Training document for the company-wide automation solution

Totally Integrated Automation (T I A)

Appendix IV

Fundamentals of fieldbus systems

with SIMATIC S7-300

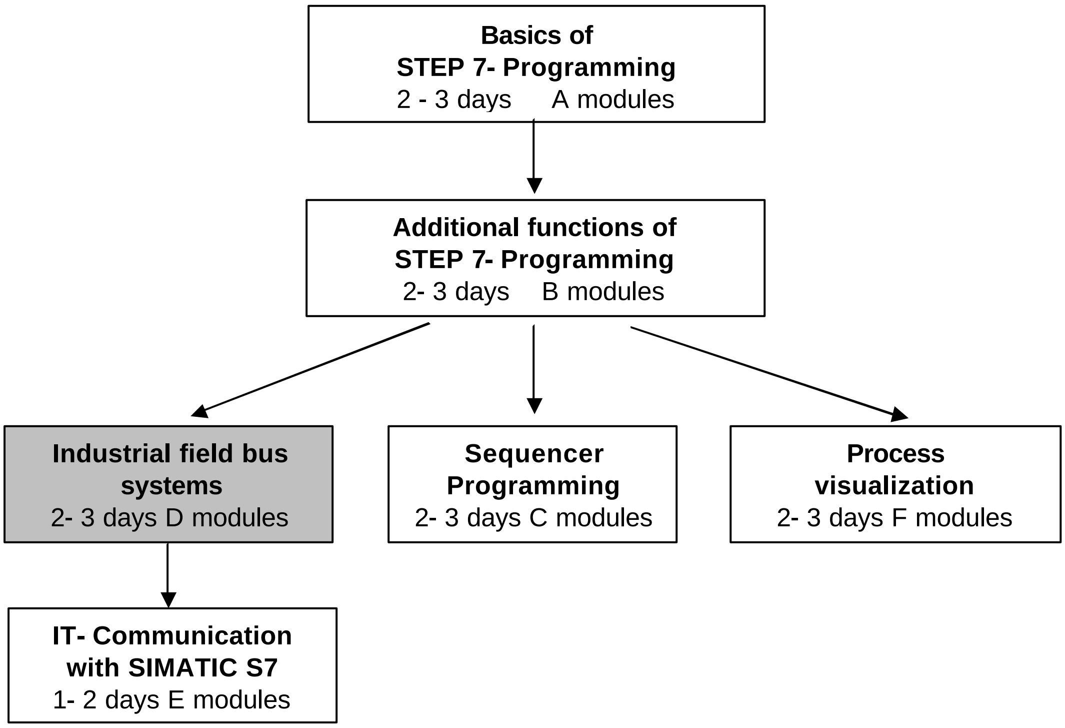
We thank the company Michael Dziallas Engineering and the instructors of vocational schools as well as further persons for the support with the production of the document.

PAGE:

PAGE:

1. FORWARD

Appendix IV is the requirement for the processing of the module of the theme **Industrial field bus systems**.



Learning goal:

This document should give you an overview of field bus systems in general and should introduce the integrated bus systems of the automation system S7-300. They are:

- Multi Point Interface (MPI)
- AS-Interface
- PROFIBUS

Requirements:

Since the basics are found within this appendix, no special requirements are necessary.

1.1 PREFACE

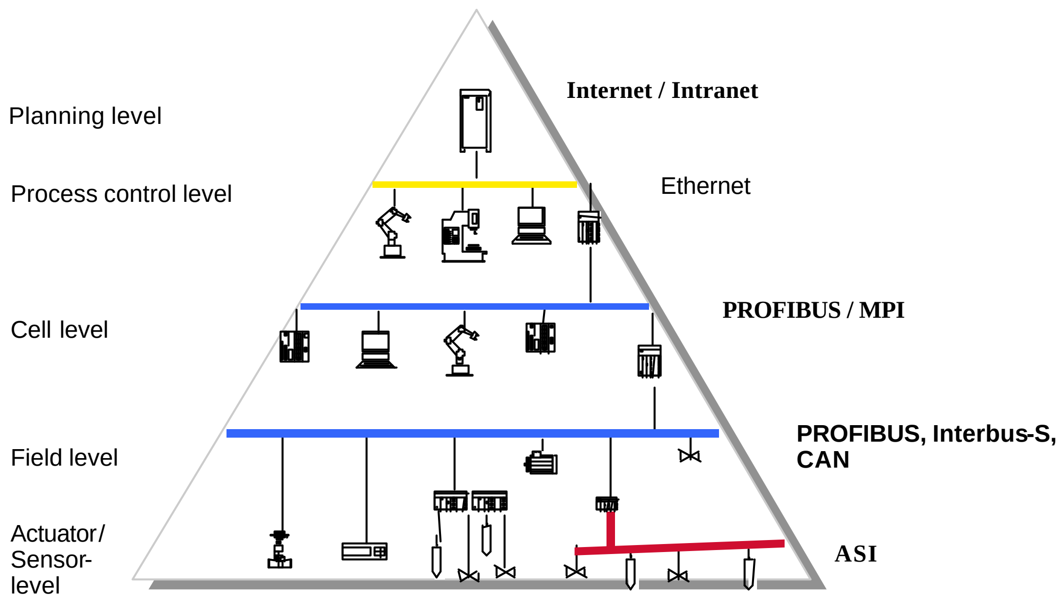
By complex systems with a large number of in-/output signals, it is no longer practical to realize the automation task with a signal, central controller.
Then one has skipped the control tasks in order to distribute more smaller automation devices. These devices are coordinated from higher order controllers or mainframes, which are integrated over a bus system in the whole process.
The in- and outputs are no longer connected to the central signal modules directly on the controller, but are distributed in a process from location by I/O modules that are connected over a field bus with a signal controller.

This **distribution of the automation task** with the **connection of the peripherals from location** has the following advantages:

- Simpler programming through smaller programs,
- Minimization of the cabling cost, (Cabling errors are reduced),
- Breaking down system structures,
- Simple expansion resp. modification,
- Short positions by critical signals e.g. analog values or counter frequencies, digitalized into I/O range,
- Flexibility from automation systems is increased by the assignment of the peripherals
- Higher system availability by faults through self-sustaining controllers,
- Comprehensive self monitoring and error diagnostic of the transmission system,
- Installation and maintenance is simplified,

In the following pages, the different field bus systems should be visualized for realization of each solution in order to then be responsive to the bus systems of the SIMATIC S7-300.

2. HIERARCHY LEVEL IN AUTOMATIONS TECHNOLOGY



In order to receive the complex information in large enterprise in the handle, different hierarchy levels are built inside of the whole automation range.
The information exchange takes place inside of and between a signal hierarchy plane (vertical and horizontal).

Each hierarchy level is assigned a further level in which the requirements by the communication are arranged. There the different communication tasks can no longer be released with a network. Different communication systems would be developed.

In the upper levels, you find complex computer systems. Large data heaps with uncritical reaction time, large node counters and a further expansion of networks dominate it.

The communication in the lower level is embodied through minor data heaps and message throughput such as smaller node counters. Here lie real time requirements in the foreground. The length of the network is rather small.

One differentiates the following five hierarchy levels:

- The analysis of the information from the production process that the application plan, such as the stipulation of guidelines and strategies for manufacturing take place in the **planning level**. In longer periods, larger data heaps are transmitted here over a large range.

- The coordination of a single production range takes place in the **process control level.** Here the cell level is provided with jobs and program data and it is decided, how the production has to occur. This process control computer such as a computer for configuration, diagnosis, operation and protocol, resides in this level.

- The **cell level** integrates the single manufacturing cells that are controlled from the cell computer or the PLCs. Here in the foreground lies the specific communication between intelligent systems.

- In the **field level**, you find programmable devices for open or closed loop control and monitor like PLCs or industry computers that evaluate the data of the sensor/actuator level. For the connection to the superposed systems, larger data heaps with critical reaction time are transmitted.

- The **actuator-/sensor level** is an integral part of the field level and integrates the technical process with the controller. This occurs with simple field devices like sensors and actors. The faster cyclic actualization of the in- and output data lies in the middle point, where small messages are transmitted. The duration for the actualization of the in- and output data must circumstantially be smaller as the cycle time of the controller.

3. FIELD BUS SYSTEMS (A CHOICE)

Direct in the range of the field bus systems there is a multitude of systems with competitive standards that will be assured in this hard-fought market. Without demand for completeness, the most important field bus systems in Europe should be shortly visualized.

3.1 INTERBUS-S – ONE OF THE FIRST

In 1985, the Interbus-S was developed from the Phoenix Contact Company with the goal being to avoid costly parallel cabling in the PLC peripherals.
The Interbus-S does not want to represent a universal communications medium, but rather single PLCs, CNC controllers or process automation systems with their peripherals connected. The strength of the Interbus-S is a very high transmission efficiency with very small data heaps per node.
The Interbus-S is qualified only for the lowest hierarchy level. It connects sensors and actors with the corresponding controllers. It is not designed for the linking controllers to one another in a network.

3.2 PROFIBUS - VERSATILITY

The PROFIBUS (Process Field Bus) is qualified for the networking of complex devices with its multi master protocol.
The PROFIBUS is named after DIN 19245, where it extends its user range from the field level to the process control level.
In principle it is applicable with its protocol profile PROFIBUS-DP (Distributed I/O) down to the sensor/actor level. For the cost efficient activation of a large number of sensors and actors, it offers the integration of the bus on low-order levels like the AS-I.

3.3 AS-I – SMALL AND FAST

The AS-I (Actuator Sensor Interface) is matched to the requirements in the lowest level.
AS-I operates actors and sensors with the first control level and replaces them with cable harnesses, distributor cabinets and connecting terminal plates. Since then, the AS-I is an open standard. In the meantime, many manufacturers offer intelligent, AS-I compatible actors and sensors in order to be able to transfer more information than only 1/0.
AS-I is especially easy in data manipulation. Field devices are simply clamped into cut terminal technology on an unprotected 2 way conductive flat cable. As a result, the installation can then be accomplished by people without any expertise.
AS-I is fast, simple, cost effective and also future safe because it meets more than half of the world market's requirements for sensors from manufacturers that support it.

3.4 CAN – FIELD BUS ON WHEELS

The CAN bus system (Controller Area Network) was primarily developed by Bosch in cooperation with Intel, in order to reduce cable harnesses in automobile building. When one compares the requirements of KFZ bus systems with those of industrial field bus systems, then one sees striking similarities:

- Minor costs,
- Functional safety under difficult environmental conditions,
- High real-time capability and
- Simple data manipulation.

CAN is therefore very well suited for the networking of intelligent sensors and actors within machines.

4. INTEGRATED BUS SYSTEMS FOR THE SIMATIC S7-300

In the frame of 'Totally Integrated Automation', different bus systems as an integral component were included in the SIMATIC S7-300.
Therefore, the three following bus systems shall be explained in more depth:

4.1 THE MULTIPOINT- INTERFACE (MPI)

This bus system was chiefly developed as a programming interface. MPI serves likewise for the communication with components that work for the 'man/machine interface' and for homogenous communication between automation devices.

4.2 THE AS- INTERFACE (AS-I)

The AS-Interface is a network system for binary sensors and actors in the lowest field range.

4.3 THE PROFIBUS

PROFIBUS is a bus system that is used in the field range as well as for cell networks with a small amount of nodes.
There are three protocol profiles for the PROFIBUS that can be operated together on a circuit (RS 485 fiber-optic cable).

- **PROFIBUS-FMS** (Fieldbus Message Specification) is suited for the communication of automation devices in small cell nets under one another and for the communication with field devices with a FMS interface.
- **PROFIBUS-DP** (Distributed Peripheral) is the protocol profile for the connection of distributed I/Os in the field area e.g. ET 200 modules with very fast reaction time.
- **PROFIBUS-PA** (Process Automation) is the communication compatible addition from the PROFIBUS-DP about a transmission technology that allows the users into the EX-area. The transmission technology of the PROFIBUS-PA corresponds to the international Standard IEC 1158-2.

In the frame of this document, we will only take a further look into the **PROFIBUS-DP**. There this frequently applicable case is together with the SIMATIC S7-300 and also exists for an integrated interface.

5. THE MULTIPOINT- INTERFACE (MPI)

This bus system was developed as a program interface for the SIMATIC S7. The MPI serves likewise for the communication with components that are used for the 'man/machine interface' and for the homogenous communication between automation devices.
The operation area from the MPI and PROFIBUS is divided into many areas, where MPI is considerably cost effective. This interface is already available in all SIMATIC S7 products.
The considerable advantage over the PROFIBUS is that the transmission protocol displays an abstract "SIEMENS-Standard" so that no product from "outside manufacturers" can be integrated into each bus system.

5.1 TECHNICAL DATA FOR THE MPI

The MPI (Multipoint Interface) is one of the many integrated communication interface devices of the SIMATIC S7 that is simultaneously connected with more program devices/PCs with STEP 7, HMI systems (Operator panel/operator station), S7-300, M7-300, S7-400 and M7-400.
It can be used for simple linking in networks and enables the following forms of communication:

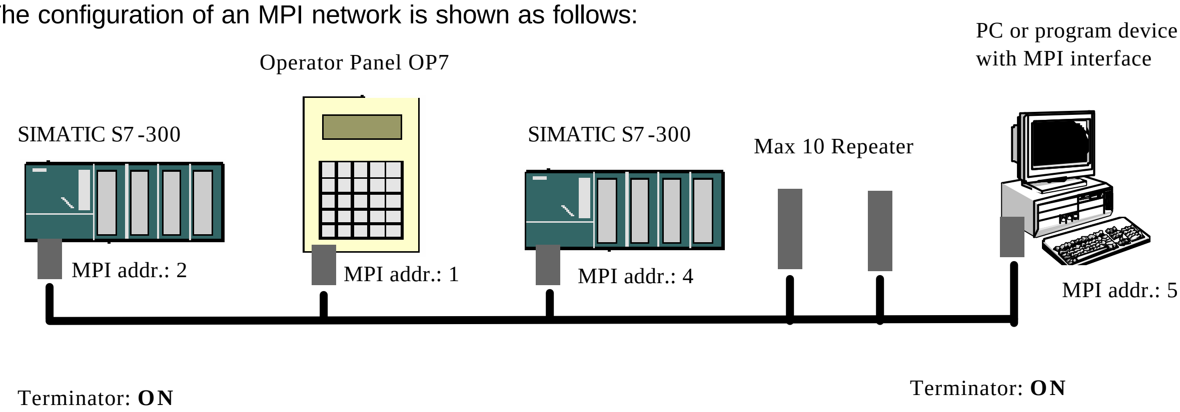
- With the service **global communications**, the networked CPUs can cyclically exchange data under one another. A S7-300 CPU can therefore exchange a maximal of 4 packets with at least 22 bytes, where with STEP 7 V4.x a max. of 15 CPUs can participate on the data exchange.
- **Programming-** and **diagnostic functions** can be executed with MPI from other programmed devices/PCs to all networked PLCs. There the MPI interface of the CPU is directly connected with the internal communications bus (K-BUS) of the S7-300. The function modules (FM) and communications modules (CP) are switched directly over the MPI with the K-Bus connection from the PG.
- The connection from **operator panels/operator stations** to the SIMATIC S7 PLCs is simply realistic with the MPI. There the communication services default can be supported and standard FBs are no longer necessary by the SIMATIC S5.

The following performance data from the company SIEMENS is entered for the MPI:

- Max. of 32 MPI nodes
- Each CPU has a possibility of a max. of 8 dynamic communication connections for the basic communication to SIMATIC S7/M7-300/-400.
- Each CPU can operate a max. of 4 statistic communication connections for the additional communication to the PG/PC, SIMATIC HMI-Systems and SIMATIC S7/M7-300/400.
- Data transmission speed 187,5 kbit/s or 12Mbit/s
- Flexible configuration possibilities in the bus or tree structure (with repeaters)
- Max. wire length 10km
- Interface: RS485

5.2 CONFIGURATION OF A MPI NETWORK

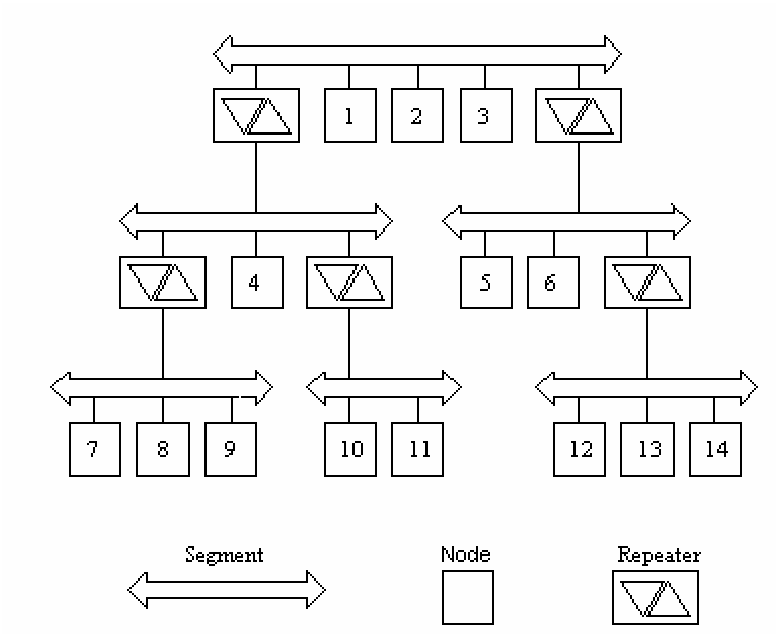
The configuration of an MPI network is shown as follows:



Up to 32 nodes can be connected with one another. Each is monitored. It should be taken into account that communication processors (CPs) and function modules that are in the SIMATIC S7-300 must also have MPI addresses and be counted towards the number of nodes.
The addresses of several nodes can be assigned between 0 and 31 (standard setting).
It makes sense, however, not to assign the address 0 (standard setting for program devices) in a closed network configuration in order to carry though a diagnosis by the MPI with additional program devices, without needing to adjust the MPI addresses of the program devices.
For an eventually available operator panel, the address 1 (standard setting for an operation panel) should be assigned.

To avoid cable reflections, make sure the first and last nodes of the MPI network are hooked up to the slots‘ integrated terminator.

The MPI network is compiled with a shielded and a stranded filter 2 wire cable and can be used up to a length of 50 m. These 50 m are measured from the first node to the last node of the MPI network. Should this large distance be used, then the PS 485 repeaters must be used. A cable length can be up to 1000 m in length between 2 RS 485 repeaters when no other node is found between the 2 repeaters. One can place up to 10 repeaters in a row.
The components that come by the configuration of the MPI configuration for operation, are the same bus cables like the bus connectors and RS 485 repeaters that are used by the electrical net of the PROFIBUS.
Therefore the electrical net can be configured as either cables or a tree structure with the help of the repeater.



Example of a tree structure under the use of repeaters

When a PROFIBUS-DP is ready in a system, it should be considered in the configuration phase, whether to develop PG functions over the PROFIBUS-DP and save the MPI cabling.

5.3 COMMISSIONING OF A MPI NETWORK

So that all connected nodes can communicate with one another over the MPI, each node must contain an **MPI address**, a maximum **MPI address** such as a **subnet** with a **transmission rate**. Therefore the following rules are required:

- All MPI addresses in a subnet must be different.
- The highest possible MPI address such as the transmission rate must be larger than or equal to the largest actual MPI address and must be equal by all nodes.

5.3.1 COMMISSIONING OF A SIMATIC S7-300 TO THE MPI NETWORK

By the SIMATIC S7-300 it should be noted that many communication processors (CPs) and function modules (FMs) have a single MPI address. This address is automatically given from the CPU, by means of a sequence in which the modules are ordered on the module rack, acquired and allocated in the following pattern:

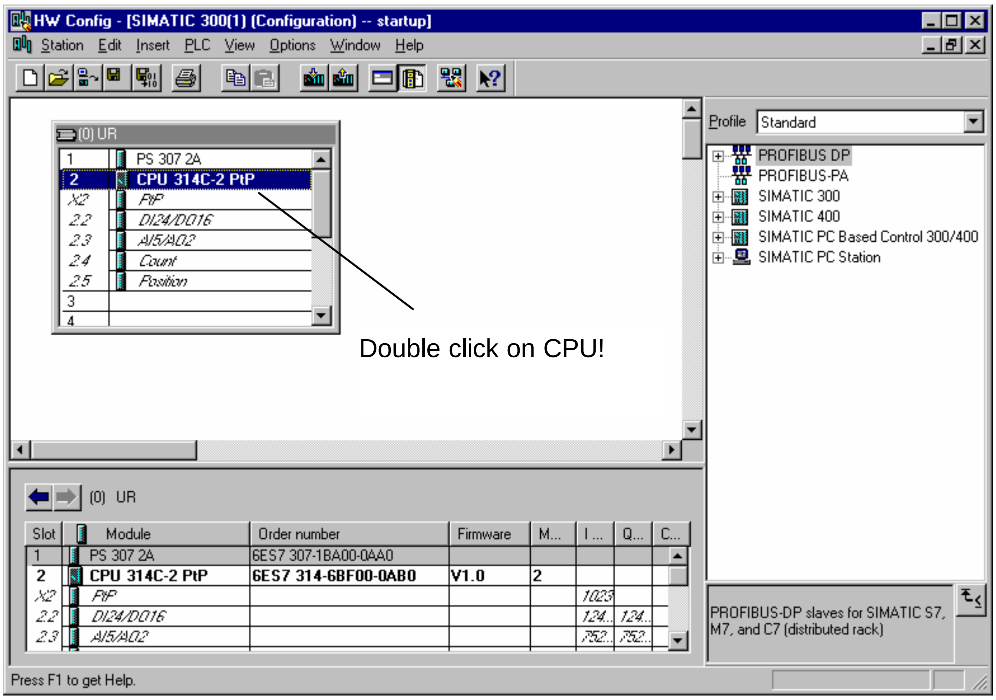
CPU:	MPI address
CP/FM1:	MPI address + 1
CP/FM2:	MPI address + 2

Through this, direct access is provided from the program device over the CPU on the associated module. This access takes place in the S7-300 over the internal communication bus (K-Bus).

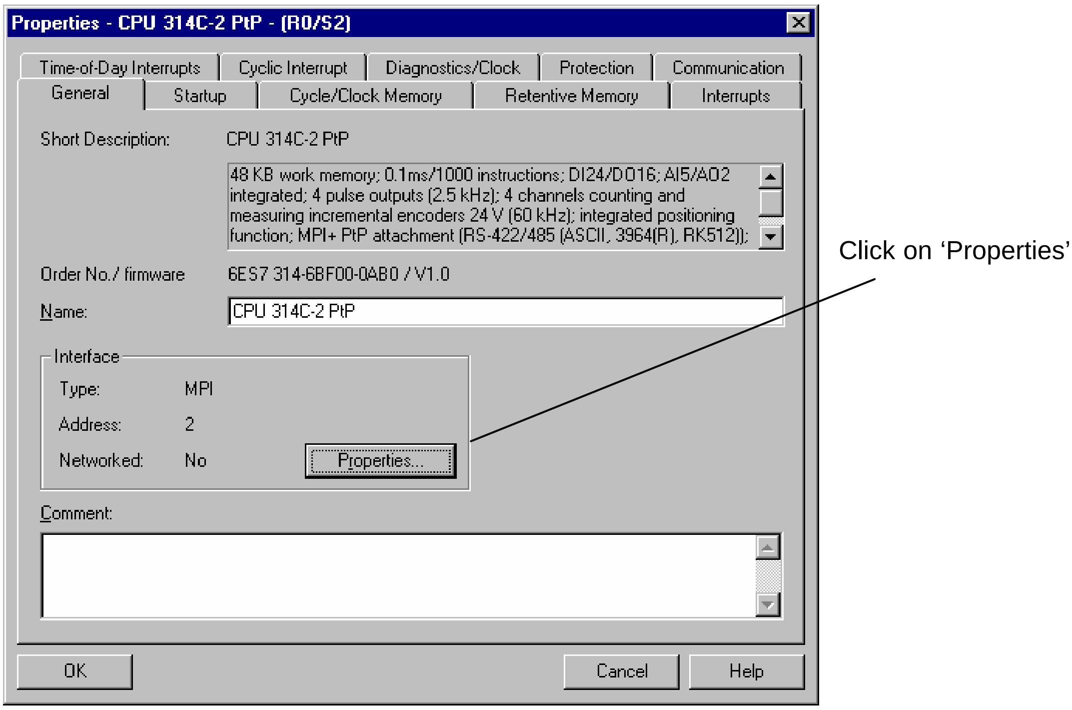
After the entry of the hardware module in **HW Config** is completed, the connection of the PLC can be configured by the MPI network. The following steps need to be followed:

1. The PLC must be connected over the MPI to the program device. It is not yet found in the MPI network.

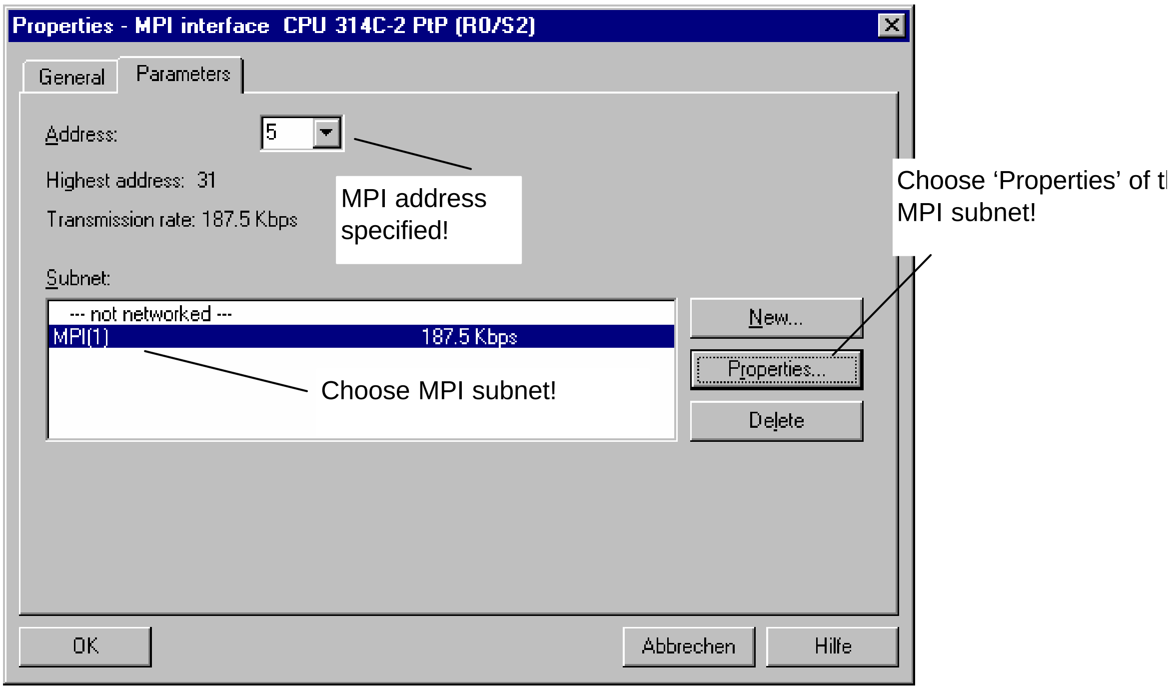
2. Double click on the CPU in the configuration table.

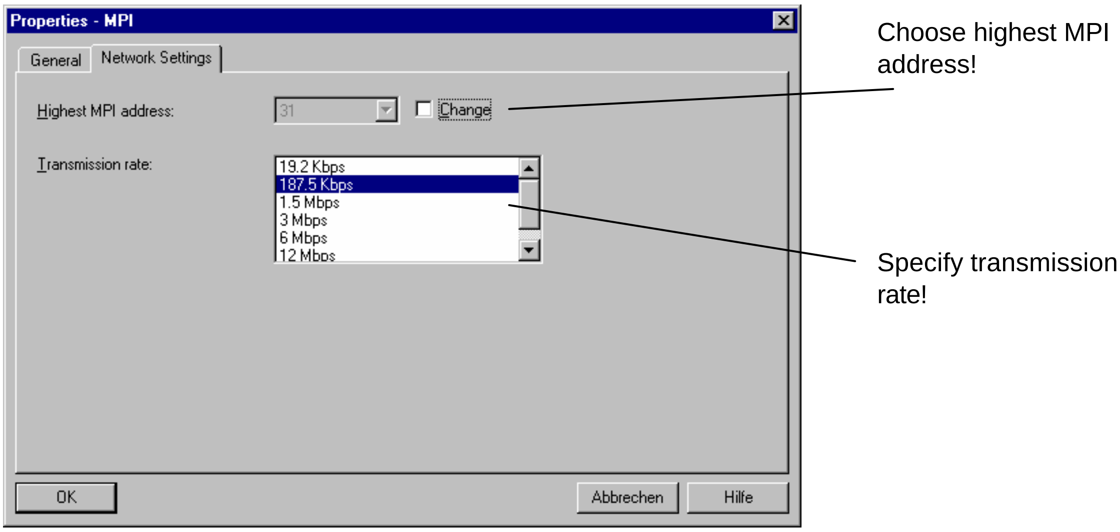


3. Then click on→ Properties.



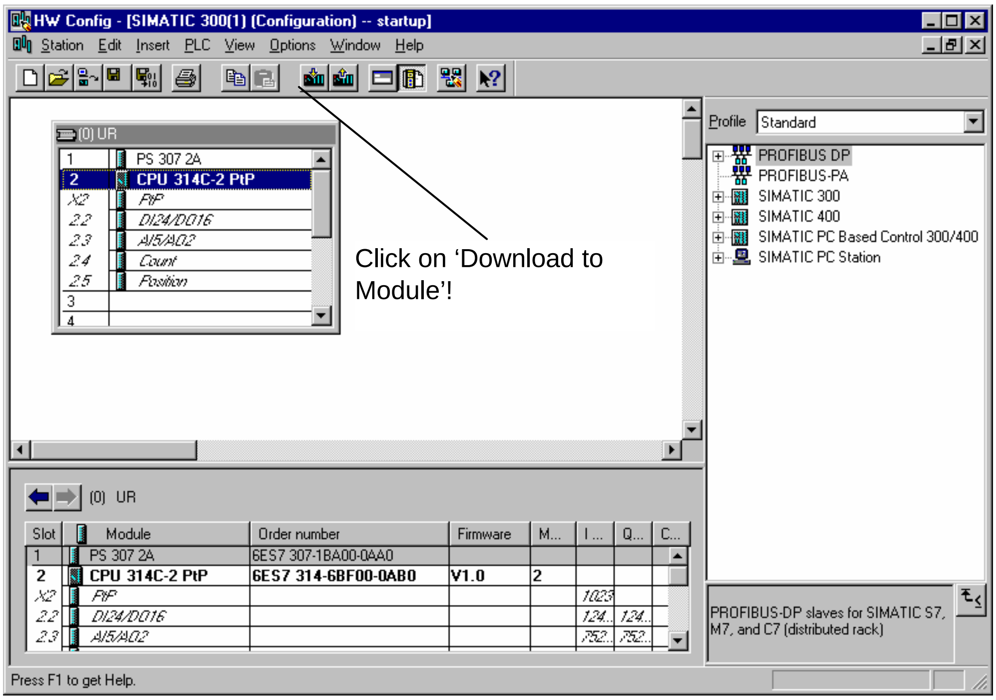
4. By the properties of the MPI interface, only the MPI address of the CPU is specified (The MPI address must not be larger than the highest set MPI address!) and the MPI subset is chosen. When the setting for the highest MPI address or transmission rate of the subnet must be adjusted, click on → Properties with the mouse.





5. The settings must be accepted → OK → OK → OK.

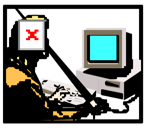
6. Now the downloading of the altered configuration takes place in the CPU and in the eventual, available CPs and FMs (The mode switch on the CPU must be on STOP!).
   In the hardware configuration the MPI addresses of the CPs and FMs corresponding to the chosen MPI address for the CPU were entered.



For the checking of the operation capability of the MPI connection one can, for example after the choice of the following MPI node, take into effect a change of operation mode from the program device/PC. By erroneous transmission, the set values must be checked. It must be ensured that the right MPI address was set and chosen. Furthermore the properties of the transmission rate and highest MPI address must agree with all nodes.
There can also be an error by the cabling by the switching of the terminal resistances. Therefore one should have first checked if the bus terminal resistances are set to the right setting and then if they have a bus connector contact, a ground connection or a short bus connection.

5.3.2 COMMISSIONING OF AN OPERATOR PANEL/AN OPERATOR STATION ON A MPI NETWORK

The operator panels/ operator stations must also be configured for the operation by the MPI configuration. The setting of the **MPI address**, the **highest MPI address**, and the **transmission rate**, such as the assignment to a **subnet** and the corresponding **communication partner** takes place here with special parameter tools like ProTool.

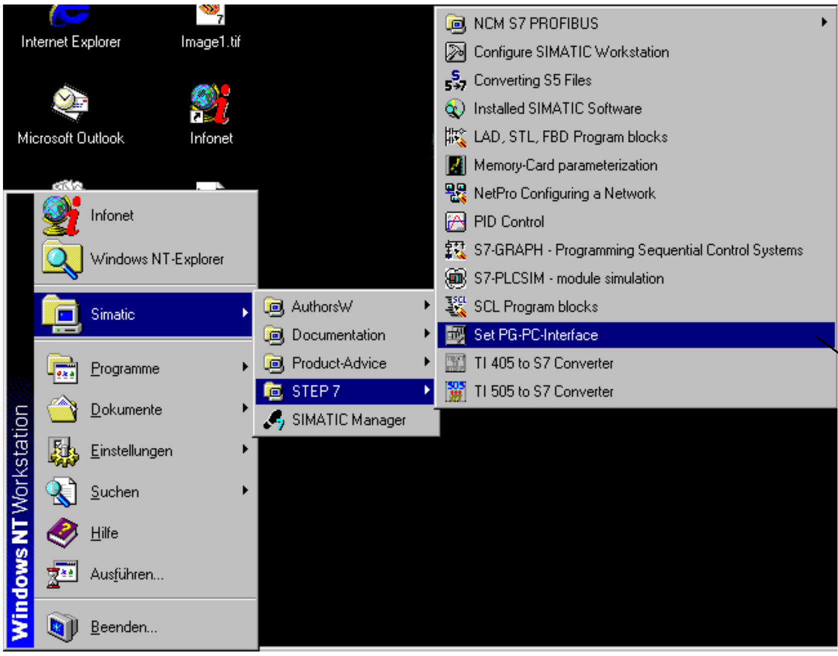


Additional information about the configuration of operator panels/ operator stations can be found in the corresponding manuals.

5.3.3 COMMISSIONING OF A PROGRAM DEVICE / PC ON THE MPI NETWORK

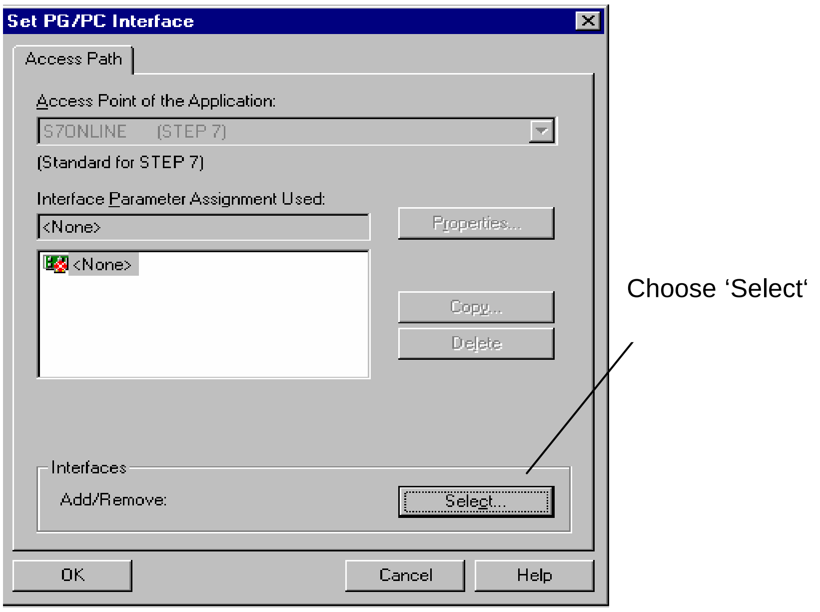
There are many possibilities for the connection of a program device/PC on the MPI network. There are MPI slot cards for the PCI, PCMIA or the alternative PC-Adapter that can be connected on a serial interface.
By the commissioning of a program device/PC to a MPI network the parameters **MPI address** and **highest MPI address** such as the **transmission rate** must be assigned. In the following example, the configuration for a PC Adapter is shown:

1. Call ‘Set PG-PC-Interface‘. (→ Start → SIMATIC → STEP 7 → Set PG-PC-Interface)

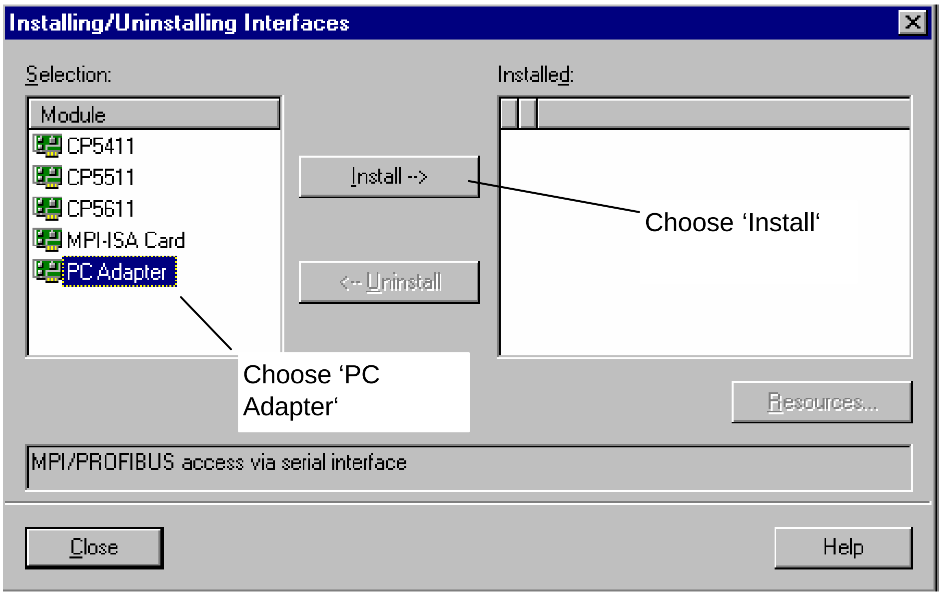


Click ‘Set PG-PC-Interface‘!

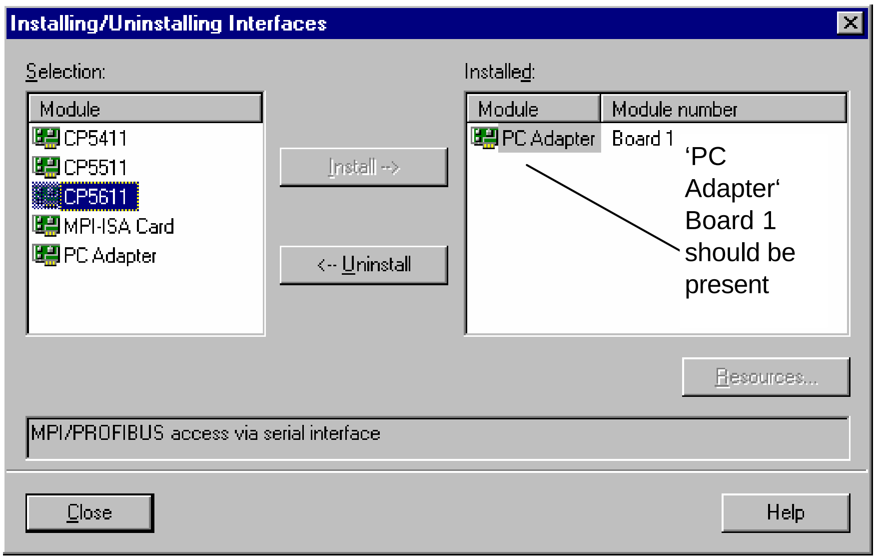
2. The module Select (→ Select) is available as the MPI-interface.



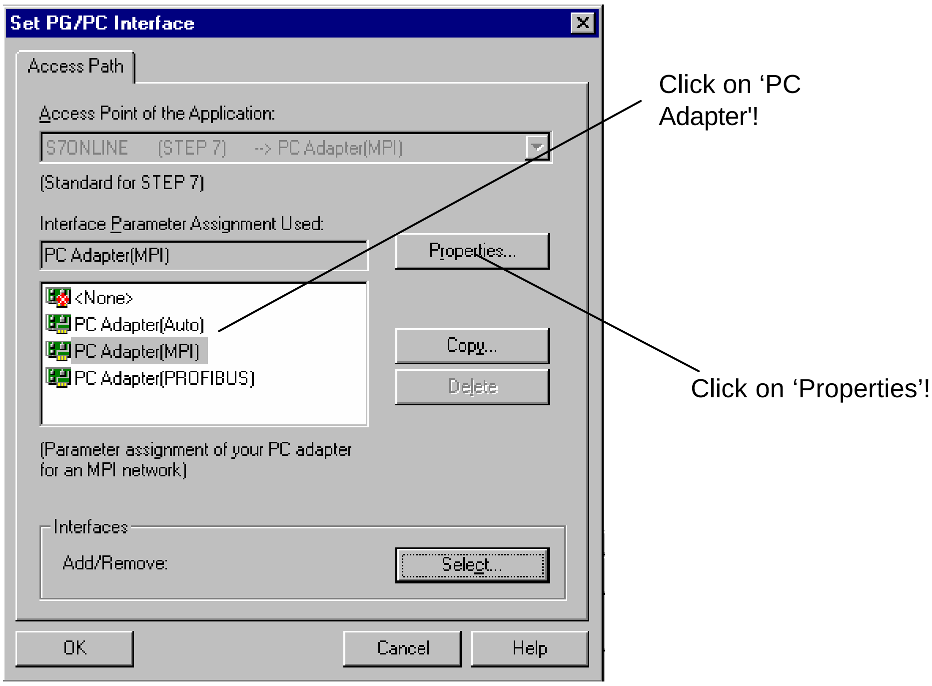
3. Select the desired module (e.g. → PC-Adapter) and install (→Install).



4. Choose the desired device (e.g. → PC Adapter → Close)

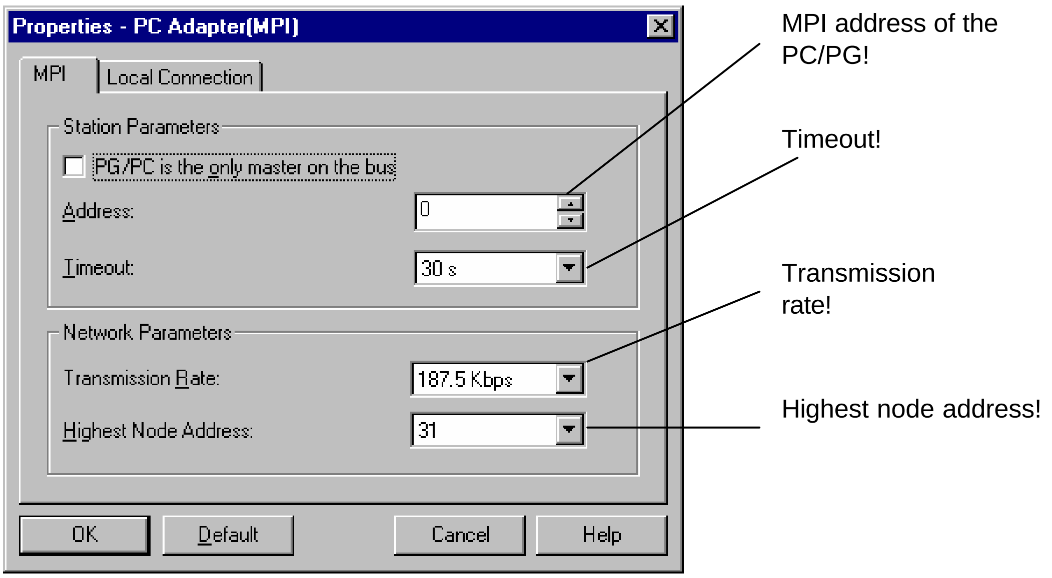


5. Choose Properties of the MPI module (e.g. → PC Adapter(MPI) → Properties).



6. Specify MPI address, timeout, highest node address etc.

Note:
It is recommended to use the preset values!



7. Accept settings (→ OK → OK).

5.4 GLOBAL DATA COMMUNICATION WITH MPI

The global data communication is simple possible data like inputs, outputs, memory bits such as areas in data blocks exchanged between S7-300 and S7-400 CPUs over the MPI interface.
This data in the operation system of the S7-300/400-CPU is an integrated communication possibility that makes a cyclic data exchange possible without having to connect extra blocks to it.
The commissioning takes place by simple parameterizing.

5.4.1 FUNCTIONALITY OF A CYCLIC DATA EXCHANGE

The cyclic data exchange takes place by the cycle control point together with the process image exchange. The CPU sends the global data by the end of a cycle and reads this data at the beginning of a cycle.
Therefore a S7-300/400-CPU sends your data simultaneously to all S7-300/400-CPUs to the MPI subnet (**Broadcast**). Up to 15 different nodes can be entered into the global data table.
With the help of a scan rate factor that you indicate in a global data table, you can set after how many cycles the data transmission of data receiving should take place.

The max. number of the transmitted data depends on the type of CPU:

CPU 31x	CPU 412	CPU 413	CPU 414	CPU 416
8Bytes	32Bytes	32Bytes	32Bytes	32Bytes

Broadcast processing:
Standard calls and data processing are carried through a broadcast in a network without waiting for a return message. When different global data is sent over the CPU, there is overlap.

.

5.4.2 REACTION TIME AND TRANSMISSION SECURITY

The reaction time is dependent on the cycle time of the user program and it averages to a n order (GD scan rate factor) time.
The reaction time can be calculated approximately with the following formula:

Tmax. = Cycle time sent * U-Factor sent + cycle received + MPI number-Tln.

This processing is broadcast processing and guarantees no data security. Thus the global data communication is not qualified for security relevant release between system components.

5.4.3 GLOBAL DATA CONFIGURATION

Global data communication is not programmed, but rather configured.
With STEP 7, a global data table is created in which the configuration data for the data exchange is specified. All SIMATIC S7-300/400-CPUs that should participate by global data communication must be in the same STEP 7 project and like described in point 5.3.1, have been taken by the MPI network into operation.
(You can find additional information in the STEP 7 reference manual).

The configuration of the MPI network and the setting of the global data table takes place as follows:

1. Open a project.

2. Choose object properties of the MPI network (Menu instruction Edit).

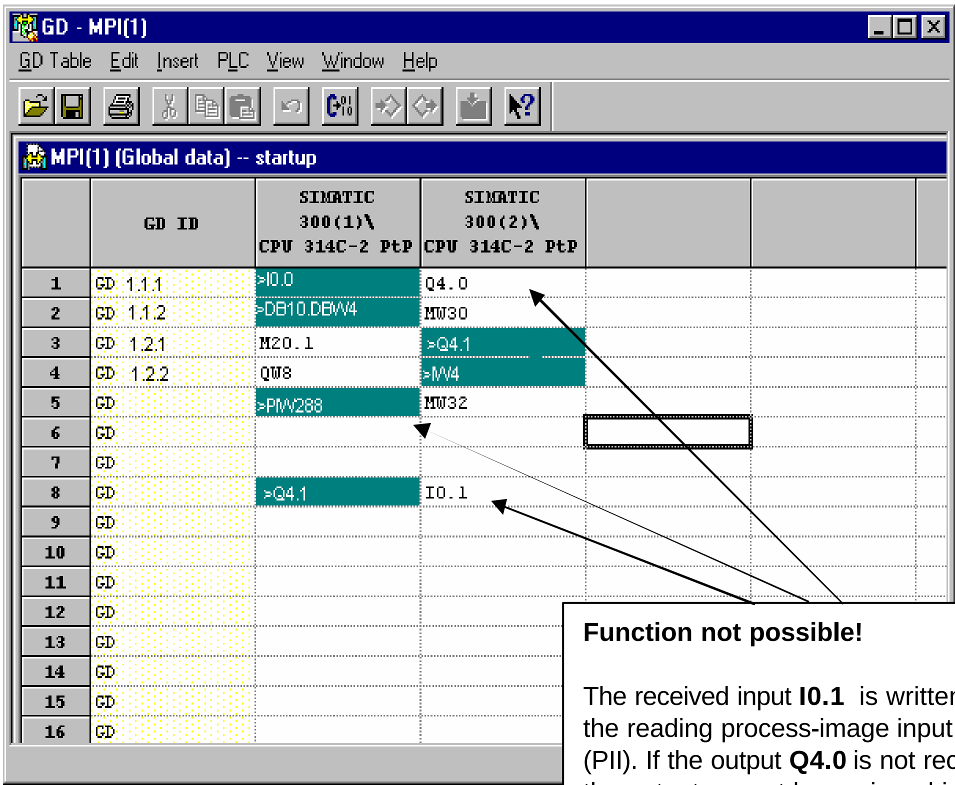
   - Highest MPI address and
   - Transmission rate setting

3. Assure that the CPU's different MPI addresses have not been altered in the 'HW Config'.

4. Choose MPI network (→ MPI(1)).

5. Define the global data in the GD table (→ Options → Define Global Data)

- Enter CPU (→ Highlight field → Edit → CPU)
- Enter e.g. memory bits-, outputs- and input ranges
- Each line belongs to a sender (→ Edit → Sender)
- Compile global data table (→ GD-Table →Compile)



Global data table

Function not possible!

The received input **I0.1** is written by the reading process-image input table (PII). If the output **Q4.0** is not received, the output cannot be assigned in the control program of the receiving CPU. The I/O range cannot be transferred.

6. Download configuration data in the CPUs (→ PLC → Download to Module)

Now the sending CPU sends the global data at the end of a cycle and the receiving CPU reads this data at the beginning of a cycle.

Note

Regard that the reading of the global data takes place before the reading of the PII and before the program processing.

The sending of the global data is first accomplished after the processing of the control program and the reading out of the PIQ in the task modules.

6. THE AS-INTERFACE

The Actuator-Sensor-Interface (AS-I) serves as the information transmission in the lowest field area and like the PROFIBUS, is an open standard. A multitude of manufactures offer products and interfaces to the AS-Interface. The AS-Interface enables a simple and extremely cost efficient integration of sensors and actors in the industrial communication and provides these sensors and actors simultaneously with the important auxiliary power. With this system, predominately binary sensors and actors are operated with the controllers. So far it is important for process signals that arise before a location to transfer with conventional parallel wiring over in-/output modules into the controller. AS-I replaces the expensive cable tree by a simple and complete unprotected 2 wire cable for all sensors and actors.
Through the robust design in a degree of protection IP65 or IP67, the AS-Interface increases straight in the lowest field area of usual and hard operating conditions.

6.1 TECHNICAL DATA FOR AN AS- INTERFACE

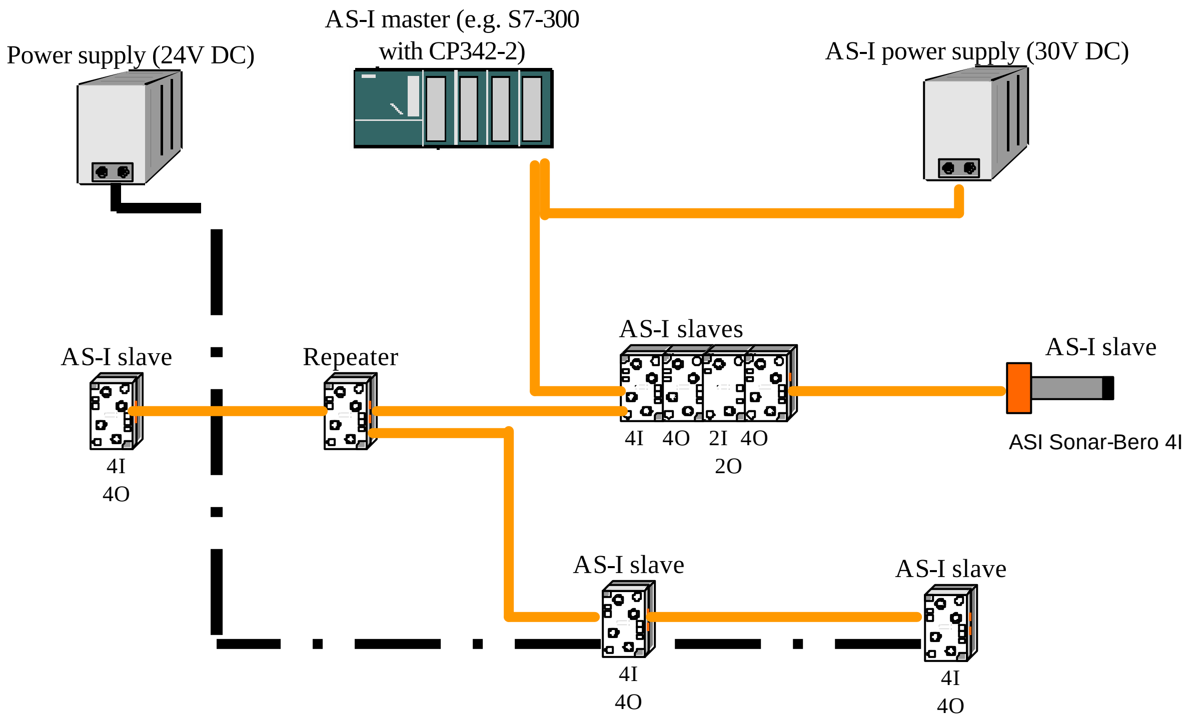
The technical data and transmission protocol of the AS-Interface are specified in the Norm EN 50 295.

The following performance data for the AS-Interface is specified here:

- Max. 31 AS-I nodes with 4 Bit I/O user data
- Max. 124 I/O sensors and actors
- Access processing with cyclic polling in the master/slave process
- Max. cycle time 5ms
- Error safe identification and iteration of faulty frames.
- Transmission medium is a usual 2 wire cable (2 x 1,5 mm^2) for data and a max. of 2A auxiliary power per AS-I pro AS-I cable. The power supply consists of 30 V DC. The signal of the data transmission is modulated. An additional power supply of 24V DC (auxiliary power) is possible.
- Connection and assembly of the AS-I components in an insulation displacement method.
- AS-I slave module with integrated circuit (AS-I chip) that requires no processor and no software. This results in an approximate, delay free processing of the frames and a small construction volume of the slave
- Special AS-I sensors and actors are directly integrated with the AS-I chips.
- Flexible configuration possibilities like in the conduit with cables, stars or tree structures
- Max. wire length of 100m or 300m (with repeaters)

6.2 CONFIGURATION OF AN AS-INTERFACE

The configuration of an AS-Interface can appear as follows:



Additionally the addressing of the AS-I slaves requires one more addressing device:

+

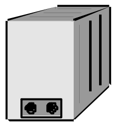
Addressing device
for AS-I slaves

The AS-Interface is a single-master system. Therefore there always exists exactly one master and up to 31 slaves in each system with the CP342-2. If more slaves are required, an additional AS-Interface system with an additional master must be inserted.

6.2.1 Basic components of an AS-Interface configuration:

The AS-Interfaces occurs modularly under the use of the following components:

- Power supply 30V DC (Power supply)



The 30V power supply is attached directly to the data circuit.

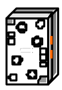
- AS-I data circuit as unprotected 2 wire cable.

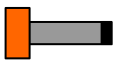
The connection of the AS-I components takes place in an insulation displacement method, where the AS-I cable is flattened in order to avoid wiring errors by assembly.

- AS-I master as a connection device for the controlling by the user or a higher level bus system with the corresponding master chips.

Over the AS-I master, the user can have access to the I/O data of the AS-I slave. This occurs at the S7-300 in the user program of the CPU.

- AS-I slaves with slave ASIC

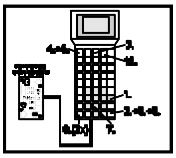




ASI Sonar-Bero 4I

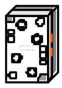
For the AS-Interface, there is a large choice in slaves from different manufacturers. Each slave must be assigned by the commissioning of a target AS-I address that is then saved there. The addressing occurs either with the configuration device or over the master in which each slave is written a single connection by an addressed frame. This also functions when <u>one</u> slave is exchanged.

- Configuration device for the setting of the slave addresses



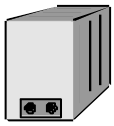
With the programming and service unit (PSG), the AS-I slave addresses can be set very easily.

- Optional: Repeater for additional wire length up to 300 m (100m without repeater)



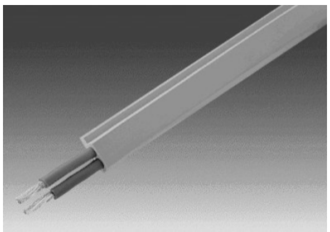
In order to implement a bus configuration with a larger expansion (e.g. by material systems), the repeaters must be interposed. This is connected with the AS-I data circuit.

- Optional: additional power supply 24V DC (power supply) for auxiliary power



When an AS-I slave requires more as 100mA or all slaves require more than 2A of auxiliary power pro AS-I cable, an additional power supply of 24V DC is required. This is connected over the AS-I network cable (black) with the auxiliary power contacts of the slaves.

- AS-I network cable for the auxiliary power as an unprotected black 2 wire cable.



The connection of the auxiliary power occurs in an insulation displacement method where the AS-I cable is flattened in order to avoid wire errors by assembly.

6.3 TECHNICAL DATA FOR THE CP 342-2

The AS-Interface master CP342-2 can be used in the S7-300 or also in a PROFIBUS slave ET 200M by any activation either in the central device or in one of the 3 additional devices and occupies a slot there.

It offers the following functions and characteristics:

- Simpler operation in the I/O address range of the SIMATIC S7-300 and ET 200M
- Configuration of the CPs is not necessary
- Triggering of up to 31 AS-Interface slaves corresponding to the AS-I specification V2.0
- Up to 248 binary elements by the operation of triggerable bi-directional slaves
- Monitoring of the power supply on the AS-Interface cable
- Requires 1 Slot
- 16 bytes are occupied in the I/O operation in the analog address place
- LEDs for the displaying of operation states such as the operational readiness of the connected slaves
- Button for the switching of the operation mode and for the altering of the current configuration
- Connection possibility for the AS-Interface cable over the standard front connector

The CP342-2 recognizes two modes:

- **Configuration mode:**
  This mode is set in the delivered state of the CP342-2 (LED CM).
  The configuration mode serves for the commissioning of an AS-I installation. In this mode, the CP342-2 can exchange data with each of the connected slaves on the AS-I cable. New incoming slaves are quickly recognized from the master and recorded in the cyclic data exchange.

- **Protected mode:**
  One switches into protected mode with the SET- Button.
  In this mode the CP342-2 exchanges data only with slaves that are “configured“. “Configured“ means that slave addresses stored in the CP and configuration data stored in the CP agree with the values of the appropriate slaves.

6.4 BUS ACCESS PROCESS

The master contains a processor, whose software is provided from the manufacturer so that the communication between master and slaves executes fully independent with an addressed and programmed device.

The addressed slaves are cyclically spoken to and questioned by the master module during data exchange.
Therefore the net data rate averages to 4 bits pro call of a slave.
The serial information transmission between the master and the slaves that is implemented by an extremely small frame length with minor overhead, determines a max. cycle time of 5 ms. The real-time requests are fulfilled for most of the control programs in the exercise.

The I/O addressing of a master module for a control program's purpose is identical with the addressing of conventional digital or analog in-/output task modules.
The master module also displays an address range in the CPU, which it can access in the program. Each single AS-I slave allocates a nibble (4 bit large data unit) for inputs and outputs in this address range.

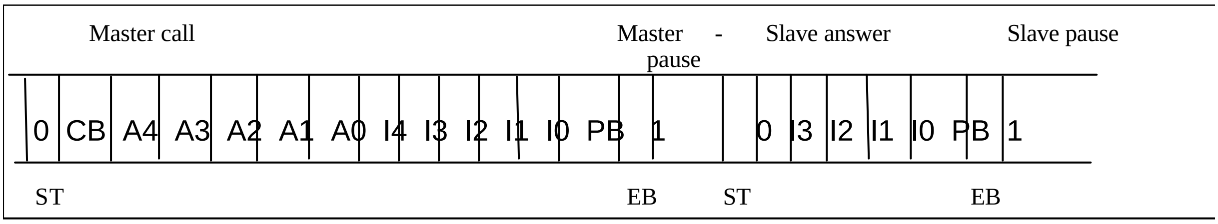
Parameterized frames are accomplished acyclically like configured or addressed frames where no real-time request exists. It is also only possible to run a parameter call to a slave per cycle.

To recognize transmission errors faster, all frames are checked quickly through a routine handler for their validness. If needed the frames run once again.

6.5 DATA TRANSMISSION AND TRANSMISSION SECURITY

The data transmission occurs over the 2-conductor, unprotected and oil resistant AS-I data circuit which is connected to a power supply of 30V DC. The signal is modulated on this power level.

The AS-I messages are shown as follows:



The following bits are relevant for the data transmission:

- ST = Start bit
- CB = Control bit
- A4 ... A0 = Address of the slave (5 Bit)
- I4 ..... I0 = Information unit from the master to the slave (5 Bit)
- I3 ..... I0 Information unit form the slave to the master (4 Bit)
- PB = Parity bit
- EB = End bit

The master can start a call alone if the frame is very short with little corresponding protocol overhead. Through this and through the limited number of slaves the in-/output data can be actualized very quickly in all slaves without the AS-Interface having to operate with a higher data rate. By this principle, the AS-Interface is less accident prone by a feed by foreign electromagnetic fields.
This ruggedness, next to the cheap price, is one of the advantages compared to other systems (e.g. the PROFIBUS) that must carry along many large protocol overheads due to their versatile communication possibilities.

A master call with a slave response executes by the AS-Interface with the following measurements:

Master call :

- The Start Bit ST marks the start of a master call (ST = 0).
- The Control Bit CB recognizes the data.- (CB = 0) Address- (CB = 0), Parameter- (CB = 0) and command call (CB = 1).
- The address of the called slave remains in the 5 bits A4...A0.
- The information unit of the master to the slave is transmitted into the 5 bits I4...I0.
- The Parity Bit PB makes sure that the sum of all “1en“ in the master call is even.
  Through this, the slave can recognize if the transmission of the call occurred error free.
- The End Bit EB highlights the end of the master call (EB = 1).
- The master pause is pushed between 3..10 bit time for the guarantee of transmission safety.

Slave response:

- The Start Bit ST highlights the start of the slave response (ST = 0).
- The information unit of the slave to the master is transmitted into 4 bits I3...I0.
- The Parity Bit PB makes sure that the sum of all “1en“ in the master call is even.
  Through this, the master can recognize if the transmission of the reply occurred error free.
- The End Bit EB highlights the end of the slave response (EB = 1).
- The slave pause is pushed between 3..10 bit times for the guarantee of transmission safety.

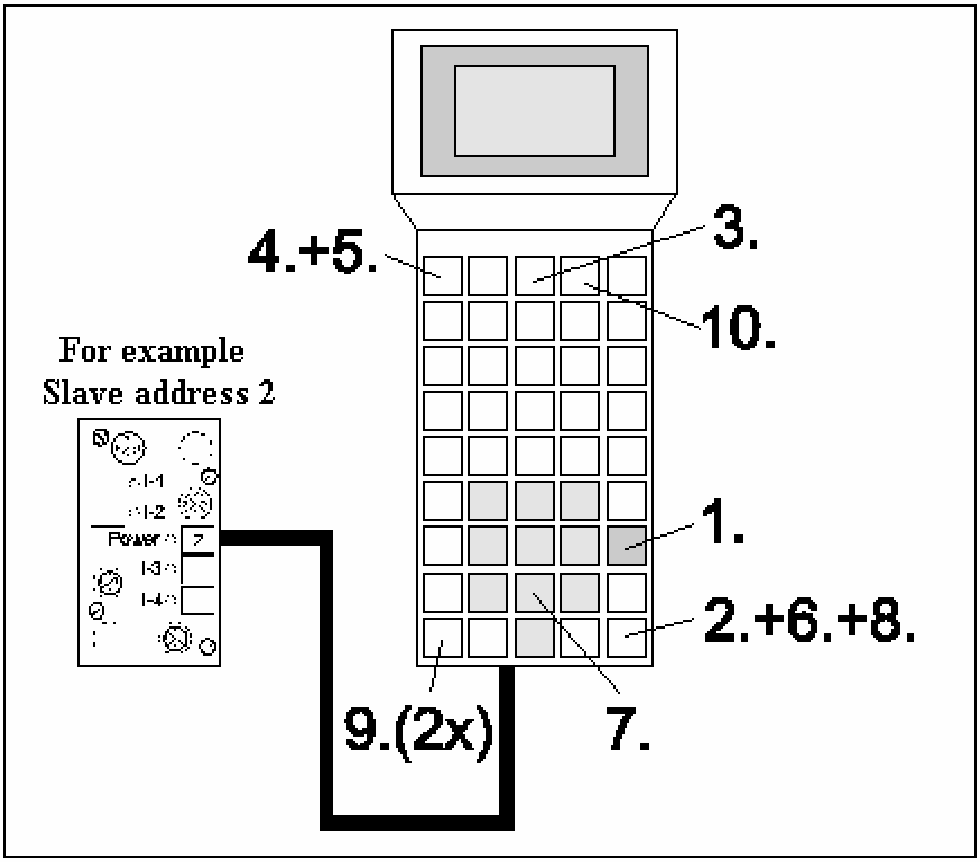
With this transmission process, a very high transmission security is guaranteed. Single, double, and triple errors are always recognized. 4/5ouple errors are always recognized with a probability of 99.9999%. There all slaves are called from the master by each cycle. The failure of a component is quickly recognized.

Through a permanent comparison of the preset/actual configuration in the master, the warning errors like error addressing, are fixed and reported.

6.6 COMMISSIONING OF AN AS-INTERFACE WITH THE CP342-2

The following steps must be followed by the user in order to bring the AS-Interface into operation with the CP342-2, to setup a project and to set the hardware configuration with the CP342-2 AS-I.

1. First all slaves must be assigned explicit addresses with the programming and service unit (PSG):



AS-I slave connected or placed on an integrated base of the PSG

1.1 Turn on PSG (START)
1.2 Activation (ENTER)
1.3 Choose ‘Master‘ (F3)
1.4 Choose ‘Single operation’ (F1)
1.5 Choose ‘New address’ (F1)
1.6 Activate AS-I address (ENTER)
1.7 Enter new address (e.g.: 2)
1.8 Activate inputs (ENTER)
1.9 Return to the main menu (2x ESC)
1.10 Turn off PSG (F4)

2. Then the transfer of the yellow data cable and the connection of all slaves of the power supply (30V DC) and the master as well the repeater in the insulation displacement method occur. Therefore the profile of the data circuit must be accounted for.

3. When an additional power supply (24V DC) is required , it can be connected to the AS-I slaves with the black AS-I power cable. Therefore the profile must be accounted for by the connection in the insulation displacement method of the power cable.

4. Finally you can connect the sensors to the M12 connector for the AS-I slaves and they will be mounted to the slaves.

5. Now the AS-I line is ready and the CP342-2 can be setup and parameterized.

6. In order to bring the S7-300 with the CP342-2 into operation, you must switch the mode switch on the CPU to STOP.

7. Bring the CP342-2 into the configuration mode in which you activate the SET- Button of the CP342-2. The display CM lights now and the recognized slaves are displayed on the diagnostic LEDs of the CP342-2.

   Note:
   You can also insert or remove additional slaves on the AS-I cable. Newly inserted slaves are quickly recognized and activated from the CP3423-2.

8. Activate the SET- Button of the CP342-2. The CP now stores those activated slaves that were indicated. The “actual configuration“ as non volatile preset configuration is switched in the protected operation. The LED “CM“ lights.

9. Now switch the mode switch of the CPU to RUN-P. The system startup of the CP342-2 is now complete .

6.7 ADDRESS ASSIGNMENT OF THE AS-INTERFACE SLAVES

The CP342-2 is assigned over the hardware configuration of a slot.
Pro slot, the CPU activates a memory area with a size of 16 bytes in the memory I/O for the domain. When, for example, the CP342-2 is configured on slot 6 then 16 bytes starting from address 288 are used for the data exchange.
Each slave has a maximum of 4 inputs and 4 outputs. Therefore only 4 bits per slave (nibble) can be assigned in the memory of the CP342-2. The assignment of several slaves to the address area is specified as follows:

Inputs	IN / OUT				IN / OUT				Address	Outputs
PIQ	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**	CP342-2 (PI/PQ)	PIQ
	In4	In3	In2	In1	In4	In3	In2	In1		
	Out4	Out3	Out2	Out1	Out4	Out3	Out2	Out1		
24	Reserved for diagnostics				Slave01				288	64
25	Slave02				Slave03				289	65
26	Slave04				Slave05				290	66
27	Slave06				Slave07				291	67
28	Slave08				Slave09				292	68
29	Slave10				Slave11				293	69
30	Slave12				Slave13				294	70
31	Slave14				Slave15				295	71
32	Slave16				Slave17				296	72
33	Slave18				Slave19				297	73
34	Slave20				Slave21				298	74
35	Slave22				Slave23				299	75
36	Slave24				Slave25				300	76
37	Slave26				Slave27				301	77
38	Slave28				Slave29				302	78
39	Slave30				Slave31				303	79

These assignments are obtained by the AS-I slaves when the addresses from I24.0 are used for inputs and addresses Q64.0 for outputs. Through the program instructions download and transfer, this assignment can be dealt within the OB1.

e.g. L PID288, T ID24 etc. for the inputs.
L QD64, T PQD288 etc. for the outputs.

In order to acquire the address of the second output by the AS-I Slave 2 (Slave 2, Out 2), the following steps needs to be performed:

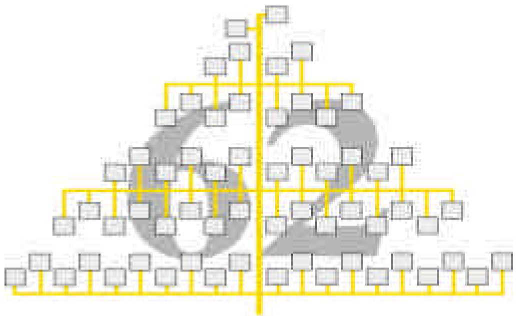
Byte address for Slave02 from the PIQ: 65
Bit address for Out2: 5
Resulting Address: Q 65.5

For the third input by AS-I Slave 7 (Slave7,In3), the address is I 27.2.

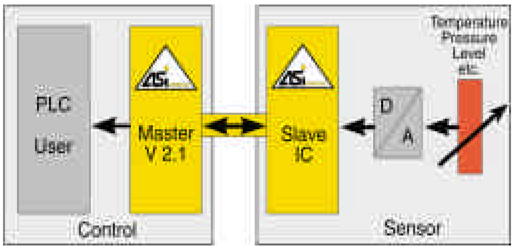
6.8 AS-INTERFACE VERSION 2.1

Additions to the past system properties (Version 2.0):

Addition of up to 62 Slaves



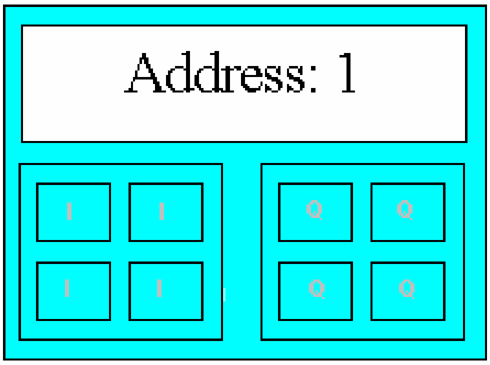
Simpler connection of analog slaves



AS-Interface Version 2.1 in comparison

	So far: Version 2.0	Version 2.1
Number of slaves	Max. 31	Max. 62
Signal amount	124 I + 123 O	248 I + 186 O
Transmission	Data and power up to 8A	Data and power up to 8A
Medium	Unprotected wire 2X1.5mm^2	Unprotected wire 2X1.5mm^2
Max. cycle time	5 ms	10 ms
Analog value transmission	With function block	Integrated into the master
Number of analog values	16 bytes for digital and analog values	124 analog values possible
Access method	Master/slave	Master/slave
Cable length	100 m, up to 300m with repeater	100 m, up to 300m with repeater

System addition: max. 62 slaves

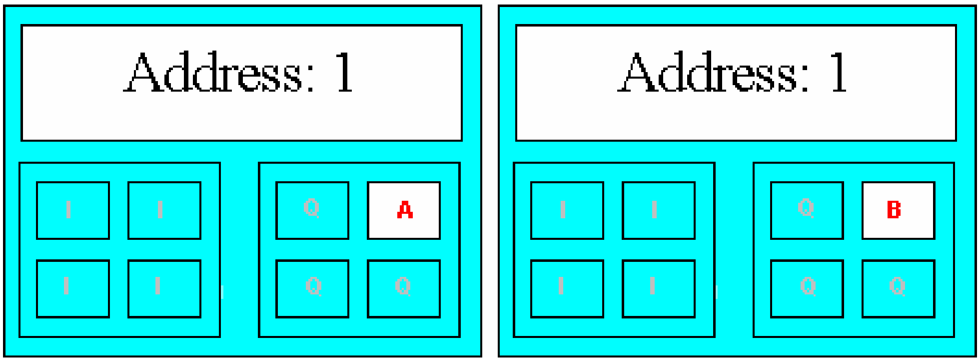


So far:

Each slave has an address(max. 4I / 4Q)

Version 2.1:

2 Slaves, each address with A or B recognition:
A/B slaves ! (Max. 4I / 3Q)



Note:

With an AS-I addressing and diagnostic device for AS-Interface version 2.1 or for CP342-2 (AS-I master with additional specification 2.1), the addresses of the slaves can be set to the value 1 to 31 (resp. 1A to 21A and 1B to 31B).

Further slaves are identified as single slaves.
A/B and single slaves can be operated together.

Communication cycle:

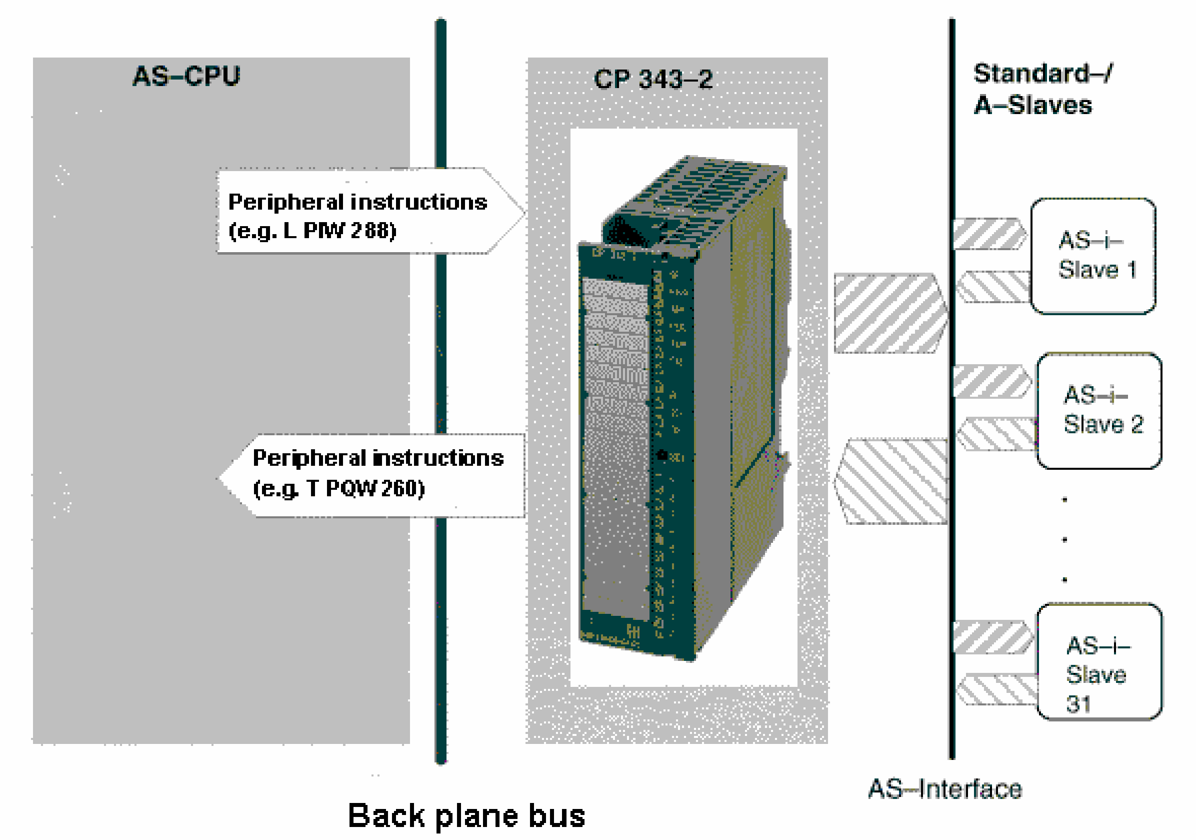
1) Queries all single slaves and all A slaves
2) Queries all single slaves and all B slaves

Cycle duration:

Between 5 ms with use of a max. 31 A or single slaves,
up to 10 ms with use of a max. 62 A and B slaves

6.8.1 AS-I binary values exchange with standard A slaves

You access the binary values of the AS-I standard slaves (resp. A slaves) in the user program over the corresponding STEP 7 peripheral instructions.

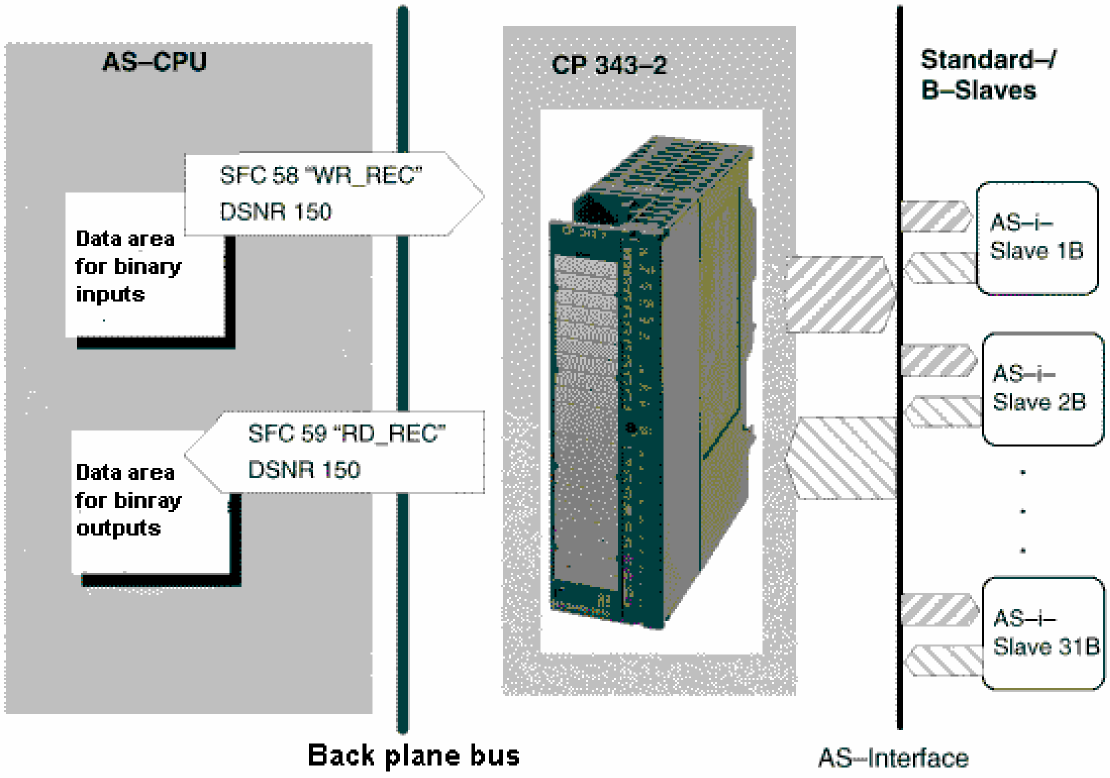


Note

The access of the inputs and outputs of the standard/A slaves occurs by the CP343-2 exactly like it does by the AS-Interface specification 2.0.

6.8.2 AS-I binary values exchange with B slaves

You access the binary values from B slaves in the user program over the system function block SFC 58 (”Write_Record”) / SFC 59 (“Read_Record”) .
You always use record number 150 for access.



Note

The CP 343–2 administrates the binary data of the B slaves in 2 16 byte large areas (an area for the input data and an area for the output data).

Access to binary data from B slaves

In record 150, this structure contains these ranges of the structure of the binary data for the standard and A slaves.

I/O byte number	Bit 7–4	Bit 3–0
n+0	reserved	Slave 1B Bit 3 | Bit 2 | Bit 1 | Bit 0
n+1	Slave 2B	Slave 3B
n+2	Slave 4B	Slave 5B
n+3	Slave 6B	Slave 7B
n+4	Slave 8B	Slave 9B
n+5	Slave 10B	Slave 11B
n+6	Slave 12B	Slave 13B
n+7	Slave 14B	Slave 15B
n+8	Slave 16B	Slave 17B
n+9	Slave 18B	Slave 19B
n+10	Slave 20B	Slave 21B
n+11	Slave 22B	Slave 23B
n+12	Slave 24B	Slave 25B
n+13	Slave 26B	Slave 27B
n+14	Slave 28B	Slave 29B
n+15	Slave 30B Bit 3 | Bit 2 | Bit 1 | Bit 0	Slave 31B Bit 3 | Bit 2 | Bit 1 | Bit 0

The following example program shows the access on the binary data from the B slaves.

Network 1: Binary input data for B slaves read

Comment:

```
CALL "RD_REC"                  //RD_REC
REQ     :=TRUE                 //trigger
IOID    :=B#16#54              //steady value
LADDR   :=W#16#100             //CP address (here 256 dec.)
RECNUM  :=B#16#96              //DS nr.=150 (record for binary data)
RET_VAL :=MW80                 //code for error status
BUSY    :=M82.0                //read activity active (Busy=1)
RECORD  :=P#I 60.0 BYTE 16     //target range of binary data
```

Network 2: Example for binary access on B slaves

Comment:

```
A    I    61.4                 //Slave 2B, connection 1
A    I    61.5                 //Slave 2B, connection 2
A    Q    50.6                 //Slave 4B, connection 3
A    I    63.1                 //Slave 7B, connection 2
A    Q    50.6                 //Slave 4B, connection 3
```

Network 3: Binary ouotput data of B slaves sent

Comment:

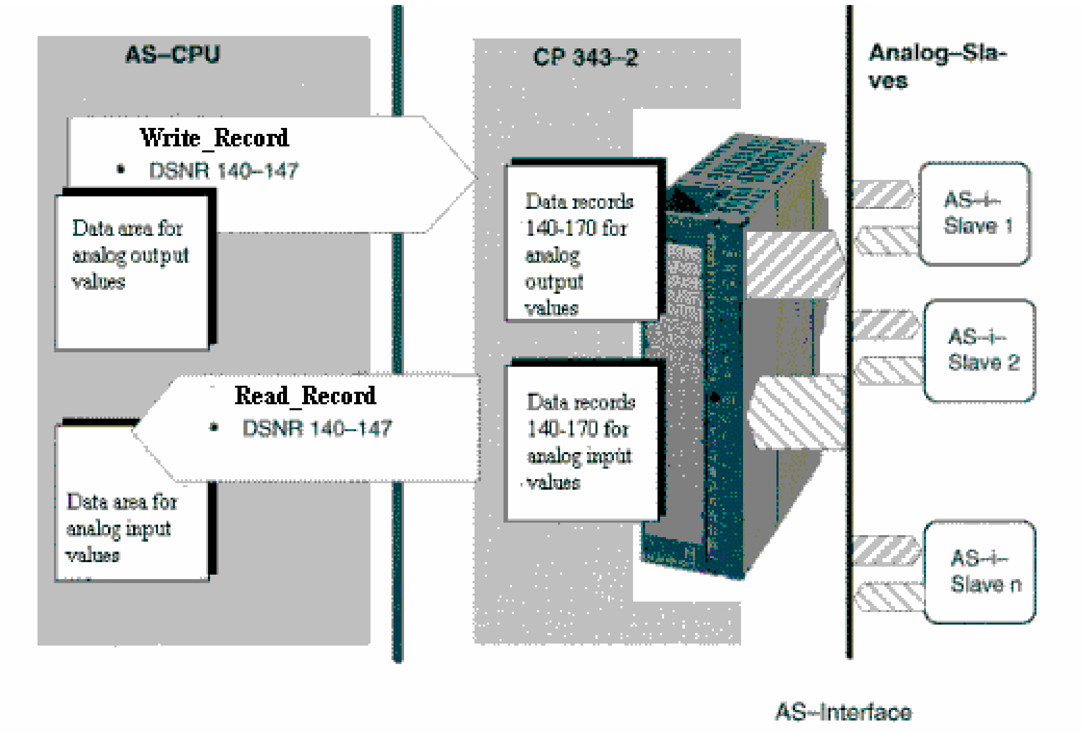
```
CALL "WR_REC"                  //WR_REC
REQ     :=TRUE                 //trigger
IOID    :=B#16#54              //steady value
LADDR   :=W#16#100             //CP address (here 256 dec.)
RECNUM  :=B#16#96              //DS nr.=150 (record for binary data)
RECORD  :=P#Q 48.0 BYTE 16     //source range of binary data
RET_VAL :=MW84                 //code for error status
BUSY    :=M86.0                //write activity active (Busy=1)
```

6.8.3 Transmitting AS-I analog values

You can set up to 31 AS-I slaves with up to 4 analog inputs or outputs. There is no load of the CPU, in which the signal is assigned as a complete analog value to the PLC(12 bit).

You access the analog values from the AS-I analog slaves in the user program over the system function block SFC 58 ("Write_Record") / SFC 59 ("Read_Record"). For this access, you use data record numbers 140 through 147.

For each data record you can use between 2 bytes and a max. of 128 bytes. For each slave address, an 8 byte comprehensive range for the addressing of 4 analog channels is used.



Note
The following types apply only for AS-I slaves that process an analog value transmission for the AS-I slave profile 7.3 and 7.4.
The analog value transmission for the AS-I slave profile 7.1 to 7.2 is not supported by the CP343-2. In this case, the analog transfer software technology must be implemented.

The following table indicates with which data record and from which AS-I slave the analog values are transmitted. The way that the analog values of the analog slaves are ordered is taken from the following table.

The tables are applied for the **analog inputs** and similarly for the **analog outputs**.

	Assigned bytes- Addresses for analog values in the data record							
Address AS–i–Slave	DS 140	DS 141	DS 142	DS 143	DS 144	DS 145	DS 146	DS 147
1	0–7							
2	8–15							
3	16–23							
4	24–31							
5	32–39	0–7						
6	40–47	8–15						
7	48–55	16–23						
8	56–63	24–31						
9	64–71	32–39	0–7					
10	72–79	40–47	8–15					
11	80–87	48–55	16–23					
12	88–95	56–63	24–31					
13	96–103	64–71	32–39	0–7				
14	104–111	72–79	40–47	8–15				
15	112–119	80–87	48–55	16–23				
16	120–127	88–95	56–63	24–31				
17		96–103	64–71	32–39	0–7			
18		104–111	72–79	40–47	8–15			
19		112–119	80–87	48–55	16–23			
20		120–127	88–95	56–63	24–31			
21			96–103	64–71	32–39	0–7		
22			104–111	72–79	40–47	8–15		
23			112–119	80–87	48–55	16–23		
24			120–127	88–95	56–63	24–31		
25				96–103	64–71	32–39	0–7	
26				104–111	72–79	40–47	8–15	
27				112–119	80–87	48–55	16–23	
28				120–127	88–95	56–63	24–31	
29					96–103	64–71	32–39	0–7
30					104–111	72–79	40–47	8–15
31					112–119	80–87	48–55	16–23

Examples / Reading note to the table

1. Configuration:
The analog slaves have the AS-I addresses 1–6.
You use the data record 140 and the value 48 as the data record length.

2. Configuration:
It uses 1 analog slave with the AS-I address 7.
You use the data record 141 and the value 24 as the data record length.

3. Configuration:
The whole address range for 31 analog slaves is used.
You use the data record 140 and the value 128 as the data record length.
Therefore you include the analog slaves 1–16.
For the rest of the analog slaves 17-31, you use the data record 144 and the value 120 as the data record length for the next application.

4. Configuration:
Analog slaves lying in address range 29–31.
You use the data record 147 and the value 24 as the data record length.

Program example:

```
OB1 :  "Main Program Sweep (Cycle)"
Comment:

Network 1: Analog input data for slaves 5 to 8 read
Comment:

   CALL "RD_REC"              //RD_REC
   REQ     :=TRUE             //trigger
   IOID    :=B#16#54          //steady value
   LADDR   :=W#16#120         //CP address (here 288 dec.)
   RECNUM  :=B#16#8D          //DS nr.=141 (record for analog slaves 5-20)
   RET_VAL :=MW80             //code for error status
   BUSY    :=M82.0            //read activity active (Busy=1)
   RECORD  :=P#I 20.0 BYTE 32 //target range of analgo data

Network 2: Example for analog values processing
Comment:

      L     EW    22          //Slave 5, connection 2 (read analog value)
      L     2                 //read value 2 into ACCU1
      *I                      //double read in value
      T     AW    28          //Slave 6, connection 1 (send analog value)

Network 3: Analog output data for slaves 5 to 8 sent
Comment:

   CALL "WR_REC"              //WR_REC
   REQ     :=TRUE             //trigger
   IOID    :=B#16#54          //steady value
   LADDR   :=W#16#120         //CP address (here 288 dec.)
   RECNUM  :=B#16#8D          //DS nr.=141 (record for analog slaves 5-20)
   RECORD  :=P#Q 20.0 BYTE 32 //source range of analog data
   RET_VAL :=MW84             //code for error status
   BUSY    :=M86.0            //write activity active (Busy=1)
```

7. THE PROFIBUS

Out of the processing of the “BMFT- Combined project field bus”, in which 13 companies and 5 colleges are associated with, the DIN 19245, otherwise known as the ”**PROFIBUS**“ (**PRO**cess **FI**eld **BUS**) was developed in the beginning of 1991.

The goal of the project was to develop a field bus system that made it possible to link up a network of automation devices of the lowest field level from sensors and actors up to the process control in the cell level.
This national scaling was met in the European Norm EN 50170 in 1996.

A field bus standard was made with the PROFIBUS which is open and company neutral. Devices of different manufacturers are equipped with a proper interface.
Due to it’s comprehension and also differentiated functionality, the PROFIBUS masks large areas of sensors/actors next to the field and cell levels and guarantees a good uniformity for the higher level bus systems of the process control level.

The characteristics of the PROFIBUS are thoroughly described in the following sections.

- PROFIBUS- FMS
- PROFIBUS- PA
- PROFIBUS- DP

These 3 compatible variants of the PROFIBUS co-ordinate properties and user ranges. They make a transparent communication from sensor/actor to the systems in the process control level possible. Planning, installation and maintenance are therefore economic and technically simple to implement.

7.1 PROFIBUS-FMS (Fieldbus Message Specification)

PROFIBUS-FMS builds a bridge between the cell and the field level. It corresponds to the DIN 19245 and is integrated into the European field bus norm EN 50170. As a result of its performance capable user functions, it is qualified for sophisticated communication tasks like for the data exchange of the intelligent automation devices to one another. Therefore one differentiates which data between active nodes (master) and passive nodes (slave) under the use of token passing with documented master-slave processing cyclic or acyclic exchanges. A transmission rate of 1.5 Mbit/s is possible.

The token passing method guarantees the allocation of the bus access authorization in which the token is contained inside of an exactly specified time frame.

The master-slave method enables the master to possess the direct send authorization when the assigned slaves communicate with the master.

PROFIBUS-FMS works object oriented and makes standardized access possible to variables, programs and large data areas (domains). All communication objects of a node are entered by the configuration of a bus system in the object directory. The access to the communication object occurs though a short representation (index) or optimally over symbolic names. The data transmission occurs on the base of logical connections.

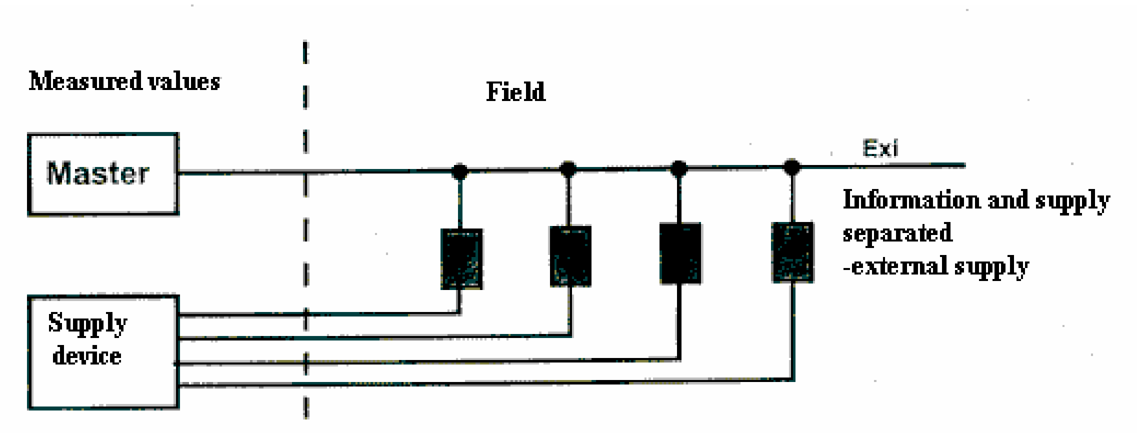
7.2 PROFIBUS-PA (Process Automation)

PROFIBUS-PA is the PROFIBUS variant for the process automation in process engineering. Originally the PROFIBUS-PA was specified in the specification ISP 3.0 (Interoperable Systems Project). Since June 1994, it was currently named PROFIBUS-ISP. In the beginning of 1995 this variant was renamed to PROFIBUS-PA. It uses processes specified in the international NORM IEC 1158-2 transmission technology and enables these processes through intrinsic safety and remote power feeding of nodes. These properties permit that also during the current operation, field devices can be cut off. For this, a non intrinsically safe field bus must have been completely disconnected.

Basic data of the Norm IEC 1158-2 (the PROFIBUS-PA) given:

- Digital, bit synchronous data transmission
- Data rate 31,25 kBit/s
- Error safe start and end delimiter
- Send level 0,75 V_{SS} to 1 V_{SS}
- Remote power feeding over signal download
- Lines- Stars- and tree topology are supported
- Power transmission: DC
- Up to 32 nodes per cable segment
- Length of up to 1900 m per cable segment (without repeater)
- Bus expandable with a maximum of 4 repeaters in a row

Fieldbus in the intrinsically safe field range with an incoming supply over the field bus (top) and the external incoming supply (bottom).



The intrinsically safe transmission technology for the IEC 1158-2 should be available for the user profiles of the PROFIBUS-FMS and PROFIBUS-DP. Next to these user profiles, communication and device profiles are defined.

7.3 PROFIBUS-DP (Distributed I/O)

The PROFIBUS-DP is positioned in the DIN E 19245 Part 3 and integrated into the European field bus norm EN 50170. It is tailored on the requirements for faster, more efficient data exchange between the automation devices and the distributed devices like binary or analog in-/output modules and actuators. This shift of the peripherals in the field level enables the incoming supply by the cables. For this reason the user field of the PROFIBUS is added underneath thereafter. The PROFIBUS-DP uses the approved properties of the PROFIBUS transmission technology and of the bus access protocol (DIN 19245 Part 1). It adds this to the functions with which the high requirements are fulfilled for the system reaction time in the range of the distributed I/O. Therefore it is possible for the PROFIBUS-FMS and PROFIBUS-DP to simultaneously execute on a single cable.

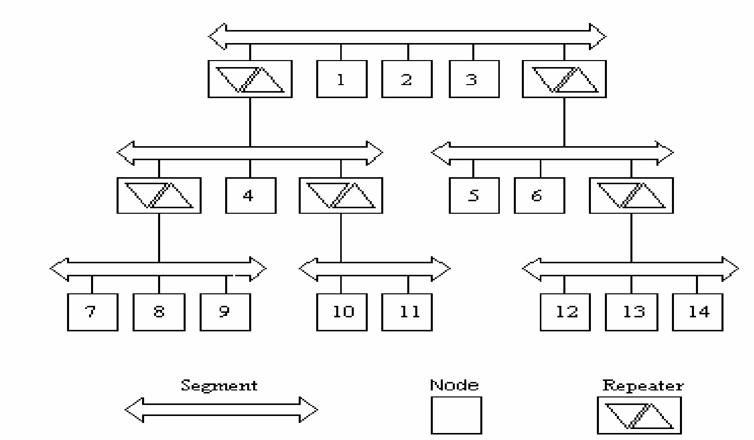
In the following section, the PROFIBUS-DP will be dealt with in more detail.

7.3.1 Technical data for the PROFIBUS-DP

The following Parameter are specified for the PROFIBUS-DP in the Norm 50170.

- The bus allocation occurs by the PROFIBUS-DP after the processing of ''Token passing with supported master-slave'.
- Typical cycle time is given with 5 -10 ms.
- A maximum of 127 nodes with a frame length of 0-246 bytes user data can be connected.
- Standard-transmission rates are defined as 9,6 KBaud / 19,2 KBaud / 93,75 KBaud / 187,5 KBaud / 500 KBaud / 1,5 MBaud / 3 MBaud / 6 MBaud / 12 MBaud.
- The bus configuration is modular expandable where as the peripherals and field devices are connected and unconnected during the operation.
- The data transmission occurs either over a 2 wire cable with a RS-485 interface or over a fiber-optic cable. We restrain ourselves here to the 2 wire cable data transmission possibility.
- The unprotected and twisted 2 wire cable (Twisted Pair) has a minimum cross section of 0.22 mm² and must be connected at the end with the shaft resistor.
- An area-wide network occurs by the PROFIBUS-DP through the compartmentalization of the bus system in the bus segments that can be connected over repeaters.

- The topology of the single bus segment is the line structure (up to 1200 m) with short drop cables (<0.3m). With the help of repeaters, a tree structure can also be constructed as shown:



- The maximum number of nodes per bus segment amounts to 32. More lines can connected under one another through performance enhancements (repeaters) where by it is noted that each repeater counts as a node. In total a maximum of 128 nodes are connectable (over all bus segments).

- By 1,5 MBaud repeaters 10 bus segments can be operated in a row (9 repeaters), by 12 MBaud-repeaters only 4 bus segments (3 repeaters).

- Transmission distances to 12 km are possible by electrical configuration and to 23.8 km with optical configuration. The distances are dependent on the transmission rate, like shown here (electrical configuration).

Transmission rate in Kbaud	9,6	19,2	93,75	187,5	500	1500	3000	6000	12000
Length per segment in m	1200	1200	1200	1000	400	200	100	100	100
Max. length in m	12000	12000	12000	10000	4000	2000	400	400	400
By number of bus segments:	10	10	10	10	10	10	4	4	4

- By the PROFIBUS-DP, extensive diagnostic possibilities are possible with the help of software tools.

7.3.2 Configuration of the Profibus-DP

7.3.3 Devices types by Profibus-DP

DP master class 1 (DPM1)

Here the PROFIBUS concerns a central controller which exchanges information with the distributed stations (DP slaves) in a specified message cycle. The following master-slave user functions are concretely supported:

- Collection of diagnostic information of the DP slaves.
- Cyclic user data operation
- Parameterizing and configuration of the DP slaves
- Controlling of DP slaves with control commands

These functions are independently handled from the user interface of the DP master (class 1). The interface between the user and the user interface is calibrated as a data service interface. Typical devices are programmable logic controllers (PLC), computerized numerical control (CNC) or robot control (RC).

DP master class 2 (DPM2)

Devices of this type are programming, configuring or diagnostic devices. They are set by commissioning in order to specify the configuration of the DP system that stands from the number of DP devices, of allocation between node addresses by the bus and of I/O addresses such as indication over data consistency, diagnostic format and bus parameters.

Between the DP slave and the DP master (class 2), the following additional functions are possible next to the master-slave functions of the DP master (class 1):

- Reading of the DP slave configuration
- Reading of the in- and output values
- Address allocation by DP slaves

Between the DP master (class 2) and the DP master (class 1), the following functions are available (most of these functions execute acyclically):

- Entry of the available diagnostic information of the assigned DP slaves in the DP master (class 1)
- Upload and download of data records
- Activation of the bus parameter records
- Activation and deactivation of the DP slaves
- Adjustment of the operation type of the DP master (class 1)

DP-Slave

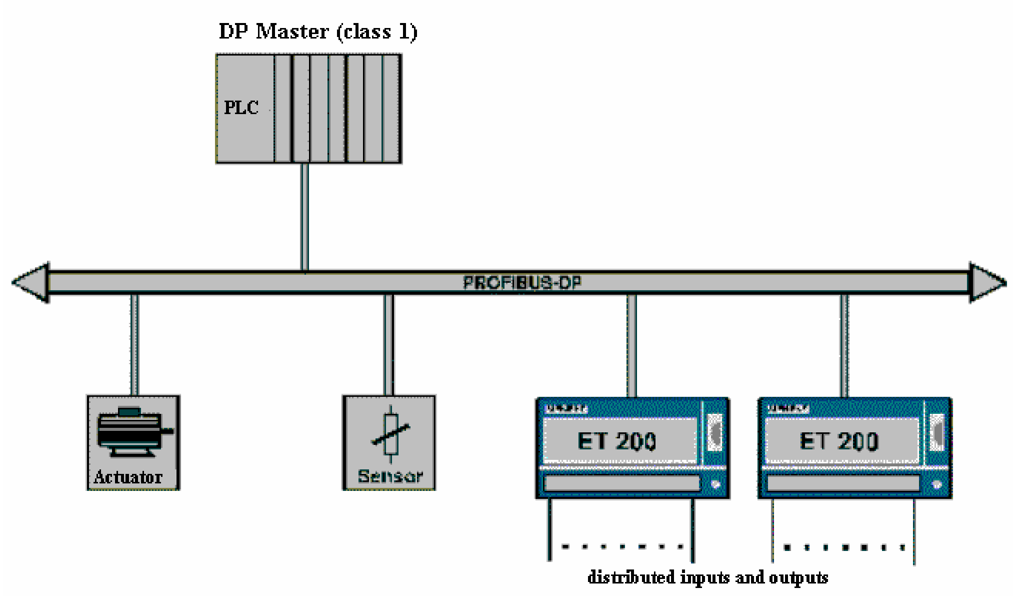
A peripheral device (sensor/actuator) is identified as a DP slave when the input data is read in and the output data is given to the peripherals. It is also possible for the devices to provide only input or output information. Typical DP slaves are devices with binary in-/outputs for 24 or 200 V, analog inputs, analog outputs, counters, and also:

- Pneumatic valve islands
- Code reading devices
- Proximity switches
- Measurement value sensors
- Drive controllers

Most of the input and output data is device dependent and is allowed to contain a maximum of 246 bytes for inputs and 246 bytes for outputs. From cost and implementation technological principles, many of the available devices have a maximal user data length of 32 bytes.

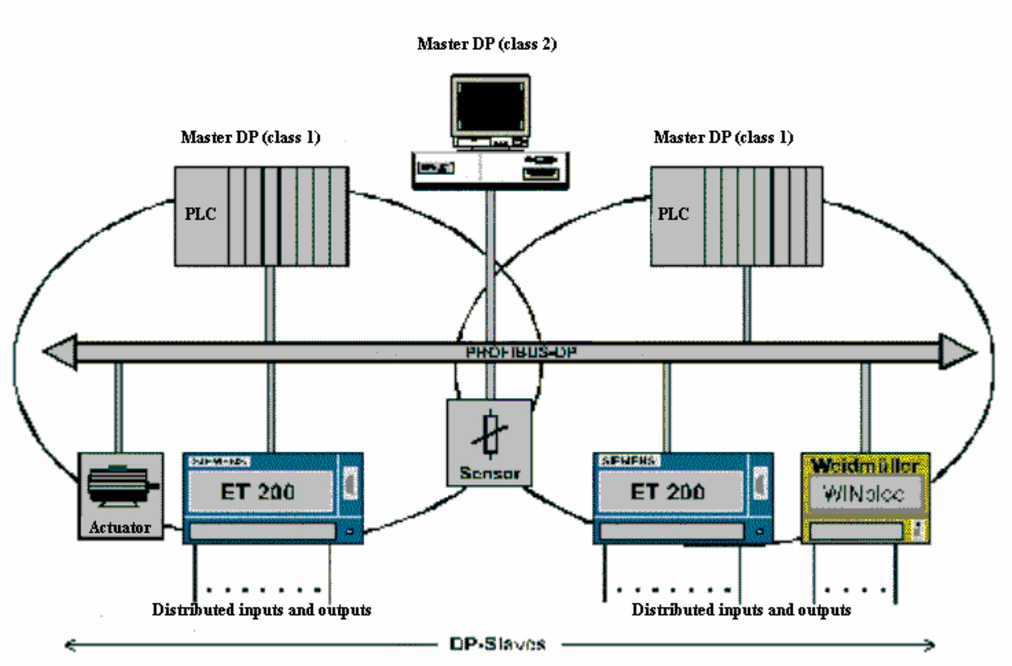
7.3.4 System configuration

The PROFIBUS DP lets mono or multi systems be implemented. Through this stands a high degree in flexibility by the system configuration. A maximum of 126 devices can be connected to a Profibus. The bus structure offers the possibility to couple and uncouple a single node reaction-free in order for the system to be taken stepwise into operation. Further enhancements have no influence on the configuration ready installed devi ces. By the mono master system, only a master is active on the bus in the operation phase of the bus system. The PLC is the central control component in which the DP slaves are coupled over the transmission medium to the PLC. It presents a clean master-slave access process. With this system configuration, the smallest bus cycle is accomplished.



PROFIBUS-DP-Mono-Master-System

You find more masters on the bus in a multi master operation. You can either construct independent sub systems existing from each master and the additional slaves or function these systems as additional configuration and diagnostic devices. The input and output images of the slaves can be read by all masters. The description of the outputs is only possible for a master (class 1). Naturally, the masters can also exchange data frames with one another over the AGAG-Connection. Multi master systems obtain an average bus cycle.



PROFIBUS-DP-Multi-Master -System

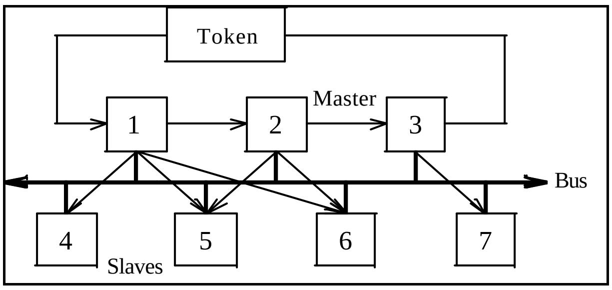
7.4 BUS ACCESS PROCESS

The bus access process is naturally in tight connection with the topology of the PROFIBUS system. Basically the communication network is differentiated from the star, ring, and bus network.
With a star configuration, all attached nodes communicate over a central computer that appoints the whole performance capability and function security. The nodes of a ring network build a closed ring configuration. The advantage is that a node always knows from where it's information comes. The disadvantages are described below:

a) When a node is filled in, the whole system fills as a result of the ring interruption.
b) The cable cost is relatively high because the first node must be connected with the last node.

The PROFIBUS therefore uses the bus (resp. the line network). By these systems all nodes are attached over a small drop cable on a whole cable. From this principle each message from each node is recognized.

The sending capability must be governed through the bus access process. Two processes come to the PROFIBUS for use. **Token passing** and **Master/Slave** processes. Therefore the PROFIBUS access process is also referred to as "hybrid". The master occupies the right of the bus access by the master/slave process. The passive slaves only answer to the instruction of the master. The other method is by the token passing process. Here the access right over the "Token" is allocated and assigned to the active nodes one after the other. Only the master that occupies the tokens can communicate with the other active and passive nodes on the bus access.



Representation of the hybrid access process

So two important factors of the bus are implemented:

- Automation devices receive enough time with appropriate intelligence in order to follow through with your communication tasks (through token passing process).

- The data exchange between the automation devices is implemented with the simple process peripherals (I/O level) under real-time conditions (through master/slave process).

7.4.1 Frame configuration

Not several characters but a complete character packet are transferred over the PROFIBUS. The configuration of this character packet is firmly provided and standardized.

Across from the conditions of the higher data transmission security stands the minimization of the protocol overhead in order to be able to transmit a high net data rate. The possible frame variants are listed in picture 5-3:

- Frame with firm information field length without a data field.

Call frame resp. Response frame (without SYN):

SYN	SD1	DA	SA	FC	FCS	ED

Short acknowledgement:

SC

- Frame with firm information field length with data field (data field length 8 Byte)

Call frame resp. Response frame (without SYN):

SYN	SD3	DA	SA	FC	Data-unit	FCS	ED

- Frame with variable information field length (max. data field length 246 bytes)

Call frame resp. Response frame (without SYN):

SYN	SD2	LE	LEr	SD2	DA	SA	FC	Data-unit	FCS	ED

- Token frame

SYN	SD4	DA	SA

Representation and commenting of different frames:

SDx	Start byte	Start Delimiter	Determination of frame format
LE	Length byte	Length	Value between 4 ... 249
LEr	Length byte	repeated	For security with reiteration
DA	Target address	Destination Address	Receiver of the information
SA	Source address	Source Address	Sender of the information
FC	Control byte	Frame Control	Recognition of the frame type
FCS	Check byte	Frame Check Sequence	Check information
ED	End byte	End Delimiter	End boundary
SC	Single character	Single Character	Code E5H
SYN	Synchronization bit		Min. 33 bit release condition

The start bytes SDx characterize the frame type where as the control bytes FC recognize the frame type. The codes SD1 : 10H, SD2 : 68H, SD3 : A2H and SD4 : DCH are assigned to the start bytes.

The configuration of the control byte FC should be explained further in the following chapter. The bit b8 (res) is reserved for the IEC 870-5-2. Bit b7 recognizes the frame type, where upon the conditions b6 and b5 are compiled (control information). The transmission functions are coded and set in the bits b4...b1.

Frame = 1 Call frame (Request-, Send/Request- Frame)

b8	b7	b6	b5	b4	b3	b2	b1
Res	Frame **1**	FCB	FCV	Function			

FCB	Frame Count Bit	: 0/1, alternating call result bit
FCV	Frame Count Bit valid	: 0 alternating function of the FCB not valid
		: 1 alternating function of the FCB valid

Code Nr.	Function
0, 1, 2	FC-Code 0, 1, 2 (for IEC870-5-2)
3	Send Data with Acknowledge low
4	Send Data with no Acknowledge low
5	Send Data with Acknowledge high
6	Send Data with no Acknowledge high
7	Reserve
8	FC-Code 8 (for IEC 870-5-2)
9	Request FDL-Status with Reply
10, 11	Reserve
12	Send and Request Data low
13	Send and Request Data high
14	Request Ident with Reply
15	Request LSAP-Status with Reply

Frame = 0 Acknowledgement or Response frame

b8	b7	b6	b5	b4	b3	b2	b1
Res	Frame **0**	Station type		Function			

Station type Node type : Station type and FDL status

b6	b5	<== Bit position
0	0	Passive node
0	1	Active node not ready
1	0	Active node ready for the logical token ring
1	1	Active node in the logical token ring

Configuration of the control byte:

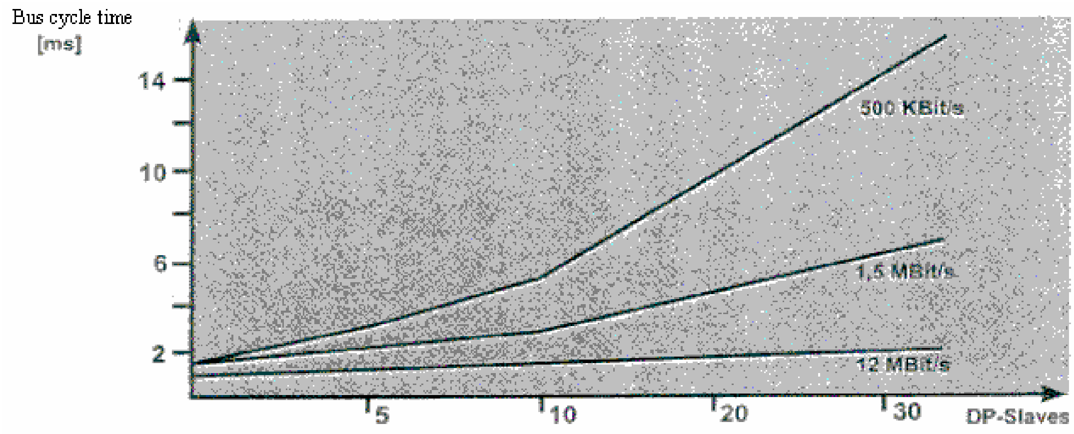
Code Nr.	Function
0	Acknowledgement positive (OK)
1	Acknowledgement negative (UE) User-Error
2	Acknowledgement negative (RR) no Resource for Send Data
3	Acknowledgement negative (RS) no Service activated
4, 5, 6, 7	Reserve
8	Response FDL/FMA1/2-Data low (DL)
9	Acknowledgement negative (NR) no Response FDL/FMA1/2-Data
10	Response FDL-Data high (DH)
11	Reserve
12	Response FDL-Data low (RDL) no Resource for Send Data
13	Response FDL-Data high (RDH) no Resource for Send Data
14, 15	Reserve

7.5 DATA TRANSMISSION AND TRANSMISSION SECURITY

7.5.1 Time response of the PROFIBUS-DP

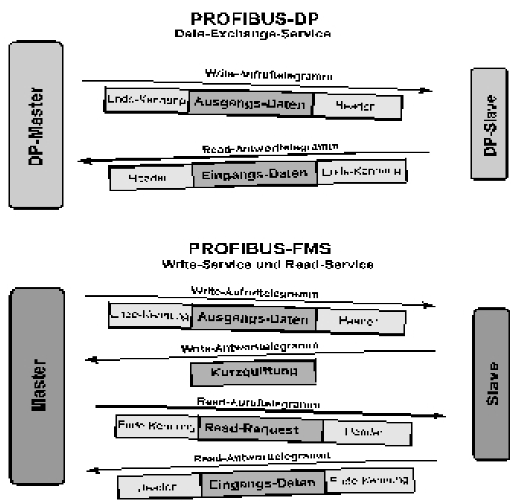
The “Slim“ architecture from the PROFIBUS-DP and the resulting low-order bus cycle times make it interesting for the time critical applications.

A more important method for improving the time response is the increasing of the transmission rate to 12 Mbit/s. The time response of the field bus with 12 Mbit/s by the PROFIBUS-DP is also negligible by a large number of slaves and I/O data. This high transmission rate leaves itself however with industrial, strong electromagnetic “contamination”. The environment does not always realize this. In this case the transmission rate is reduced progressively.



Bus cycle time of a PROFIBUS-DP-Mono-Master-System

The significant increase in speed compared to the PROFIBUS-FMS is particularly attributed to the PROFIBUS so that the transmission of the in- and output data is accomplished in a message cycle through the use of the SRD service (Send and receive data service) of the 2nd layer.



Message cycles of DP and FMS in comparison

7.5.2 Protection mechanisms

Protective devices are required against error parameterizing or failure of the transmission devices from safety reasons. Monitoring mechanisms which are implemented in the form of time monitoring, are used by the master and also by the slave in the PROFIBUS-DP. The monitoring interval is therefore specified by the configuration of the DP system.

By DP master:

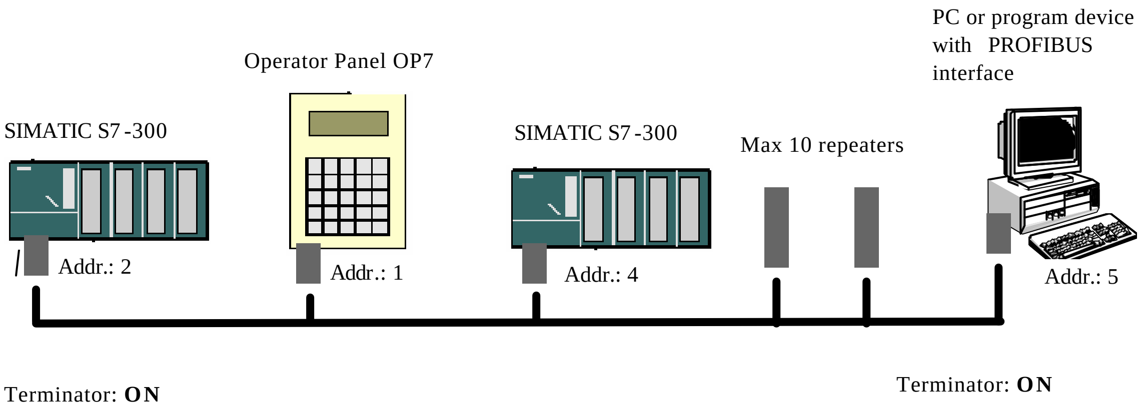
The master accomplishes a time monitoring for each associated time monitoring of the user data transfer. Therefore the master is checked if a normal user data transfer with the slave took place at least one time inside of an appointed, specific time frame (the data control time). If this is not the case, the user is informed. If the automatic error handling is disabled, then the master exits the state operate and switches the outputs of all slaves into the protection state.

By DP slave:

The slave accomplishes the address time monitoring for the recognition of errors of the master or of the transmission line. No data processing takes place inside of a specified address interval with the assigned DP master. Then the DP slave switches the outputs in the safe state independently. Additionally by the multi master operation, access protection is required for the in- and outputs of the slaves in order to ensure that the direct access goes out only from the qualified master. The slaves set an image of the inputs and outputs for operation for all other masters so that the image can be read from each remaining master without access capability.

7.6 COMMISSIONING OF THE PROFIBUS-DP

The configuration of a PROFIBUS network with an electrical network (Where we restrict ourselves here!) can be seen as follows:



Noticeable is the similarity to the MPI network. The physical configuration is identical. The addresses of the single node can be freely assigned between 0 and 126 (standard setting). It is important that each node possesses an explicit address.

In order to avoid cable reflections, the addresses on the first and last nodes of the PROFIBUS network must be switched in the slotted, integrated terminator.

7.6.1 Commissioning of the PROFIBUS- DP with the CPU 315-2DP

The CPU 315-2DP is a CPU that is delivered with an integrated PROFIBUS DP interface. For the CPU 315-2DP, the following PROFIBUS protocol profiles are available:

- DP interface as master according to EN 50170.
- DP interface as a slave according to EN 50170.

The PROFIBUS-DP (Distributed I/O) is the protocol profile for the connection of distributed I/O/field devices with very fast reaction times.

A further characteristic is that the address of the in- and output modules by this CPU can be parameterized. This makes sure that the nodes on the PROFIBUS-DP in the process image of the CPU are connected.
The data of a PROFIBUS-DP slave is deposited e.g. on the free input and output devices in the process image.

Note:
A MASTER- MASTER communication over the FDL protocol is not possible by the CPU 315-2DP.

7.6.2 Commissioning of the PROFIBUS-DP with the CP342-5DP

The PROFIBUS communication processor CP 342-5DP makes it possible for the S7-300 to be connected to the PROFIBUS with the protocol profile distributed peripheral (DP).

The parameterization of the PROFIBUS parameter for the PLC such as the configuration of the PROFIBUS network occurs with the Software STEP 7. In addition the software "NCM S7 PROFIBUS" (Already contained in STEP 7 V5.x!) is required for the CP342-5DP. Therefore the user has an operative configuration tool for the central and distributed configuration.

For the SIMATIC S7-300 with the CP342-5 as a combimaster, the following protocol profiles are available:

- DP interface as a master or slave according to EN 50170.
- SEND/RECEIVE interface (AG/AG) according to the SDA service
- S7-Functions. These offer an optimal communication in the SIMATIC S7/M7/PC network.

Note
SEND/RECEIVE (FDL interface) offers functions with which the communication between SIMATIC S5 and S7 and to a PC can be simply and quickly implemented.

7.6.3 Blocks for the CP342-5DP

On part of the user program, the transmission of the data area for the DP and FDL communication is activated through programmed FC block calls and the following execution is monitored. The important FC blocks for the communication are in the library **"SIMATIC_NET_CP"**. In order to user these functions, the functions must be copied into the specified project.

For the communication between the master and the slaves

DP-Send (FC1), DP-Receive (FC2)

Note
These blocks can only be called once!
The data range must be efficient for all slaves.

For the communication between the master AGs

AG-Send (FC5), AG-Receive (FC6)

Note
With this block, a communication between the Masters AGs is possible over a FDL connection. More calls are possible with different FDL-IDs.

7.6.3.1 DP-SEND (FC1)

The FC block **DP-SEND** transfers data from the user program in the CPU to the PROFIBUS-CP. The DP-SEND has the following importance for the operation type of the PROFIBUS-CP:

- By the use in the DP master

The block gives out data of a given DP output area to the PROFIBUS-CP for the tasks in the distributed peripheral. .

- By the use in the DP slave

The block gives out a given DP data range of the CPU in the send buffer of the PROFIBUS-CP for the transmission in the DP master.
By the call of the FC block DP-SEND the following parameters must be entered:

Name	Type	Range of values	Comment
CPLADDR	WORD		Modules-start address (can be extracted in STEP 7 in the configuration table under Edit ⇨ Configuration) Must be entered in HEX format e.g. 256 = 100Hex
SEND	ANY		Declaration of addresses and lengths of the DP send range (the address can refer to the PA areas, the bit memory address area, and data block area) e.g. P# Q10.0 Byte 4
DONE	BOOL	0: - 1: New data	State parameter displays, if the application was handled error free.
ERROR	BOOL	0: - 1: Error	Error display
STATUS	WORD		Status display

Formal parameters for the function DP-SEND

7.6.3.2 DP-RECEIVE (FC2)

The FC block **DP-RECV** receives data over the PROFIBUS DP. The DP-SEND has the following importance for the operation type of the PROFIBUS-CP:

- By the use in the DP master

The block accepts process data of the distributed peripheral such as status information in a given DP input range.

- By the use in the DP slave

The block accepts the transmitted DP data from the DP master out of the receive buffer of the PROFIBUS-CP in a given DP data area of the CPU.
By the call of the FC block DP-RECV, the following parameters must be entered.

Name	Type	Range of values	Comment
CPLADDR	WORD		Modules-start address (can be extracted in STEP 7 in the configuration table under Edit ⇨ Configuration) Must be entered in HEX format e.g. 256 = 100Hex
RECV	ANY		Declaration of the address and length of the DP-receive range (the address can refer to the PA areas, the bit memory address area, and data block area) e.g. P# I10.0 Byte 4
NDR	BOOL	0: - 1: New data accepted	The state parameter displays if new data was accepted.
ERROR	BOOL	0: - 1: Error	Error display
STATUS	WORD		Status display
DPSTATUS	BYTE		DP status display

Formal parameters for the function DP-RECV

Note
In the status display, codes are filed and states or errors are displayed. The codes can be determined by the table in the online help.

7.6.3.3 AG-SEND (FC5)

The FC block **AG-SEND** assign data to the PROFIBUS-CP for the transmission over a configured AGAG connection.

The entered data range can be a PA range, a bit memory address area or a data block area.

The error free execution is signaled when the whole AGAG data range over the PROFIBUS DP was transmitted.

Name	Type	Range of values	Comment
ACT	BOOL	0, 1	By ACT=1, the LEN bytes are sent out to the entered AGAG data range with the parameter SEND. By ACT=0 the status displays DONE, ERROR and STATUS are updated.
ID	INT	1, 2,...16	Connection number of the AGAG connection
LADDR	WORD		Modules-start address (can be extracted in STEP 7 in the configuration table under Edit ⇨ Configuration)Hex-Format
SEND	ANY		Declarations of addresses and lengths of AFAF send range (the address can refer to the PA areas, the bit memory address area, and data block area) Pointer-Format
LEN	INT	1, 2,...240	Number of bytes that should be sent from the AGAG data area with the application.
DONE	BOOL	0: - 1: New data	State parameter displays if the application was handled error free.
ERROR	BOOL	0: - 1: Error	Error display
STATUS	WORD		Status display

Formal parameters for the function AG-SEND

7.6.3.4 AG-RECEIVE (FC6)

The FC block **AG-RECV** accepts data from the PROFIBUS-CP over a configured AGAG connection.

The accepted data range can be a PA range, a bit memory address area or a data block area.

The error free execution is signaled when the whole AGAG data range was transmitted over the PROFIBUS DP.

Name	Type	Range of values	Comment
ID	INT	1, 2,...16	Communication number of the AGAG connection
LADDR	WORD		Modules-start address (can be extracted in STEP 7 in the configuration table under Edit ⇨ Configuration)Hex-Format
RECV	ANY		Declarations of addresses and lengths of DP send range (the address can refer to the PA areas, the bit memory address area, and data block area) Pointer-Format
NDR	BOOL	0: - 1: New data	The state parameter is displayed if new data was accepted.
ERROR	BOOL	0: - 1: Error	Error display
STATUS	WORD		Status display
LEN	INT		Number of bytes that were accepted from the PROFIBUS-CP in the AGAG data range.

Formal parameters of the function AG-RECV

Call of the functions in the control program

The function calls take place in the STEP 7 user program over **CALL FC xxx**.
The call of the functions for the DP communication takes place cyclically from the OB1. Therefore the DP-RECV is called at the beginning of each cycle and the DP-SEND at the end.
The calls of the function for transmission (receiving of AGAG data ranges) takes place in the rule event-driven from other functions or out of function blocks. Therefore the AG-RECV is also always called before the function AG-SEND in the running of a cycle.