numerous signalling and error detection devices ensure high transmission security.

Application layer

3 types of standard services:

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

2 . Transmission of **low-volume process data** in real time:

I/O messages

I/O messages can be transmitted on changes of state, cyclically, on receipt of the Strobe message or via master polling, etc.

3 . Transmission of high-volume **parameter data** (> 8 bytes) by segmentation without time restrictions: **Explicit messages** in client/server mode.

Allocation of identifiers

IDENTIFIER BITS											DESCRIPTION
10	9	8	7	6	5	4	3	2	1	0	
0	Group 1 Message ID				Source MAC ID						**Group 1 Messages**
0	1	1	0	1	Source MAC ID						Slave's I/O Change of State or CyclicMessage
0	1	1	1	0	Source MAC ID						Slave's I/O Bit-Strobe Response Message
0	1	1	1	1	Source MAC ID						Slave's I/O Poll Response Message
1	0	MAC ID						Group 2 Message ID			**Group 2 Messages**
1	0	Source MAC ID						0	0	0	Master's I/O Bit-Strobe Command Message
1	0	Source MAC ID						0	0	1	Reserved for Master's Use -- Use is TBD
1	0	Source MAC ID						0	1	0	Master'sChg of state/cyclic acknowledge msgs
1	0	Source MAC ID						0	1	1	Slave's Explicit Response Messages
1	0	Destination MAC ID						1	0	0	Master's Connected Explicit Request Messages
1	0	Destination MAC ID						1	0	1	Master's I/O Poll Cmd/Chg of State/Cyclic Msgs
1	0	Destination MAC ID						1	1	0	Group 2 Only Unconnected Explicit Req.. Msgs
1	0	Destination MAC ID						1	1	1	Duplicate MAC ID Check Messages

Profiles

- DeviceNet uses **object type modelling** for:
  - The list of available communication services
  - Device characteristics
  - A standard means of describing how to access the internal variables of a product
- A **DeviceNet node** is modelled as a **collection of objects.**

Object addressing

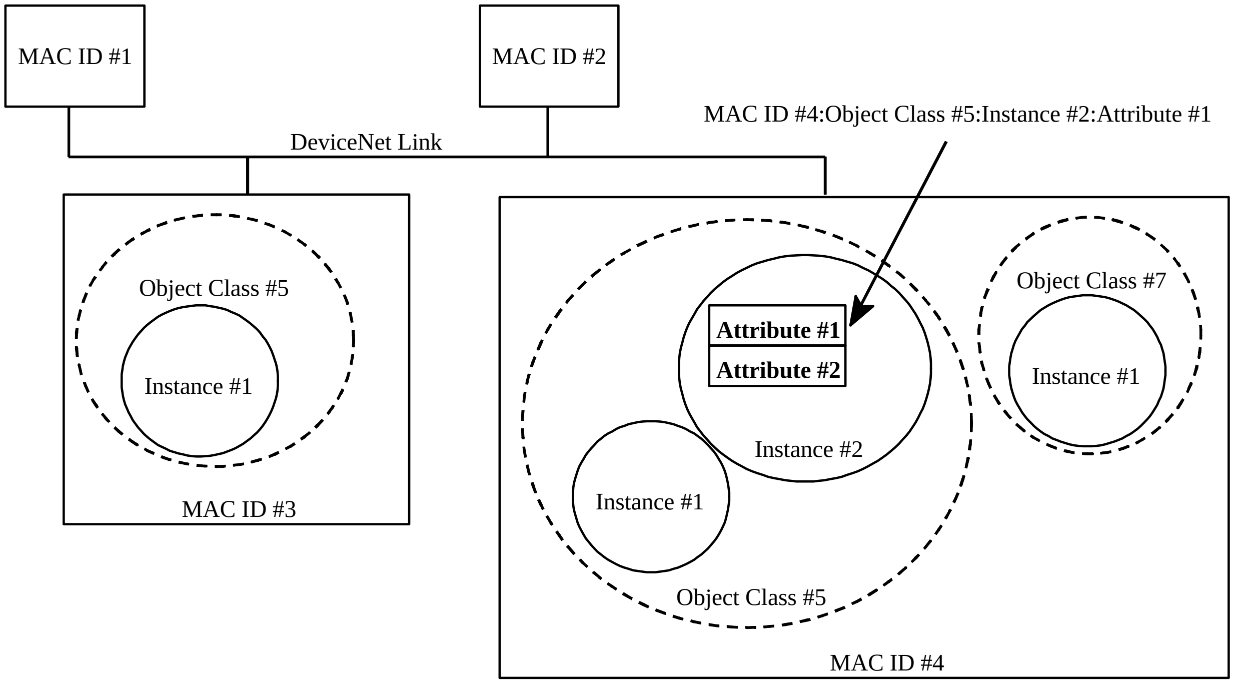
DeviceNet uses a **4-level addressing** method:

MAC ID

Class ID

Instance ID

Attribute ID



The variables of a node can be accessed via a **path** which comprises:

Class ID

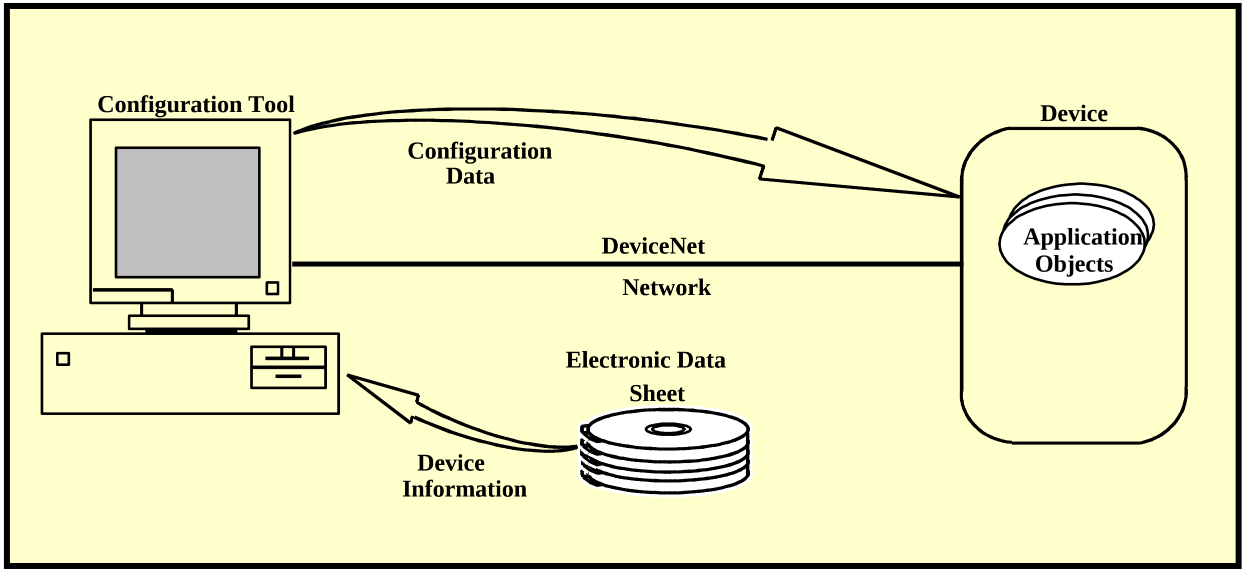
Instance ID

Attribute ID

EDS file

A DeviceNet profile is defined in an **EDS (Electronic Data Sheet)** file **supplied with the product**.

This file provides a precise description of all the component objects of the product.



Extract from DeviceNet LUFP9 gateway EDS file

```
$ DeviceNet Manager Generated Electronic Data Sheet
[File]
    DescText  = "LUFP9 Gateway";
    CreateDate = 12-08-98;
    CreateTime = 10:31:30;
    ModDate   = 10-07-2002;
    ModTime   = 16:39:54;
    Revision  = 1.02;
[Device]
    VendCode  = 90;   $ Vendor Code
    ProdType  = 12;   $ Product Type
    ProdCode  = 60;   $ Product Code
    MajRev   = 1;     $ Major Rev
    MinRev   = 3;     $ Minor Rev
    VendName  = "Schneider Electric Gateways";
    ProdTypeStr = "Communications Adapter";
    ProdName  = "LUFP9";
    Catalog   = "LUFP9";

$ Parameter Class Section
[ParamClass]
    MaxInst   = 29;     $ Max Instances - total # configuration parameters
    Descriptor = 0x00;   $ Parameter Class Descriptor - No parameters
    CfgAssembly = 0x00; $ The config assembly is not supported.
```

```
[Params]
$ ****************************************************************************
$ Polled production
$ ****************************************************************************
   Param1=
      0,                        $ parameter value slot
      6, "20 05 24 00 30 64",
      0x0002,                   $ descriptor (Scaling)
      8, 1,                                 $ USINT, 4 bytes
      "Polled production",      $ parameter name
      "",                                   $ units string
      "",
      0, 5, 0,                              $ min, max, default (0)
      0, 0, 0, 0,                           $ mult, div, base, offset
scaling
      , , , ,                   $ scaling links not used
          0;                    $ decimal places
$ ****************************************************************************
$ Polled consumtion
$ ****************************************************************************
   Param2=
      0,                        $ parameter value slot
      6, "20 05 24 00 30 65",
      0x0002,                   $ descriptor (Scaling)
      8, 1,                                 $ USINT, 4 bytes

          "Polled consumption",             $ parameter name
      "",                       $ units string
      "",
      0, 5, 0,                  $ min, max, default (0)
      0, 0, 0, 0,               $ mult, div, base, offset scaling
      , , , ,          $ scaling links not used
          0;                    $ decimal places
```

Modbus Ethernet TCP/IP

History

Modbus Ethernet TCP/IP and the ISO model

Physical layer

Link layer

Application layer

Profiles

Strengths - Weaknesses

History

	TCP - IP	Ethernet
1960	The DoD finances a project about "packet switching"	
1970	Development of the ARPANET network (IBM)	Experimental version of Ethernet defined by XEROX
1975		Ethernet principles defined by XEROX
1980	The INTERNET is launched: TCP/IP developed in current formats	First specification of Ethernet by XEROX, DEC and INTEL
1982		Version 2 of the Ethernet specification
1983	TCP/IP becomes the standard for long-distance networks	
1985		IEEE 802.3 standardization of CSMA/CD networks
1987	Growth rate 15%	
1996	Growth rate 60%	

Modbus

1999	Schneider Transparent Factory

http://www.transparentfactory.com/

Modbus Ethernet TCP/IP and the OSI model

Ethernet only covers the first 2 layers of the OSI model

7	APPLICATION	Modbus	HTTP	FTP	BootP DHCP	---
6	PRESENTATION	EMPTY				
5	SESSION	EMPTY				
4	TRANSPORT	TCP				
3	NETWORK	IP				
2	LINK = LLC + MAC	CSMA/CD				
1	PHYSICAL	Ethernet V2 or 802.3				

Physical layer

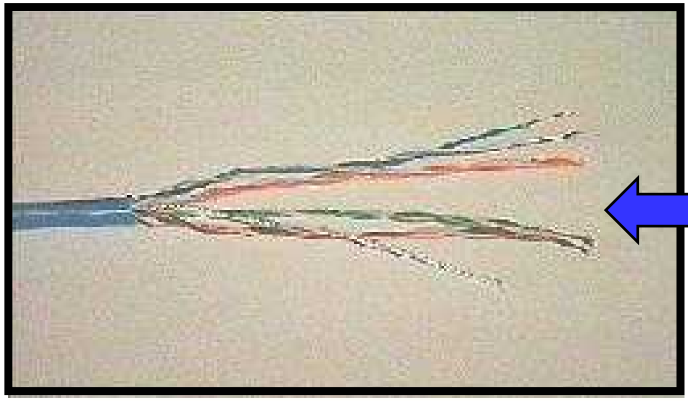
Topology:	**Free** Bus, star, tree or ring
Maximum distance:	**Depends on medium and speed** Minimum: 200 m on 100 base TX Maximum: 40,000 m on 10 base F
Speed:	**10 Mbps - 100 Mbps - 1 Gbps** 1 Gbps in office automation
Max. no. of devices:	**Depends on medium** Minimum: 30 per segment on 10 base 2 Maximum: 1024 on 10 base T or 10 base F

Transmission media

Ethernet is available on three types of medium:

	Name	Description	Speed	Max. length	Max. no. of stations/segment
Coaxial cable	10 base 5	Thick Ethernet	10 Mbps	500 m	100
	10 base 2	Thin Ethernet	10 Mbps	185 m	30
Shielded twisted pair	10 base T	Twisted pair	10 Mbps	100 m	1024
	100 base TX	Twisted pair cat. 5	100 Mbps	100 m	? ? ?
Optical fibre	10 base F	2 fibres	10 Mbps	2000 m	1024
	100 base FX	2 fibres	100 Mbps	2000 m	? ? ?

Twisted pair



Used increasingly, even at 100 Mbps

UTP **- Insulated pairs of copper wires twisted together**

Multiple colour-coded pairs enclosed in a plastic sleeve

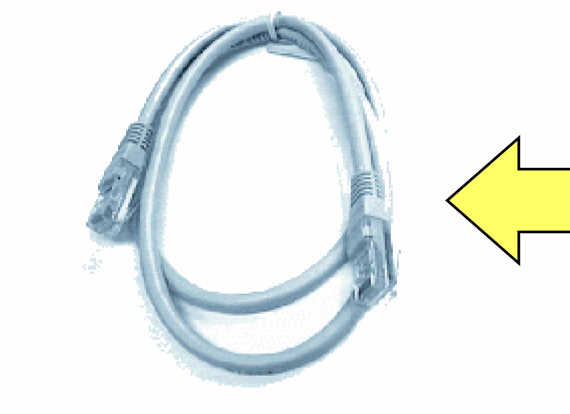
Faster than coaxial cable

STP - **Indivisible pairs enclosed in a shielding with aluminium foil**

Category 5 (Cat 5) **– The most common for IT networks**

Cat 5 = 100 Mbps (specification pending)

Cat 3 = 10 Mbps



Uses RJ45 connector

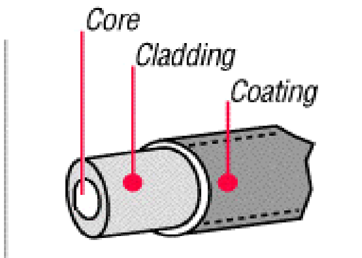
Optical fibres

Optical fibres are popular because they are secure (absence of electrical currents), compact and immune to noise and electromagnetic interference.

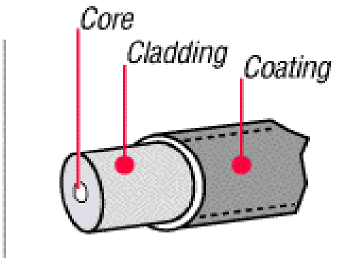
They support very long segment lengths (max. 2 km).

They are often used as backbones.

Three component parts:



Multimode Fiber (MM)



Single-Mode Fiber (SM)

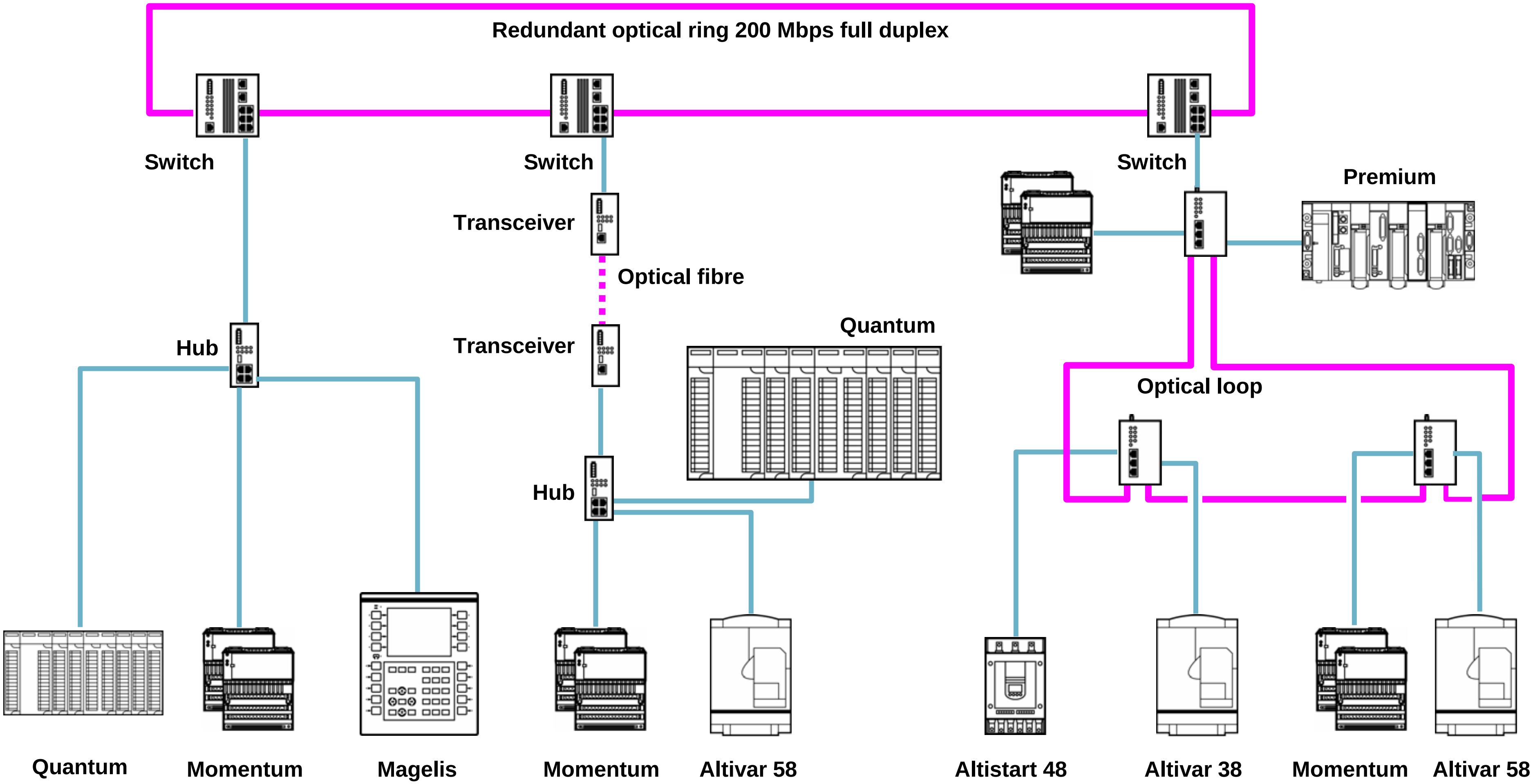
Core – Carries the light beam (glass or plastic)

Cladding – Glass tube which reflects any interference light in the core

Coating – Protects the core and the optical cladding

Multimode fibre is the most popular type as it is the least expensive and easier to use.

Example architecture



Transport network link layers

Medium access method: CSMA/CD

Carrier Sense Multiple Access with Collision Detection

The stations listen to the transmission medium and wait until it is free to send.

If a collision is detected, each station continues to send in order that the collision is seen by the entire network.

The stations resend their message after a random period of time has elapsed.

Determinism: **Resolved using segmentation**
Load factor < 10%

Transmission method: **In packets**
or IP datagrams, 64 to 1500 bytes

Max. size of useful data: **1442 bytes per packet (APDU)**

Transmission security: **CRC32** at link layer level
Acknowledgement at TCP link level
Response at application level (UNITE/Modbus)

The major application protocols

HTTP: **HyperText Transfer Protocol = Web**

File transfer in HTML format

FTP: **File Transfer Protocol**

File transfer based on the client/server model

SNMP: **Simple Network Management Protocol**

Network management: Configuration, monitoring, administration

DNS: **Domain Name Service**

Translates the symbolic name of a network node into an IP address

Application protocols

BOOTP:	**Bootstrap Protocol** IP address assignment by a server
TELNET:	Terminal interfacing with devices in half duplex mode Encapsulated ASCII format
UNITE:	Protocol based on the client/server model created by Telemecanique
MODBUS:	Protocol based on the client/server model created by Modicon
I/O scanning:	Period I/O updated by automatic sending of Modbus requests

Transparent Ready implementation classes

Implementation classes define a **list of services** to be implemented in order to ensure the interoperability of Schneider **Transparent Ready** products.

These classes are defined for 4 device families:

- Controllers: PLC, numerical controllers, etc.
- Devices: Drives, motor starters, remote I/O
- Gateways:
- HMI/SCADA

Implementation classes are identified by:

a **letter from A to Z** relating to WEB services

followed by a **number from 00 to 99** relating to user services and communication

and an **ASCII suffix** relating to the physical layer.

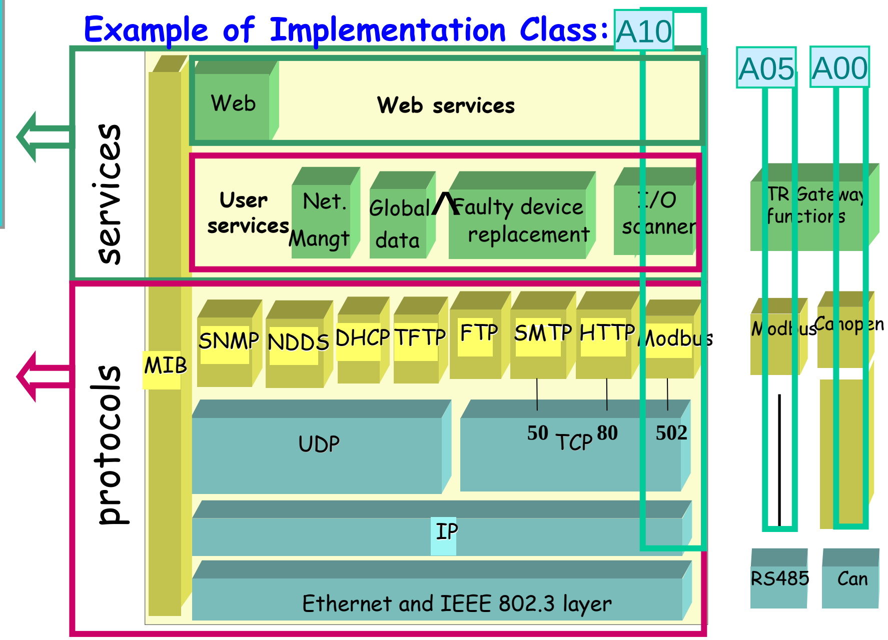
Implementation classes

Web services level

server	client
A: without Web	A: without Web
B: Web Basic	Z: Web Basic
C: Web Configurable	Y: Web Regular
D: Web Active	X: Web Active
E: Web Distributed	w: Web Distributed

User & communication level services

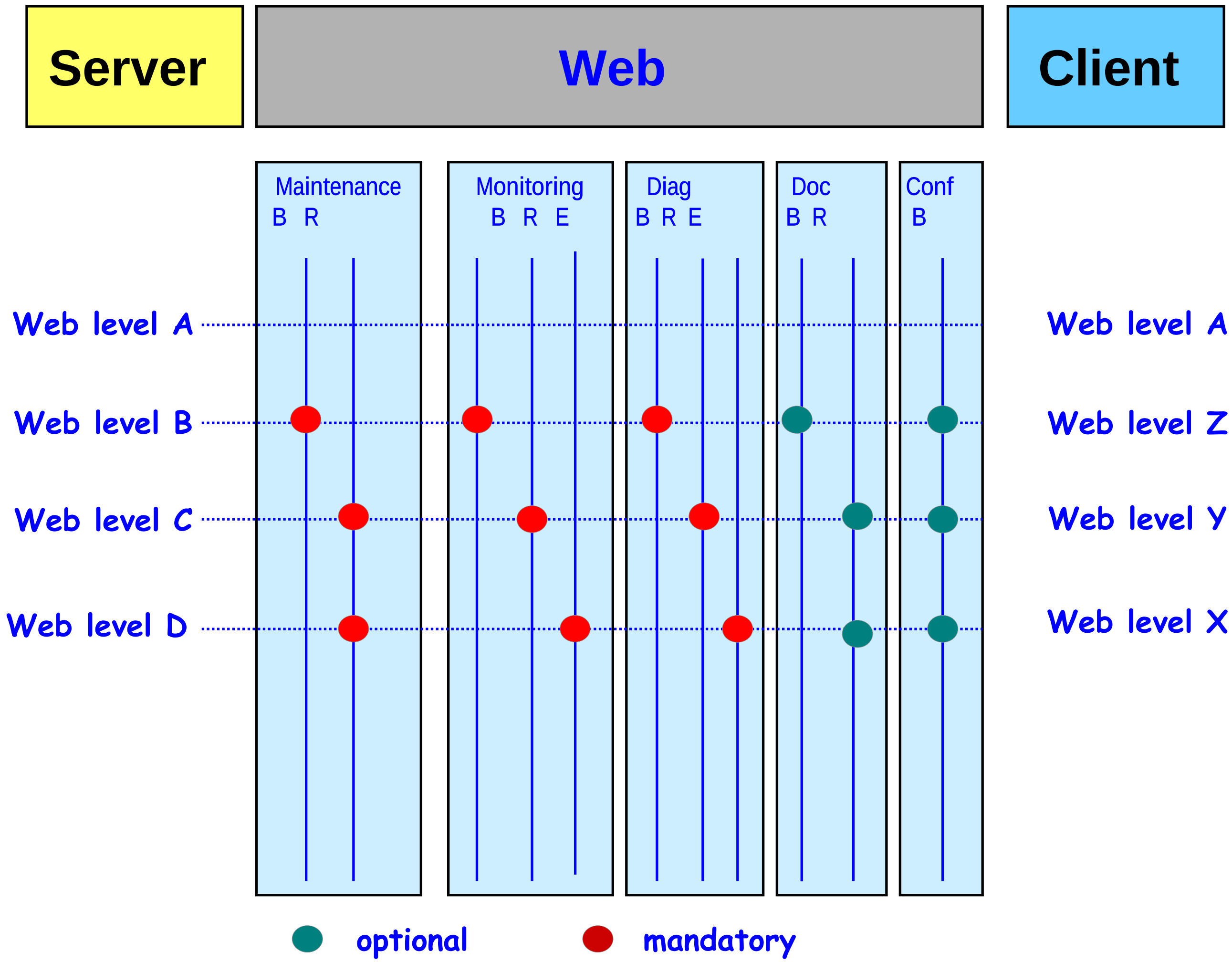
- 00: without Modbus
- 01: modbus Basic access
- 05: modbus Regular access
- 10: modbus on TCP-IP basic access
- 20: modbus on TCP-IP management access
- 30: modbus on TCP-IP added values access
- 40: distributed control on TCP-IP



Examples:

A10-Eth10/100	Modbus on Ethernet TCP-IP (10/100 Mbs), no Web
A05-SL-RS485	Modbus on RS485, no Web
A00-Can	for Can Open: profiles to be defined
C30-Eth100	Modbus on Ethernet TCP-IP (100 Mbs) + com & Web services

Web services



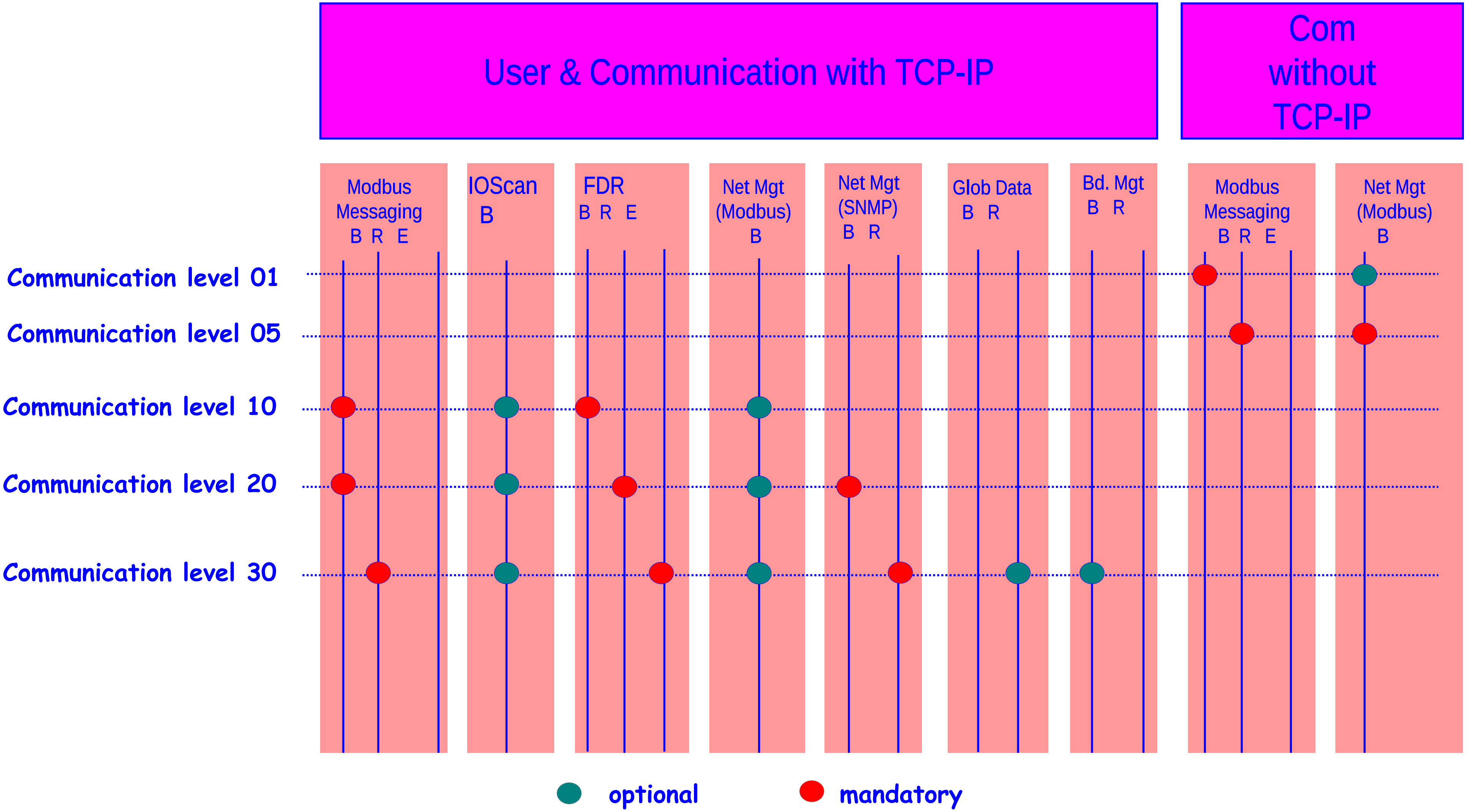
Server

A: without Web
B: Web Basic
C: Web Configurable
D: Web Active
E: Web Distributed

Client

A: without Web
Z: Web Basic
Y: Web Regular
X: Web Active
w: Web Distributed

User and communication services



Profibus-DP

History

Profibus-DP and the ISO model

Physical layer

Link layer

Application layer

Profiles

Strengths - Weaknesses

History

- In 1987, the German federal minister for technological research and development creates a "Fieldbus" working group comprising 13 organizations including SIEMENS and 5 research institutes.

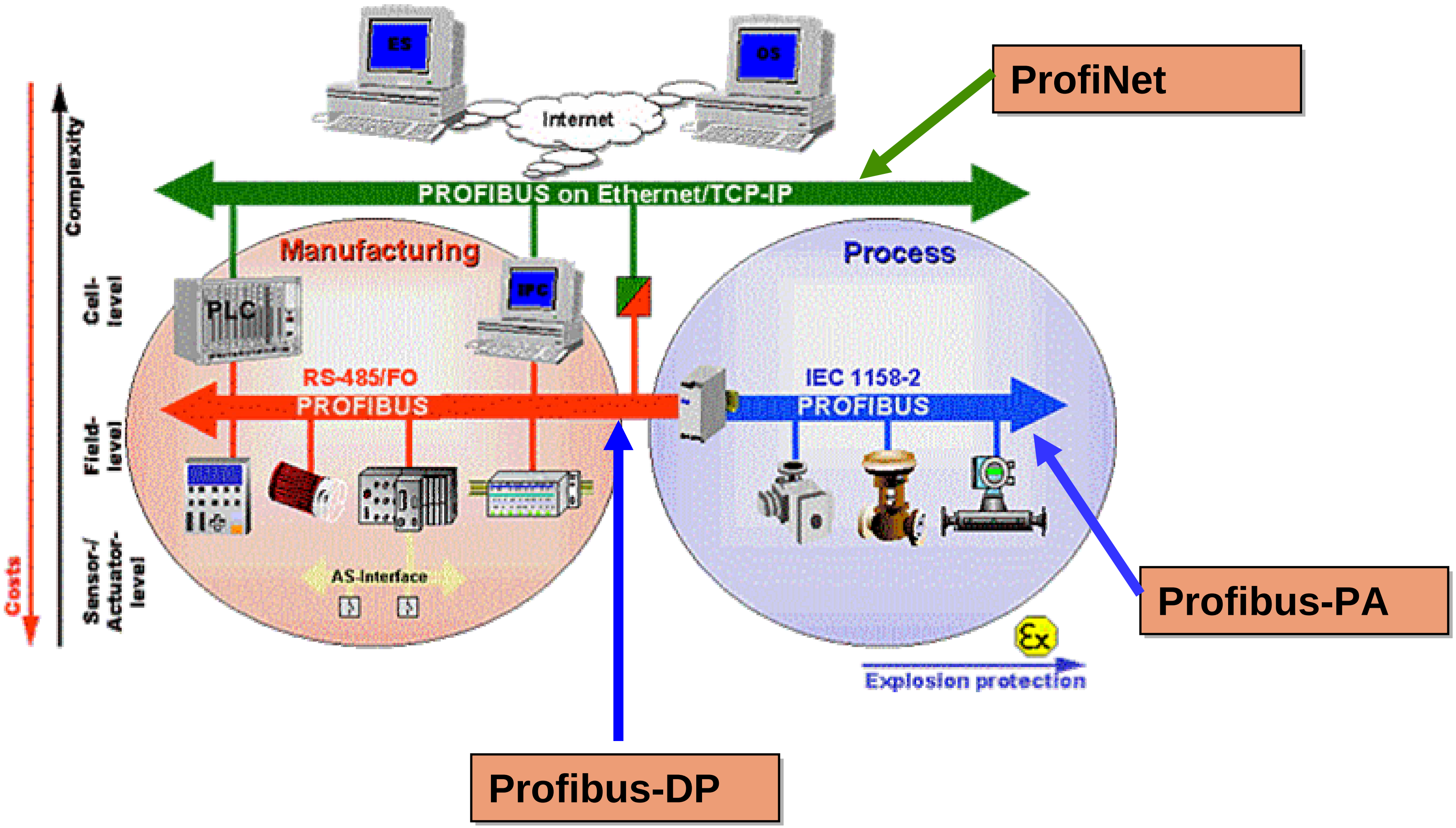
  Profibus (PROcess FIeldBUS) is born.

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

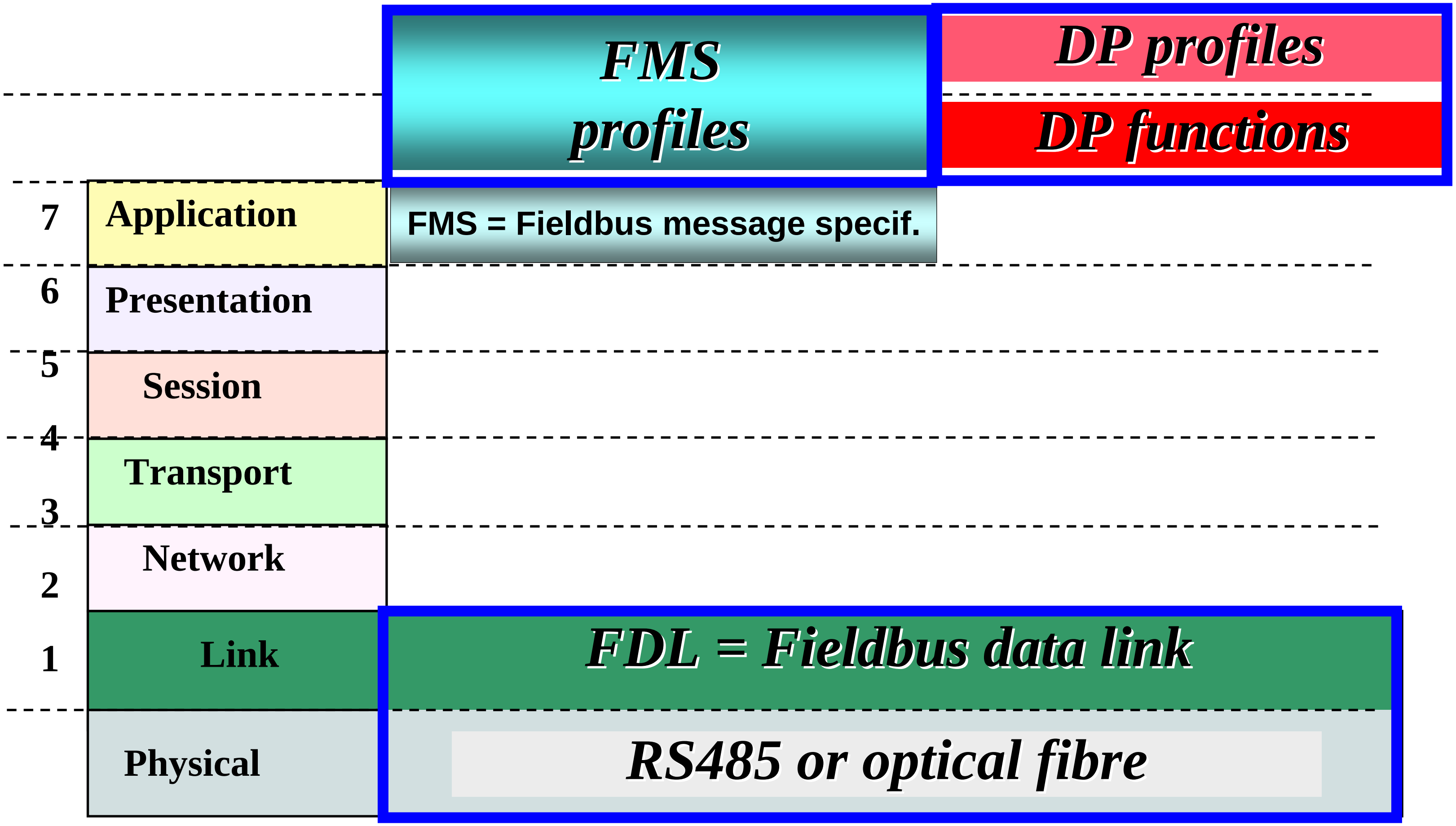
- User clubs in 20 of the world's most industrialized countries provide support in native languages. These centres of competence are governed by the "PROFIBUS International" (PI) organization, which has more than 750 members.

  http://www.profibus.com/

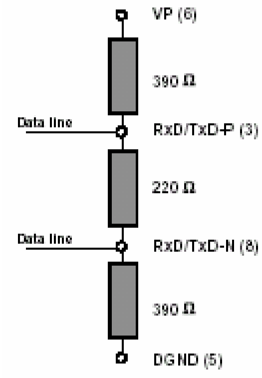
The 3 versions of Profibus



Profibus and the ISO model



Physical layer



Topology: **Bus with active line terminators**

Maximum distance: **Depends on medium and speed**

Minimum: 100 m at 12 Mbps without repeaters

Maximum: 4800 m at 9.6 kbps with 3 repeaters

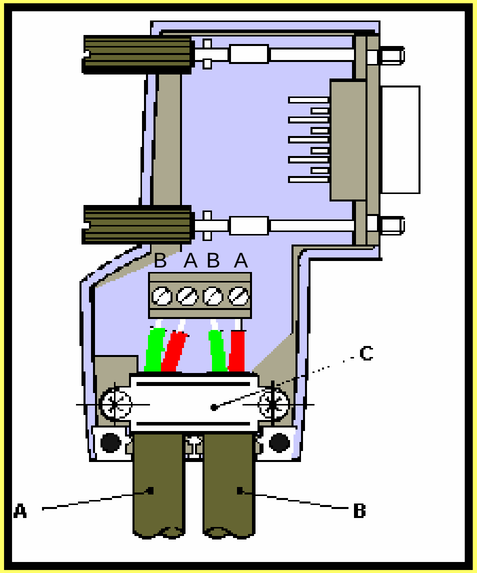
Speed: **9.6 Kbps to 12 Mbps**

Max. no. of stations: **32 without repeaters**

124 with 3 repeaters

Types of connection

IP20

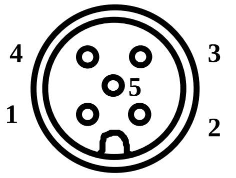


9-pin Sub D

Female, product side with or without line terminator

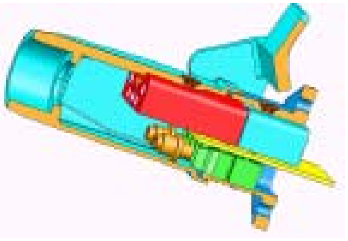
IP65

M12 connector



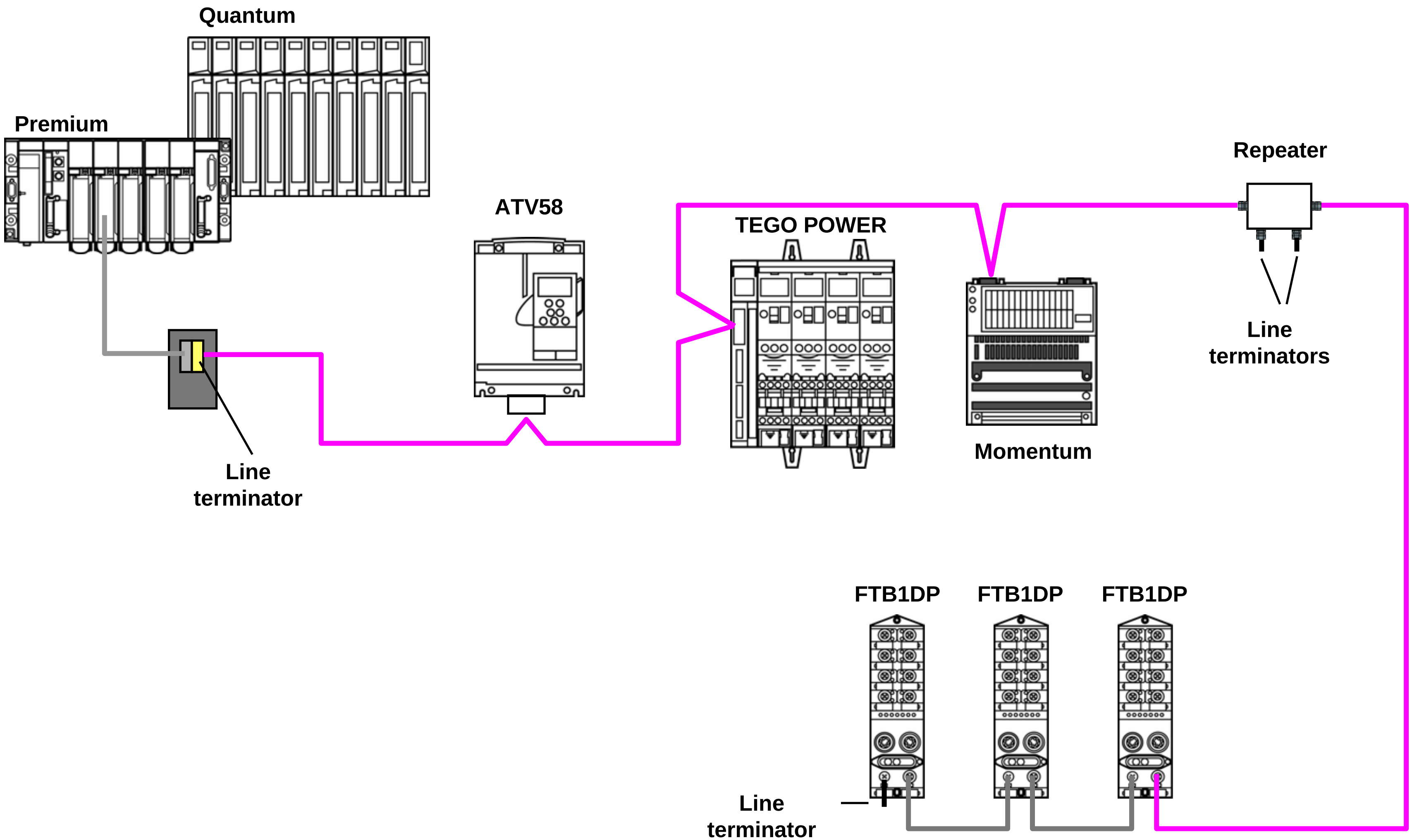
Female, product side

Han-Brid



DESINA recommendation

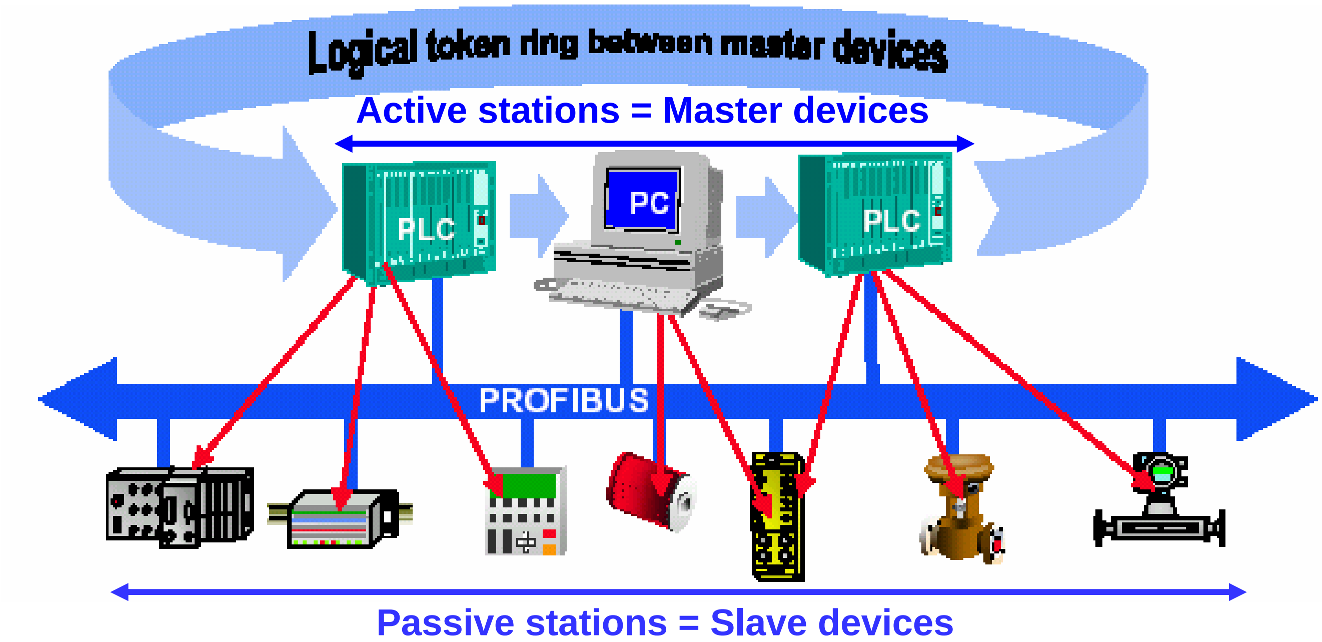
Example architecture



Link layer

PROFIBUS uses a **hybrid** access method:

- Communication between active stations is based on the **token ring** concept.
- Passive stations (slaves) use the **master-slave** concept.



Token ring

The **token ring concept** ensures that access to the bus is provided to each master device in a predefined time window.

The token is a special telegram sent by a master which must be distributed to the other masters on the ring in a maximum configurable period of time.

Master - Slave

The **master-slave concept** enables the master in possession of the token to access slaves assigned to it (passive stations) as well as other masters (FMS message handling).

Messages destined for slaves and responses to them are called
PPOs: Parameter Process Objects.

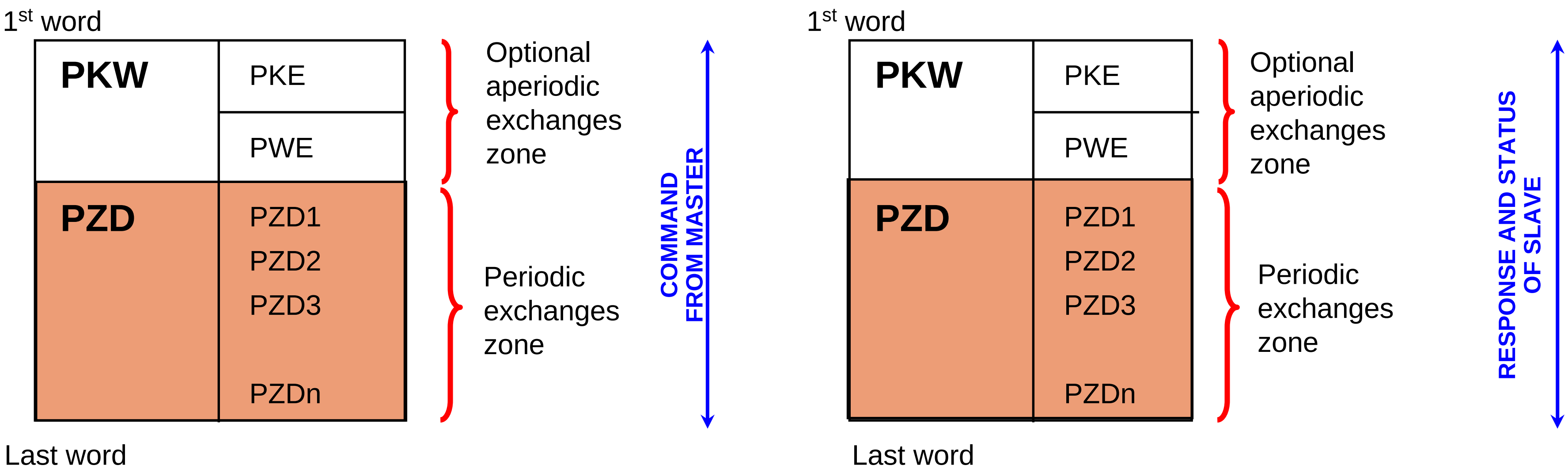
Profibus-DP can operate with a single master (mono-master mode).

The Profibus-DP Premium master module does not support master-to-master communication (FMS).

Description of PPO

The master sends a **cyclic request** to the slave

The master receives a **cyclic response** from the slave



All words are exchanged cyclically although aperiodic exchanges are used when necessary.

PKW = Parameter - Kennung - Wert = Parameter - Address - Value
PKE = Parameter - Kennung = Parameter address
PWE = Parameter - Wert = Value of the parameter whose address is contained in the PKE
PZD = Prozeßdaten = Process data

Using PKWs

Output data	
PKW	Description
Word 1	PKE output
Word 2	R/W output
Word 3	0
Word 4	Output

Input data	
PKW	Description
Word 1	PKE input
Word 2	R/W/N input
Word 3	0
Word 4	PWE input

PKE output:
Bits 0 to E: Address of variable
Bit F: = 0 Single read or write
= 1 Continuous read or write
R/W output:
= 16#0052 = Read
= 16#0057 = Write
PWE output:
= If write: Write value

PKE input:
Copy of the PKE output value
R/W/N input:
= 16#0052 Read correct
= 16#0057 Write correct
= 16#004E Read or write error
PWE input:
: If read correct value of variable
: If write correct copy of PWE output value
: If error
= 0: Address incorrect
= 1: Write refused

Application layer

Data exchanges:	**Process: Cyclic exchanges** **Parameters, diagnostics: Aperiodic (PKW)**
Max. size of data:	**244 bytes of PPO**
Interoperability:	**Product certified by the Profibus organization**
Interchangeability:	**Communication and application profiles**

DP communication profiles

Three types of station are defined:

DP master class 1 (DPM1):	Programmable controllers such as PLCs, PCs, etc.
DP master class 2 (DPM2):	Development or diagnostics tool
DP slave:	Peripheral device performing cyclic exchanges with "its" active station

The Profibus-DP TSX PBY 100 Premium module is a subset of DPM1

DP application profiles

Application profiles complete the standard for a given area of application.

Examples:

- **Numerical controllers and robots**

Based on sequential diagrams, movements and commands are described from the point of view of the control system.

- **Encoders**

Based on the connection of rotary, angle and linear encoders, and based on the definition of functions (scaling, diagnostics, etc.).

- **PROFIDRIVE variable speed drives**

Based on the basic functions of the drive: drive commands and states are described.

- **Process control and supervision (HMI)**

Specifies how control (and supervision) devices are linked with higher-level control system components. Uses the extended functions of PROFIBUS-DP relating to communication.

GSD files

The characteristics of a PROFIBUS device are described in the form of an "electronic device data sheet" (GSD) in a predefined format.

GSD files must be provided by all PROFIBUS device manufacturers.

General specifications
This section contains information about the manufacturer, the product name, hardware and software versions, speeds supported, etc.

Specifications relating to masters
This section contains all the parameters relating to masters, such as the maximum number of slaves and up/downloading options. This section does not exist for slave devices.

Specifications relating to slaves
This section contains the specifications relating to slaves, such as the number and type of I/O variables, diagnostic texts, information about modules for modular products, etc.

FIPIO

History
FIPIO and the ISO model
Physical layer
Link layer
Application layer
Profiles
Strengths - Weaknesses

History

- It all started with a working group managed by the Science and Technology Department of the Ministry for Industry and Research (France) including the manufacturers TELEMECANIQUE, MERLIN GERIN, CGEE, ALSTHOM and CSEE.

This group worked on the FIP specification during the years 1983-1985.

- The WorldFIP users and manufacturers group was created in 1987 under the name CLUB FIP.

http://www.worldfip.org/

WorldFIP meets the requirements of standards EN 50170 and IEC 61158.

FIPIO and the ISO model

DRIVECOM drives | HMI MMI COM | Welding robots | Etc.

7	APPLICATION	Process data channel + PCP message handling
6	PRESENTATION	EMPTY
5	SESSION	EMPTY
4	TRANSPORT	EMPTY
3	NETWORK	EMPTY
2	LINK = LLC + MAC	Master-slave with a single frame (shift register)
1	PHYSICAL	RS 485

Physical layer

Medium: **Shielded twisted pair or optical fibre**

Topology: **Bus type**

With tap link or daisy chain connections + line terminators

Maximum distance: **1000 m for an electrical segment**
3000 m for an optical segment
15 000 m with electrical repeaters

No. of repeaters + no. of stations =< 36
No. of repeaters x 0.5 + total length in km < 22

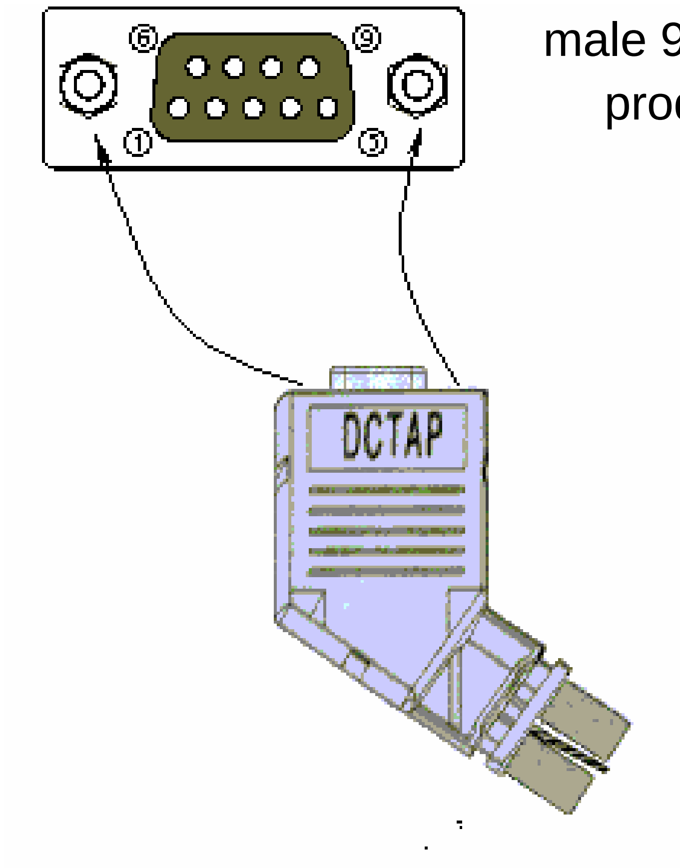
Speed: **1 Mbps**

Regardless of cable length

Max. no. of devices: **127**

1 master and 126 slaves
Max. 32 devices per segment

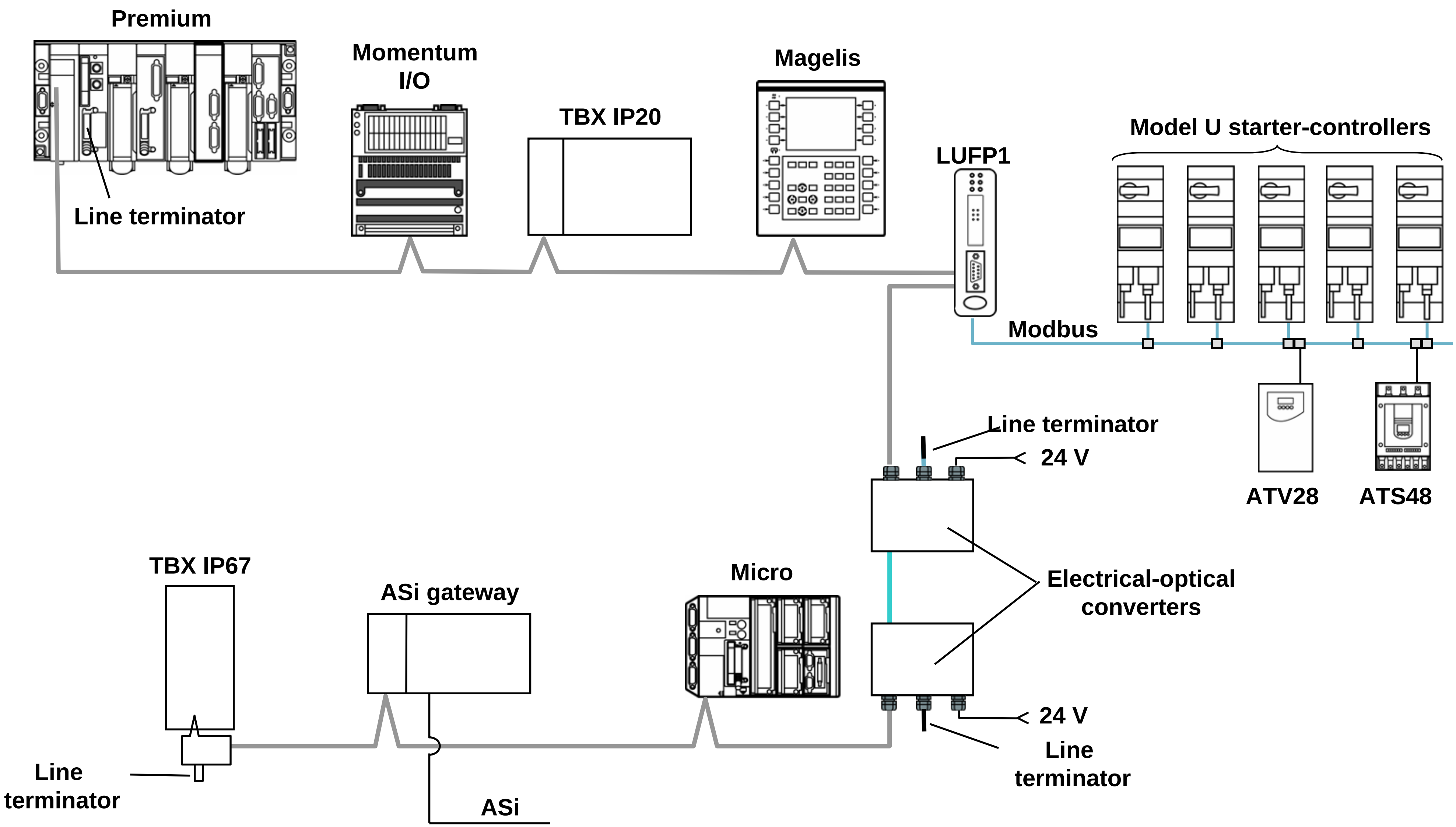
Standardized 9-pin SUB D connector



male 9-pin SUB D
product side

To FIPIO main cable or
tap junction

Example architecture



Link layer

Medium access method: **Master/Slaves (bus arbitrator)**

The bus arbitrator derives the list of variables (identifiers) to be scanned as well as their periodicity (data contained in the device profile) from the system configuration.

Communication model:

Periodic exchanges: **Producer/Consumer**

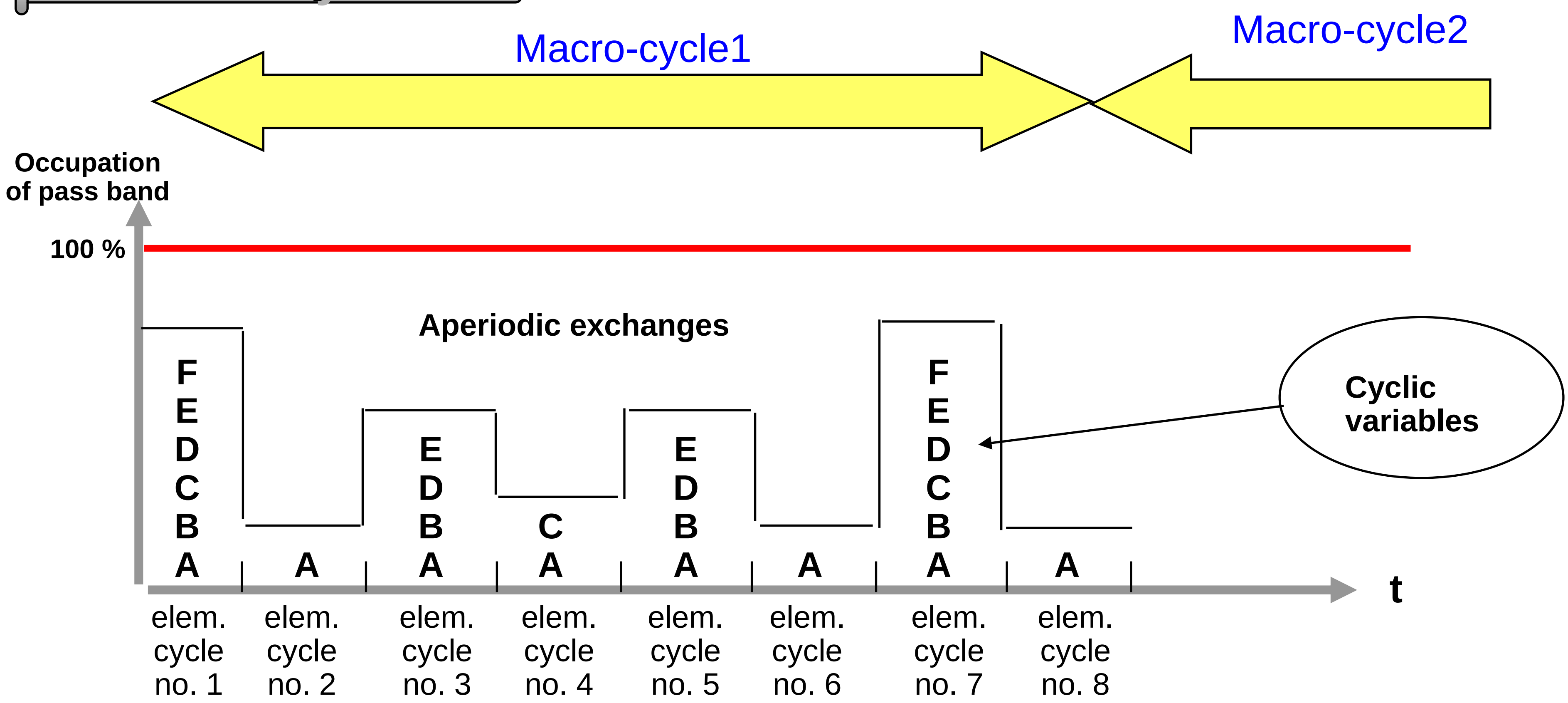
When the bus arbitrator requests the distribution of a variable (identifier), the unique producer of this variable detects this and distributes the variable.

The consumer station or stations detect the variable and the bus arbitrator moves to the next identifier.

Aperiodic exchanges: **Client/Server**

Once the periodic exchanges are complete, the bus arbitrator processes the aperiodic requests stored in a separate buffer (list of identifiers).

Link layer



Each variable is scanned at its own pace without being disturbed by aperiodic exchanges.

Profile families

3 profile families are defined:

FRD = FIPIO **Reduced** Device Profile

FSD = FIPIO **Standard** Device Profile FSD P: FIPIO Simple Device Profile

FED = FIPIO **Extended** Device Profile

The profile selected depends on:

- The number of cyclic variables to be exchanged
- The number of configuration variables
- The number of adjustment variables
- The number of diagnostic variables
- The structure of the device

Profile overview

Standard profile	FRD	FSD	FED
Cyclic variables			
Input acquisition	2 words	8 words	32 words
Output control	2 words	8 words	32 words
Configuration variables	-	16 words	30 words
Adjustment variables	-	32 words	30 words
Commands			
Specific command	-	-	8 words
Diagnostics			
Validity of inputs	1 byte	1 byte	1 byte
Specific status	-	-	8 words

Profile denomination syntax

4 fields are used to identify a profile:

FSD C8 P

Family

Structure

No. of I/O

Parameters can/cannot be set

- FRD Reduced
- FSD Standard
- FED Extended

- C Compact
- M Modular

- 2 words
- 8 words
- 32 words

- P Parameters can be set
- \- Parameters cannot be set

Interbus

- History
- Interbus and the ISO model
- Physical layer
- Link layer
- Application layer
- Profiles
- Strengths - Weaknesses

History

- **1983** — Protocol specifications by Phoenix Contact
- **1985** — First prototypes
- **1987** — First products launched
- **1990** — First profiles
- **1993** — Creation of the international Interbus Club
- **2001** — EN 50254 approval
  350,000 networks installed
  4 million nodes connected
  2700 products

Interbus and the ISO model

DRIVECOM drives | HMI MMI COM | Welding robots | Etc.

7	APPLICATION	Process data channel + PCP message handling
6	PRESENTATION	EMPTY
5	SESSION	EMPTY
4	TRANSPORT	EMPTY
3	NETWORK	EMPTY
2	LINK = LLC + MAC	Master-slave with single frame (shift register)
1	PHYSICAL	RS 485

Physical layer

Medium: **Shielded twisted double pair**

1 pair for receiving, 1 pair for sending

Topology: **Ring type**

Viewed from the outside, resembles a bus topology with the connecting cable containing the signal loop-back.

Maximum distance: **400 m between 2 devices**
12.8 km total

Speed: **500 Kbps**

Max. no. of devices: **512**

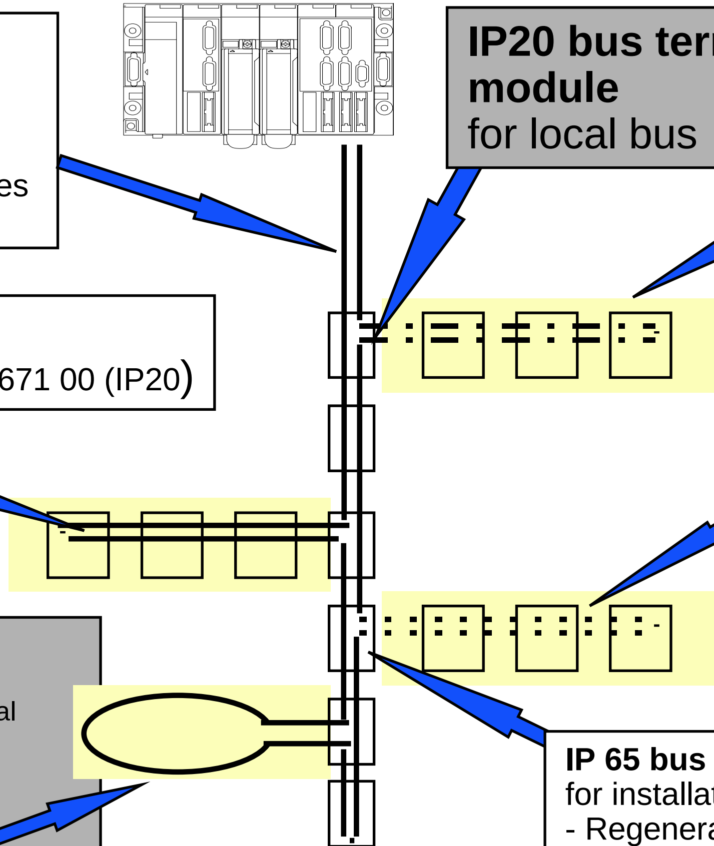
The different types of bus

Remote bus
(main bus)
- RS 485 point-to-point
- Max. 256 devices
- Max. 400 m between 2 devices
- Total length: 12.8 km

IP20 bus terminal module
for local bus

Local bus TTL
(designed for a cost-effective installation of a remote sub-station in an enclosure)
- Max. 8 devices
- Max. 1.5 m between 2 devices
- Total length: 10m
- Max. current: 800 mA

Remote bus branch
Bus terminal module: 170 BNO 671 00 (IP20)



Installation bus
(variant of remote bus
+ sensor power supply voltage)
- RS 485
- With 24 V, 4.5 A max. power supply
- Max. 40 I/O modules
- Max. 50 m between 2 devices
- Total length: 50 m

Interbus sensor loop
(direct connection of digital and analog sensors on Interbus-S via a bus terminal module)
- 1 unshielded pair + 24 V
- Max. 32 devices
- Max. 10 m between 2 devices
- Total length: 100 m

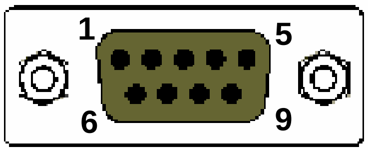
IP 65 bus terminal module
for installation bus
- Regenerates data
- Supplies 24 V/4.5 A
Bus terminal module: 170 ENO 396 00 (IP65)

No Schneider devices on the local bus or "*sensor loop*"

Types of connection

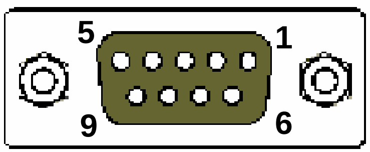
IP20

9-pin Sub D IN



Male, product side

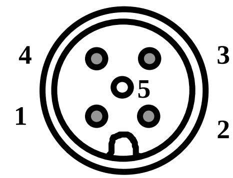
9-pin Sub D OUT



Female, product side

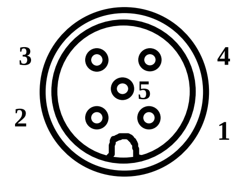
IP65

M12 IN connector



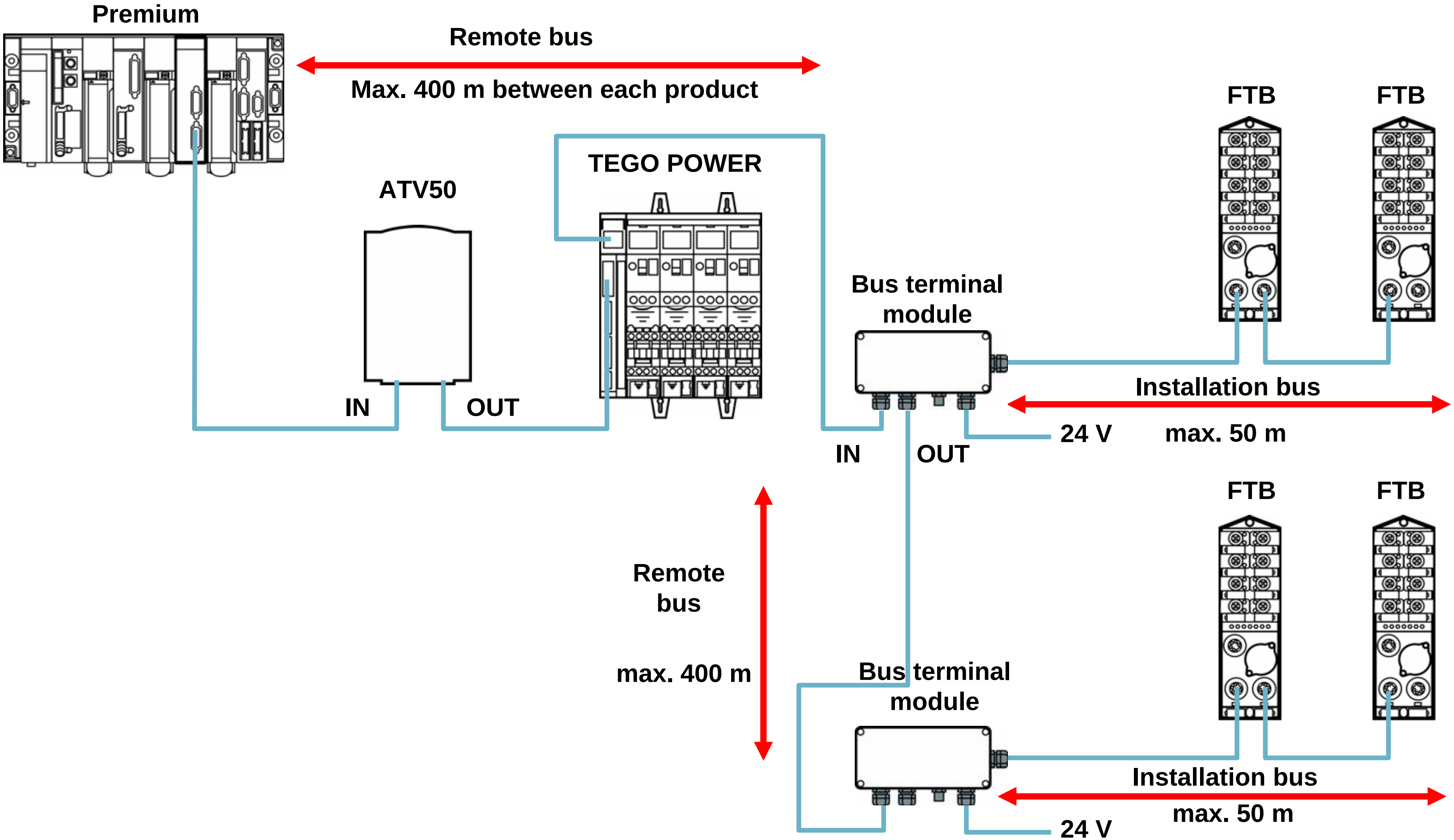
Male, product side

M12 OUT connector



Female, product side

Example architecture



Link layer

Medium access method: **Master/Slaves**

Transmission of a single frame containing both sensor (input) and actuator (output) data.

This single frame is managed like a shift register with a maximum of 256 words. Each slave (station) is a component of the register.

The frame structure is hybrid, enabling 2 data classes to be supported (maximum 32 words per device):

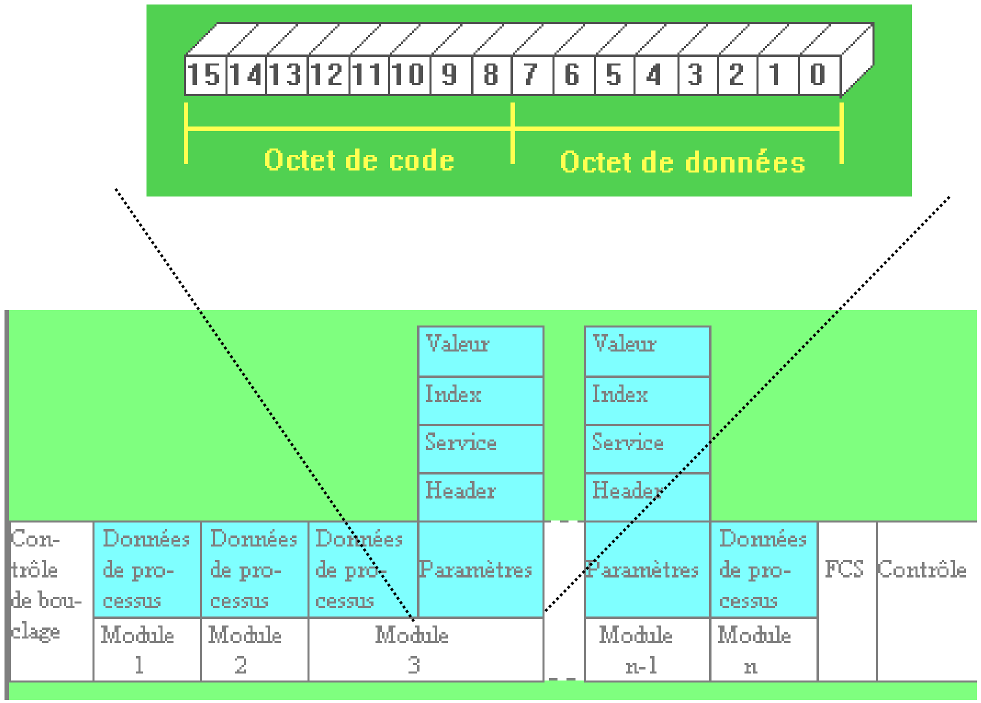
- **Cyclic process data** (periodic slave I/O words)
- **Acyclic parameter data** (fixed memory space)

Acyclic exchanges

Acyclic data is transmitted using PCP

PCP = Peripherals Communication Protocol

which fragments parameter data.



Profiles

Interbus profiles define for a product family:

- The recognition of a device by means of its ID code
- The format of command data (outputs) and status words (inputs) exchanged
- The status chart

A new device can only be integrated into the **CMD Tool** network configuration tool by adding it to a database managed by **PHOENIX CONTACT** (no EDS file).

Modbus

- History
- Modbus and the ISO model
- Physical layer
- Link layer
- Application layer
- Profiles
- Strengths - Weaknesses

History

The **MODBUS protocol** is a message handling structure created by **MODICON** in **1979** to connect PLCs to programming tools.

Today, this protocol is mainly used to set up master/client type communications with slaves/servers between intelligent devices.

MODBUS is **independent of the physical layer**.

It can be implemented on **RS232, RS422, or RS485** links as well as on a **wide variety of other media** (e.g.: optical fibre, radio, etc.).

Modbus serial link and the ISO model

MODBUS on a serial link operating at 1200 to 56 Kbps with a master/slave access method.

7	Application	*Modbus*
6	Presentation	
5	Session	
4	Transport	
3	Network	
2	Link	*Master/Slave*
1	Physical	*RS485*

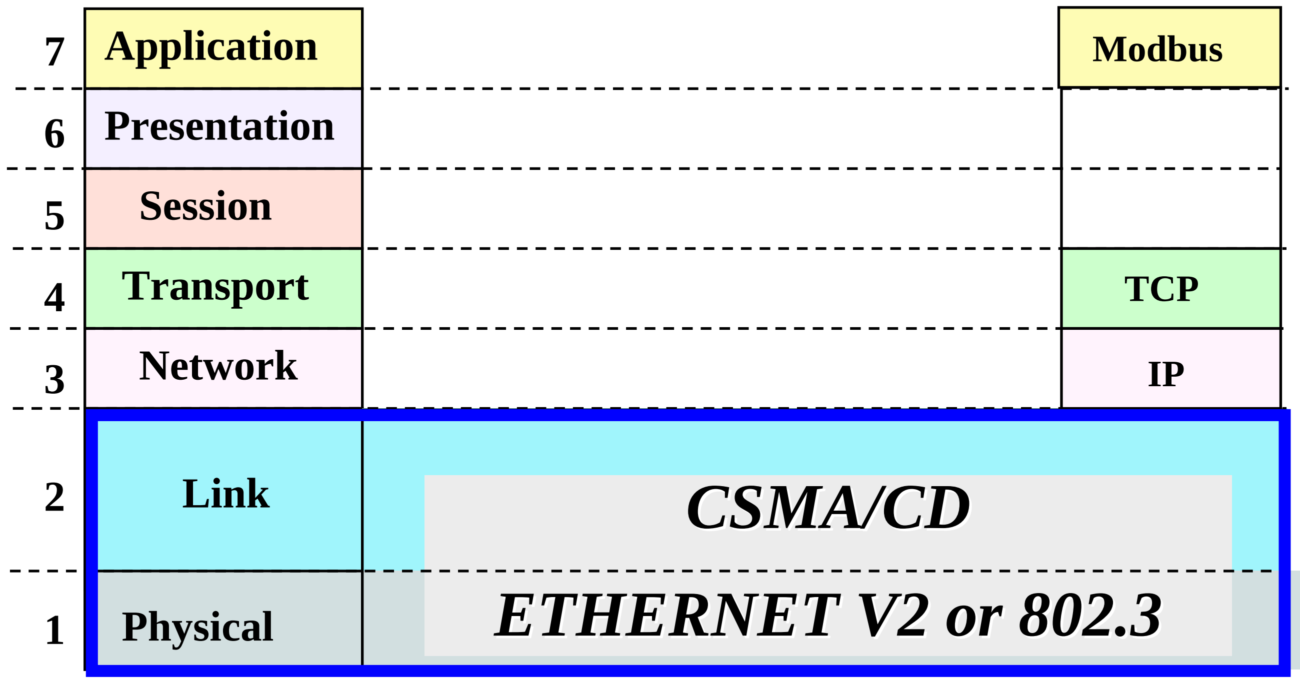
Modbus Plus and the ISO model

MODBUS PLUS is a bus operating at 1 Mbps based on a token ring access method which uses the MODBUS message handling structure.

7	Application	*Modbus*
6	Presentation	
5	Session	
4	Transport	
3	Network	
2	Link	*802.4 token ring*
1	Physical	*RS485*

Modbus Ethernet TCP/IP

MODBUS Ethernet TCP/IP uses TCP/IP and Ethernet 10 Mbps or 100 Mbps to carry the MODBUS message handling structure.

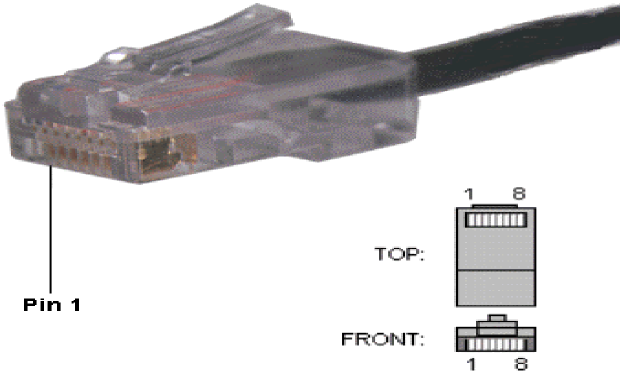


RS485 physical layer

Medium:	**Shielded twisted pair**
Topology:	**Bus type** With tap links and line terminators
Maximum distance:	**1300 m without repeaters**
Speed:	**19,200 bps (56 Kbps on some products)**
Max. no. of devices:	**32** 1 master and 31 slaves

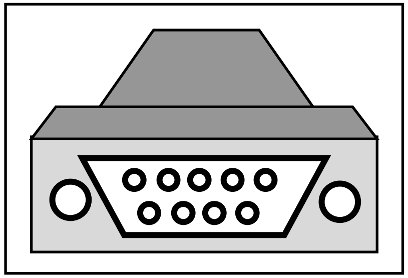
Connectors recommended by Schneider

TIA/EIA-485/RJ45



Female, product side

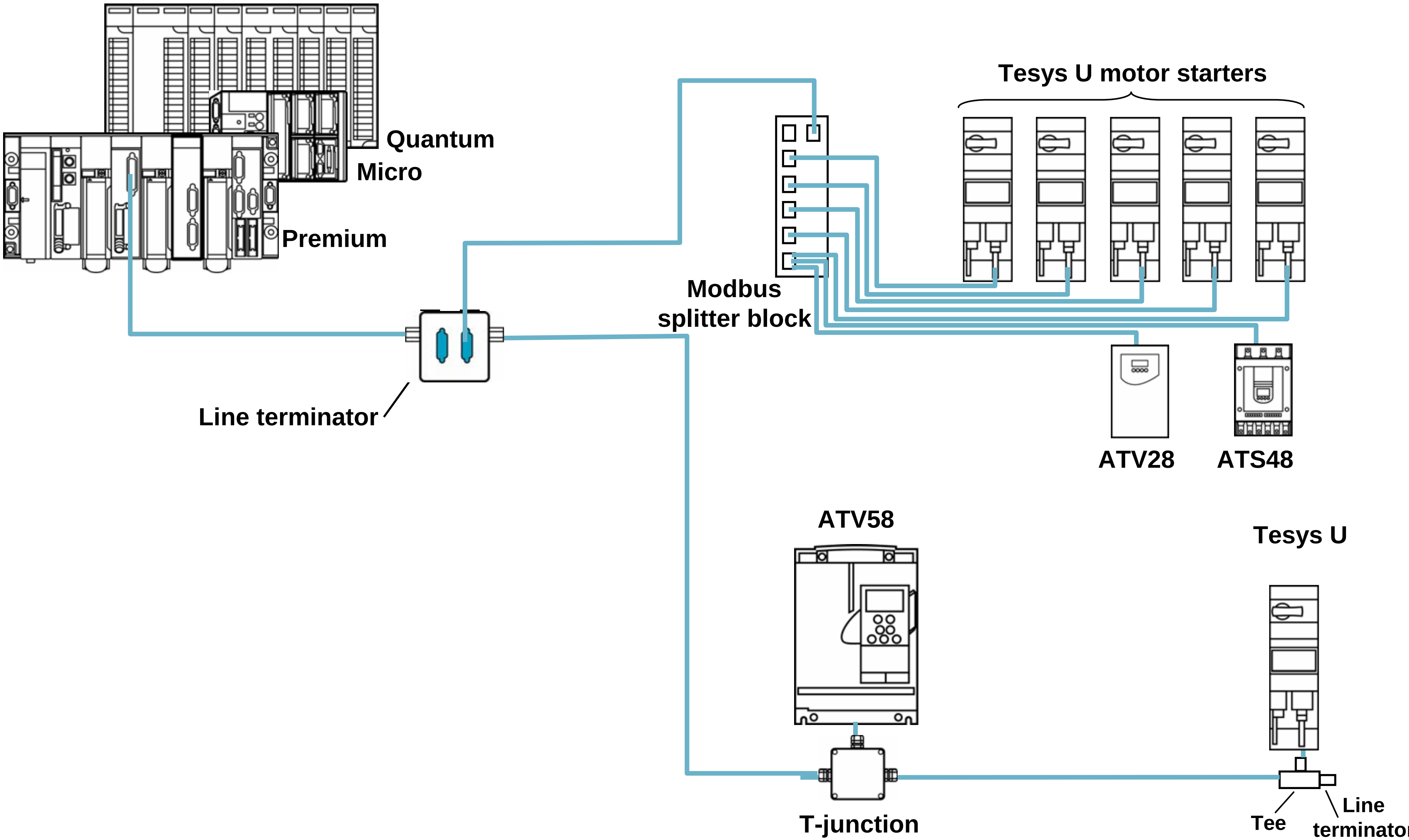
TIA/EIA-485/9-pin SUB-D



Female, product side

Male, product side

Example architecture



Link layer

Medium access method:	**Master/slave**
Transmission method:	**Client/server**
Max. size of useful data:	**120 PLC words**
Transmission security:	**LRC or CRC** **Start and stop delimiters** **Parity bit** **Continuous flow**

Modbus ASCII and Modbus RTU

There are 2 versions of the MODBUS protocol:

- ASCII mode
  Each byte in the frame is sent in 2-character ASCII format.

- RTU mode
  Each byte in the frame is sent in 2-character 4-bit hexadecimal format.

The main advantage of RTU mode is that it sends data more quickly.

ASCII mode allows the insertion of a time interval of one second between 2 characters without generating a transmission error.

Structure of a Modbus frame

The structure of a Modbus frame is the same for requests (message from the master to the slave) and responses (message from the slave to the master).

Modbus RTU

silence	Address	Function	Data	Checksum	silence

Silence >= 3.5 characters

Modbus ASCII

:	Address	Function	Data	Checksum	CR	LF

3A Hex

0D Hex 0A Hex

Example frame in RTU mode

- Function code = 3: Read n words

Request:

1 byte	1 byte	2 bytes	2 bytes	2 bytes
Slave address	Funct. code = 3	Address of 1st word	Number of words to read	**CRC16**

Response:

1 byte	1 byte	2 bytes	2 bytes		2 bytes	2 bytes
Slave address	Funct. code = 3	Number of bytes read	Value of 1st word	....	Value of last word	**CRC16**

Implementation classes

Modbus message handling implementation classes are a subset of the **Transparent Ready** project which defines a **list of services** to be implemented in order to ensure the interoperability of Schneider products.

3 classes are defined for the server device family (drives, motor starters, remote I/O, etc.).

The classes correspond to a **list of Modbus requests to be supported**.

Basic:	Access to words and identification
Regular:	Basic + bit access + network diagnostics
Extended:	Regular + other types of access

Comparison at physical level

	ASi	CANopen	DeviceNet	Ethernet TCP/IP Modbus	Profibus-DP	FIPIO	Interbus	Modbus
Medium	Yellow flat ribbon cable Round unshielded cable Round shielded cable	Shielded twisted pair	Double shielded twisted pairs	Coaxial cable : 10base2 - 10base5 Shielded twisted pair 10baseT - 10baseTX Optical fibre 10baseF - 10 baseFX	Shielded twisted pair Optical fibre	Shielded twisted pair Optical fibre	Double shielded twisted pairs	Shielded twisted pairs
Max. distance without repeaters	100m	Acc. to speed: 25m to 1 Mbps 1km to 10 Kbps	Acc. to speed: 100m to 500Kbps 500m to 125Kbps	Twisted pair 100m Optical fibre 2000m	Acc. to speed: 100m to 12Mbps 1.2km to 10Kbps	1000 m twisted pair 3000 m optical fibre	400m	1300m
Max. distance with repeaters	300m	Depends on the type of repeater	Depends on the type of repeater	10km optical fibre	400 to 4800m acc. to speed	15km	12.8km	Depends on the type of repeater
Speed	166 Kbps	9 possible speeds from 10 Kbps to 1 Mbps	125, 250 or 500 Kbps	10/100Mbps	9.6 Kbps to 1 Mbps	1 Mbps	500 Kbps	up to 19200 bps
Max. number of devices	ASi V1: 1 master + 31 slaves ASi V2: 1 master + 62 slaves	128 1 master and 127 slaves	64 1 master and 63 slaves	64 I/O scanning and Modbus	Mono or Multi-masters 126 devices max	1 manager + 126 devices	512	32 1 master and 31 slaves

Comparison at link and application level

	ASi	CANopen	DeviceNet	Ethernet TCP/IP Modbus	Profibus-DP	FIPIO	Interbus	Modbus
Medium access method	Master Slaves	CSMA/CA	CSMA/CA	CSMA/CD	Token ring and master/slaves	Bus manager	Master Slaves Single frame	Master Slaves
Type and size of data exchanged	ASi V1: Cyclic: 4 IN bits 4 OUT bits Acyclic: 4 P bits ASi V2: Cyclic: 4 IN bits 3 OUT bits Acyclic: 3 P bits	Cyclic I/O: PDO 8 IN bytes 8 OUT bytes Acyclic: SDO Param./adjust. >8 bytes due to fractioning of information	Cyclic I/O: I/O messages 8 IN bytes 8 OUT bytes or >8 if fragmentation Acyclic: Explicit messages Param./adjust. >8 bytes due to fractioning of information	Cyclic I/O: I/O scanning 125 IN words 125 OUT words Acyclic: Param./adjust. via asynchronous messaging 507 words	Cyclic I/O: PZD 244 IN words 244 OUT words PKW = 1 word at once	Cyclic I/O: 32 IN words 32 OUT words Acyclic: Param.= 30 words Adust. = 30 words	Cyclic I/O: 256 I/O words Acyclic: 256 words via fragmentation	Acyclic variables 1920 bits 120 words

PLCs

	ASi	CANopen	DeviceNet	Ethernet TCP/IP Modbus	Profibus-DP	FIPIO	Interbus	Modbus
Zelio	Slave							
Twido	Master V2	2004						Master or Slave
Micro	Master V1			Yes		Yes Agent		Master or Slave
Premium	Master V1 Master V2	Yes		Yes	Yes	Yes Manager	Yes	Master or Slave
Quantum	Master V1	Pending	Yes Third-party module	Yes	Yes		Yes	Master or Slave

Industrial control

		ASi	CANopen	DeviceNet	Ethernet TCP/IP Modbus	Profibus-DP	FIPIO	Interbus	Modbus
Motion control	LEXIUM MHDA		Yes			Yes	Yes		
Motor starters	TEGO Quickfit	Yes	Yes	Yes		Yes	Yes	Yes	
	Tesys U	Yes		Via gateway		Via gateway	Via gateway		Yes
Remote I/O	ASI IP20 and IP 67	Yes							
	IP20 Momentum			Yes	Yes	Yes	Yes	Yes	
	Advantys IP20		Yes	Yes	Yes	Yes	Yes	Yes	
	Advantys IP67		Yes	Yes		Yes		Yes	
HMI	XBT-H XBT-P XBT-E	Via gateway							Yes
	XBT-F	Via gateway			Yes		Yes		Yes
Drives	ATS46			Via gateway		Via gateway	Via gateway		Yes
	ATV28								Yes
	ATV58	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
	ATV68					Yes	Yes		Yes

Setup procedure

On the installation

Wire the installation

Configure the slaves: Address, communication speed, etc.

Via switches, rotary switch or console. Some products automatically detect the communication speed and format.

With PL7 and configurator

Declare the master module in the PLC

PL7 is the programming software for Micro and Premium PLCs

Configure the master module

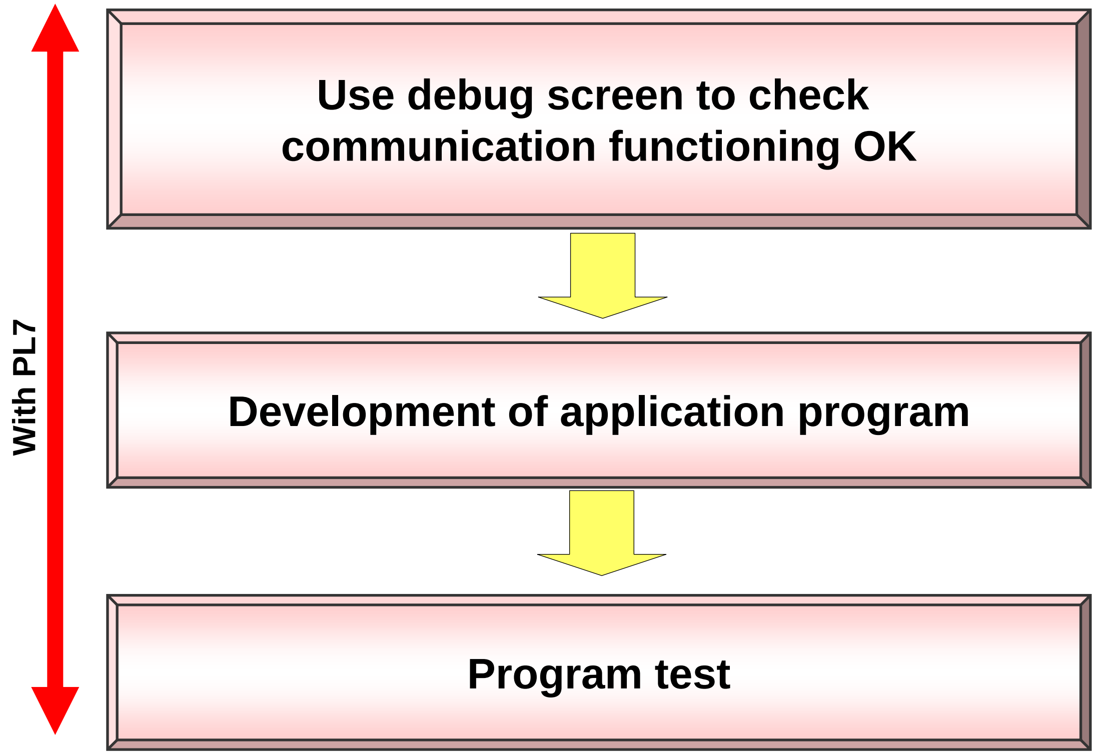
With **PL7** for ASi, Ethernet, FIPIO and Modbus

With **SycCon** for CANopen and Profibus

With **CMD Tool** for Interbus

Save and transfer the configuration to the PLC

Setup procedure



The different types of exchange

Adding a communication module to a PLC **increases possible object applications**, which can be of **2 types**:

Implicit objects:

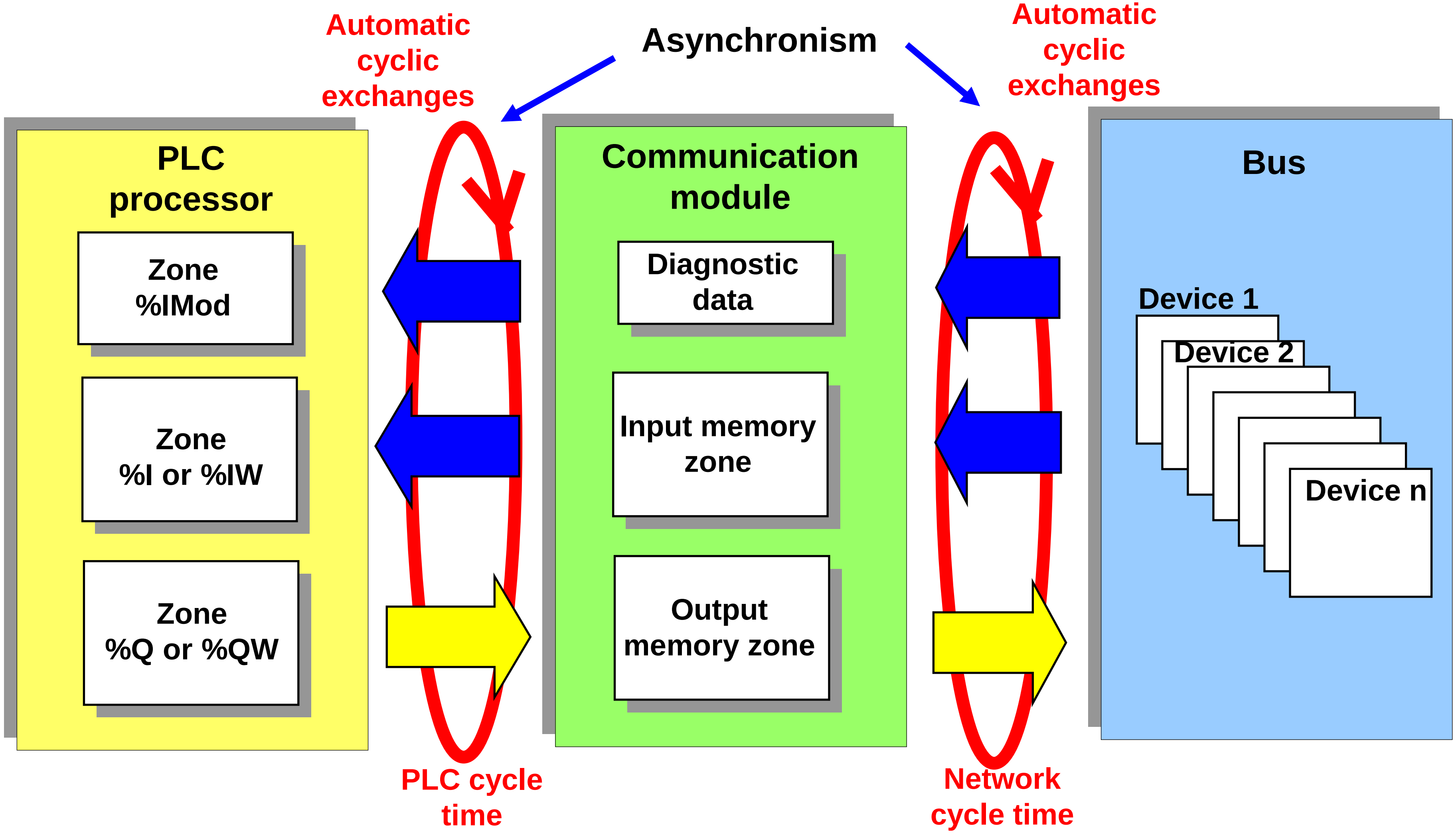
These input or output variables are **updated automatically** by the PLC CPU and the communication module asynchronously.

Explicit objects:

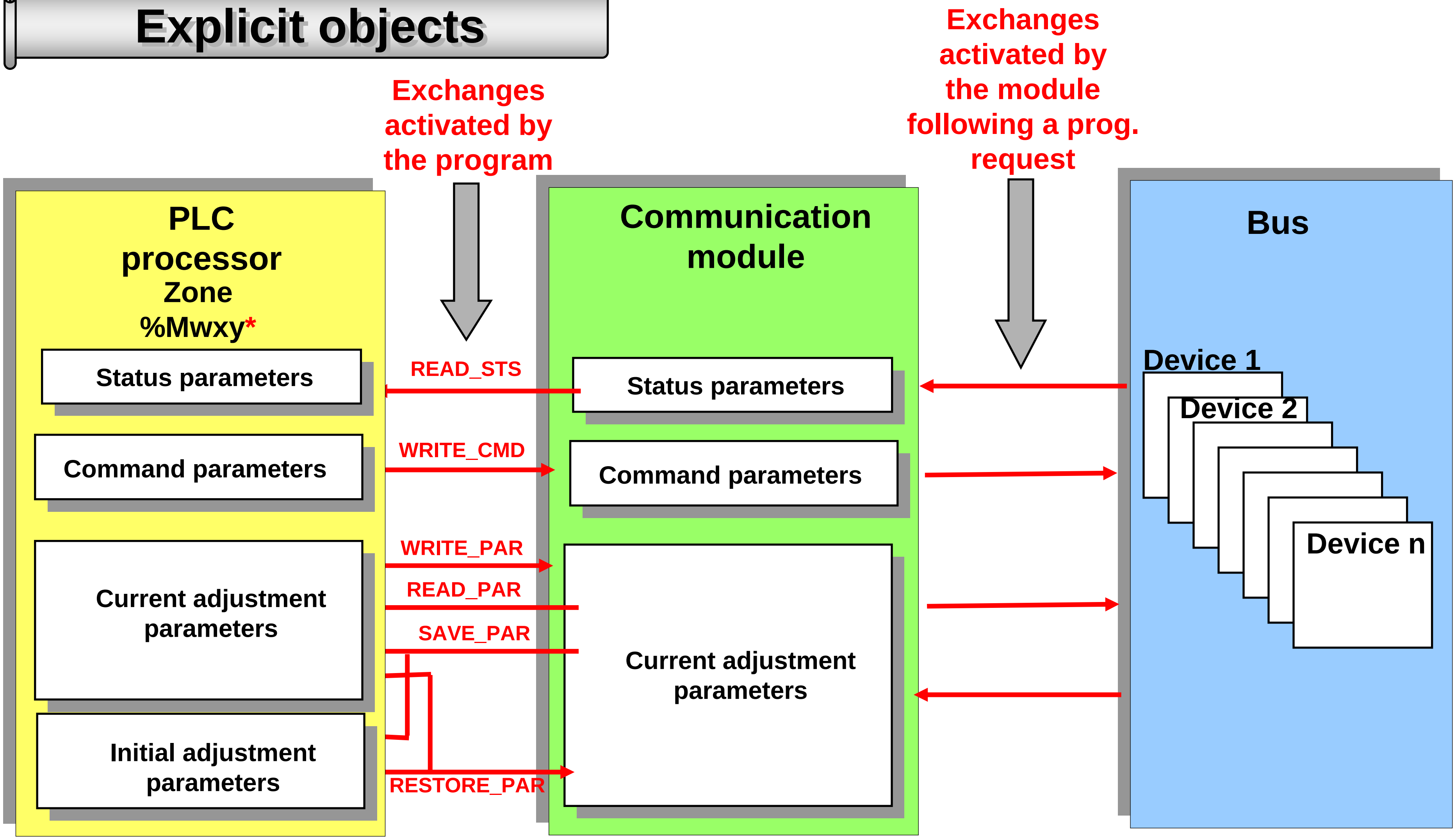
These input or output variables are updated at the request of the user program.

It is also possible to **exchange** data **directly** between the application and remote devices using **communication functions (Read_var, Write_var, Send_Req, etc.)**

Implicit objects

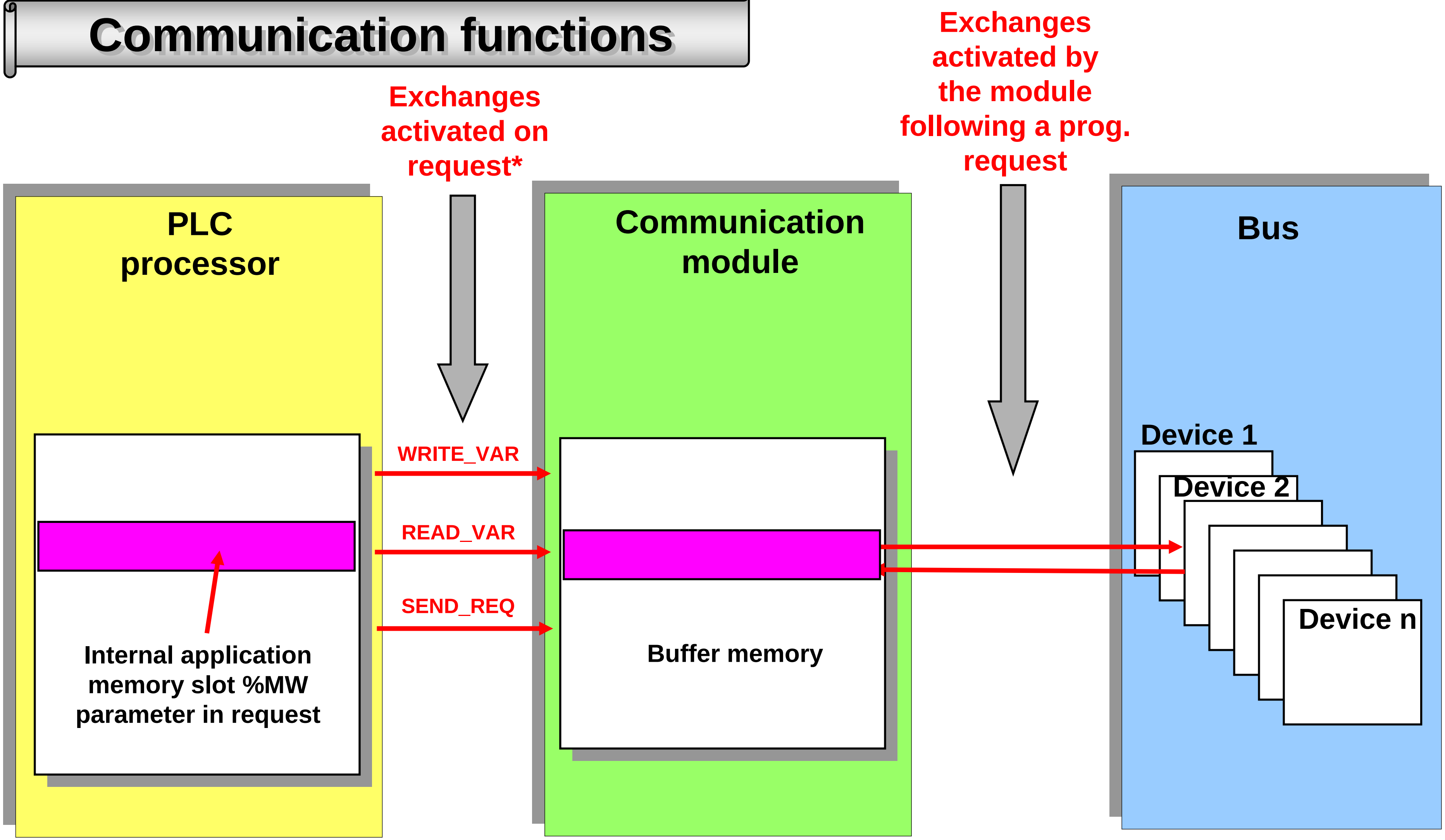


Explicit objects



* %Mwxy: Where x = Rack number - y = Communication module slot number

Communication functions



* The request enables a parameter to be set defining the device being addressed and where the data is stored.