ifm electronic

Supplementary device manual

AS-i controllere with Ethernet programming interface

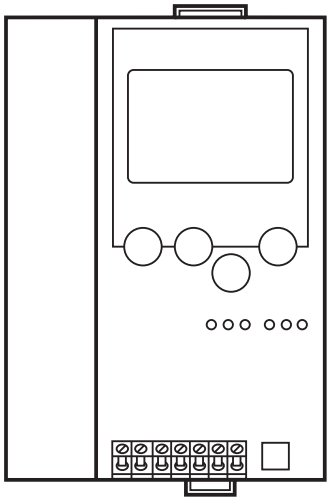
ecomat300

AC1353 / AC1354
AC1355 / AC1356
AC1357 / AC1358

Firmware version RTS 2.x

Target from V15 onwards

for CoDeSys® from version 2.3 onwards



7390565 / 02 05 / 2011

As on: 05 May 2011

Contents

1 On this manual

In this chapter you will find an overview of the following points:

- What do the symbols and formats stand for?
- What devices are described in this manual?
- How is this manual structured?

1.1 What do the symbols and formats stand for?

The following symbols or pictograms shall illustrate our remarks in this manual:

1.1.1 Warning levels, signal words

⚠ DANGER

Death or serious irreversible injuries are *to be expected*.

⚠ WARNING

Death or serious irreversible injuries *may result*.

⚠ CAUTION

Slight reversible injuries *may result*.

NOTICE

Property damage is to be expected or possible.

NOTE

Important note for the correct handling of this product or the manual.

1.1.2 Symbols and formats

○ …	A state to be prevented to avoid a danger.
► …	Instruction
> …	Reaction from the device or software
→ …	Means: "see"
abc	Active cross-reference (link) to another part of the text or an external target on the internet

[…]	[Designation] of key, signalling lamp, button, menu item .... Several buttons or menu items to be selected successively are indicated as follows: ► [1st step] > [2nd step] > [3rd step] Several buttons to be clicked simultaneously are indicated as follows: ► [Ctrl] + [Alt] + [Del]
ABC	DESIGNATION of a parameter (in block capitals)
ABC	`DESIGNATION` of file names (in Monospace font)

1.2 What devices are described in this manual?

This manual presents the AS-i controller family from **ifm electronic gmbh**

- with AS-i version 2.1 master
- with a firmware from version RTS 2.2 onwards
- with the target from V15 onwards
- with the option Ethernet programming interface (Ethernet PG)

In this supplementary manual only the Ethernet programming interface is described. Higher-level or general information → separate basic instructions of the device manual.

1.3 How is this manual structured?

This manual is a combination of different instruction types. It is for beginners and also a reference for advanced users.

How to use this manual:

- To find a certain subject straight away, please use the table of contents at the beginning of this manual.
- You can also find a requested term quickly with the index at the end of the manual.
- At the beginning of a chapter we will give you a brief overview of its contents.

Headers	You can find the title of the current chapter in bold in the header of each page. Below is the current title of the second order.
Footers	You can find the chapter-related number of the page in the footer of each page. Example: 12-7 means page 7 in chapter 12.

Abbreviations and technical terms → page 7-1, (chapter Terms, abbreviations at the end of the manual.

We reserve the right to make alterations which can result in a change of contents of the manual. You can find the current version on **ifm's** website at:
→ www.ifm.com > Select country/language > [Service] > [Download] > [Bus system AS-Interface]

Nobody is perfect. Send us your suggestions for improvements to this manual and you will receive a little gift from us to thank you.

1.4 Overview: where is what?

figure: overview of controller

2 Safety instructions

In this chapter you can find general safety instructions such as:

- General rules
- Required previous knowledge
- Safety instructions for mounting and installation
- When are you allowed to use this device and when not?

2.1 General

→ separate basic instructions of the device manual

No characteristics are warranted with the information, notes and examples provided in this manual. The drawings, representations and examples imply no responsibility for the system and no application-specific particularities.

The manufacturer of the machine/equipment is responsible for the safety of the machine/equipment.

⚠ WARNING

Property damage or bodily injury when the notes in this manual are not adhered to!

ifm electronic assumes no liability for this.

► The acting person must have read and understood the safety instructions and the corresponding chapters of this manual before performing any work on or with this device.

► The acting person must be authorised to work on the machine/equipment.

2.2 What previous knowledge is required?

These instructions are for persons with knowledge and previous knowledge of control technology and PLC programming with IEC 61131-3 as well as the CoDeSys® software.

The manual is intended for persons authorised to mount, connect and set up the controllere according to the EMC and low voltage directives. The controllers must be installed and put into operation by a qualified electrician.

In case of malfunctions or uncertainties please contact the manufacturer at: → back of the instructions.

2.3 Functions and features

→ separate basic instructions of the device manual

3 System requirements

3.1 Information concerning the device

→ separate basic instructions of the device manual

This manual describes the AS-i controller device family from ifm electronic gmbh with the option Ethernet TCP/IP interface.

3.2 Information concerning the software

→ separate basic instructions of the device manual

3.3 Required accessories

Basic functions → separate basic instructions of the device manual

For configuration and programming you also need:

- the software "CoDeSys for Automation Alliance™" version 2.3 or higher
  → CD article no. AC0340
- in case of direct connection of the controller to a PC with Ethernet interface (LAN):
  a *cross-over* CAT5 Ethernet patch cable with an RJ45 connector on both sides:
  2 m article no. EC2080
  5 m article no. E30112
- in case of connection of the controller to a PC with Ethernet interface (LAN) via a hub or switch:
  a common CAT5 Ethernet patch cable with an RJ45 connector on both sides
- in case of direct connection of the controller to a PC with serial interface:
  programming cable article no. E70320

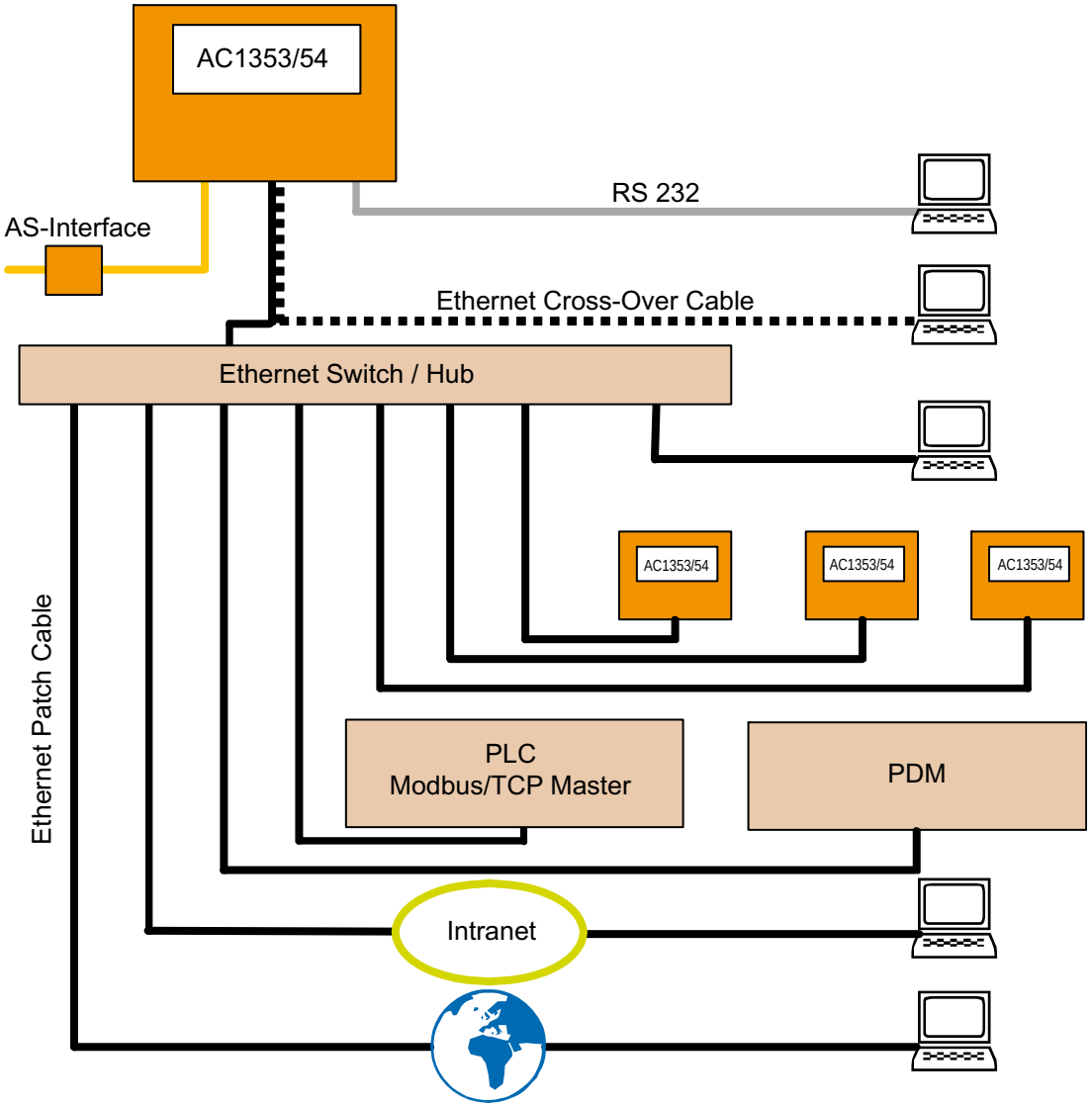
4 Function

Basic functions → separate basic instructions of the device manual

Fieldbus interface (option) → separate supplementary device manual

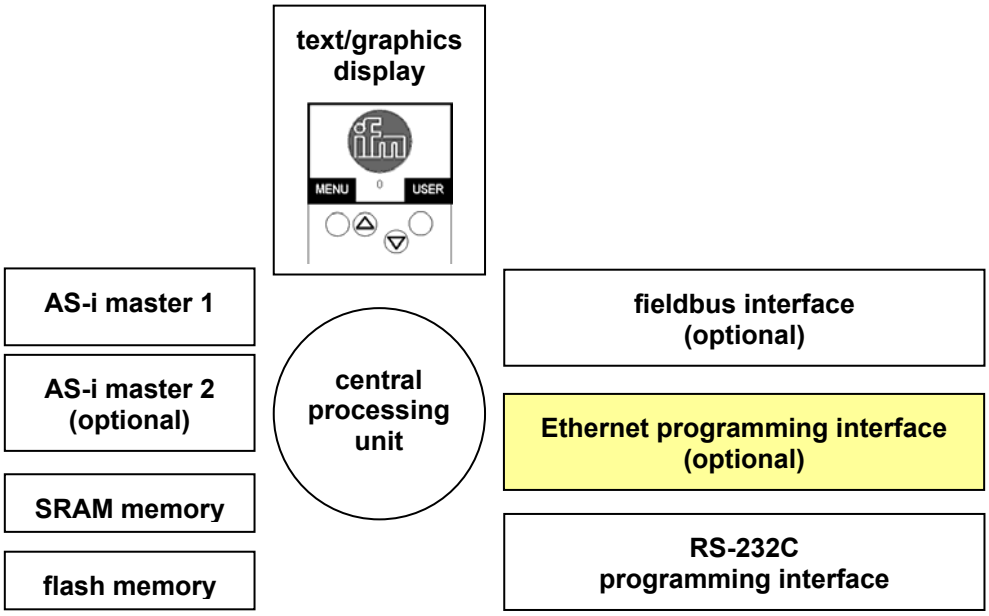
4.1 Overview

- The programming Ethernet interface of the controllere can be used for project and data transmission.
  - from the PC to the controllere, as well as
  - from the controllere to the PC.
- In the network one or several PCs as well as one or several controllere devices can communicate.
- controllere devices of this type further contain a MODBUS/TCP server which allows data exchange with a MODBUS/TCP client.
- The controllere can be connected via intranet or internet (→ following figure). The required information for all above-mentioned transmissions can be found in these instructions.



4.2 Data management

The controllere consists of different devices:



This manual exclusively describes the following subject:

- With the optional **Ethernet programming interface**, (10/100 MBd, twisted pair), the device can, in addition to even faster programming and diagnosis, also be networked to other controllere devices.

4.3 Which operating modes are there for the PLC in the controllere?

Operating mode	Meaning	Behaviour at Modbus / fieldbus
Run	SPS program start > The PLC program stored in the controllere is processed. > LED [PLC RUN] lights	At Modbus AS-i Slaves in the controllere application program can be written: Mapping of the PLC address ranges %IB4.512…%IB4.639 %IW4.320…%IW4.639
Stop	SPS program stop > The PLC program stored in the controllere is stopped. > LED [PLC RUN] flashes	
Gateway	Controllere as gateway > The PLC program stored in the controllere is not processed. > LED [PLC RUN] goes out	Only for devices with the option fieldbus connection: The fieldbus has exclusive write access on the AS-i outputs. Device with fieldbus: Modbus has no access here! Device without fieldbus: Modbus has write access on the AS-i outputs. The timeouts for the analogue and digital AS-i outputs only run in the operation mode Gateway. For the other data areas which are be written via Modbus there is no timeout monitoring.

Note

During changes to the PLC program or to the slaves the PLC program should be stopped to avoid malfunctions.

Note

In devices with Profibus and Ethernet programming interface Modbus is not be used as fieldbus but as interface for operation and configuration.

4.4 AS-Interface as well as project transmission and diagnosis via RS232

Also the controllere devices of type AC1353/54 contain one ore more AS-Interface masters. An RS232 programming interface is also available.

AS-Interface system → separate basic instructions of the device manual
Project transmission and diagnosis via RS232 interface → separate basic instructions of the device manual

4.5 Project transmission and diagnosis via Ethernet interface

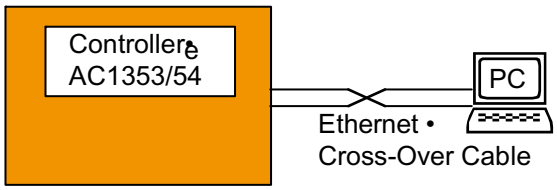
This section describes the project transmission and diagnosis (AS-i networks and projects) via a simple structure (PC - controllere with a point-to-point connection via Ethernet) as well as in an Ethernet network.

4.5.1 Point-to-point connection

Connection between		via	→ page
controllere	PC	point-to-point connection	here
controllere	controllere	network connection	4-18
controllere	client	MODBUS/TCP server / client	4-33
controllere HTML page	PC	HTML data exchange	4-74

Overview point-to-point connection

A simple point-to-point connection is to be implemented (→ figure):



For this, the following steps are required:

Step 1 ► Connect the PC to the controllere by means of a cross-over cable (→ page 4-5)

Step 2 ► Set IP addresses and subnet mask in the controllere and the PC (→ page 4-5)

Step 3 ► Select the target system and write the project (→ page 4-9)

Step 4 ► Set the communication parameters (→ page 4-11)

Step 5 ► Transmit and start the project (→ page 4-13)

Step 6 ► Set-up, monitoring and diagnosis of the AS-i system (→ page 4-14)

Step 7 ► Create the boot project (→ page 4-15)
Transmit and save the source code from the PC to the controllere

Step 8 ► Transmit the source code from the controllere to the PC
(service case, → page 4-16)

Here is the detailed description of the steps:

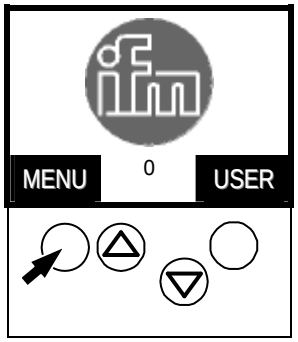
Step 1: Connect the PC to the controller

► Connect the LAN connection of the PC to the controller.

► To do so, use a cross-over CAT5 Ethernet patch cable with an RJ45 connector on both sides,
e.g.: 2 m article no. EC2080
5 m article no. E30112
A cross-over cable only enables a point-to-point connection. It cannot be used to establish a connection to a network.

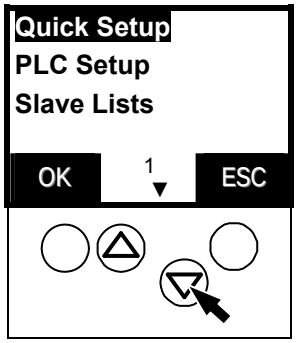
Step 2: Set IP addresses and subnet mask

1.



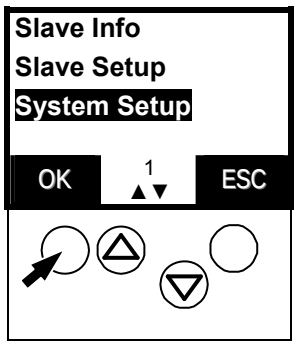
► Press [MENU]

2.



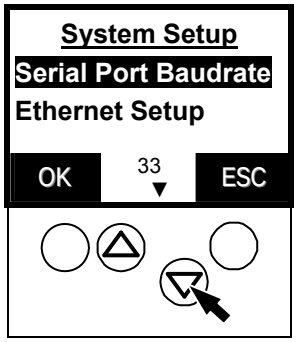
► Use [▼] to scroll to [System Setup]

3.



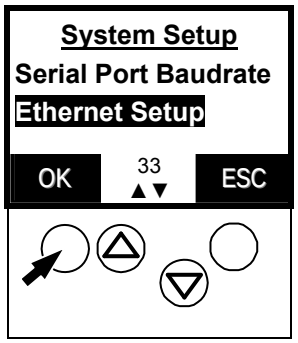
► Select [System Setup] with [OK]

4.



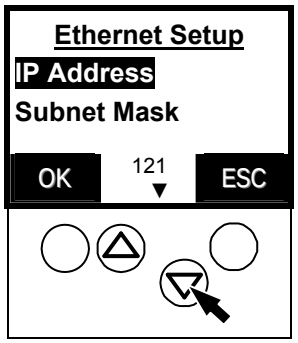
► Use [▼] to scroll to [Ethernet Setup]

5.



► Select [Ethernet Setup] with [OK]

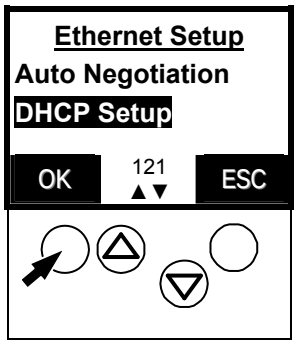
6.



To set the IP address, DHCP must be switched off.

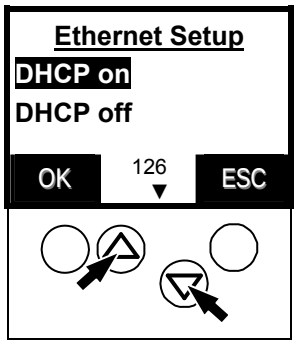
► Use [▼] to scroll to [DHCP Setup]

7.



► Select [DHCP Setup] with [OK]

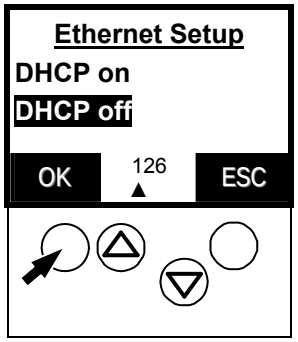
8.



> Display of the current DHCP selection

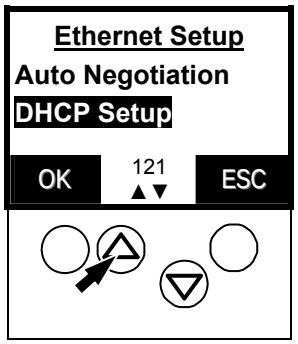
► Use [▼]/[▲] to scroll to [DHCP OFF]

9.



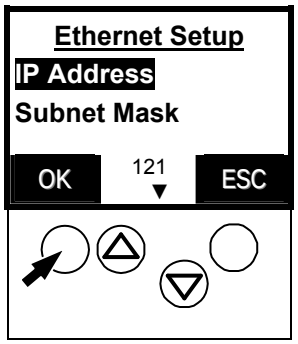
► Select [DHCP OFF] with [OK]
> "Wait!" is displayed

10.



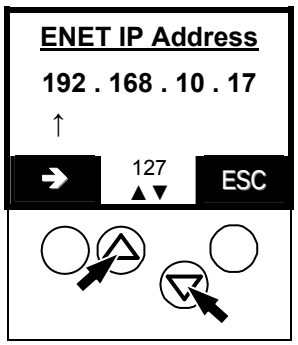
> Display of the menu [Ethernet Setup]
► Use [▲] to scroll to [IP-Adresse]

11.



► Select [IP Adress] with [OK]

12.

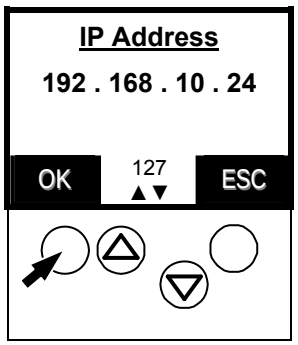


> Display of the current IP address
The arrow indicates the editable address group
► Use [▲] or [▼] to scroll to the requested address
► Use [➔] to go to the next address group
► Use [▲] or [▼] to scroll to the requested address

After the last address group:

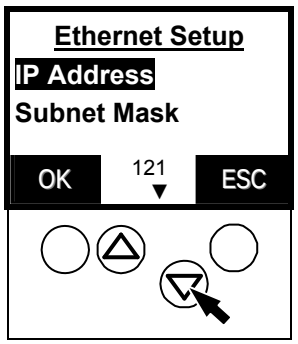
► Use [➔] to exit the editing mode

13.



> Display of new IP address
► Confirm the new IP address with [OK]
> "Wait!" is displayed
► (Use [ESC] to exit without making a change)

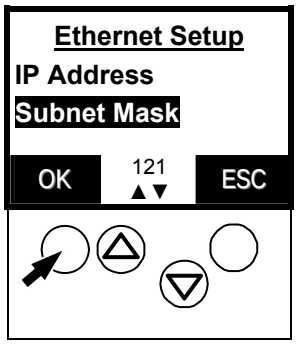
14.



> Display of the menu [Ethernet Setup]

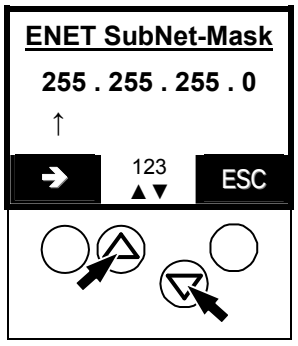
► Use [▼] to scroll to [Subnet Mask]

15.



► Select [Subnet Mask] with [OK]

16.



> Display of the current subnet mask

► Use [▲] or [▼] to scroll to the requested address

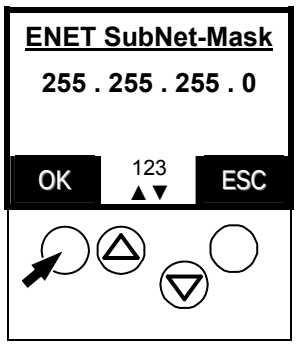
► Use [➔] to go to the next address group

► Use [▲] or [▼] to scroll to the requested address

After the last address group:

► Use [➔] to exit the editing mode

17.



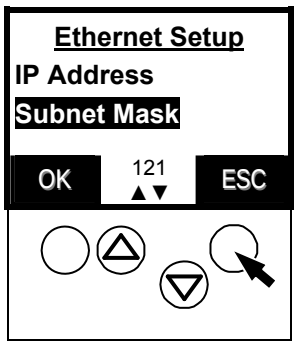
> Display of the current subnet mask

► Confirm the new subnet mask with [OK]

> "Wait!" is displayed

► (Use [ESC] to exit without making a change)

18.



► Use [ESC] 4 times to return to the start screen

In the Windows operating system the setting of the PC addresses is carried out correspondingly. In our example the settings of the PC are 192.168.10.20 as IP address and 255.255.255.0 as subnet mask.

NOTE

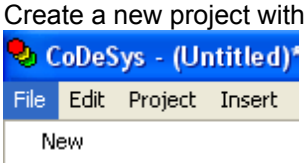
In a local network the participants can only communicate if their IP addresses are from the same "family".

Example: subnet mask = 255.255.255.**0**
Then the IP addresses of the first 3 address groups (where "255" is) must be identical for all participants. The IP address may (and must) only be different in the last block (where "0" is) (permitted values): 0…254).
Here: IP address in the controllere = 192.168.10.**24**, IP address in the PC = 192.168.10.**20**

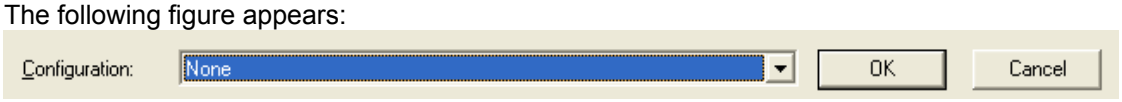
Step 3: Select the target system and write the project

► Start CoDeSys (version 2.3.5.0 or higher) on the PC

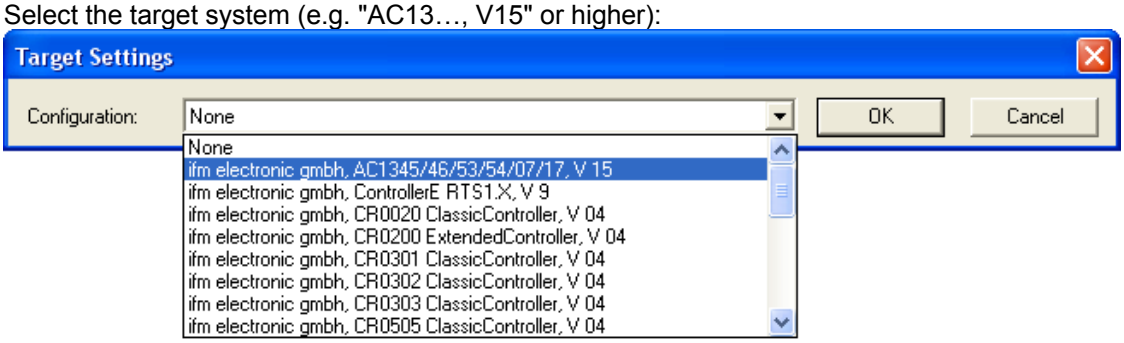
► Create a new project with [File] > [New]:



> The following figure appears:

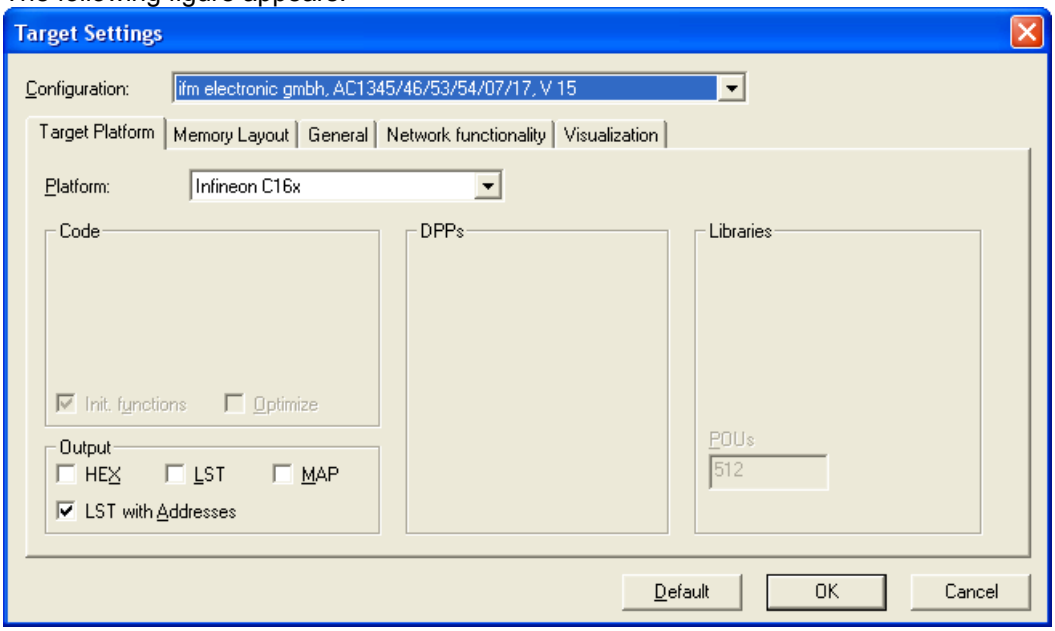


► Select the target system (e.g. "AC13…, V15" or higher):



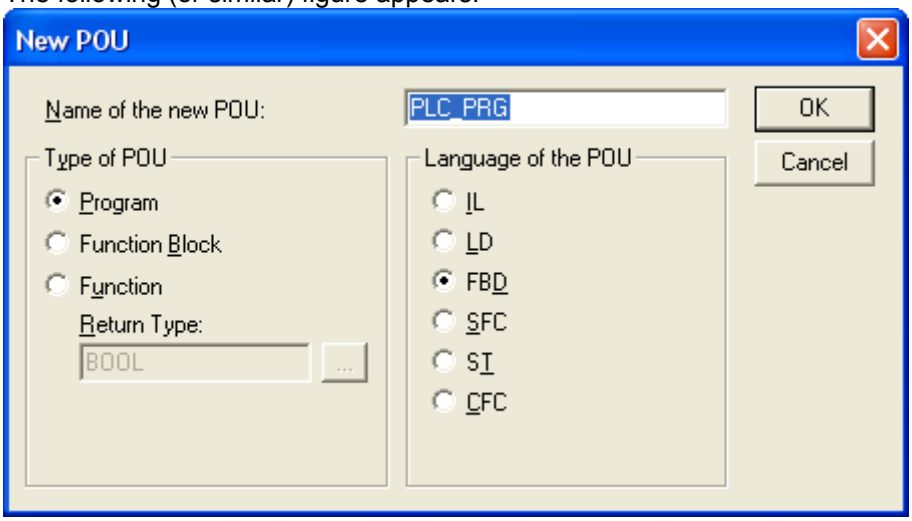
► Confirm with [OK]

> The following figure appears:



► Confirm with [OK]

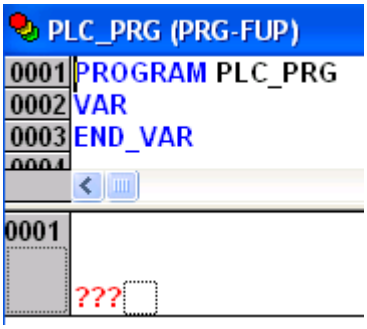
> The following (or similar) figure appears:



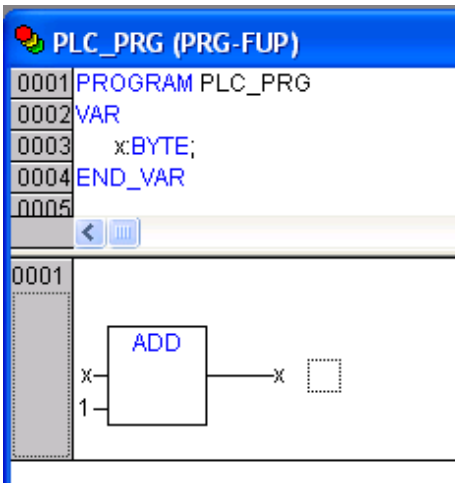
► Create the first POU. To do so, adopt the entries from the figure (→ above).

► Confirm with [OK]

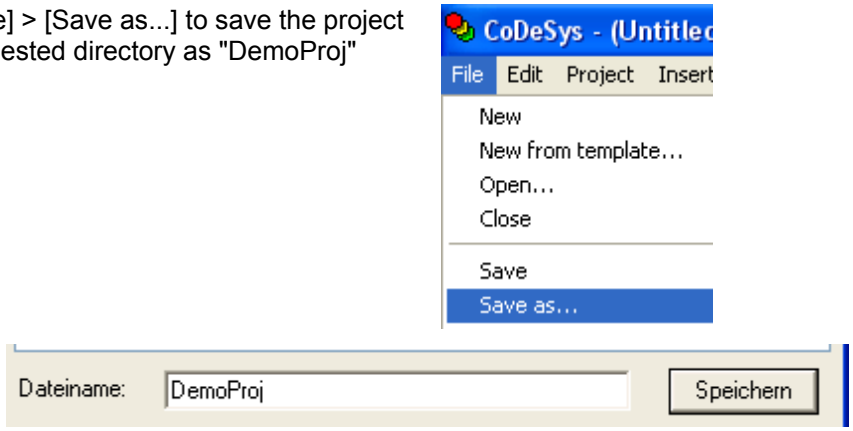
> The following figure appears:



► Complement your POU PLC_PRG by the entries as in the figure to the right:



► Select [File] > [Save as...] to save the project in the requested directory as "DemoProj" (→ right):

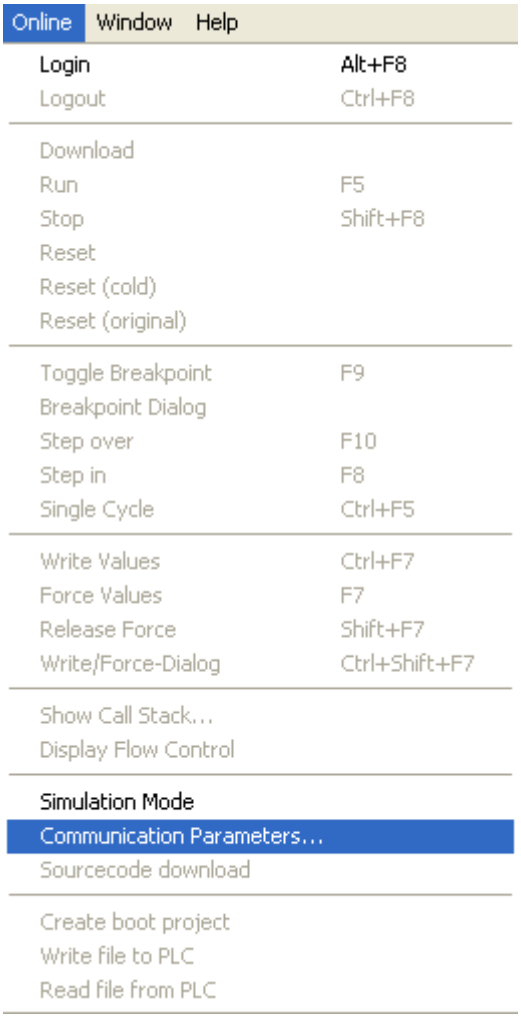


Step 4: Set the communication parameters

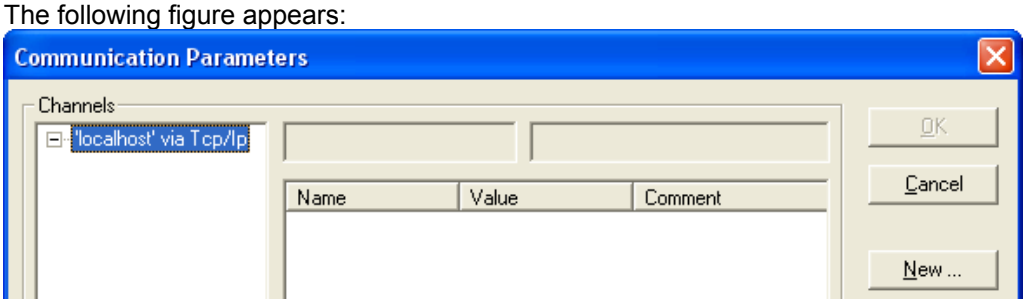
The cable alone does not enable the communication between the controllere and the PC. The same communication parameters must be set for both devices and the project.

Note: the set communication parameters of the project are saved together with the project and are therefore part of the project.

► Select [Online] > [Communication Parameters...] to call the following dialogue:

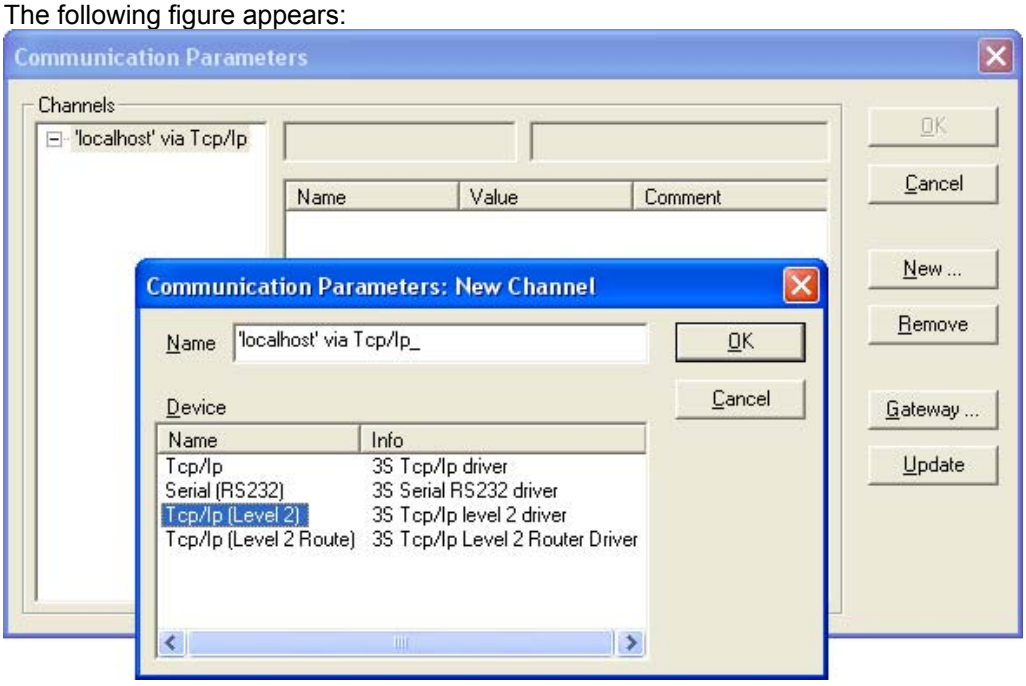


> The following figure appears:



► Click [New…]
Enter the parameters in the following dialogue window as shown in the window below

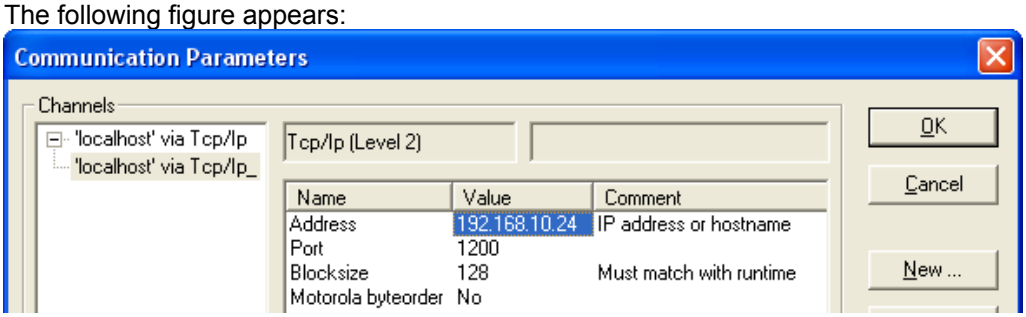
> The following figure appears:



► Select the entry [Tcp/Ip(Level 2)]

► Confirm with [OK]

> The following figure appears:



► Activate the address field by double-clicking

► Enter the corresponding IP address of the controller (see step 2)

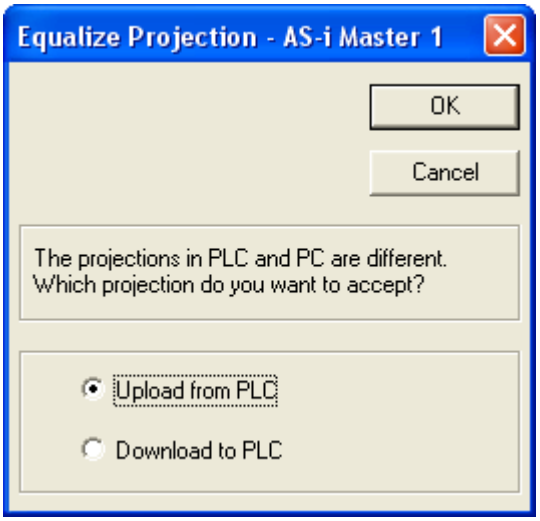
► Use [ENTER] to exit the editing mode

► Confirm with [OK]

Step 5: Transmit and start the project

► Click [Online] > [Login] to activate the connection from the PC to the controller:

> The following message appears:



Reason: in the project which is saved on the PC details have not yet been defined.

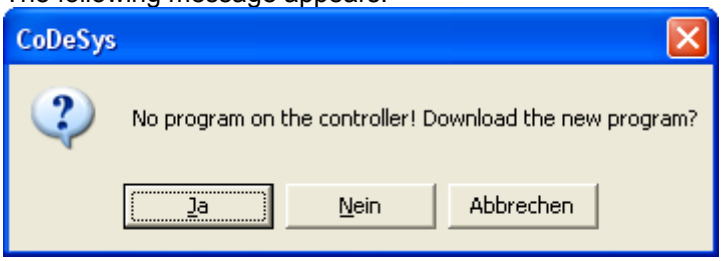
► Adopt the setting [Upload from PLC] (→ figure)

► Confirm with [OK]

Now there are two possibilities to proceed:

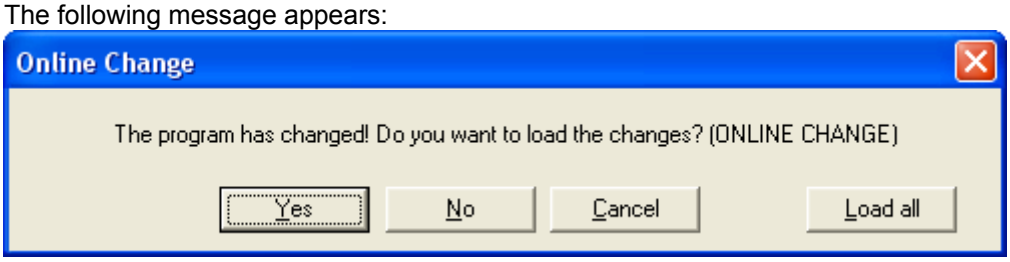
a) no program has yet been saved in the controller

> The following message appears:



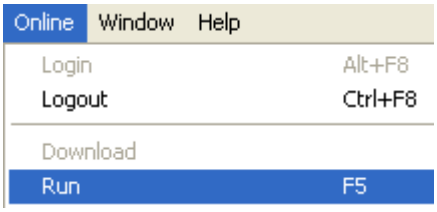
b) a program has already been saved in the controller

> The following message appears:



► In both cases you send your project to the controller by confirming with [yes].

► Click [Online] > [Run] to start the project:



> The project in the controller starts

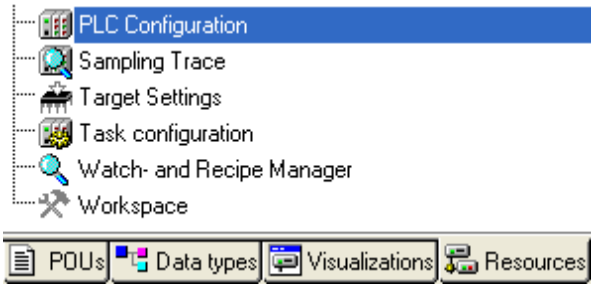
The project can then be tested.

Step 6: Set-up, monitoring and diagnosis of the AS-i system

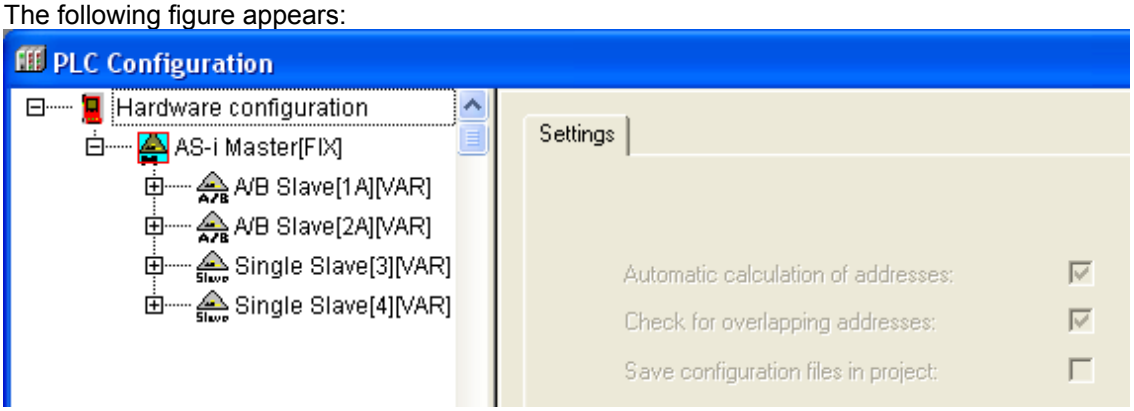
First use the PLC configuration window:

► Click the tab [Resources] (below)

► Click the option [PLC configuration]



> The following figure appears:

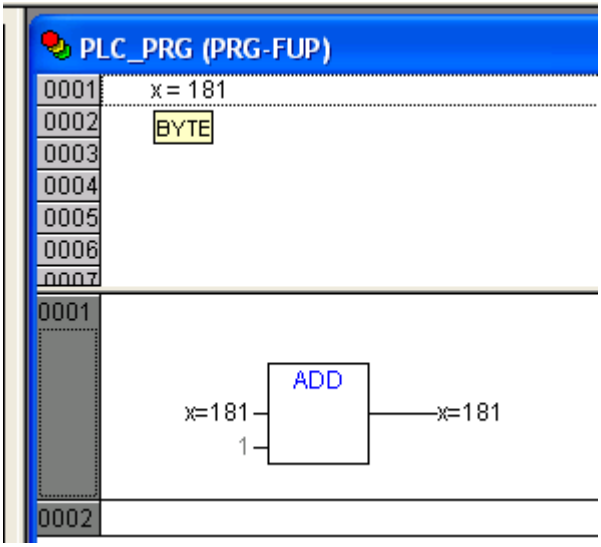


In step 5 you copied the configuration data from the controller to your PC into your project. This is why the data shown in the figure above is already available.

Detailed description of the PLC configuration → separate basic instructions of the device manual

In the controller you can observe the processing of your project and debug the program if necessary.

> Here:
Display of the POU PLC_PRG with the example of the adder.



Step 7: Create the boot project and save the source code

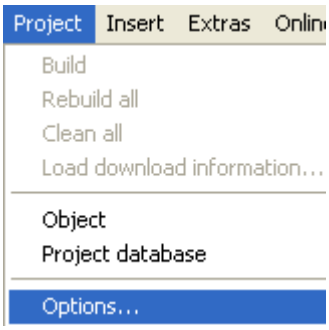
When the controller is switched off the device forgets all setting parameters. In the controller you can non-volatilely save a boot project which loads all current settings when the device is switched on. At the latest when completing the project you must create a boot project to non-volatilely save the project in the controller.

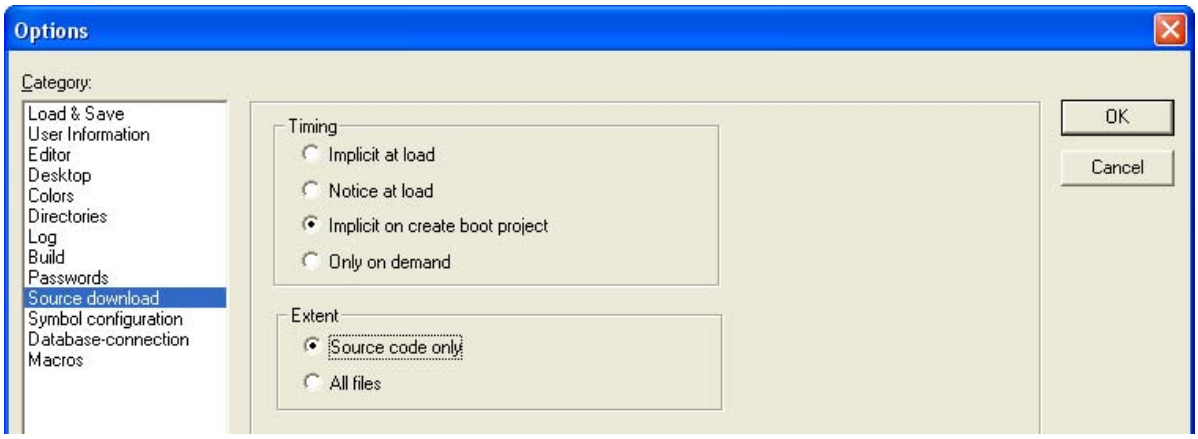
► Menu [Online] > [Create boot project]

> The boot project is saved in the controller

The **source code** does not only contain the program code of the project but also all comments and symbolic parameter names saved in the project. This allows a service technician later to copy the current program with all information to his PC and to edit the project.

► Click the menu points [Project] > [Options] > [Source download] to save the source code in the controller





► Use [OK] to confirm the settings

> When (again) creating the boot project the source code is transmitted to the controller

► Menu [Online] > [Create boot project]

> The boot project is saved in the controller

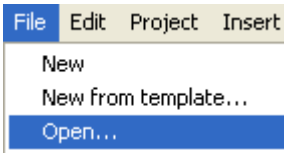
Step 8: Transmit the source code from the controller$_e$ to the PC (service case)

Your project was transmitted as source code from the PC to the controller$_e$ and is available there (→ step 7).

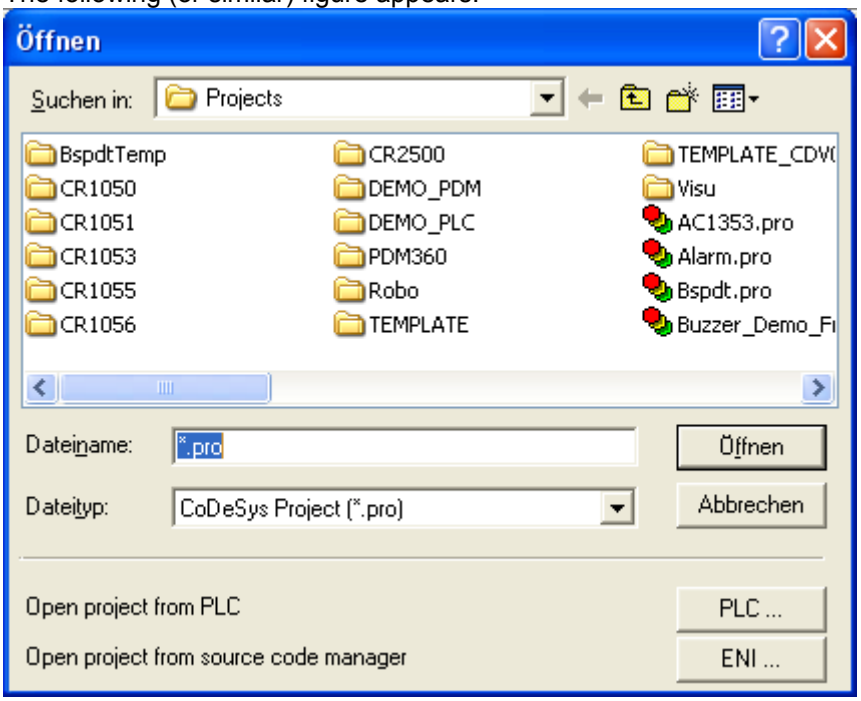
► Save your current project and close it.

Imagine you are a service technician who has to change the work flow of a machine. Probably your PC has not saved the current status of your project because other colleagues operated the machine in the meantime. For this reason you copy the project from the machine (controller$_e$) to your PC:

► Menu [File] > [Open…]:

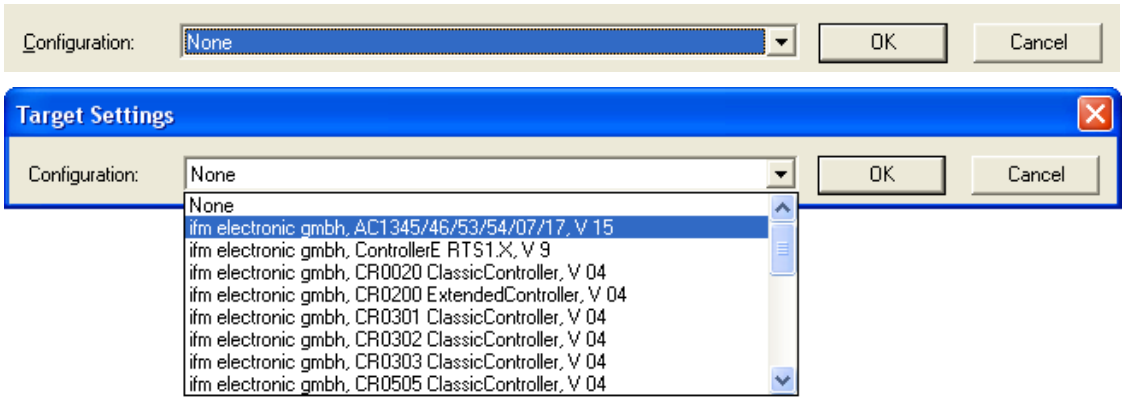


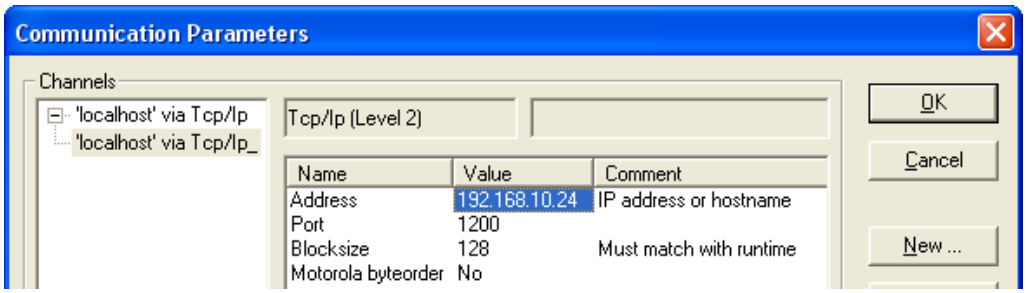
> The following (or similar) figure appears:



► Click [PLC…] (open the project from the controller)

► Carry out the steps as in the following figures (details → steps 3...6):





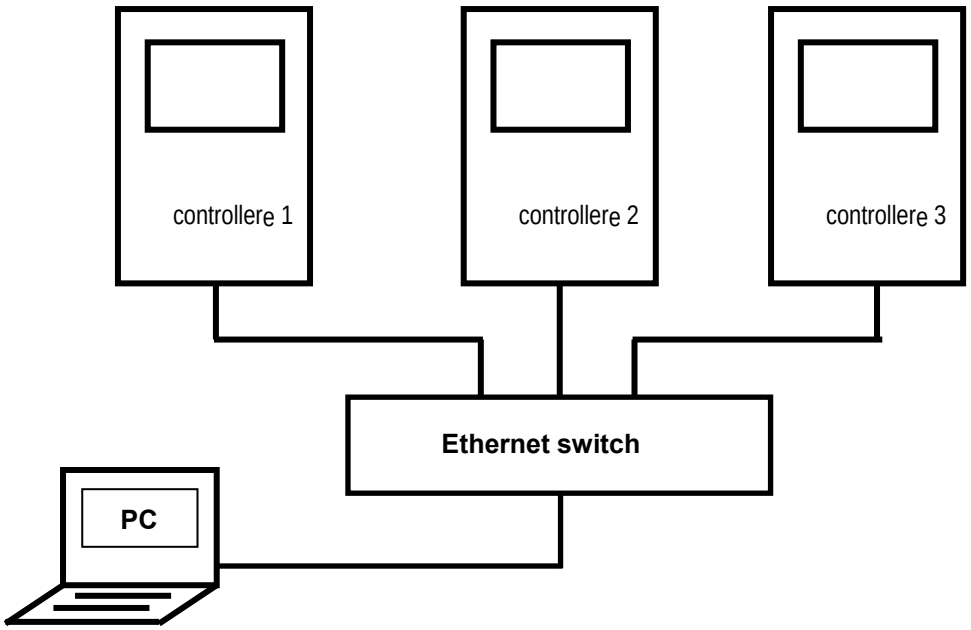
► Convince yourself that the copied project which you have transmitted from the controllere corresponds to your original project.

4.5.2 Ethernet network connection

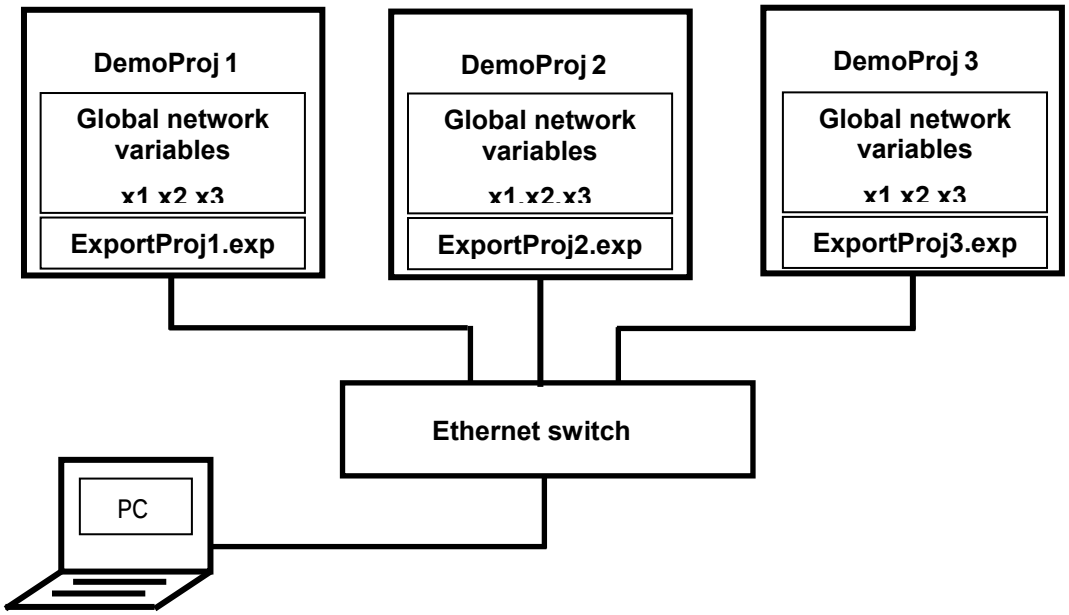
Connection between		via	→ page
controllere	PC	point-to-point connection	4-4
controllere	controllere	network connection	here
controllere	client	MODBUS/TCP server / client	4-33
controllere HTML page	PC	HTML data exchange	4-74

Overview Ethernet network connection

An Ethernet network connection is to be implemented (→ example below).



Between 3 controllere devices and 1 PC, information exchange shall take place. We implement this by means of the "global network variables" (→ illustration below). The examples only show the variables x1, x2 and x3 in the global network variable lists of the corresponding projects.



Excursion: Global network variables / EXP files

Global network variables are used for data exchange between controllers in the network. There is a difference between export and import variables:

- Export variables originate from the **local** project.
  Their values can be locally influenced.
  It shall also be possible to read and use the variables in one or more other projects. Therefore theses variables must be **exported** (made available) from the local project.
- Import variables do **not originate from the local project** but from another project.
  Their values cannot be locally influenced.
  The variables are read and used in the **local** project (and possibly also in several other projects). For this, these variables must be **imported** into the corresponding local project.

For exporting or importing, the global network variables must be summarised in lists. An EXP file is assigned to each global variable list.

In this example each project has 3 lists of global network variables:
- 1 with export variables (with local data for reading in other controllers) and
- 2 with import variables (with data of the two other controllers for local reading).

Please note the following when creating a global list of variables:

By ticking [Export before compile] in the window [Properties] CoDeSys updates the corresponding EXP file (e.g. `ExportProj1.exp`) when rebuilding the project.

By ticking [Import before compile] in the window [Properties], CoDeSys takes into account the corresponding EXP file (e.g. `ExportProj1.exp`) when rebuilding the project, and updates the list.

Example:

The project "DemoProj1" contains the global network variable list "Global_Variables_Export_Proj1".

► In the properties of this list you enter the file `ExportProj1.exp`.

► Activate [Export before compile].

> When rebuilding this project, `ExportProj1.exp` is updated.

The project "DemoProj2" contains the global network variable list "Global_Variables_Import_Proj1".

► In the properties of this list you enter the file `ExportProj1.exp`.

► Activate [Import before compile].

> When starting to rebuild this project, the list "Global_Variables_Import_Proj1" is updated by means of the file ExportProj1.exp and then used for the rebuild.

NOTE

If a project with variables to be exported is changed, all projects which import these global network variable lists must then be rebuilt to update the lists:

► Menu [Project] > [Rebuild all]

► Menu [Online] > [Create boot project]

> The boot project is saved in the controllere

Overview: Steps for implementing an Ethernet network connection

An Ethernet network connection is to be implemented (→ page 4-18). For this the following steps are required:

Step 1 ► Connect the devices via Ethernet (→ page 4-20)

Step 2 ► Set IP addresses and subnet mask (→ page 4-20)

Step 3 ► Select the first target system and create a project (→ page 4-21)

Step 4 ► Set the communication parameters (→ page 4-23)

Step 5 ► Activate the network variables support (→ page 4-25)

Step 6 ► Integrate libraries (→ page 4-26)

Step 7 ► Complete and transmit the project (global network variables) (→ page 4-27)

Step 8 ► Write projects for further controllere devices (→ page 4-30)

Step 9 ► Transmit projects (→ page 4-32)

Step 10 ► Test the transmission of global network variables (→ page 4-32)

Here is the detailed description of the steps:

Step 1: Connect the devices via Ethernet

► Set up the Ethernet network by connecting the PC and the controllere to the hub (or switch).

► To do so, use common CAT5 Ethernet patch cables with an RJ45 connector on both sides.

Step 2: Set IP addresses and subnet mask

► Set appropriate IP addresses and subnet masks on all three controllere devices (procedure → page 4-5).

NOTE

In a local network the participants can only communicate if their IP addresses are from the same "family".

Example: Subnet mask = 255.255.255.**0**
Then the IP addresses of the first 3 address groups (where "255" is) must be identical for all participants. The IP address may (and must) only be different in the last block (where "0" is) (permitted values): 0…254).

Ask the network administrator for the specifications!

In our example we assume the following values:
Subnet mask = 255.255.255.0
IP address of the controllere 1 = 192.168.10.21
IP address of the controllere 2 = 192.168.10.22
IP address of the controllere 3 = 192.168.10.23
IP address of the PC = 192.168.10.20

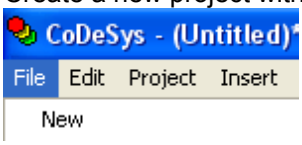
Step 3: Select the first target system and create a project

Three projects are to be written: one for each controllere in the network. The projects differ only slightly, the main differences concern the global variables and the executable part. Below please find a more detailed description of the projects.

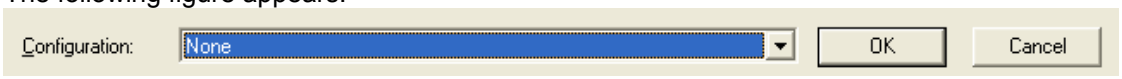
Let's start with the project for controllere 1:

► Start CoDeSys (version 2.3.5.0 or higher) on the PC

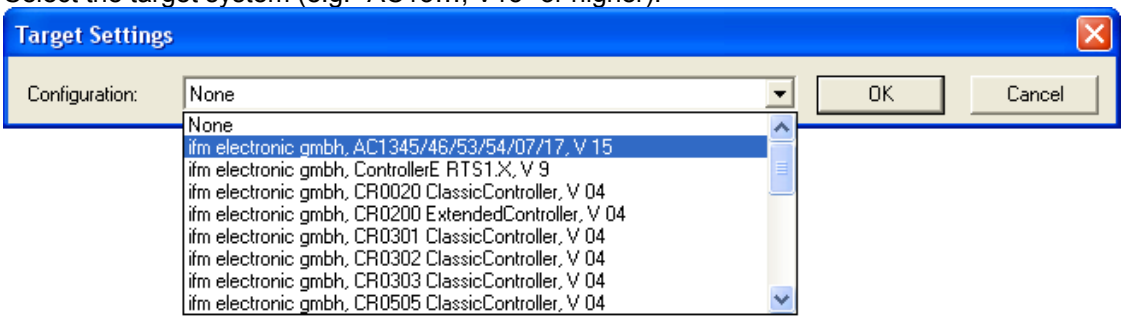
► Create a new project with [File] > [New]:



> The following figure appears:

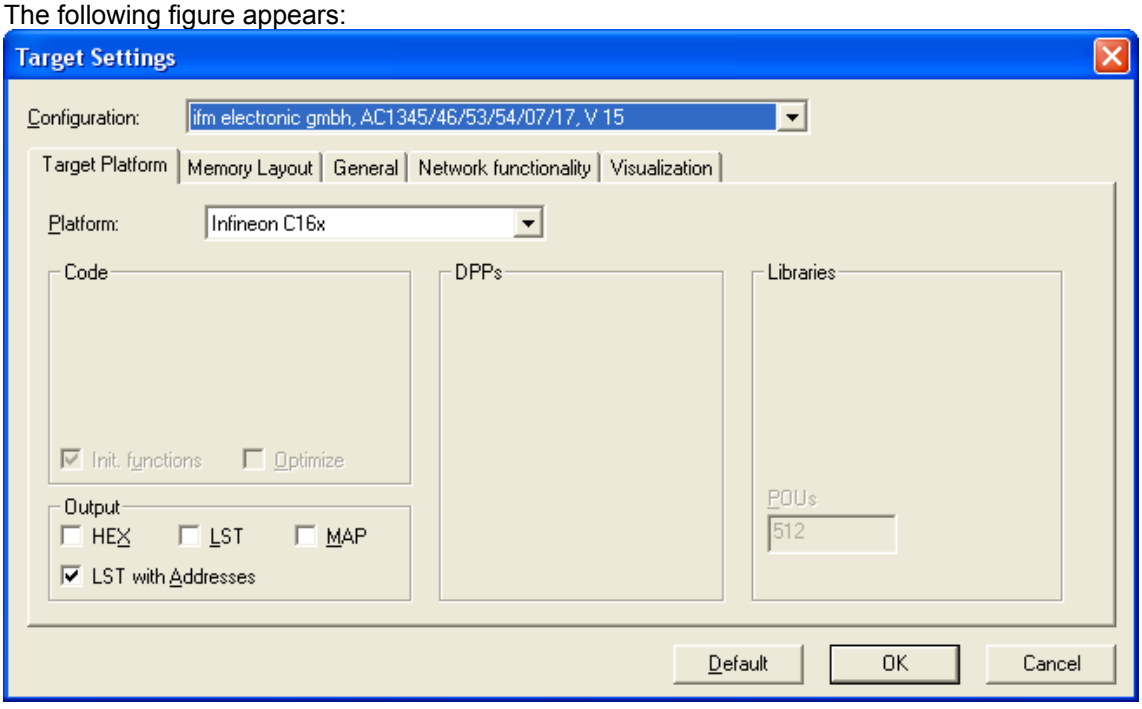


► Select the target system (e.g. "AC13…, V15" or higher):



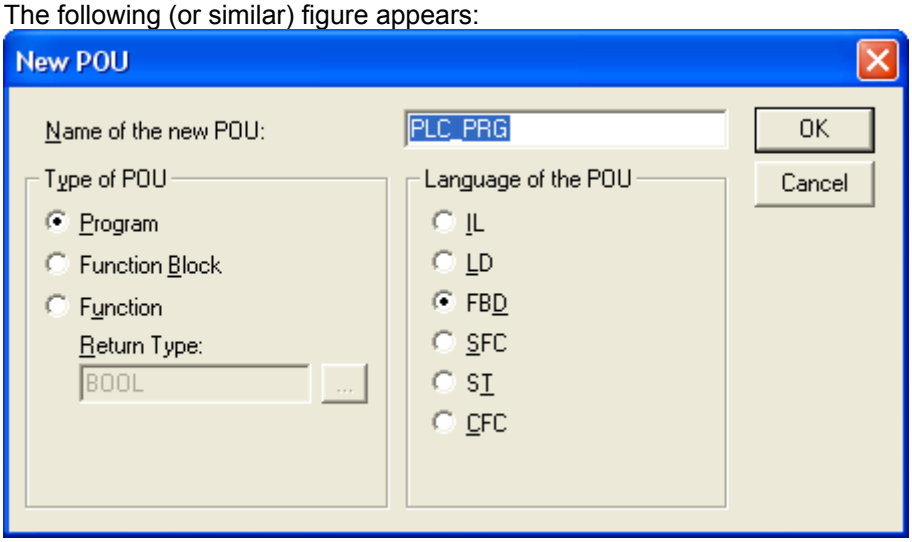
► Confirm with [OK]

> The following figure appears:



► Confirm with [OK]

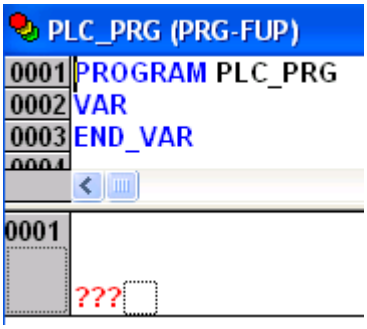
> The following (or similar) figure appears:



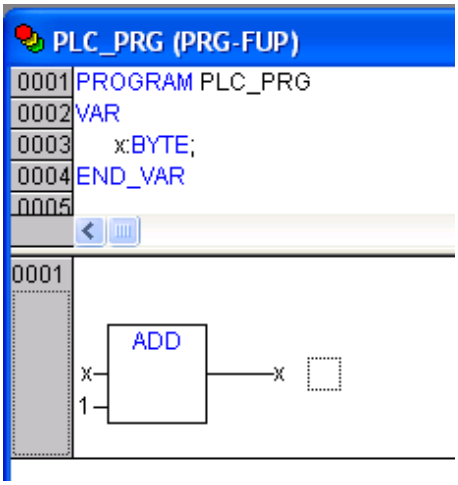
► Create the first POU. To do so, adopt the entries from the figure (→ above).

► Confirm with [OK]

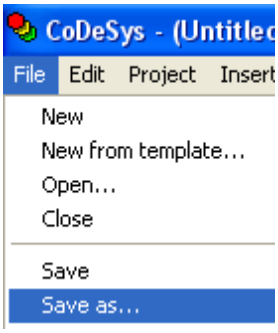
> The following figure appears:



► Complement your POU PLC_PRG by the entries as in the figure to the right:



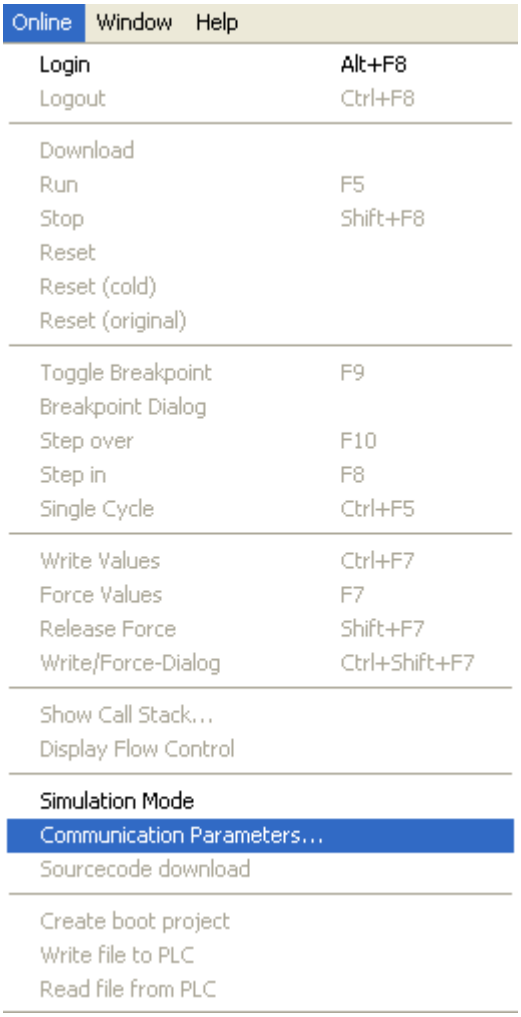
► Click [File] > [Save as...] to save the project in the requested directory.
Here:
File name = "DemoProj1" (for the first controllere in the network)



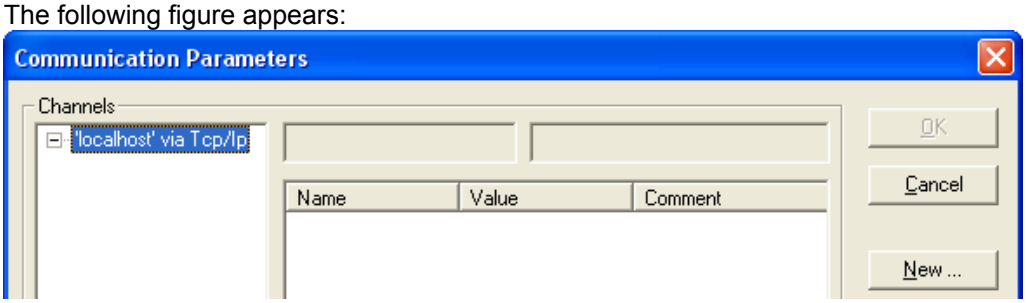
Step 4: Set the communication parameters

The same communication parameters must be set for the PC, the controllere and the project.

► Select [Online] > [Communication Parameters...] to call the following dialogue:

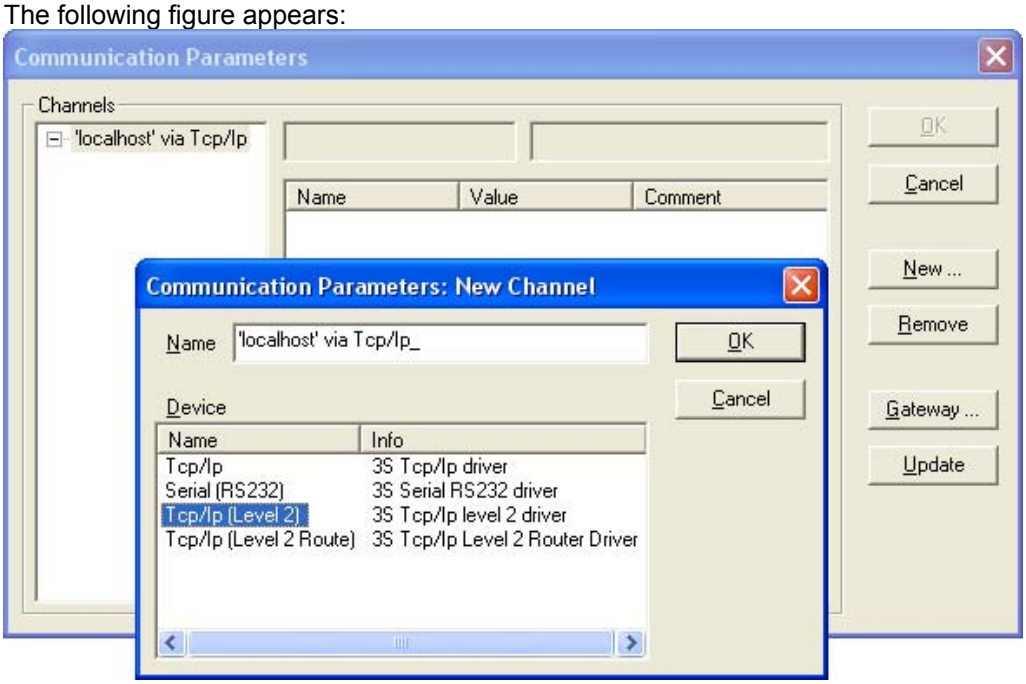


> The following figure appears:



► Click [New…]
Enter the parameters in the following dialogue window as shown in the window below

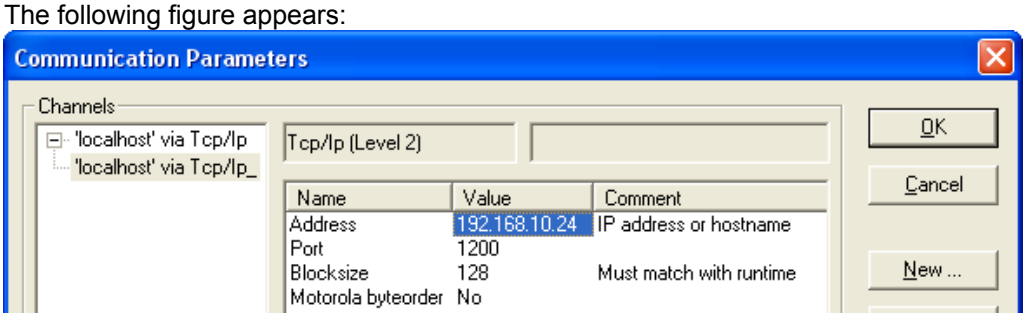
> The following figure appears:



► Select the entry [Tcp/Ip(Level 2)]

► Confirm with [OK]

> The following figure appears:



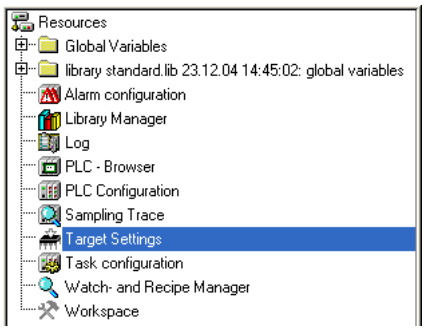
► Enter the corresponding IP address of the controller (see step 2)

► Confirm with [OK]

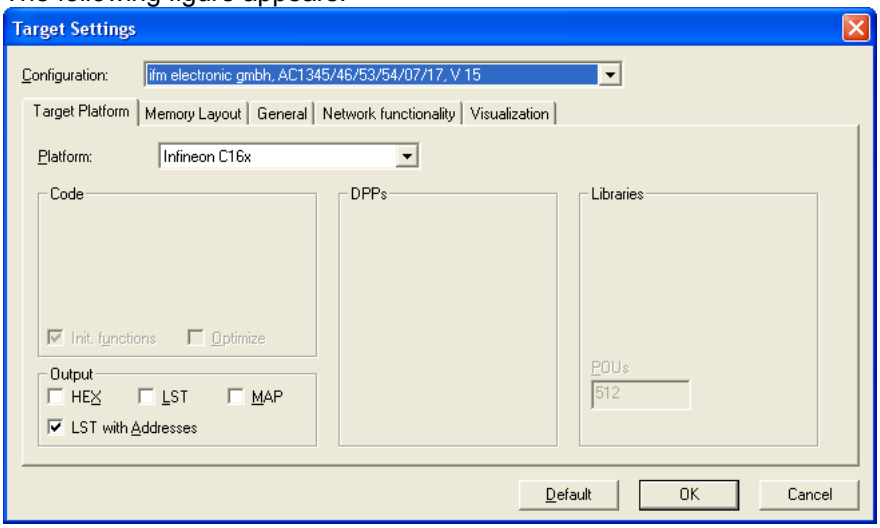
Step 5: Activate the network variable support

► Click the tab [Resources] in CoDeSys

► Double-click [Target Settings]



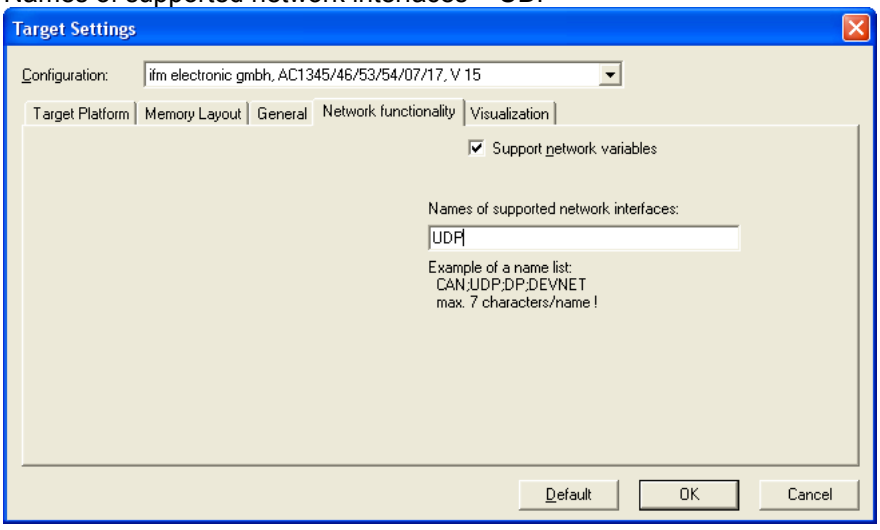
> The following figure appears:



► Double-click the tab [Network functionality]

► Activate the field [Support network variables]

► Names of supported network interfaces = UDP

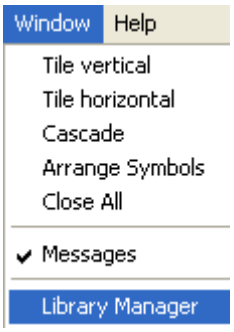


► Confirm with [OK]

> Data exchange via global network variables is now possible

Step 6: Integrate libraries

► Menu [Window] > [Library Manager]

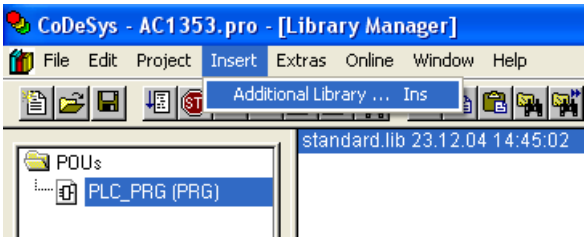


> Display of the libraries already loaded (here: only `standard.lib`)

► Menu [Insert] > [Additional Library... Ins]

or:

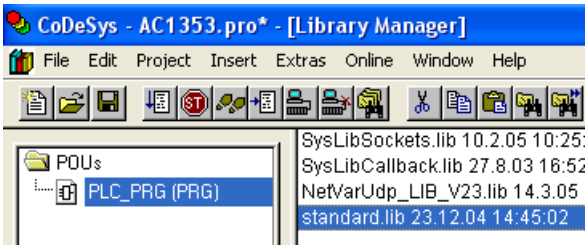
► Key [Ins]



► Insert the following libraries:

- `SysLibSockets.lib`
- `SysLibCallback.lib`
- `NetVarUdp_LIB_V23.lib`

> Now the library manager should look like this or similar (the order is not relevant):



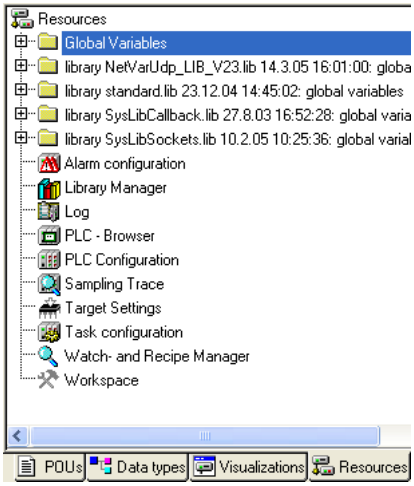
Step 7: Complete and transmit the project (Global network variables)

To demonstrate the data exchange via global network variables you now write a project for each of the three controllere devices. These projects are suitable for this purpose mainly due to the global variable lists which they contain.

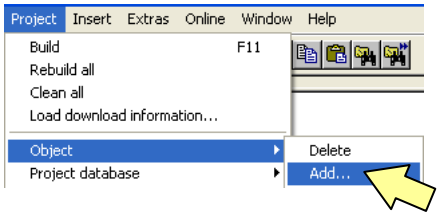
The example shows the project for controllere 1.

► Click the tab [Resources] in CoDeSys

► Click [Global Variables]
→ right figure



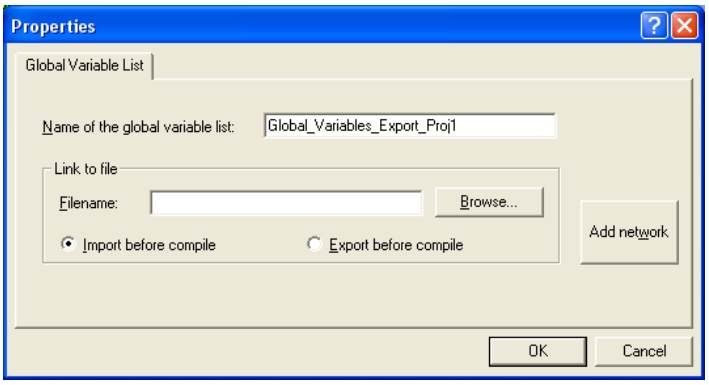
► Menu [Projekt] > [Object] > [Add…]



> Display of the window "Properties":

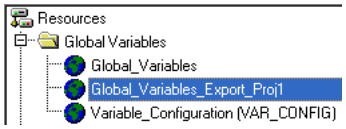
► Enter the name of the global variable list as shown

► Confirm with [OK]

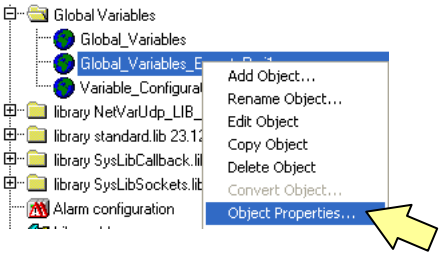


► Enter the variable x1 in the window that appears (→ figure)

► Right-click on the resources element [Global_Variables_Export_Proj1]

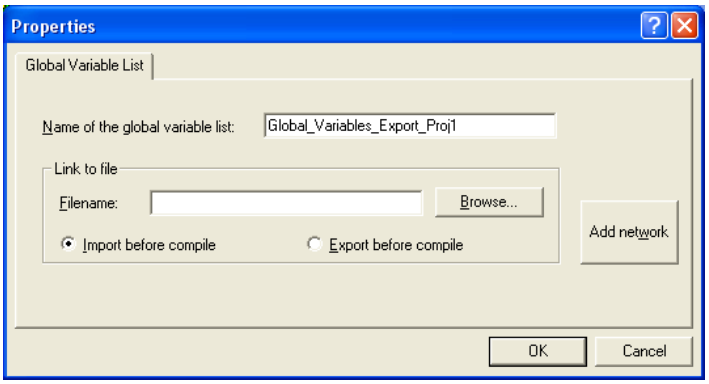


► Click [Object Properties...] (→ figure)



> Display of the window "Properties":

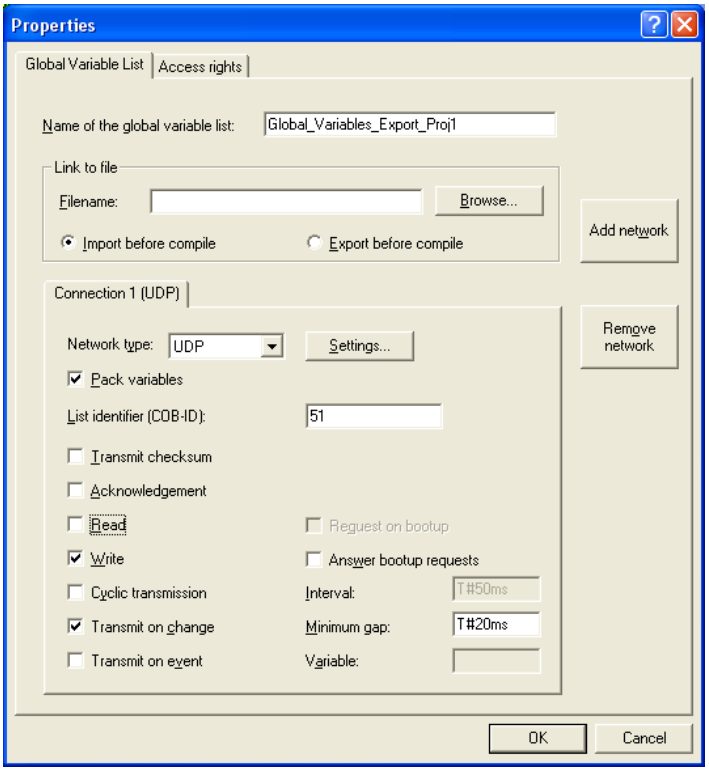
► Click [Add network]



> The following figure appears:

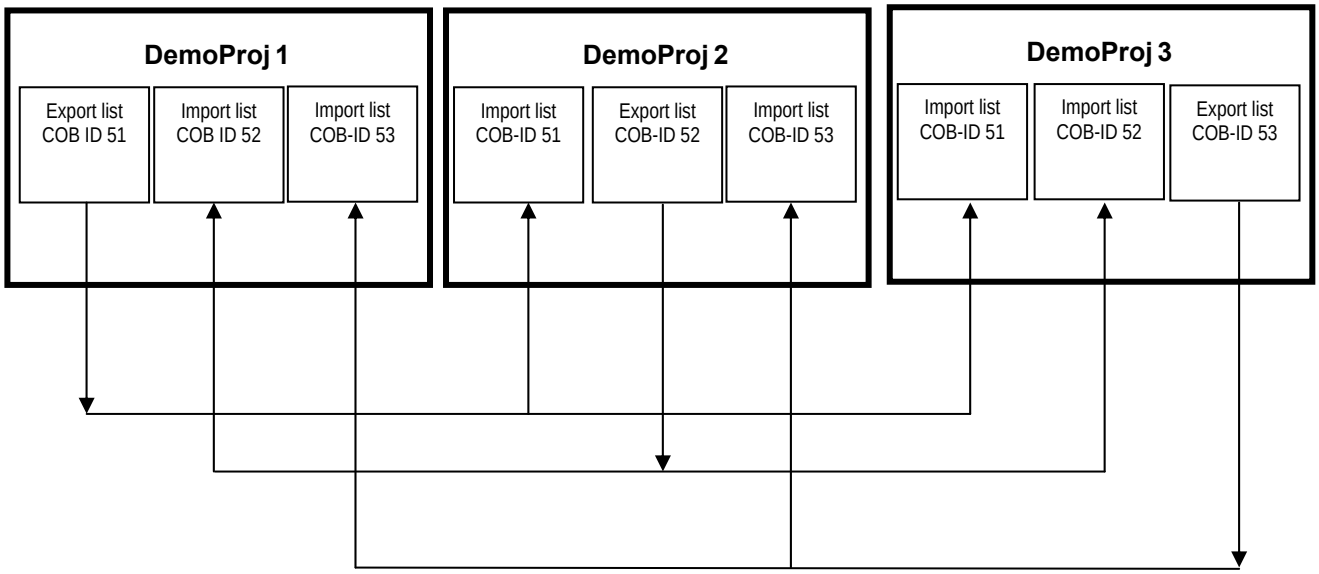
► Enter the properties of the list similarly as shown, but:

- The exact path of the link to file depends on the PC.
- The entered minimum gap depends on the application.
- For export: activate [Export before compile] and [Write]!



Excursion: variable list identifier (COB ID)

Here we operate with variable lists which are exported from one controller device and imported to one or more controller devices. This assignment of the variable lists is marked by a COB ID. This correlation is shown in the figure below.



Here it can be seen that the export variables of the DemoProj1 in controller1 have been assigned the COB ID=51 and that these variables can be found each in DemoProj2 and DemoProj3 in controller 2 and controller 3. We have used this schematic illustration for the definition of the COB IDs.

NOTE

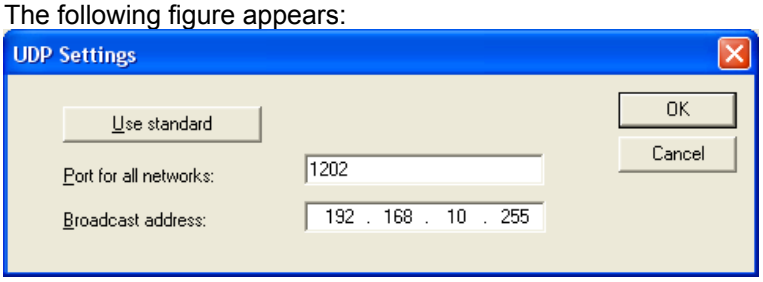
When selecting the COB IDs observe the following:

- The export list and its corresponding import lists must be assigned to the same COB ID.
- Each COB ID may be assigned only to one export list.

Continuation of step 7:

► In the window [Properties] of the global variable list:
Click [Settings...]

> The following figure appears:



► As broadcast address enter the IP address of controller1, but:
replace the value in the last group by "255" (→ figure).

► Confirm with [OK]

Step 8: Write projects for further controllere devices

► Create **now** corresponding projects with export files also for the two other controllere devices ("DemoProj2", "DemoProj3"). These files are required for the following steps. Data names and entries → following table:

controllere	File name of the project	Name of the global variable list	COB ID	File name of the link to file	Global variable
1	DemoProj1.pro	Global_Variables_Export_Proj1	51	\ExportProj1.exp	x1:INT
2	DemoProj2.pro	Global_Variables_Export_Proj2	52	\ExportProj2.exp	x2:INT
3	DemoProj3.pro	Global_Variables_Export_Proj3	53	\ExportProj3.exp	x3:INT

> For all three controllere devices the export lists have been created.

For the project "DemoProj1" in controller1 we have so far only created the "global network variables" which are to be **exported**.

Now we create two lists of "global network variables" which will be imported by "DemoProj1". Use the above-mentioned methods:

► Open the project "DemoProj1" in CoDeSys

► Click the tab [Resources]

► Tick (click) the entry [Global Variables]

► Menu [Project] > [Object] > [Add…]

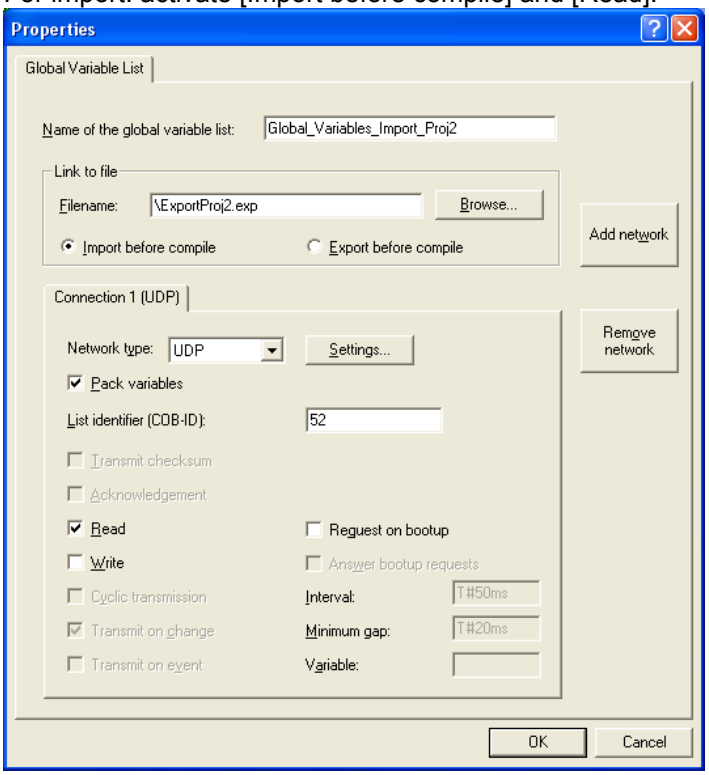
► Define the list „Global_Variables_Import_Proj2”

► Confirm with [OK]

► Enter the variable x2 in the window that appears (→ figure)

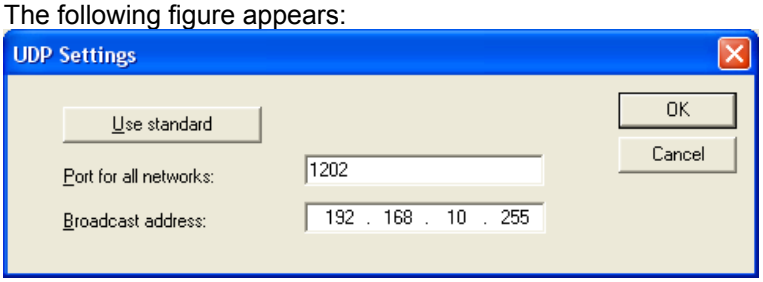
```
0001 VAR_GLOBAL
0002     x2:INT;
0003 END_VAR
0004
```

► Enter the properties and settings of this list (→ figure below)
For import: activate [Import before compile] and [Read]!



► Click [Settings...]

> The following figure appears:



► As broadcast address enter the IP address of controllere2, but:
replace the value in the last group by "255".

► Confirm with [OK]

The same operation is to be carried out for controllere3:

► Define the list "Global_Variables_Import_Proj3"

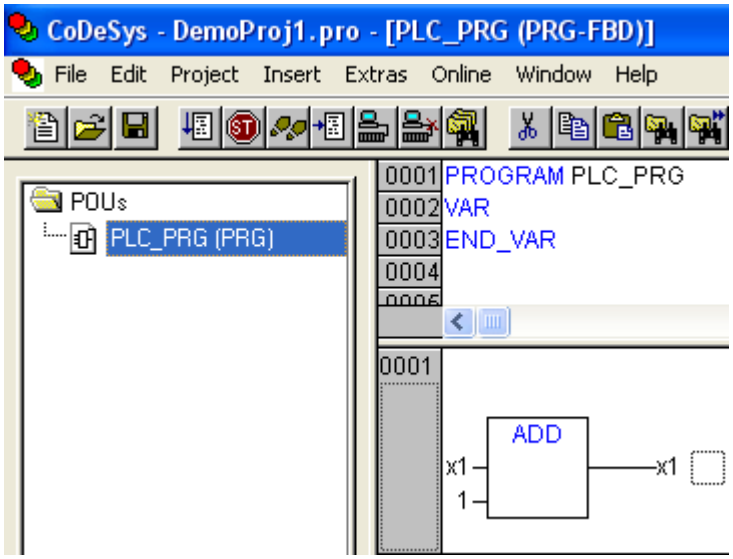
► Content of the variable list: "x3:INT"

► File name of the link to file = `\ExportProj3.exp`

► Correspondingly create the global import variable lists also for the two other projects.

So far we have "only" organised the data management. Now we will look at the executable parts of the projects.

For each project we will write only one POU which increases the contents of the corresponding variables (x1, x2 or x3) by 1 in each PLC cycle. In our example we show the POU PLC_PRG for `DemoProj1.pro` (→ figure).



► Create this program for each of these three controllere devices (`DemoProj1.pro, DemoProj2.pro and DemoProj3.pro`).

Step 9: Transmit projects

► Transmit the projects created in steps 6 + 7 to the corresponding controllere.

► Start these projects there.

Step 10: Test the global network variable transmission

► Check the behaviour of the data transmission by looking at the corresponding global network variable lists.
E.g. if you open the variables "Global_Variables_Import_Proj2" of DemoProj1, you should find that the value of x2 increases.

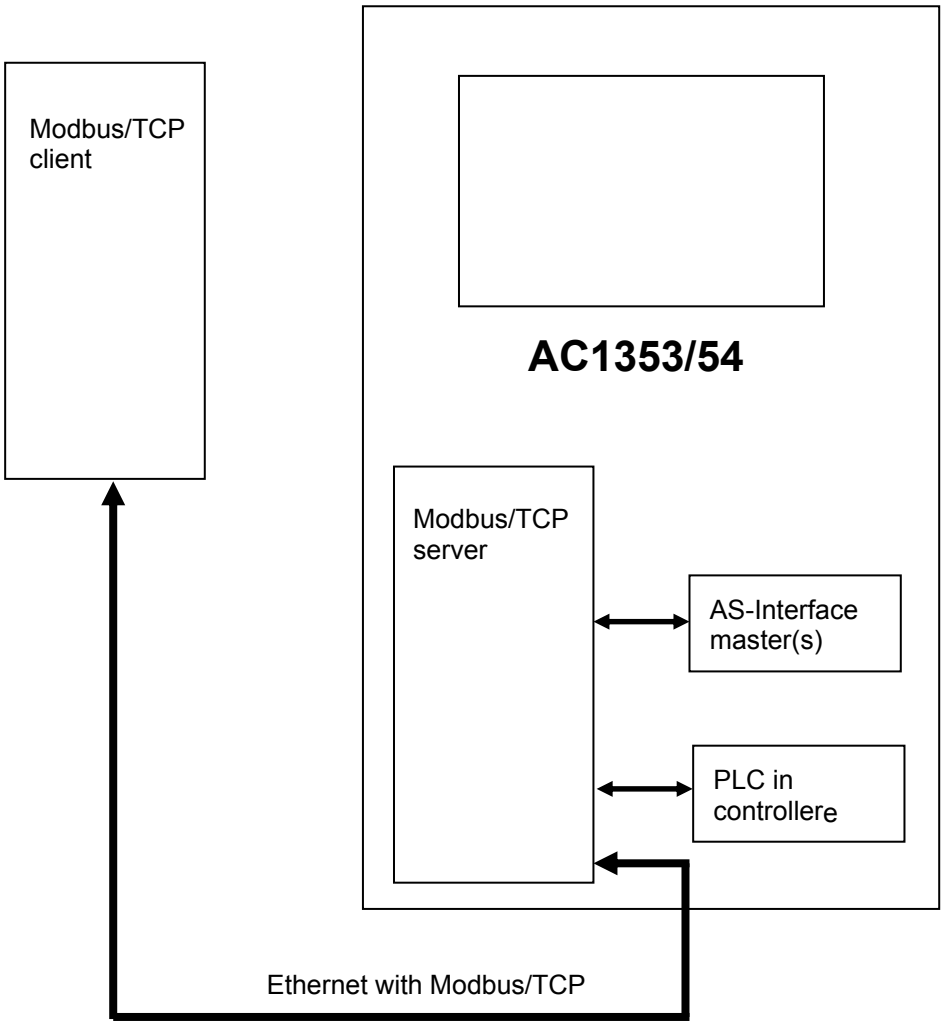
► Check the other projects and make sure that the transmission of the global network variables functions.

4.5.3 MODBUS/TCP server / client

Connection between		via	→ page
controllere	PC	point-to-point connection	4-4
controllere	controllere	network connection	4-18
controllere	client	MODBUS/TCP server / client	here
controllere HTML page	PC	HTML data exchange	4-74

Overview MODBUS/TCP server / client

The controllere and AC1353/AC1354 contain a MODBUS/TCP server which enables the data exchange with a MODBUS/TCP client. Schematic illustration → figure



The algorithm for the data exchange depends on what client is used. All clients require the addresses of the memory locations from which the data are taken from the server and in which the data are saved in the server. Here these addresses are indicated as Modbus addresses.

The client carries out the corresponding operation on these Modbus addresses to receive the desired result.

Valid Modbus addresses and their meaning

Modbus address [words]			Content	Access	Size [words]	IEC addresses	
start		end		r = read			
dec.	hex.	dec.		w = write		from	to
1024	400	1024	controller PLC status (→ page 4-36)	r	1	—	—
2048	800	2048	Modbus timeout (→ page 4-36)	r/w	1	—	—
2049	801	2049	Modbus write timeout (→ page 4-36)	r/w	1	—	—
2050	802	2050	delete the write timeout register (→ page 4-36)	r/w	1	—	—
Data AS-i master 1							
4096	1000	4127	digital slave inputs (→ page 4-37)	r	32	%IB1.1 %IB11.1	%IB1.31 %IB11.31
4128	1020	4129	master flags (→ page 4-40)	r	2	%IW31.240	%IW31.241
4130	1022	4284	analogue slave inputs (→ page 4-41)	r	155	%IW21.1.0	%IW21.31.4
4285	10BD	4348	current configuration data (→ page 4-53)	r	64	%IW31.0	%IW31.63
4349	10FD	4364	current parameters (→page 4-55)	r	16	%IW31.64	%IW31.79
4365	110D	4368	LAS (→ page 4-56)	r	4	%IW31.80	%IW31.83
4369	1111	4372	LDS (→ page 4-57)	r	4	%IW31.84	%IW31.87
4373	1115	4376	LPF (→ page 4-58)	r	4	%IW31.88	%IW31.91
4377	1119	4380	LPS (→ page 4-59)	r	4	%IW31.92	%IW31.95
4381	111D	4444	projected configuration data (→ page 4-53)	r	64	%IW31.96	%IW31.159
4445	115D	4460	reflected parameters (→ page 4-55)	r	16	%IW31.160	%IW31.175
4461	116D	4522	telegram error counter (→ page 4-60)	r	62	%IW31.176	%IW31.237
4523	11AB	4523	configuration error counter (→ page 4-62)	r	1	%IW31.238	—
4524	11AC	4524	AS-i cycle counter (→ page 4-62)	r	1	%IW31.239	—
4525	11AD	4556	digital slave outputs (→ page 4-37)	r/w	32	%QB1.1 %QB11.1	%QB1.31 %QB11.31
4557	11CD	4558	reserved	—	2	—	—
4559	11CF	4713	analogue slave outputs (→ page 4-41)	r/w	155	%QW21.1.0	%QW21.31.4
4714	126A	4777	reserved	—	64	—	—
4778	12AA	4793	reserved	—	16	—	—
4794	12BA	4812	host command channel request (→ page 4-63)	r/w	19	—	—
4813	12CD	4831	host command channel response (→ page 4-64)	r	19	—	—
Data AS-i master 2							
8192	2000	8223	digital slave inputs (→ page 4-37)	r	32	%IB2.1 %IB12.1	%IB2.31 %IB12.31
8224	2020	8225	master flags (→ page 4-40)	r	2	%IW32.240	%IW32.241
8226	2022	8380	analogue slave inputs (→ page 4-41)	r	155	%IW22.1.0	%IW22.31.4
8381	20BD	8444	current configuration data (→ page 4-53)	r	64	%IW32.0	%IW32.63
8445	20FD	8460	current parameters (→page 4-55)	r	16	%IW32.64	%IW32.79
8461	210D	8464	LAS (→ page 4-56)	r	4	%IW32.80	%IW32.83
8465	2111	8468	LDS (→ page 4-57)	r	4	%IW32.84	%IW32.87

Modbus address [words]			Content	Access	Size [words]	IEC addresses	
start		end		r = read			
dec.	hex.	dec.		w = write		from	to
8469	2115	8472	LPF (→ page 4-58)	r	4	%IW32.88	%IW32.91
8473	2119	8476	LPS (→ page 4-59)	r	4	%IW32.92	%IW32.95
8477	211D	8540	projected configuration data (→ page 4-53)	r	64	%IW32.96	%IW32.159
8541	215D	8556	reflected parameters (→ page 4-55)	r	16	%IW32.160	%IW32.175
8557	216D	8618	telegram error counter (→ page 4-60)	r	62	%IW32.176	%IW32.237
8619	21AB	8619	configuration error counter (→ page 4-62)	r	1	%IW32.238	—
8620	21AC	8620	AS-i cycle counter (→ page 4-62)	r	1	%IW32.239	—
8621	21AD	8652	digital slave outputs (→ page 4-37)	r/w	32	%QB2.1 %QB12.1	%QB2.31 %QB12.31
8653	21CD	8654	reserved	—	2	—	—
8655	21CF	8809	analogue slave outputs (→ page 4-41)	r/w	155	%QW22.1.0	%QW22.31.4
8810	226A	8873	reserved	—	64	—	—
8874	22AA	8889	reserved	—	16	—	—
8890	22BA	8908	host command channel request (→ page 4-63)	r/w	19	—	—
8909	22CD	8927	host command channel response (→ page 4-64)	r	19	—	—
General data							
12288	3000	12351	inputs from fieldbus (→ page 4-65)	r	64	%IW0.0	%IW0.63
12352	3040	12415	outputs to fieldbus (→ page 4-65)	r	64	%QW0.0	%QW0.63
12416	3080	12671	extended data to the PLC in the controller (→ page 4-67)	r/w	256	%IW4.0	%IW4.255
12672	3180	12927	extended data from the PLC in the controller (→ page 4-67)	r	256	%QW4.0	%QW4.255

Modbus address for controllere PLC status

Modbus address	Data content (16 bits = 1 word)
1024	status value = 1 → PLC is in the operating mode RUN status value = 2 → PLC is in the operating mode STOP status value = 8 → PLC is in the operating mode GATEWAY

Modbus address for Modbus timeout

Modbus address	Data content (16 bits = 1 word)
2048	timeout value in [ms]

- ► The PLC of the controllere must be in the gateway mode.
- ► If value ≠ 0: the outputs are reset if no Modbus telegram (read or write request) has been received in the specified time [ms].
- ► If value = 0: this function is deactivated.
- ► The register is predefined by the corresponding setting on the ifm standard HTML page of the web server. The value set via the ifm standard HTML page is stored non-volatilely in the controllere. Changes of this register via Modbus however are volatile. After rebooting the controllere the value defined by the HTML page is again activated.

Modbus address for Modbus write timeout

Modbus address	Data content (16 bits = 1 word)
2049	timeout value in [ms]

- ► The PLC of the controllere must be in the gateway mode.
- ► Function is identical to "Modbus timeout" (→ page 4-36), but for the Modbus write timeout only Modbus write requests are taken into account to trigger the timeout time counter.

Modbus address for "delete Modbus write timeout register"

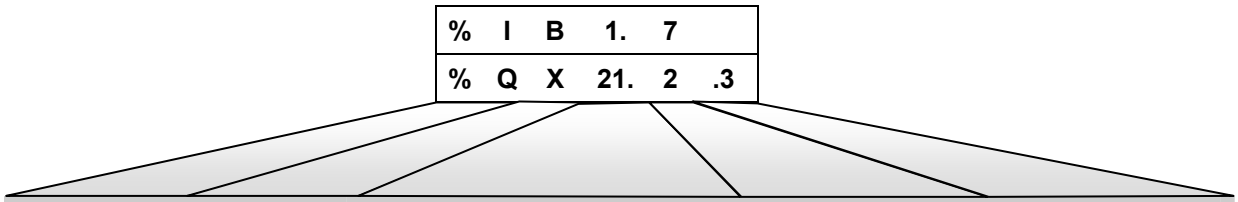
Modbus address	Data content (16 bits = 1 word)
2050	user-defined

- ► Writing on this Modbus address results in a reset of the Modbus write timeout register, thus the timeout time counter is reset.
- ► This function allows to prevent the triggering of the Modbus write timeout without having to write on the outputs used.
- ► The value written in this register is ignored in the controllere.

Modbus addresses of the digital slave inputs and outputs

Modbus addresses				Bits of the Modbus address									
Master 1 inputs	Master 1 outputs	Master 2 inputs	Master 2 outputs	15…12	11	10	9	8	7…4	3	2	1	0
			Slave data bits →		D3	D2	D1	D0		D3	D2	D1	D0
4096	4525	8192	8621	0	slave 2 (A)				0	slave 1 (A)			
4097	4526	8193	8622	0	slave 4 (A)				0	slave 3 (A)			
4098	4527	8194	8623	0	slave 6 (A)				0	slave 5 (A)			
4099	4528	8195	8624	0	slave 8 (A)				0	slave 7 (A)			
4100	4529	8196	8625	0	slave 10 (A)				0	slave 9 (A)			
4101	4530	8197	8626	0	slave 12 (A)				0	slave 11 (A)			
4102	4531	8198	8627	0	slave 14 (A)				0	slave 13 (A)			
4203	4532	8199	8628	0	slave 16 (A)				0	slave 15 (A)			
4104	4533	8200	8629	0	slave 18 (A)				0	slave 17 (A)			
4105	4534	8201	8630	0	slave 20 (A)				0	slave 19 (A)			
4106	4535	8202	8631	0	slave 22 (A)				0	slave 21 (A)			
4107	4536	8203	8632	0	slave 24 (A)				0	slave 23 (A)			
4108	4537	8204	8633	0	slave 26 (A)				0	slave 25 (A)			
4109	4538	8205	8634	0	slave 28 (A)				0	slave 27 (A)			
4110	4539	8206	8635	0	slave 30 (A)				0	slave 29 (A)			
4111	4540	8207	8636	0	reserved				0	slave 31 (A)			
4112	4541	8208	8637	0	slave 2 B				0	slave 1 B			
4113	4542	8209	8638	0	slave 4 B				0	slave 3 B			
4114	4543	8210	8639	0	slave 6 B				0	slave 5 B			
4115	4544	8211	8640	0	slave 8 B				0	slave 7 B			
4116	4545	8212	8641	0	slave 10 B				0	slave 9 B			
4117	4546	8213	8642	0	slave 12 B				0	slave 11 B			
4118	4547	8214	8643	0	slave 14 B				0	slave 13 B			
4119	4548	8215	8644	0	slave 16 B				0	slave 15 B			
4120	4549	8216	8645	0	slave 18 B				0	slave 17 B			
4121	4550	8217	8646	0	slave 20 B				0	slave 19 B			
4122	4551	8218	8647	0	slave 22 B				0	slave 21 B			
4123	4552	8219	8648	0	slave 24 B				0	slave 23 B			
4124	4553	8220	8649	0	slave 26 B				0	slave 25 B			
4125	4554	8221	8650	0	slave 28 B				0	slave 27 B			
4126	4555	8222	8651	0	slave 30 B				0	slave 29 B			
4127	4556	8223	8652	0	reserved				0	slave 31 B			

IEC addresses in the PLC of the controller$_E$ for the digital slave inputs and outputs



Data area	Type of access	Identification	Slave address	Data bit
		1 = S/A slave on master 1	1 = slave 1	if type of access = X:
I = input	B = byte	2 = S/A slave on master 2	2 = slave 2	0 = D0
Q = output	X = bit	11 = B slave on master 1	…	1 = D1 2 = D2
		21 = B slave on master 2	31 = slave 31	3 = D3

S/A slave = single slave or A slave

IEC addresses (PLC in controller$_E$)				Inputs / outputs to slave address
Master 1 inputs	**Master 1 outputs**	**Master 2 inputs**	**Master 2 outputs**	
%IB1.1	%QB1.1	%IB2.1	%QB2.1	1 (A)
%IB1.2	%QB1.2	%IB2.2	%QB2.2	2 (A)
%IB1.3	%QB1.3	%IB2.3	%QB2.3	3 (A)
%IB1.4	%QB1.4	%IB2.4	%QB2.4	4 (A)
%IB1.5	%QB1.5	%IB2.5	%QB2.5	5 (A)
%IB1.6	%QB1.6	%IB2.6	%QB2.6	6 (A)
%IB1.7	%QB1.7	%IB2.7	%QB2.7	7 (A)
%IB1.8	%QB1.8	%IB2.8	%QB2.8	8 (A)
%IB1.9	%QB1.9	%IB2.9	%QB2.9	9 (A)
%IB1.10	%QB1.10	%IB2.10	%QB2.10	10 (A)
%IB1.11	%QB1.11	%IB2.11	%QB2.11	11 (A)
%IB1.12	%QB1.12	%IB2.12	%QB2.12	12 (A)
%IB1.13	%QB1.13	%IB2.13	%QB2.13	13 (A)
%IB1.14	%QB1.14	%IB2.14	%QB2.14	14 (A)
%IB1.15	%QB1.15	%IB2.15	%QB2.15	15 (A)
%IB1.16	%QB1.16	%IB2.16	%QB2.16	16 (A)
%IB1.17	%QB1.17	%IB2.17	%QB2.17	17 (A)
%IB1.18	%QB1.18	%IB2.18	%QB2.18	18 (A)
%IB1.19	%QB1.19	%IB2.19	%QB2.19	19 (A)
%IB1.20	%QB1.20	%IB2.20	%QB2.20	20 (A)
%IB1.21	%QB1.21	%IB2.21	%QB2.21	21 (A)
%IB1.22	%QB1.22	%IB2.22	%QB2.22	22 (A)
%IB1.23	%QB1.23	%IB2.23	%QB2.23	23 (A)
%IB1.24	%QB1.24	%IB2.24	%QB2.24	24 (A)
%IB1.25	%QB1.25	%IB2.25	%QB2.25	25 (A)
%IB1.26	%QB1.26	%IB2.26	%QB2.26	26 (A)
%IB1.27	%QB1.27	%IB2.27	%QB2.27	27 (A)
%IB1.28	%QB1.28	%IB2.28	%QB2.28	28 (A)
%IB1.29	%QB1.29	%IB2.29	%QB2.29	29 (A)
%IB1.30	%QB1.30	%IB2.30	%QB2.30	30 (A)

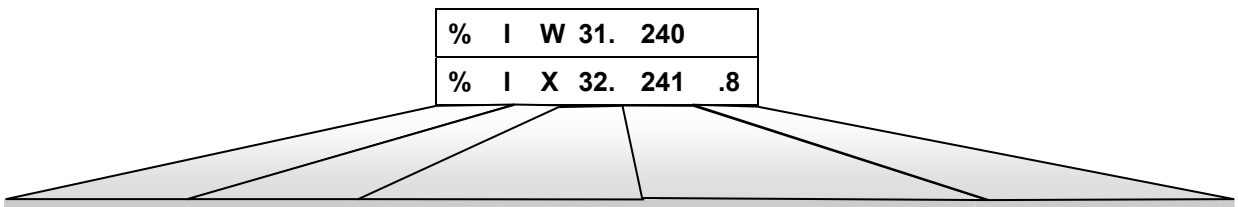
IEC addresses (PLC in controller$_E$)				Inputs / outputs to slave address
Master 1 inputs	**Master 1 outputs**	**Master 2 inputs**	**Master 2 outputs**	
%IB1.31	%QB1.31	%IB2.31	%QB2.31	31 (A)
%IB11.1	%QB11.1	%IB21.1	%QB21.1	1 B
%IB11.2	%QB11.2	%IB21.2	%QB21.2	2 B
%IB11.3	%QB11.3	%IB21.3	%QB21.3	3 B
%IB11.4	%QB11.4	%IB21.4	%QB21.4	4 B
%IB11.5	%QB11.5	%IB21.5	%QB21.5	5 B
%IB11.6	%QB11.6	%IB21.6	%QB21.6	6 B
%IB11.7	%QB11.7	%IB21.7	%QB21.7	7 B
%IB11.8	%QB11.8	%IB21.8	%QB21.8	8 B
%IB11.9	%QB11.9	%IB21.9	%QB21.9	9 B
%IB11.10	%QB11.10	%IB21.10	%QB21.10	10 B
%IB11.11	%QB11.11	%IB21.11	%QB21.11	11 B
%IB11.12	%QB11.12	%IB21.12	%QB21.12	12 B
%IB11.13	%QB11.13	%IB21.13	%QB21.13	13 B
%IB11.14	%QB11.14	%IB21.14	%QB21.14	14 B
%IB11.15	%QB11.15	%IB21.15	%QB21.15	15 B
%IB11.16	%QB11.16	%IB21.16	%QB21.16	16 B
%IB11.17	%QB11.17	%IB21.17	%QB21.17	17 B
%IB11.18	%QB11.18	%IB21.18	%QB21.18	18 B
%IB11.19	%QB11.19	%IB21.19	%QB21.19	19 B
%IB11.20	%QB11.20	%IB21.20	%QB21.20	20 B
%IB11.21	%QB11.21	%IB21.21	%QB21.21	21 B
%IB11.22	%QB11.22	%IB21.22	%QB21.22	22 B
%IB11.23	%QB11.23	%IB21.23	%QB21.23	23 B
%IB11.24	%QB11.24	%IB21.24	%QB21.24	24 B
%IB11.25	%QB11.25	%IB21.25	%QB21.25	25 B
%IB11.26	%QB11.26	%IB21.26	%QB21.26	26 B
%IB11.27	%QB11.27	%IB21.27	%QB21.27	27 B
%IB11.28	%QB11.28	%IB21.28	%QB21.28	28 B
%IB11.29	%QB11.29	%IB21.29	%QB21.29	29 B
%IB11.30	%QB11.30	%IB21.30	%QB21.30	30 B
%IB11.31	%QB11.31	%IB21.31	%QB21.31	31 B

Modbus addresses for the master flags

Modbus address IEC address (PLC in controllere)		Bit	Bit = TRUE means:
Master 1	**Master 2**		
4128 %IW31.240	8224 %IW32.240	0	"No Slave Reset" When executing the function "Config all" (via the menu or command channel of the controllere) the slaves are NOT reset, as described in the AS-i specification.
		1...15	reserved
4129 %IW31.241	8225 %IW32.241	0	"Config OK" There is no configuration error. The configuration of all AS-i slaves in the network complies with the configuration data.
		1	"LDS.0" One slave with the AS-i address 0 was detected on the master.
		2	reserved
		3	reserved
		4	"Configuration_Active" The controllere is in the configuration mode
		5	"Normal_Operation_Active" The AS-i master is in normal operation: it communicates with at least one slave.
		6	„AS-i_Power_Fail“ The AS-i voltage is too low.
		7	reserved
		8	„Periphery_OK“ None of the active AS-i slaves signals a periphery fault.
		9	„Auto_Address_Enable“ The mode "automatic addressing of the slaves" on this master is activated.
		10...15	reserved

Examples:
To retrieve the bit LDS.0 "slave 0 detected" for master 1, the address %IX31.241.1 is used; for master 2 the address %IX32.241.1 is used.

IEC addresses in the PLC of the controllere for the master flags



Data area	Type of access	Identification	Word selection	Data bit no.
I = input	W = word	31 = master 1	240 = master flags word 1	if type of access = X:
	X = bit	32 = master 2	241 = master flags word 2	0…15

Modbus addresses for the analogue slave inputs and outputs

Modbus addresses				Data content (16 bits = word)		Slave address
Master 1 inputs	Master 1 outputs	Master 2 inputs	Master 2 outputs	Channel	Description	
4130	4559	8226	8655	0	1st channel S/A slave	1
4131	4560	8227	8656	1	2nd channel S/A slave	
4132	4561	8228	8657	2	3rd channel single slave or: 1st channel B slave	
4133	4562	8229	8658	3	4th channel single slave or: 2nd channel B slave	
4134	4563	8230	8659	4	status	
4135	4564	8231	8660	0	1st channel S/A slave	2
4136	4565	8232	8661	1	2nd channel S/A slave	
4137	4566	8233	8662	2	3rd channel single slave or: 1st channel B slave	
4138	4567	8234	8663	3	4th channel single slave or: 2nd channel B slave	
4139	4568	8235	8664	4	status	
4140	4569	8236	8665	0	1st channel S/A slave	3
4141	4570	8237	8666	1	2nd channel S/A slave	
4142	4571	8238	8667	2	3rd channel single slave or: 1st channel B slave	
4143	4572	8239	8668	3	4th channel single slave or: 2nd channel B slave	
4144	4573	8240	8669	4	status	
4145	4574	8241	8670	0	1st channel S/A slave	4
4146	4575	8242	8671	1	2nd channel S/A slave	
4147	4576	8243	8672	2	3rd channel single slave or: 1st channel B slave	
4148	4577	8244	8673	3	4th channel single slave or: 2nd channel B slave	
4149	4578	8245	8674	4	status	
4150	4579	8246	8675	0	1st channel S/A slave	5
4151	4580	8247	8676	1	2nd channel S/A slave	
4152	4581	8248	8677	2	3rd channel single slave or: 1st channel B slave	
4153	4582	8249	8678	3	4th channel single slave or: 2nd channel B slave	
4154	4583	8250	8679	4	status	
4155	4584	8251	8680	0	1st channel S/A slave	6
4156	4585	8252	8681	1	2nd channel S/A slave	
4157	4586	8253	8682	2	3rd channel single slave or: 1st channel B slave	
4158	4587	8254	8683	3	4th channel single slave or: 2nd channel B slave	
4159	4588	8255	8684	4	status	
4160	4589	8256	8685	0	1st channel S/A slave	7
4161	4590	8257	8686	1	2nd channel S/A slave	

Modbus addresses				Data content (16 bits = word)		Slave address
Master 1 inputs	**Master 1 outputs**	**Master 2 inputs**	**Master 2 outputs**	**Channel**	**Description**	
4162	4591	8258	8687	2	3rd channel single slave or: 1st channel B slave	
4163	4592	8259	8688	3	4th channel single slave or: 2nd channel B slave	
4164	4593	8260	8689	4	status	
4165	4594	8261	8690	0	1st channel S/A slave	
4166	4595	8262	8691	1	2nd channel S/A slave	
4167	4596	8263	8692	2	3rd channel single slave or: 1st channel B slave	8
4168	4597	8264	8693	3	4th channel single slave or: 2nd channel B slave	
4169	4598	8265	8694	4	status	
4170	4599	8266	8695	0	1st channel S/A slave	
4171	4600	8267	8696	1	2nd channel S/A slave	
4172	4601	8268	8697	2	3rd channel single slave or: 1st channel B slave	9
4173	4602	8269	8698	3	4th channel single slave or: 2nd channel B slave	
4174	4603	8270	8699	4	status	
4175	4604	8271	8700	0	1st channel S/A slave	
4176	4605	8272	8701	1	2nd channel S/A slave	
4177	4606	8273	8702	2	3rd channel single slave or: 1st channel B slave	10
4178	4607	8274	8703	3	4th channel single slave or: 2nd channel B slave	
4179	4608	8275	8704	4	status	
4180	4609	8276	8705	0	1st channel S/A slave	
4181	4610	8277	8706	1	2nd channel S/A slave	
4182	4611	8278	8707	2	3rd channel single slave or: 1st channel B slave	11
4183	4612	8279	8708	3	4th channel single slave or: 2nd channel B slave	
4184	4613	8280	8709	4	status	
4185	4614	8281	8710	0	1st channel S/A slave	
4186	4615	8282	8711	1	2nd channel S/A slave	
4187	4616	8283	8712	2	3rd channel single slave or: 1st channel B slave	12
4188	4617	8284	8713	3	4th channel single slave or: 2nd channel B slave	
4189	4618	8285	8714	4	status	
4190	4619	8286	8715	0	1st channel S/A slave	
4191	4620	8287	8716	1	2nd channel S/A slave	
4192	4621	8288	8717	2	3rd channel single slave or: 1st channel B slave	13
4193	4622	8289	8718	3	4th channel single slave or: 2nd channel B slave	
4194	4623	8290	8719	4	status	

Modbus addresses				Data content (16 bits = word)		Slave address
Master 1 inputs	**Master 1 outputs**	**Master 2 inputs**	**Master 2 outputs**	**Channel**	**Description**	
4195	4624	8291	8720	0	1st channel S/A slave	14
4196	4625	8292	8721	1	2nd channel S/A slave	
4197	4626	8293	8722	2	3rd channel single slave or: 1st channel B slave	
4198	4627	8294	8723	3	4th channel single slave or: 2nd channel B slave	
4199	4628	8295	8724	4	status	
4200	4629	8296	8725	0	1st channel S/A slave	15
4201	4630	8297	8726	1	2nd channel S/A slave	
4202	4631	8298	8727	2	3rd channel single slave or: 1st channel B slave	
4203	4632	8299	8728	3	4th channel single slave or: 2nd channel B slave	
4204	4633	8300	8729	4	status	
4205	4634	8301	8730	0	1st channel S/A slave	16
4206	4635	8302	8731	1	2nd channel S/A slave	
4207	4636	8303	8732	2	3rd channel single slave or: 1st channel B slave	
4208	4637	8304	8733	3	4th channel single slave or: 2nd channel B slave	
4209	4638	8305	8734	4	status	
4210	4639	8306	8735	0	1st channel S/A slave	17
4211	4640	8307	8736	1	2nd channel S/A slave	
4212	4641	8308	8737	2	3rd channel single slave or: 1st channel B slave	
4213	4642	8309	8738	3	4th channel single slave or: 2nd channel B slave	
4214	4643	8310	8739	4	status	
4215	4644	8311	8740	0	1st channel S/A slave	18
4216	4645	8312	8741	1	2nd channel S/A slave	
4217	4646	8313	8742	2	3rd channel single slave or: 1st channel B slave	
4218	4647	8314	8743	3	4th channel single slave or: 2nd channel B slave	
4219	4648	8315	8744	4	status	
4220	4649	8316	8745	0	1st channel S/A slave	19
4221	4650	8317	8746	1	2nd channel S/A slave	
4222	4651	8318	8747	2	3rd channel single slave or: 1st channel B slave	
4223	4652	8319	8748	3	4th channel single slave or: 2nd channel B slave	
4224	4653	8320	8749	4	status	
4225	4654	8321	8750	0	1st channel S/A slave	20
4226	4655	8322	8751	1	2nd channel S/A slave	
4227	4656	8323	8752	2	3rd channel single slave or: 1st channel B slave	

Modbus addresses				Data content (16 bits = word)		Slave address
Master 1 inputs	**Master 1 outputs**	**Master 2 inputs**	**Master 2 outputs**	**Channel**	**Description**	
4228	4657	8324	8753	3	4th channel single slave or: 2nd channel B slave	
4229	4658	8325	8754	4	status	
4230	4659	8326	8755	0	1st channel S/A slave	
4231	4660	8327	8756	1	2nd channel S/A slave	
4232	4661	8328	8757	2	3rd channel single slave or: 1st channel B slave	21
4233	4662	8329	8758	3	4th channel single slave or: 2nd channel B slave	
4234	4663	8330	8759	4	status	
4235	4664	8331	8760	0	1st channel S/A slave	
4236	4665	8332	8761	1	2nd channel S/A slave	
4237	4666	8333	8762	2	3rd channel single slave or: 1st channel B slave	22
4238	4667	8334	8763	3	4th channel single slave or: 2nd channel B slave	
4239	4668	8335	8764	4	status	
4240	4669	8336	8765	0	1st channel S/A slave	
4241	4670	8337	8766	1	2nd channel S/A slave	
4242	4671	8338	8767	2	3rd channel single slave or: 1st channel B slave	23
4243	4672	8339	8768	3	4th channel single slave or: 2nd channel B slave	
4244	4673	8340	8769	4	status	
4245	4674	8341	8770	0	1st channel S/A slave	
4246	4675	8342	8771	1	2nd channel S/A slave	
4247	4676	8343	8772	2	3rd channel single slave or: 1st channel B slave	24
4248	4677	8344	8773	3	4th channel single slave or: 2nd channel B slave	
4249	4678	8345	8774	4	status	
4250	4679	8346	8775	0	1st channel S/A slave	
4251	4680	8347	8776	1	2nd channel S/A slave	
4252	4681	8348	8777	2	3rd channel single slave or: 1st channel B slave	25
4253	4682	8349	8778	3	4th channel single slave or: 2nd channel B slave	
4254	4683	8350	8779	4	status	
4255	4684	8351	8780	0	1st channel S/A slave	
4256	4685	8352	8781	1	2nd channel S/A slave	
4257	4686	8353	8782	2	3rd channel single slave or: 1st channel B slave	26
4258	4687	8354	8783	3	4th channel single slave or: 2nd channel B slave	
4259	4688	8355	8784	4	status	
4260	4689	8356	8785	0	1st channel S/A slave	27
4261	4690	8357	8786	1	2nd channel S/A slave	

Modbus addresses				Data content (16 bits = word)		Slave address
Master 1 inputs	**Master 1 outputs**	**Master 2 inputs**	**Master 2 outputs**	**Channel**	**Description**	
4262	4691	8358	8787	2	3rd channel single slave or: 1st channel B slave	
4263	4692	8359	8788	3	4th channel single slave or: 2nd channel B slave	
4264	4693	8360	8789	4	status	
4265	4694	8361	8790	0	1st channel S/A slave	
4266	4695	8362	8791	1	2nd channel S/A slave	
4267	4696	8363	8792	2	3rd channel single slave or: 1st channel B slave	28
4268	4697	8364	8793	3	4th channel single slave or: 2nd channel B slave	
4269	4698	8365	8794	4	status	
4270	4699	8366	8795	0	1st channel S/A slave	
4271	4700	8367	8796	1	2nd channel S/A slave	
4272	4701	8368	8797	2	3rd channel single slave or: 1st channel B slave	29
4273	4702	8369	8798	3	4th channel single slave or: 2nd channel B slave	
4274	4703	8370	8799	4	status	
4275	4704	8371	8800	0	1st channel S/A slave	
4276	4705	8372	8801	1	2nd channel S/A slave	
4277	4706	8373	8802	2	3rd channel single slave or: 1st channel B slave	30
4278	4707	8374	8803	3	4th channel single slave or: 2nd channel B slave	
4279	4708	8375	8804	4	status	
4280	4709	8376	8805	0	1st channel S/A slave	
4281	4710	8377	8806	1	2nd channel S/A slave	
4282	4711	8378	8807	2	3rd channel single slave or: 1st channel B slave	31
4283	4712	8379	8808	3	4th channel single slave or: 2nd channel B slave	
4284	4713	8380	8809	4	status	

Status information of analogue slaves

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
0	**analogue channel 0 from/for slave or: analogue data channel 0 from/for slave 1A**															
1	**analogue data channel 1 from/for slave 1 or: analogue data channel 1 from/for slave 1A**															
2	**analogue data channel 2 from/for slave 1 or: analogue data channel 0 from/for slave 1B**															
3	**analogue data channel 3 from/for slave 1 or: analogue data channel 1 from/for slave 1B**															
4	**TIB**	**TOB**	**TIA**	**TOA**	**TVB**	**OVB**	**TVA**	**OVA**	**O3**	**V3**	**O2**	**V2**	**O1**	**V1**	**O0**	**V0**

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
5	**analogue data channel 0 from/for slave 2 or: analogue data channel 0 from/for slave 2A**															
6	**analogue data channel 1 from/for slave 2 or: analogue data channel 1 from/for slave 2A**															
7	**analogue data channel 2 from/for slave 2 or: analogue data channel 0 from/for slave 2B**															
8	**analogue data channel 3 from/for slave 2 or: analogue data channel 1 from/for slave 2B**															
9	**TIB**	**TOB**	**TIA**	**TOA**	**TVB**	**OVB**	**TVA**	**OVA**	**O3**	**V3**	**O2**	**V2**	**O1**	**V1**	**O0**	**V0**
…	…															
150	**analogue data channel 0 from/for slave 31 or: analogue data channel 0 from/for slave 31A**															
151	**analogue data channel 1 from/for slave 31 or: analogue data channel 1 from/for slave 31A**															
152	**analogue data channel 2 from/for slave 31 or: analogue data channel 0 from/for slave 31B**															
153	**analogue data channel 3 from/for slave 31 or: analogue data channel 1 from/for slave 31B**															
154	**TIB**	**TOB**	**TIA**	**TOA**	**TVB**	**OVB**	**TVA**	**OVA**	**O3**	**V3**	**O2**	**V2**	**O1**	**V1**	**O0**	**V0**

Legend:

Vn
validity bit "valid" for channel number n = 0...3

NOTE: set Vn = "1" for analogue **out**put slaves!

On
bit "overflow" for channel number n = 0…3

OVA
channel-independent bit "output data valid" from A slave

CTT1:
0 = more than 3.5 s have elapsed since the last update of the output values
1 = slave requests new output data within the next 3 s

CTT2…CTT5:
0 = slave receives no new output data
1 = slave receives new output data

TVA
channel-independent bit "transmission valid" from A slave/single slave:

0 = error during transmission or: timeout
1 = transmission of analogue input/output data OK

OVB
channel-independent bit "output data valid" from B slave:

CTT1:
0 = more than 3.5 s have elapsed since the last update of the output values
1 = slave requests new output data within the next 3 s

CTT2…CTT5:
0 = slave receives no new output data
1 = slave receives new output data

NOTE: only valid for analogue **out**put slaves. Set OVB = 0 for **in**put slaves!

TVB
channel-independent bit "transmission valid" from B slave:

0 = error during transmission or: timeout
1 = transmission of analogue input/output data OK

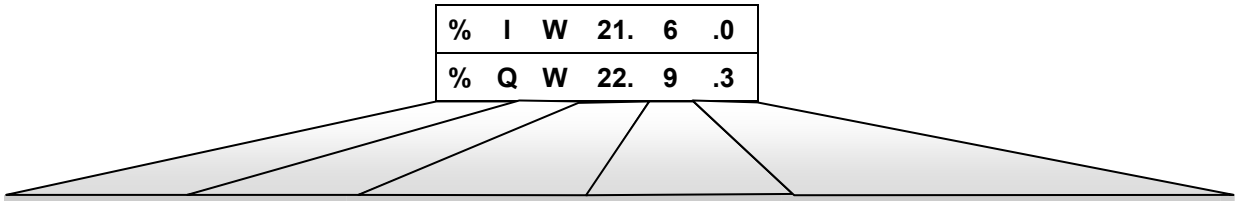
TIA
slave transmits analogue input data...

TIB
0 = in the analogue mode (15 bits, with sign)
1 = in the transparent mode (16 bits, without sign)

TOA
slave receives analogue output data...

TOB
0 = in the analogue mode (15 bits, with sign)
1 = in the transparent mode (16 bits, without sign)

IEC addresses in the PLC of the controller$_e$ for the analogue slave inputs and outputs



Data area	Type of access	Identification	Slave address	Data channel
I = input Q = output	W = word	21 = master 1 22 = master 2	1 = slave 1 2 = slave 2 … 31 = slave 31	0 = 1st channel S/A slave 1 = 2nd channel S/A slave 2 = 3rd channel single slave or: 1st channel B slave 3 = 4th channel single slave or: 2nd channel B slave 4 = status

S/A slave = single slave or A slave

A/B slaves have max. 2 analogue channels

IEC addresses (PLC in controller$_e$)				Data content (16 bits = word)		Slave address
Master 1 inputs	**Master 1 outputs**	**Master 2 inputs**	**Master 2 outputs**	**Channel**	**Description**	
%IW21.1.0	%QW21.1.0	%IW22.1.0	%QW22.1.0	0	1st channel S/A slave	1
%IW21.1.1	%QW21.1.1	%IW22.1.1	%QW22.1.1	1	2nd channel S/A slave	
%IW21.1.2	%QW21.1.2	%IW22.1.2	%QW22.1.2	2	3rd channel single slave or: 1st channel B slave	
%IW21.1.3	%QW21.1.3	%IW22.1.3	%QW22.1.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.1.4	%QW21.1.4	%IW22.1.4	%QW22.1.4	4	status	
%IW21.2.0	%QW21.2.0	%IW22.2.0	%QW22.2.0	0	1st channel S/A slave	2
%IW21.2.1	%QW21.2.1	%IW22.2.1	%QW22.2.1	1	2nd channel S/A slave	
%IW21.2.2	%QW21.2.2	%IW22.2.2	%QW22.2.2	2	3rd channel single slave or: 1st channel B slave	
%IW21.2.3	%QW21.2.3	%IW22.2.3	%QW22.2.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.2.4	%QW21.2.4	%IW22.2.4	%QW22.2.4	4	status	
%IW21.3.0	%QW21.3.0	%IW22.3.0	%QW22.3.0	0	1st channel S/A slave	3
%IW21.3.1	%QW21.3.1	%IW22.3.1	%QW22.3.1	1	2nd channel S/A slave	
%IW21.3.2	%QW21.3.2	%IW22.3.2	%QW22.3.2	2	3rd channel single slave or: 1st channel B slave	
%IW21.3.3	%QW21.3.3	%IW22.3.3	%QW22.3.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.3.4	%QW21.3.4	%IW22.3.4	%QW22.3.4	4	status	
%IW21.4.0	%QW21.4.0	%IW22.4.0	%QW22.4.0	0	1st channel S/A slave	4
%IW21.4.1	%QW21.4.1	%IW22.4.1	%QW22.4.1	1	2nd channel S/A slave	
%IW21.4.2	%QW21.4.2	%IW22.4.2	%QW22.4.2	2	3rd channel single slave or: 1st channel B slave	
%IW21.4.3	%QW21.4.3	%IW22.4.3	%QW22.4.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.4.4	%QW21.4.4	%IW22.4.4	%QW22.4.4	4	status	
%IW21.5.0	%QW21.5.0	%IW22.5.0	%QW22.5.0	0	1st channel S/A slave	5

IEC addresses (PLC in controller$_E$)				Data content (16 bits = word)		Slave address
Master 1 inputs	**Master 1 outputs**	**Master 2 inputs**	**Master 2 outputs**	**Channel**	**Description**	
%IW21.5.1	%QW21.5.1	%IW22.5.1	%QW22.5.1	1	2nd channel S/A slave	
%IW21.5.2	%QW21.5.2	%IW22.5.2	%QW22.5.2	2	3rd channel single slave or: 1st channel B slave	
%IW21.5.3	%QW21.5.3	%IW22.5.3	%QW22.5.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.5.4	%QW21.5.4	%IW22.5.4	%QW22.5.4	4	status	
%IW21.6.0	%QW21.6.0	%IW22.6.0	%QW22.6.0	0	1st channel S/A slave	6
%IW21.6.1	%QW21.6.1	%IW22.6.1	%QW22.6.1	1	2nd channel S/A slave	
%IW21.6.2	%QW21.6.2	%IW22.6.2	%QW22.6.2	2	3rd channel single slave or: 1st channel B slave	
%IW21.6.3	%QW21.6.3	%IW22.6.3	%QW22.6.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.6.4	%QW21.6.4	%IW22.6.4	%QW22.6.4	4	status	
%IW21.7.0	%QW21.7.0	%IW22.7.0	%QW22.7.0	0	1st channel S/A slave	7
%IW21.7.1	%QW21.7.1	%IW22.7.1	%QW22.7.1	1	2nd channel S/A slave	
%IW21.7.2	%QW21.7.2	%IW22.7.2	%QW22.7.2	2	3rd channel single slave or: 1st channel B slave	
%IW21.7.3	%QW21.7.3	%IW22.7.3	%QW22.7.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.7.4	%QW21.7.4	%IW22.7.4	%QW22.7.4	4	status	
%IW21.8.0	%QW21.8.0	%IW22.8.0	%QW22.8.0	0	1st channel S/A slave	8
%IW21.8.1	%QW21.8.1	%IW22.8.1	%QW22.8.1	1	2nd channel S/A slave	
%IW21.8.2	%QW21.8.2	%IW22.8.2	%QW22.8.2	2	3rd channel single slave or: 1st channel B slave	
%IW21.8.3	%QW21.8.3	%IW22.8.3	%QW22.8.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.8.4	%QW21.8.4	%IW22.8.4	%QW22.8.4	4	status	
%IW21.9.0	%QW21.9.0	%IW22.9.0	%QW22.9.0	0	1st channel S/A slave	9
%IW21.9.1	%QW21.9.1	%IW22.9.1	%QW22.9.1	1	2nd channel S/A slave	
%IW21.9.2	%QW21.9.2	%IW22.9.2	%QW22.9.2	2	3rd channel single slave or: 1st channel B slave	
%IW21.9.3	%QW21.9.3	%IW22.9.3	%QW22.9.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.9.4	%QW21.9.4	%IW22.9.4	%QW22.9.4	4	status	
%IW21.10.0	%QW21.10.0	%IW22.10.0	%QW22.10.0	0	1st channel S/A slave	10
%IW21.10.1	%QW21.10.1	%IW22.10.1	%QW22.10.1	1	2nd channel S/A slave	
%IW21.10.2	%QW21.10.2	%IW22.10.2	%QW22.10.2	2	3rd channel single slave or: 1st channel B slave	
%IW21.10.3	%QW21.10.3	%IW22.10.3	%QW22.10.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.10.4	%QW21.10.4	%IW22.10.4	%QW22.10.4	4	status	
%IW21.11.0	%QW21.11.0	%IW22.11.0	%QW22.11.0	0	1st channel S/A slave	11
%IW21.11.1	%QW21.11.1	%IW22.11.1	%QW22.11.1	1	2nd channel S/A slave	
%IW21.11.2	%QW21.11.2	%IW22.11.2	%QW22.11.2	2	3rd channel single slave or: 1st channel B slave	

IEC addresses (PLC in controller$_E$)				Data content (16 bits = word)		Slave address
Master 1 inputs	**Master 1 outputs**	**Master 2 inputs**	**Master 2 outputs**	**Channel**	**Description**	
%IW21.11.3	%QW21.11.3	%IW22.11.3	%QW22.11.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.11.4	%QW21.11.4	%IW22.11.4	%QW22.11.4	4	status	
%IW21.12.0	%QW21.12.0	%IW22.12.0	%QW22.12.0	0	1st channel S/A slave	
%IW21.12.1	%QW21.12.1	%IW22.12.1	%QW22.12.1	1	2nd channel S/A slave	
%IW21.12.2	%QW21.12.2	%IW22.12.2	%QW22.12.2	2	3rd channel single slave or: 1st channel B slave	12
%IW21.12.3	%QW21.12.3	%IW22.12.3	%QW22.12.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.12.4	%QW21.12.4	%IW22.12.4	%QW22.12.4	4	status	
%IW21.13.0	%QW21.13.0	%IW22.13.0	%QW22.13.0	0	1st channel S/A slave	
%IW21.13.1	%QW21.13.1	%IW22.13.1	%QW22.13.1	1	2nd channel S/A slave	
%IW21.13.2	%QW21.13.2	%IW22.13.2	%QW22.13.2	2	3rd channel single slave or: 1st channel B slave	13
%IW21.13.3	%QW21.13.3	%IW22.13.3	%QW22.13.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.13.4	%QW21.13.4	%IW22.13.4	%QW22.13.4	4	status	
%IW21.14.0	%QW21.14.0	%IW22.14.0	%QW22.14.0	0	1st channel S/A slave	
%IW21.14.1	%QW21.14.1	%IW22.14.1	%QW22.14.1	1	2nd channel S/A slave	
%IW21.14.2	%QW21.14.2	%IW22.14.2	%QW22.14.2	2	3rd channel single slave or: 1st channel B slave	14
%IW21.14.3	%QW21.14.3	%IW22.14.3	%QW22.14.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.14.4	%QW21.14.4	%IW22.14.4	%QW22.14.4	4	status	
%IW21.15.0	%QW21.15.0	%IW22.15.0	%QW22.15.0	0	1st channel S/A slave	
%IW21.15.1	%QW21.15.1	%IW22.15.1	%QW22.15.1	1	2nd channel S/A slave	
%IW21.15.2	%QW21.15.2	%IW22.15.2	%QW22.15.2	2	3rd channel single slave or: 1st channel B slave	15
%IW21.15.3	%QW21.15.3	%IW22.15.3	%QW22.15.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.15.4	%QW21.15.4	%IW22.15.4	%QW22.15.4	4	status	
%IW21.16.0	%QW21.16.0	%IW22.16.0	%QW22.16.0	0	1st channel S/A slave	
%IW21.16.1	%QW21.16.1	%IW22.16.1	%QW22.16.1	1	2nd channel S/A slave	
%IW21.16.2	%QW21.16.2	%IW22.16.2	%QW22.16.2	2	3rd channel single slave or: 1st channel B slave	16
%IW21.16.3	%QW21.16.3	%IW22.16.3	%QW22.16.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.16.4	%QW21.16.4	%IW22.16.4	%QW22.16.4	4	status	
%IW21.17.0	%QW21.17.0	%IW22.17.0	%QW22.17.0	0	1st channel S/A slave	
%IW21.17.1	%QW21.17.1	%IW22.17.1	%QW22.17.1	1	2nd channel S/A slave	
%IW21.17.2	%QW21.17.2	%IW22.17.2	%QW22.17.2	2	3rd channel single slave or: 1st channel B slave	17
%IW21.17.3	%QW21.17.3	%IW22.17.3	%QW22.17.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.17.4	%QW21.17.4	%IW22.17.4	%QW22.17.4	4	status	
%IW21.18.0	%QW21.18.0	%IW22.18.0	%QW22.18.0	0	1st channel S/A slave	18

IEC addresses (PLC in controller$_E$)				Data content (16 bits = word)		Slave address
Master 1 inputs	**Master 1 outputs**	**Master 2 inputs**	**Master 2 outputs**	**Channel**	**Description**	
%IW21.18.1	%QW21.18.1	%IW22.18.1	%QW22.18.1	1	2nd channel S/A slave	
%IW21.18.2	%QW21.18.2	%IW22.18.2	%QW22.18.2	2	3rd channel single slave or: 1st channel B slave	
%IW21.18.3	%QW21.18.3	%IW22.18.3	%QW22.18.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.18.4	%QW21.18.4	%IW22.18.4	%QW22.18.4	4	status	
%IW21.19.0	%QW21.19.0	%IW22.19.0	%QW22.19.0	0	1st channel S/A slave	
%IW21.19.1	%QW21.19.1	%IW22.19.1	%QW22.19.1	1	2nd channel S/A slave	
%IW21.19.2	%QW21.19.2	%IW22.19.2	%QW22.19.2	2	3rd channel single slave or: 1st channel B slave	19
%IW21.19.3	%QW21.19.3	%IW22.19.3	%QW22.19.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.19.4	%QW21.19.4	%IW22.19.4	%QW22.19.4	4	status	
%IW21.20.0	%QW21.20.0	%IW22.20.0	%QW22.20.0	0	1st channel S/A slave	
%IW21.20.1	%QW21.20.1	%IW22.20.1	%QW22.20.1	1	2nd channel S/A slave	
%IW21.20.2	%QW21.20.2	%IW22.20.2	%QW22.20.2	2	3rd channel single slave or: 1st channel B slave	20
%IW21.20.3	%QW21.20.3	%IW22.20.3	%QW22.20.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.20.4	%QW21.20.4	%IW22.20.4	%QW22.20.4	4	status	
%IW21.21.0	%QW21.21.0	%IW22.21.0	%QW22.21.0	0	1st channel S/A slave	
%IW21.21.1	%QW21.21.1	%IW22.21.1	%QW22.21.1	1	2nd channel S/A slave	
%IW21.21.2	%QW21.21.2	%IW22.21.2	%QW22.21.2	2	3rd channel single slave or: 1st channel B slave	21
%IW21.21.3	%QW21.21.3	%IW22.21.3	%QW22.21.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.21.4	%QW21.21.4	%IW22.21.4	%QW22.21.4	4	status	
%IW21.22.0	%QW21.22.0	%IW22.22.0	%QW22.22.0	0	1st channel S/A slave	
%IW21.22.1	%QW21.22.1	%IW22.22.1	%QW22.22.1	1	2nd channel S/A slave	
%IW21.22.2	%QW21.22.2	%IW22.22.2	%QW22.22.2	2	3rd channel single slave or: 1st channel B slave	22
%IW21.22.3	%QW21.22.3	%IW22.22.3	%QW22.22.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.22.4	%QW21.22.4	%IW22.22.4	%QW22.22.4	4	status	
%IW21.23.0	%QW21.23.0	%IW22.23.0	%QW22.23.0	0	1st channel S/A slave	
%IW21.23.1	%QW21.23.1	%IW22.23.1	%QW22.23.1	1	2nd channel S/A slave	
%IW21.23.2	%QW21.23.2	%IW22.23.2	%QW22.23.2	2	3rd channel single slave or: 1st channel B slave	23
%IW21.23.3	%QW21.23.3	%IW22.23.3	%QW22.23.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.23.4	%QW21.23.4	%IW22.23.4	%QW22.23.4	4	status	
%IW21.24.0	%QW21.24.0	%IW22.24.0	%QW22.24.0	0	1st channel S/A slave	24
%IW21.24.1	%QW21.24.1	%IW22.24.1	%QW22.24.1	1	2nd channel S/A slave	
%IW21.24.2	%QW21.24.2	%IW22.24.2	%QW22.24.2	2	3rd channel single slave or: 1st channel B slave	

IEC addresses (PLC in controller$_E$)				Data content (16 bits = word)		Slave address
Master 1 inputs	**Master 1 outputs**	**Master 2 inputs**	**Master 2 outputs**	**Channel**	**Description**	
%IW21.24.3	%QW21.24.3	%IW22.24.3	%QW22.24.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.24.4	%QW21.24.4	%IW22.24.4	%QW22.24.4	4	status	
%IW21.25.0	%QW21.25.0	%IW22.25.0	%QW22.25.0	0	1st channel S/A slave	
%IW21.25.1	%QW21.25.1	%IW22.25.1	%QW22.25.1	1	2nd channel S/A slave	
%IW21.25.2	%QW21.25.2	%IW22.25.2	%QW22.25.2	2	3rd channel single slave or: 1st channel B slave	25
%IW21.25.3	%QW21.25.3	%IW22.25.3	%QW22.25.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.25.4	%QW21.25.4	%IW22.25.4	%QW22.25.4	4	status	
%IW21.26.0	%QW21.26.0	%IW22.26.0	%QW22.26.0	0	1st channel S/A slave	
%IW21.26.1	%QW21.26.1	%IW22.26.1	%QW22.26.1	1	2nd channel S/A slave	
%IW21.26.2	%QW21.26.2	%IW22.26.2	%QW22.26.2	2	3rd channel single slave or: 1st channel B slave	26
%IW21.26.3	%QW21.26.3	%IW22.26.3	%QW22.26.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.26.4	%QW21.26.4	%IW22.26.4	%QW22.26.4	4	status	
%IW21.27.0	%QW21.27.0	%IW22.27.0	%QW22.27.0	0	1st channel S/A slave	
%IW21.27.1	%QW21.27.1	%IW22.27.1	%QW22.27.1	1	2nd channel S/A slave	
%IW21.27.2	%QW21.27.2	%IW22.27.2	%QW22.27.2	2	3rd channel single slave or: 1st channel B slave	27
%IW21.27.3	%QW21.27.3	%IW22.27.3	%QW22.27.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.27.4	%QW21.27.4	%IW22.27.4	%QW22.27.4	4	status	
%IW21.28.0	%QW21.28.0	%IW22.28.0	%QW22.28.0	0	1st channel S/A slave	
%IW21.28.1	%QW21.28.1	%IW22.28.1	%QW22.28.1	1	2nd channel S/A slave	
%IW21.28.2	%QW21.28.2	%IW22.28.2	%QW22.28.2	2	3rd channel single slave or: 1st channel B slave	28
%IW21.28.3	%QW21.28.3	%IW22.28.3	%QW22.28.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.28.4	%QW21.28.4	%IW22.28.4	%QW22.28.4	4	status	
%IW21.29.0	%QW21.29.0	%IW22.29.0	%QW22.29.0	0	1st channel S/A slave	
%IW21.29.1	%QW21.29.1	%IW22.29.1	%QW22.29.1	1	2nd channel S/A slave	
%IW21.29.2	%QW21.29.2	%IW22.29.2	%QW22.29.2	2	3rd channel single slave or: 1st channel B slave	29
%IW21.29.3	%QW21.29.3	%IW22.29.3	%QW22.29.3	3	4th channel single slave or: 2nd channel B slave	
IW21.29.4	IW21.29.4	IW22.29.4	IW22.29.4	4	status	
%IW21.30.0	%QW21.30.0	%IW22.30.0	%QW22.30.0	0	1st channel S/A slave	
%IW21.30.1	%QW21.30.1	%IW22.30.1	%QW22.30.1	1	2nd channel S/A slave	
%IW21.30.2	%QW21.30.2	%IW22.30.2	%QW22.30.2	2	3rd channel single slave or: 1st channel B slave	30
%IW21.30.3	%QW21.30.3	%IW22.30.3	%QW22.30.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.30.4	%QW21.30.4	%IW22.30.4	%QW22.30.4	4	status	
%IW21.31.0	%QW21.31.0	%IW22.31.0	%QW22.31.0	0	1st channel S/A slave	31

IEC addresses (PLC in controller$_E$)				Data content (16 bits = word)		Slave address
Master 1 inputs	**Master 1 outputs**	**Master 2 inputs**	**Master 2 outputs**	**Channel**	**Description**	
%IW21.31.1	%QW21.31.1	%IW22.31.1	%QW22.31.1	1	2nd channel S/A slave	
%IW21.31.2	%QW21.31.2	%IW22.31.2	%QW22.31.2	2	3rd channel single slave or: 1st channel B slave	
%IW21.31.3	%QW21.31.3	%IW22.31.3	%QW22.31.3	3	4th channel single slave or: 2nd channel B slave	
%IW21.31.4	%QW21.31.4	%IW22.31.4	%QW22.31.4	4	status	

Modbus addresses for configuration data (CDI) of the slaves

CDI = **C**onfiguration **D**ata **I**mage

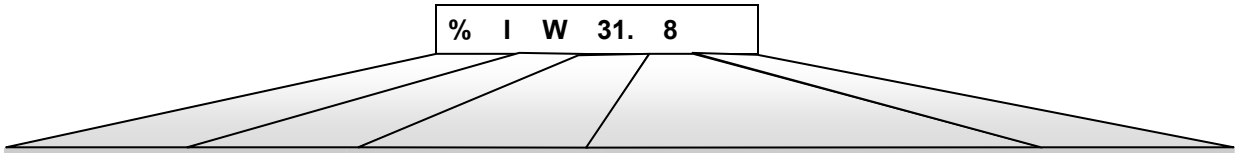
Modbus addresses				Bits / data content				IEC addresses (PLC in controllere)			
Master 1		Master 2		15…12 XID2	11…8 XID1	7…4 ID	3…0 IO	Master 1		Master 2	
current	projected	current	projected	configuration data of slave				current	projected	current	projected
4285	4381	8381	8477	0				%IW31.0	%IW31.96	%IW32.0	%IW32.96
4286	4382	8382	8478	1 (A)				%IW31.1	%IW31.97	%IW32.1	%IW32.97
4287	4383	8383	8479	2 (A)				%IW31.2	%IW31.98	%IW32.2	%IW32.98
4288	4384	8384	8480	3 (A)				%IW31.3	%IW31.99	%IW32.3	%IW32.99
4289	4385	8385	8481	4 (A)				%IW31.4	%IW31.100	%IW32.4	%IW32.100
4290	4386	8386	8482	5 (A)				%IW31.5	%IW31.101	%IW32.5	%IW32.101
4291	4387	8387	8483	6 (A)				%IW31.6	%IW31.102	%IW32.6	%IW32.102
4292	4388	8388	8484	7 (A)				%IW31.7	%IW31.103	%IW32.7	%IW32.103
4293	4389	8389	8485	8 (A)				%IW31.8	%IW31.104	%IW32.8	%IW32.104
4294	4390	8390	8486	9 (A)				%IW31.9	%IW31.105	%IW32.9	%IW32.105
4295	4391	8391	8487	10 (A)				%IW31.10	%IW31.106	%IW32.10	%IW32.106
4296	4392	8392	8488	11 (A)				%IW31.11	%IW31.107	%IW32.11	%IW32.107
4297	4393	8393	8489	12 (A)				%IW31.12	%IW31.108	%IW32.12	%IW32.108
4298	4394	8394	8490	13 (A)				%IW31.13	%IW31.109	%IW32.13	%IW32.109
4299	4395	8395	8491	14 (A)				%IW31.14	%IW31.110	%IW32.14	%IW32.110
4300	4396	8396	8492	15 (A)				%IW31.15	%IW31.111	%IW32.15	%IW32.111
4301	4397	8397	8493	16 (A)				%IW31.16	%IW31.112	%IW32.16	%IW32.112
4302	4398	8398	8494	17 (A)				%IW31.17	%IW31.113	%IW32.17	%IW32.113
4303	4399	8399	8495	18 (A)				%IW31.18	%IW31.114	%IW32.18	%IW32.114
4304	4400	8400	8496	19 (A)				%IW31.19	%IW31.115	%IW32.19	%IW32.115
4305	4401	8401	8497	20 (A)				%IW31.20	%IW31.116	%IW32.20	%IW32.116
4306	4402	8402	8498	21 (A)				%IW31.21	%IW31.117	%IW32.21	%IW32.117
4307	4403	8403	8499	22 (A)				%IW31.22	%IW31.118	%IW32.22	%IW32.118
4308	4404	8404	8500	23 (A)				%IW31.23	%IW31.119	%IW32.23	%IW32.119
4309	4405	8405	8501	24 (A)				%IW31.24	%IW31.120	%IW32.24	%IW32.120
4310	4406	8406	8502	25 (A)				%IW31.25	%IW31.121	%IW32.25	%IW32.121
4311	4407	8407	8503	26 (A)				%IW31.26	%IW31.122	%IW32.26	%IW32.122
4312	4408	8408	8504	27 (A)				%IW31.27	%IW31.123	%IW32.27	%IW32.123
4313	4409	8409	8505	28 (A)				%IW31.28	%IW31.124	%IW32.28	%IW32.124
4314	4410	8410	8506	29 (A)				%IW31.29	%IW31.125	%IW32.29	%IW32.125
4315	4411	8411	8507	30 (A)				%IW31.30	%IW31.126	%IW32.30	%IW32.126
4316	4412	8412	8508	31 (A)				%IW31.31	%IW31.127	%IW32.31	%IW32.127
4317	4413	8413	8509	(0 B)*				%IW31.32	%IW31.128	%IW32.32	%IW32.128
4318	4414	8414	8510	1 B				%IW31.33	%IW31.129	%IW32.33	%IW32.129
4319	4415	8415	8511	2 B				%IW31.34	%IW31.130	%IW32.34	%IW32.130
4320	4416	8416	8512	3 B				%IW31.35	%IW31.131	%IW32.35	%IW32.131
4321	4417	8417	8513	4 B				%IW31.36	%IW31.132	%IW32.36	%IW32.132
4322	4418	8418	8514	5 B				%IW31.37	%IW31.133	%IW32.37	%IW32.133

Modbus addresses				Bits / data content				IEC addresses (PLC in controller$_E$)			
Master 1		Master 2		15…12 XID2	11…8 XID1	7…4 ID	3…0 IO	Master 1		Master 2	
current	projected	current	projected	configuration data of slave				current	projected	current	projected
4323	4419	8419	8515	6 B				%IW31.38	%IW31.134	%IW32.38	%IW32.134
4324	4420	8420	8516	7 B				%IW31.39	%IW31.135	%IW32.39	%IW32.135
4325	4421	8421	8517	8 B				%IW31.40	%IW31.136	%IW32.40	%IW32.136
4326	4422	8422	8518	9 B				%IW31.41	%IW31.137	%IW32.41	%IW32.137
4327	4423	8423	8519	10 B				%IW31.42	%IW31.138	%IW32.42	%IW32.138
4328	4424	8424	8520	11 B				%IW31.43	%IW31.139	%IW32.43	%IW32.139
4329	4425	8425	8521	12 B				%IW31.44	%IW31.140	%IW32.44	%IW32.140
4330	4426	8426	8522	13 B				%IW31.45	%IW31.141	%IW32.45	%IW32.141
4331	4427	8427	8523	14 B				%IW31.46	%IW31.142	%IW32.46	%IW32.142
4332	4428	8428	8524	15 B				%IW31.47	%IW31.143	%IW32.47	%IW32.143
4333	4429	8429	8525	16 B				%IW31.48	%IW31.144	%IW32.48	%IW32.144
4334	4430	8430	8526	17 B				%IW31.49	%IW31.145	%IW32.49	%IW32.145
4335	4431	8431	8527	18 B				%IW31.50	%IW31.146	%IW32.50	%IW32.146
4336	4432	8432	8528	19 B				%IW31.51	%IW31.147	%IW32.51	%IW32.147
4337	4433	8433	8529	20 B				%IW31.52	%IW31.148	%IW32.52	%IW32.148
4338	4434	8434	8530	21 B				%IW31.53	%IW31.149	%IW32.53	%IW32.149
4339	4435	8435	8531	22 B				%IW31.54	%IW31.150	%IW32.54	%IW32.150
4340	4436	8436	8532	23 B				%IW31.55	%IW31.151	%IW32.55	%IW32.151
4341	4437	8437	8533	24 B				%IW31.56	%IW31.152	%IW32.56	%IW32.152
4342	4438	8438	8534	25 B				%IW31.57	%IW31.153	%IW32.57	%IW32.153
4343	4439	8439	8535	26 B				%IW31.58	%IW31.154	%IW32.58	%IW32.154
4344	4440	8440	8536	27 B				%IW31.59	%IW31.155	%IW32.59	%IW32.155
4345	4441	8441	8537	28 B				%IW31.60	%IW31.156	%IW32.60	%IW32.156
4346	4442	8442	8538	29 B				%IW31.61	%IW31.157	%IW32.61	%IW32.157
4347	4443	8443	8539	30 B				%IW31.62	%IW31.158	%IW32.62	%IW32.158
4348	4444	8444	8540	31 B				%IW31.63	%IW31.159	%IW32.63	%IW32.159

*) For the AS-i address 0B there is no configuration data. The value of this field is always zero.

IEC addresses in the PLC of the controller$_E$ for configuration data (CDI) of the slaves

CDI = **C**onfiguration **D**ata **I**mage

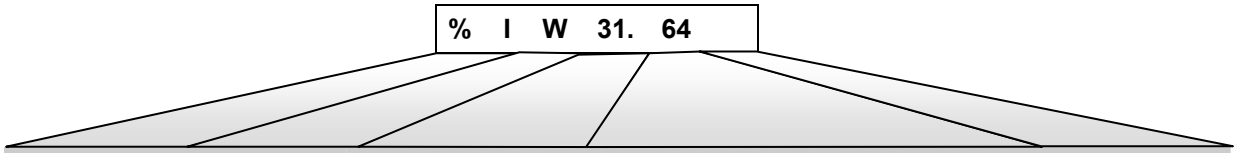


Data area	Type of access	Identification		IEC address index	—
I = input	W = word	31 = master 1	0…63	current configuration	not used
		32 = master 2	96…159	projected configuration	

Modbus addresses for parameter data of the slaves

Modbus addresses				Bits				IEC addresses (PLC in controller$_E$)			
Master 1		Master 2		15...12	11...8	7...4	3...0	Master 1		Master 2	
current	reflected	current	reflected	parameter data of slave				current	reflected	current	reflected
4349	4445	8445	8541	4(A)	3(A)	2(A)	1(A)	%IW31.64	%IW31.160	%IW32.64	%IW32.160
4350	4446	8446	8542	8(A)	7(A)	6(A)	5(A)	%IW31.65	%IW31.161	%IW32.65	%IW32.161
4351	4447	8447	8543	12(A)	11(A)	10(A)	9(A)	%IW31.66	%IW31.162	%IW32.66	%IW32.162
4352	4448	8448	8544	16(A)	15(A)	14(A)	13(A)	%IW31.67	%IW31.163	%IW32.67	%IW32.163
4353	4449	8449	8545	20(A)	19(A)	18(A)	17(A)	%IW31.68	%IW31.164	%IW32.68	%IW32.164
4354	4450	8450	8546	24(A)	23(A)	22(A)	21(A)	%IW31.69	%IW31.165	%IW32.69	%IW32.165
4355	4451	8451	8547	28(A)	27(A)	26(A)	25(A)	%IW31.70	%IW31.166	%IW32.70	%IW32.166
4356	4452	8452	8548	1B	31(A)	30(A)	29(A)	%IW31.71	%IW31.167	%IW32.71	%IW32.167
4357	4453	8453	8549	5B	4B	3B	2B	%IW31.72	%IW31.168	%IW32.72	%IW32.168
4358	4454	8454	8550	9B	8B	7B	6B	%IW31.73	%IW31.169	%IW32.73	%IW32.169
4359	4455	8455	8551	13B	12B	11B	10B	%IW31.74	%IW31.170	%IW32.74	%IW32.170
4360	4456	8456	8552	17B	16B	15B	14B	%IW31.75	%IW31.171	%IW32.75	%IW32.171
4361	4457	8457	8553	21B	20B	19B	18B	%IW31.76	%IW31.172	%IW32.76	%IW32.172
4362	4458	8458	8554	25B	24B	23B	22B	%IW31.77	%IW31.173	%IW32.77	%IW32.173
4363	4459	8459	8555	29B	28B	27B	26B	%IW31.78	%IW31.174	%IW32.78	%IW32.174
4364	4460	8460	8556	reserved		31B	30B	%IW31.79	%IW31.175	%IW32.79	%IW32.175

IEC addresses in the PLC of the controller$_E$ for parameter data of the slaves



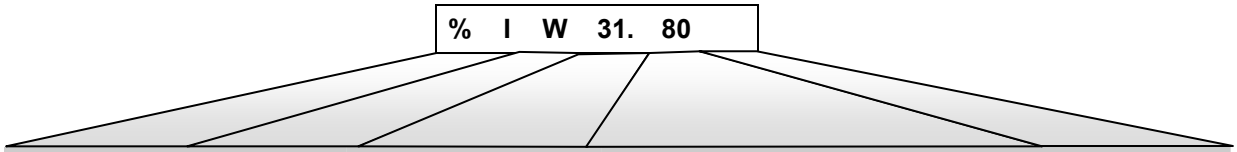
Data area	Type of access	Identification	IEC address index		—
I = input	W = word	31 = master 1	64….79	current parameters	not used
		32 = master 2	160…175	reflected parameters	

Modbus addresses for the slave list LAS (list of active slaves)

Modbus addresses **IEC addresses** (PLC in controllere)		**Bits** **AS-i slave addresses**															
Master 1	**Master 2**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
4365 %IW31.80	8461 %IW32.80	15(A)	14(A)	13(A)	12(A)	11(A)	10(A)	9(A)	8(A)	7(A)	6(A)	5(A)	4(A)	3(A)	2(A)	1(A)	0*)
4366 %IW31.81	8462 %IW32.81	31(A)	30(A)	29(A)	28(A)	27(A)	26(A)	25(A)	24(A)	23(A)	22(A)	21(A)	20(A)	19(A)	18(A)	17(A)	16(A)
4367 %IW31.82	8463 %IW32.82	15B	14B	13B	12B	11B	10B	9B	8B	7B	6B	5B	4B	3B	2B	1B	—
4368 %IW31.83	8464 %IW32.83	31B	30B	29B	28B	27B	26B	25B	24B	23B	22B	21B	20B	19B	18B	17B	16B

*) LAS has no slave 0, so these values are set to 0 by default!

IEC addresses in the PLC of the controllere for the slave list LAS (list of active slaves)

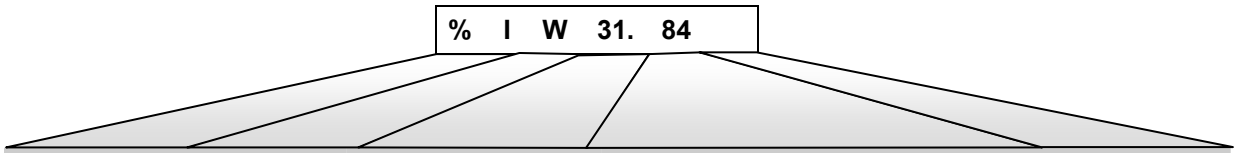


Data area	**Type of access**	**Identification**	**Slave selection**	**—**
I = input	W = word	31 = master 1 32 = master 2	80 = LAS of the slaves 1(A)…15(A) 81 = LAS of the slaves 16(A)…31(A) 82 = LAS of the slaves 1B…15B 83 = LAS of the slaves 16B…31B	not used

Modbus addresses for the slave list LDS (list of detected slaves)

Modbus addresses **IEC addresses** (PLC in controllere)		**Bits** **AS-i slave addresses**															
Master 1	**Master 2**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
4369 %IW31.84	8465 %IW32.84	15(A)	14(A)	13(A)	12(A)	11(A)	10(A)	9(A)	8(A)	7(A)	6(A)	5(A)	4(A)	3(A)	2(A)	1(A)	0
4370 %IW31.85	8466 %IW32.85	31(A)	30(A)	29(A)	28(A)	27(A)	26(A)	25(A)	24(A)	23(A)	22(A)	21(A)	20(A)	19(A)	18(A)	17(A)	16(A)
4371 %IW31.86	8467 %IW32.86	15B	14B	13B	12B	11B	10B	9B	8B	7B	6B	5B	4B	3B	2B	1B	—
4372 %IW31.87	8468 %IW32.87	31B	30B	29B	28B	27B	26B	25B	24B	23B	22B	21B	20B	19B	18B	17B	16B

IEC addresses in the PLC of the controllere for the slave list LDS (list of detected slaves)



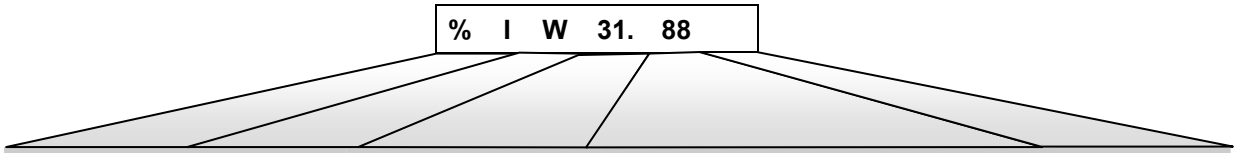
Data area	**Type of access**	**Identification**	**Slave selection**	**—**
I = input	W = word	31 = master 1 32 = master 2	84 = LDS of the slaves 1(A)…15(A) 85 = LDS of the slaves 16(A)…31(A) 86 = LDS of the slaves 1B…15B 87 = LDS of the slaves 16B…31B	not used

Modbus addresses for the slave list LPF (list of slaves with periphery faults)

Modbus addresses **IEC addresses** (PLC in controllere)		**Bits** **AS-i slave addresses**															
Master 1	**Master 2**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
4373 %IW31.88	8469 %IW32.88	15(A)	14(A)	13(A)	12(A)	11(A)	10(A)	9(A)	8(A)	7(A)	6(A)	5(A)	4(A)	3(A)	2(A)	1(A)	0*)
4374 %IW31.89	8470 %IW32.89	31(A)	30(A)	29(A)	28(A)	27(A)	26(A)	25(A)	24(A)	23(A)	22(A)	21(A)	20(A)	19(A)	18(A)	17(A)	16(A)
4375 %IW31.90	8471 %IW32.90	15B	14B	13B	12B	11B	10B	9B	8B	7B	6B	5B	4B	3B	2B	1B	—
4376 %IW31.91	8472 %IW32.91	31B	30B	29B	28B	27B	26B	25B	24B	23B	22B	21B	20B	19B	18B	17B	16B

*) LPF has no slave 0, so these values are set to 0 by default!

IEC addresses in the PLC of the controllere for the slave list LPF (list of slaves with periphery faults)



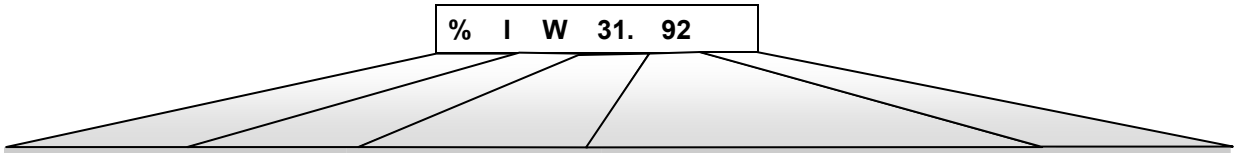
Data area	**Type of access**	**Identification**	**Slave selection**	**—**
I = input	W = word	31 = master 1 32 = master 2	88 = LPF of the slaves 1(A)…15(A) 89 = LPF of the slaves 16(A)…31(A) 90 = LPF of the slaves 1B…15B 91 = LPF of the slaves 16B…31B	not used

Modbus addresses for the slave list LPS (list of projected slaves)

Modbus addresses **IEC addresses** (PLC in controllere)		**Bits** **AS-i slave addresses**															
Master 1	**Master 2**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
4377 %IW3192	8473 %IW32.92	15(A)	14(A)	13(A)	12(A)	11(A)	10(A)	9(A)	8(A)	7(A)	6(A)	5(A)	4(A)	3(A)	2(A)	1(A)	0*)
4378 %IW31.93	8474 %IW32.93	31(A)	30(A)	29(A)	28(A)	27(A)	26(A)	25(A)	24(A)	23(A)	22(A)	21(A)	20(A)	19(A)	18(A)	17(A)	16(A)
4379 %IW31.94	8475 %IW32.94	15B	14B	13B	12B	11B	10B	9B	8B	7B	6B	5B	4B	3B	2B	1B	—
4380 %IW31.95	8476 %IW32.95	31B	30B	29B	28B	27B	26B	25B	24B	23B	22B	21B	20B	19B	18B	17B	16B

*) LPS has no slave 0, so these values are set to 0 by default!

IEC addresses in the PLC of the controllere for the slave list LPS (list of projected slaves)



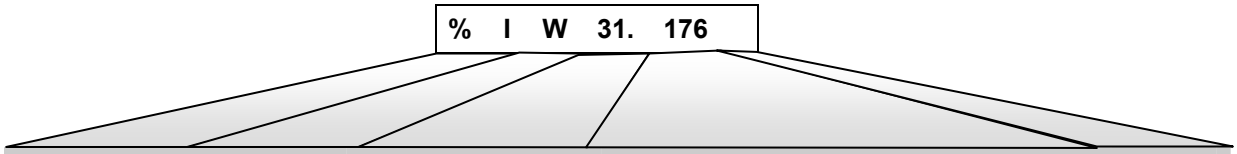
Data area	**Type of access**	**Identification**	**Slave selection**	**—**
I = input	W = word	31 = master 1 32 = master 2	92 = LPS of the slaves 1(A)…15(A) 93 = LPS of the slaves 16(A)…31(A) 94 = LPS of the slaves 1B…15B 95 = LPS of the slaves 16B…31B	not used

Modbus addresses for the slave telegram error counters

Modbus addresses		Telegram error counter of slave (16 bits = 1 word)	IEC addresses (PLC in controllere)	
Master 1	Master 2		Master 1	Master 2
4461	8557	1 (A)	%IW31.176	%IW32.176
4462	8558	2 (A)	%IW31.177	%IW32.177
4463	8559	3 (A)	%IW31.178	%IW32.178
4464	8560	4 (A)	%IW31.179	%IW32.179
4465	8561	5 (A)	%IW31.180	%IW32.180
4466	8562	6 (A)	%IW31.181	%IW32.181
4467	8563	7 (A)	%IW31.182	%IW32.182
4468	8564	8 (A)	%IW31.183	%IW32.183
4469	8565	9 (A)	%IW31.184	%IW32.184
4470	8566	10 (A)	%IW31.185	%IW32.185
4471	8567	11 (A)	%IW31.186	%IW32.186
4472	8568	12 (A)	%IW31.187	%IW32.187
4473	8569	13 (A)	%IW31.188	%IW32.188
4474	8570	14 (A)	%IW31.189	%IW32.189
4475	8571	15 (A)	%IW31.190	%IW32.190
4476	8572	16 (A)	%IW31.191	%IW32.191
4477	8573	17 (A)	%IW31.192	%IW32.192
4478	8574	18 (A)	%IW31.193	%IW32.193
4479	8575	19 (A)	%IW31.194	%IW32.194
4480	8576	20 (A)	%IW31.195	%IW32.195
4481	8577	21 (A)	%IW31.196	%IW32.196
4482	8578	22 (A)	%IW31.197	%IW32.197
4483	8579	23 (A)	%IW31.198	%IW32.198
4484	8580	24 (A)	%IW31.199	%IW32.199
4485	8581	25 (A)	%IW31.200	%IW32.200
4486	8582	26 (A)	%IW31.201	%IW32.201
4487	8583	27 (A)	%IW31.202	%IW32.202
4488	8584	28 (A)	%IW31.203	%IW32.203
4489	8585	29 (A)	%IW31.204	%IW32.204
4490	8586	30 (A)	%IW31.205	%IW32.205
4491	8587	31 (A)	%IW31.206	%IW32.206
4492	8588	1 B	%IW31.207	%IW32.207
4493	8589	2 B	%IW31.208	%IW32.208
4494	8590	3 B	%IW31.209	%IW32.209
4495	8591	4 B	%IW31.210	%IW32.210
4496	8592	5 B	%IW31.211	%IW32.211
4497	8593	6 B	%IW31.212	%IW32.212
4498	8594	7 B	%IW31.213	%IW32.213
4499	8595	8 B	%IW31.214	%IW32.214
4500	8596	9 B	%IW31.215	%IW32.215
4501	8597	10 B	%IW31.216	%IW32.216

Modbus addresses		Telegram error counter of slave (16 bits = 1 word)	IEC addresses (PLC in controller$_E$)	
Master 1	Master 2		Master 1	Master 2
4502	8598	11 B	%IW31.217	%IW32.217
4503	8599	12 B	%IW31.218	%IW32.218
4504	8600	13 B	%IW31.219	%IW32.219
4505	8601	14 B	%IW31.220	%IW32.220
4506	8602	15 B	%IW31.221	%IW32.221
4507	8603	16 B	%IW31.222	%IW32.222
4508	8604	17 B	%IW31.223	%IW32.223
4509	8605	18 B	%IW31.224	%IW32.224
4510	8606	19 B	%IW31.225	%IW32.225
4511	8607	20 B	%IW31.226	%IW32.226
4512	8608	21 B	%IW31.227	%IW32.227
4513	8609	22 B	%IW31.228	%IW32.228
4514	8610	23 B	%IW31.229	%IW32.229
4515	8611	24 B	%IW31.230	%IW32.230
4516	8612	25 B	%IW31.231	%IW32.231
4517	8613	26 B	%IW31.2xx	%IW32.2xx
4518	8614	27 B	%IW31.233	%IW32.233
4519	8615	28 B	%IW31.234	%IW32.234
4520	8616	29 B	%W31.235	%W32.235
4521	8617	30 B	%IW31.236	%IW32.236
4522	8618	31 B	%IW31.237	%IW32.237

IEC addresses in the PLC of the controller$_E$ for the slave telegram error counter

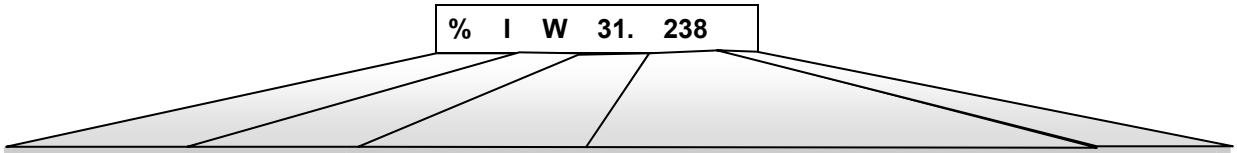


Data area	Type of access	Identification	IEC address index	—
I = input	W = word	31 = master 1 32 = master 2	176…237 = error counter of the slaves 1(A)…31B	not used

Modbus addresses for the configuration error counter

Modbus addresses		Configuration error counter of AS-i master (16 bits = 1 word)	IEC addresses (PLC in controller$_E$)	
Master 1	Master 2		Master 1	Master 2
4523	8619	configuration error counter of AS-i master	%IW31.238	%IW32.238

IEC addresses in the PLC of the controller$_E$ for the configuration error counter

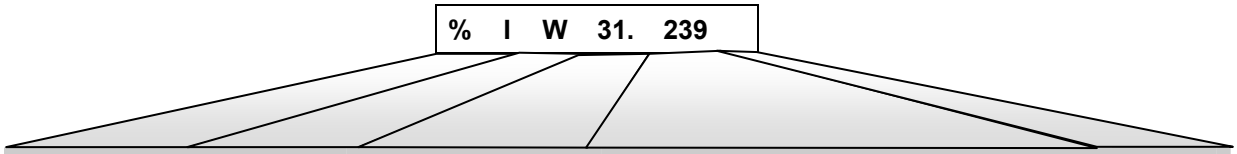


Data area	Type of access	Identification	IEC address index	—
I = input	W = word	31 = master 1 32 = master 2	238 = configuration error counter	not used

Modbus addresses for the AS-i cycle counter

Modbus addresses		AS-i cycle counter of AS-i master (16 bits = 1 word)	IEC addresses (PLC in controller$_E$)	
Master 1	Master 2		Master 1	Master 2
4524	8620	AS-i cycle counter of AS-i master	%IW31.239	%IW32.239

IEC addresses in the PLC of the controller$_E$ for the AS-i cycle counter



Data area	Type of access	Identification	IEC address index	—
I = input	W = word	31 = master 1 32 = master 2	239 = AS-i cycle counter	not used

Modbus addresses for the request data of the host command channel

Modbus addresses		Bit															
Master 1	**Master 2**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
4794	8890	echo byte of the request										request / status					
4795	8891	command code															
4796	8892	data (0)															
4797	8893	data (1)															
4798	8894	data (2)															
4799	8895	data (3)															
4800	8896	data (4)															
4801	8897	data (5)															
4802	8898	data (6)															
4803	8899	data (7)															
4804	8900	data (8)															
4805	8901	data (9)															
4806	8902	data (10)															
4807	8903	data (11)															
4808	8904	data (12)															
4809	8905	data (13)															
4810	8906	data (14)															
4811	8907	data (15)															
4812	8908	reserved															

Modbus addresses for the response data of the host command channel

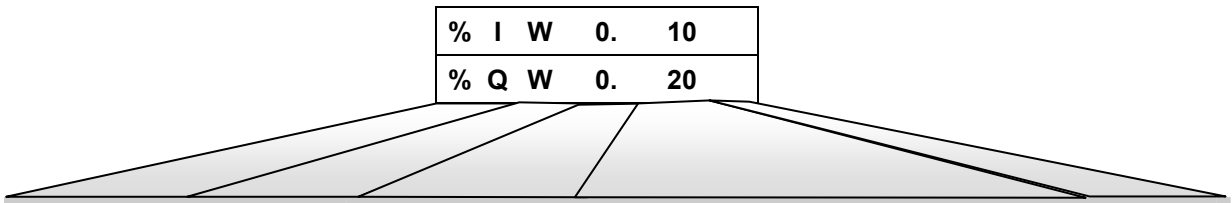
Modbus addresses		Bit															
Master 1	**Master 2**	**15**	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
4813	8909	echo byte of the response											status				
4814	8910	command code															
4815	8911	data (0)															
4816	8912	data (1)															
4817	8913	data (2)															
4818	8914	data (3)															
4819	8915	data (4)															
4820	8916	data (5)															
4821	8917	data (6)															
4822	8918	data (7)															
4823	8919	data (8)															
4824	8920	data (9)															
4825	8921	data (10)															
4826	8922	data (11)															
4827	8923	data (12)															
4828	8924	data (13)															
4829	8925	data (14)															
4830	8926	data (15)															
4831	8927	reserved															

Modbus addresses for the fieldbus data from/to the PLC of the controller

Modbus addresses		Data content (16 bits = 1 word)	IEC addresses (PLC in controller)	
Data to the PLC	Data from the PLC		Data to the PLC	Data from the PLC
12288	12352	user-defined	%IW0.0	%QW0.0
12289	12353	user-defined	%IW0.1	%QW0.1
12290	12354	user-defined	%IW0.2	%QW0.2
12291	12355	user-defined	%IW0.3	%QW0.3
12292	12356	user-defined	%IW0.4	%QW0.4
12293	12357	user-defined	%IW0.5	%QW0.5
12294	12358	user-defined	%IW0.6	%QW0.6
12295	12359	user-defined	%IW0.7	%QW0.7
12296	12360	user-defined	%IW0.8	%QW0.8
12297	12361	user-defined	%IW0.9	%QW0.9
12298	12362	user-defined	%IW0.10	%QW0.10
12299	12363	user-defined	%IW0.11	%QW0.11
12300	12364	user-defined	%IW0.12	%QW0.12
12301	12365	user-defined	%IW0.13	%QW0.13
12302	12366	user-defined	%IW0.14	%QW0.14
12303	12367	user-defined	%IW0.15	%QW0.15
12304	12368	user-defined	%IW0.16	%QW0.16
12305	12369	user-defined	%IW0.17	%QW0.17
12306	12370	user-defined	%IW0.18	%QW0.18
12307	12371	user-defined	%IW0.19	%QW0.19
12308	12372	user-defined	%IW0.20	%QW0.20
12309	12373	user-defined	%IW0.21	%QW0.21
12310	12374	user-defined	%IW0.22	%QW0.22
12311	12375	user-defined	%IW0.23	%QW0.23
12312	12376	user-defined	%IW0.24	%QW0.24
12313	12377	user-defined	%IW0.25	%QW0.25
12314	12378	user-defined	%IW0.26	%QW0.26
12315	12379	user-defined	%IW0.27	%QW0.27
12316	12380	user-defined	%IW0.28	%QW0.28
12317	12381	user-defined	%IW0.29	%QW0.29
12318	12382	user-defined	%IW0.30	%QW0.30
12319	12383	user-defined	%IW0.31	%QW0.31
12320	12384	user-defined	%IW0.32	%QW0.32
12321	12385	user-defined	%IW0.33	%QW0.33
12322	12386	user-defined	%IW0.34	%QW0.34
12323	12387	user-defined	%IW0.35	%QW0.35
12324	12388	user-defined	%IW0.36	%QW0.36
12325	12389	user-defined	%IW0.37	%QW0.37
12326	12390	user-defined	%IW0.38	%QW0.38
12327	12391	user-defined	%IW0.39	%QW0.39
12328	12392	user-defined	%IW0.40	%QW0.40

Modbus addresses		Data content (16 bits = 1 word)	IEC addresses (PLC in controller$_E$)	
Data to the PLC	Data from the PLC		Data to the PLC	Data from the PLC
12329	12393	user-defined	%IW0.41	%QW0.41
12330	12394	user-defined	%IW0.42	%QW0.42
12331	12395	user-defined	%IW0.43	%QW0.43
12332	12396	user-defined	%IW0.44	%QW0.44
12333	12397	user-defined	%IW0.45	%QW0.45
12334	12398	user-defined	%IW0.46	%QW0.46
12335	12399	user-defined	%IW0.47	%QW0.47
12336	12400	user-defined	%IW0.48	%QW0.48
12337	12401	user-defined	%IW0.49	%QW0.49
12338	12402	user-defined	%IW0.50	%QW0.50
12339	12403	user-defined	%IW0.51	%QW0.51
12340	12404	user-defined	%IW0.52	%QW0.52
12341	12405	user-defined	%IW0.53	%QW0.53
12342	12406	user-defined	%IW0.54	%QW0.54
12343	12407	user-defined	%IW0.55	%QW0.55
12344	12408	user-defined	%IW0.56	%QW0.56
12345	12409	user-defined	%IW0.57	%QW0.57
12346	12410	user-defined	%IW0.58	%QW0.58
12347	12411	user-defined	%IW0.59	%QW0.59
12348	12412	user-defined	%IW0.60	%QW0.60
12349	12413	user-defined	%IW0.61	%QW0.61
12350	12414	user-defined	%IW0.62	%QW0.62
12351	12415	user-defined	%IW0.63	%QW0.63

IEC addresses in the PLC of the controller$_E$ for the fieldbus data from/to the PLC of the controller$_E$



Data area	Type of access	Identification	IEC address index	—
I = input Q = output	W = word	0 (fixed value)	0…63 = word x of the data field	not used

Modbus addresses for the extended data from/to the PLC of the controller$_e$

Modbus addresses		Data content (16 bits = 1 word)	IEC addresses (PLC in controller$_e$)	
Data to the PLC	Data from the PLC		Data to the PLC	Data from the PLC
12416	12672	user-defined	%IW4.0	%QW4.0
12417	12673	user-defined	%IW4.1	%QW4.1
12418	12674	user-defined	%IW4.2	%QW4.2
12419	12675	user-defined	%IW4.3	%QW4.3
12420	12676	user-defined	%IW4.4	%QW4.4
12421	12677	user-defined	%IW4.5	%QW4.5
12422	12678	user-defined	%IW4.6	%QW4.6
12423	12679	user-defined	%IW4.7	%QW4.7
12424	12680	user-defined	%IW4.8	%QW4.8
12425	12681	user-defined	%IW4.9	%QW4.9
12426	12682	user-defined	%IW4.10	%QW4.10
12427	12683	user-defined	%IW4.11	%QW4.11
12428	12684	user-defined	%IW4.12	%QW4.12
12429	12685	user-defined	%IW4.13	%QW4.13
12430	12686	user-defined	%IW4.14	%QW4.14
12431	12687	user-defined	%IW4.15	%QW4.15
12432	12688	user-defined	%IW4.16	%QW4.16
12433	12689	user-defined	%IW4.17	%QW4.17
12434	12690	user-defined	%IW4.18	%QW4.18
12435	12691	user-defined	%IW4.19	%QW4.19
12436	12692	user-defined	%IW4.20	%QW4.20
12437	12693	user-defined	%IW4.21	%QW4.21
12438	12694	user-defined	%IW4.22	%QW4.22
12439	12695	user-defined	%IW4.23	%QW4.23
12440	12696	user-defined	%IW4.24	%QW4.24
12441	12697	user-defined	%IW4.25	%QW4.25
12442	12698	user-defined	%IW4.26	%QW4.26
12443	12699	user-defined	%IW4.27	%QW4.27
12444	12700	user-defined	%IW4.28	%QW4.28
12445	12701	user-defined	%IW4.29	%QW4.29
12446	12702	user-defined	%IW4.30	%QW4.30
12447	12703	user-defined	%IW4.31	%QW4.31
12448	12704	user-defined	%IW4.32	%QW4.32
12449	12705	user-defined	%IW4.33	%QW4.33
12450	12706	user-defined	%IW4.34	%QW4.34
12451	12707	user-defined	%IW4.35	%QW4.35
12452	12708	user-defined	%IW4.36	%QW4.36
12453	12709	user-defined	%IW4.37	%QW4.37
12454	12710	user-defined	%IW4.38	%QW4.38
12455	12711	user-defined	%IW4.39	%QW4.39
12456	12712	user-defined	%IW4.40	%QW4.40

Modbus addresses		Data content (16 bits = 1 word)	IEC addresses (PLC in controller$_e$)	
Data to the PLC	Data from the PLC		Data to the PLC	Data from the PLC
12457	12713	user-defined	%IW4.41	%QW4.41
12458	12714	user-defined	%IW4.42	%QW4.42
12459	12715	user-defined	%IW4.43	%QW4.43
12460	12716	user-defined	%IW4.44	%QW4.44
12461	12717	user-defined	%IW4.45	%QW4.45
12462	12718	user-defined	%IW4.46	%QW4.46
12463	12719	user-defined	%IW4.47	%QW4.47
12464	12720	user-defined	%IW4.48	%QW4.48
12465	12721	user-defined	%IW4.49	%QW4.49
12466	12722	user-defined	%IW4.50	%QW4.50
12467	12723	user-defined	%IW4.51	%QW4.51
12468	12724	user-defined	%IW4.52	%QW4.52
12469	12725	user-defined	%IW4.53	%QW4.53
12470	12726	user-defined	%IW4.54	%QW4.54
12471	12727	user-defined	%IW4.55	%QW4.55
12472	12728	user-defined	%IW4.56	%QW4.56
12473	12729	user-defined	%IW4.57	%QW4.57
12474	12730	user-defined	%IW4.58	%QW4.58
12475	12731	user-defined	%IW4.59	%QW4.59
12476	12732	user-defined	%IW4.60	%QW4.60
12477	12733	user-defined	%IW4.61	%QW4.61
12478	12734	user-defined	%IW4.62	%QW4.62
12479	12735	user-defined	%IW4.63	%QW4.63
12480	12736	user-defined	%IW4.64	%QW4.64
12481	12737	user-defined	%IW4.65	%QW4.65
12482	12738	user-defined	%IW4.66	%QW4.66
12483	12739	user-defined	%IW4.67	%QW4.67
12484	12740	user-defined	%IW4.68	%QW4.68
12485	12741	user-defined	%IW4.69	%QW4.69
12486	12742	user-defined	%IW4.70	%QW4.70
12487	12743	user-defined	%IW4.71	%QW4.71
12488	12744	user-defined	%IW4.72	%QW4.72
12489	12745	user-defined	%IW4.73	%QW4.73
12490	12746	user-defined	%IW4.74	%QW4.74
12491	12747	user-defined	%IW4.75	%QW4.75
12492	12748	user-defined	%IW4.76	%QW4.76
12493	12749	user-defined	%IW4.77	%QW4.77
12494	12750	user-defined	%IW4.78	%QW4.78
12495	12751	user-defined	%IW4.79	%QW4.79
12496	12752	user-defined	%IW4.80	%QW4.80
12497	12753	user-defined	%IW4.81	%QW4.81
12498	12754	user-defined	%IW4.82	%QW4.82

Modbus addresses		Data content (16 bits = 1 word)	IEC addresses (PLC in controller$_e$)	
Data to the PLC	Data from the PLC		Data to the PLC	Data from the PLC
12499	12755	user-defined	%IW4.83	%QW4.83
12500	12756	user-defined	%IW4.84	%QW4.84
12501	12757	user-defined	%IW4.85	%QW4.85
12502	12758	user-defined	%IW4.86	%QW4.86
12503	12759	user-defined	%IW4.87	%QW4.87
12504	12760	user-defined	%IW4.88	%QW4.88
12505	12761	user-defined	%IW4.89	%QW4.89
12506	12762	user-defined	%IW4.90	%QW4.90
12507	12763	user-defined	%IW4.91	%QW4.91
12508	12764	user-defined	%IW4.92	%QW4.92
12509	12765	user-defined	%IW4.93	%QW4.93
12510	12766	user-defined	%IW4.94	%QW4.94
12511	12767	user-defined	%IW4.95	%QW4.95
12512	12768	user-defined	%IW4.96	%QW4.96
12513	12769	user-defined	%IW4.97	%QW4.97
12514	12770	user-defined	%IW4.98	%QW4.98
12515	12771	user-defined	%IW4.99	%QW4.99
12516	12772	user-defined	%IW4.100	%QW4.100
12517	12773	user-defined	%IW4.101	%QW4.101
12518	12774	user-defined	%IW4.102	%QW4.102
12519	12775	user-defined	%IW4.103	%QW4.103
12520	12776	user-defined	%IW4.104	%QW4.104
12521	12777	user-defined	%IW4.105	%QW4.105
12522	12778	user-defined	%IW4.106	%QW4.106
12523	12779	user-defined	%IW4.107	%QW4.107
12524	12780	user-defined	%IW4.108	%QW4.108
12525	12781	user-defined	%IW4.109	%QW4.109
12526	12782	user-defined	%IW4.110	%QW4.110
12527	12783	user-defined	%IW4.111	%QW4.111
12528	12784	user-defined	%IW4.112	%QW4.112
12529	12785	user-defined	%IW4.113	%QW4.113
12530	12786	user-defined	%IW4.114	%QW4.114
12531	12787	user-defined	%IW4.115	%QW4.115
12532	12788	user-defined	%IW4.116	%QW4.116
12533	12789	user-defined	%IW4.117	%QW4.117
12534	12790	user-defined	%IW4.118	%QW4.118
12535	12791	user-defined	%IW4.119	%QW4.119
12536	12792	user-defined	%IW4.120	%QW4.120
12537	12793	user-defined	%IW4.121	%QW4.121
12538	12794	user-defined	%IW4.122	%QW4.122
12539	12795	user-defined	%IW4.123	%QW4.123
12540	12796	user-defined	%IW4.124	%QW4.124

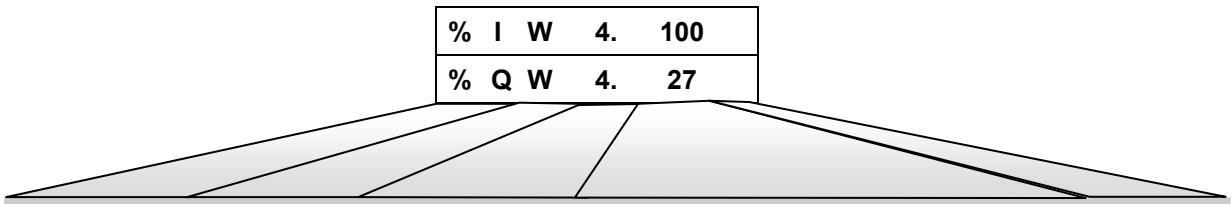
Modbus addresses		Data content (16 bits = 1 word)	IEC addresses (PLC in controller$_e$)	
Data to the PLC	Data from the PLC		Data to the PLC	Data from the PLC
12541	12797	user-defined	%IW4.125	%QW4.125
12542	12798	user-defined	%IW4.126	%QW4.126
12543	12799	user-defined	%IW4.127	%QW4.127
12544	12800	user-defined	%IW4.128	%QW4.128
12545	12801	user-defined	%IW4.129	%QW4.129
12546	12802	user-defined	%IW4.130	%QW4.130
12547	12803	user-defined	%IW4.131	%QW4.131
12548	12804	user-defined	%IW4.132	%QW4.132
12549	12805	user-defined	%IW4.133	%QW4.133
12550	12806	user-defined	%IW4.134	%QW4.134
12551	12807	user-defined	%IW4.135	%QW4.135
12552	12808	user-defined	%IW4.136	%QW4.136
12553	12809	user-defined	%IW4.137	%QW4.137
12554	12810	user-defined	%IW4.138	%QW4.138
12555	12811	user-defined	%IW4.139	%QW4.139
12556	12812	user-defined	%IW4.140	%QW4.140
12557	12813	user-defined	%IW4.141	%QW4.141
12558	12814	user-defined	%IW4.142	%QW4.142
12559	12815	user-defined	%IW4.143	%QW4.143
12560	12816	user-defined	%IW4.144	%QW4.144
12561	12817	user-defined	%IW4.145	%QW4.145
12562	12818	user-defined	%IW4.146	%QW4.146
12563	12819	user-defined	%IW4.147	%QW4.147
12564	12820	user-defined	%IW4.148	%QW4.148
12565	12821	user-defined	%IW4.149	%QW4.149
12566	12822	user-defined	%IW4.150	%QW4.150
12567	12823	user-defined	%IW4.151	%QW4.151
12568	12824	user-defined	%IW4.152	%QW4.152
12569	12825	user-defined	%IW4.153	%QW4.153
12570	12826	user-defined	%IW4.154	%QW4.154
12571	12827	user-defined	%IW4.155	%QW4.155
12572	12828	user-defined	%IW4.156	%QW4.156
12573	12829	user-defined	%IW4.157	%QW4.157
12574	12830	user-defined	%IW4.158	%QW4.158
12575	12831	user-defined	%IW4.159	%QW4.159
12576	12832	user-defined	%IW4.160	%QW4.160
12577	12833	user-defined	%IW4.161	%QW4.161
12578	12834	user-defined	%IW4.162	%QW4.162
12579	12835	user-defined	%IW4.163	%QW4.163
12580	12836	user-defined	%IW4.164	%QW4.164
12581	12837	user-defined	%IW4.165	%QW4.165
12582	12838	user-defined	%IW4.166	%QW4.166

Modbus addresses		Data content (16 bits = 1 word)	IEC addresses (PLC in controller$_e$)	
Data to the PLC	Data from the PLC		Data to the PLC	Data from the PLC
12583	12839	user-defined	%IW4.167	%QW4.167
12584	12840	user-defined	%IW4.168	%QW4.168
12585	12841	user-defined	%IW4.169	%QW4.169
12586	12842	user-defined	%IW4.170	%QW4.170
12587	12843	user-defined	%IW4.171	%QW4.171
12588	12844	user-defined	%IW4.172	%QW4.172
12589	12845	user-defined	%IW4.173	%QW4.173
12590	12846	user-defined	%IW4.174	%QW4.174
12591	12847	user-defined	%IW4.175	%QW4.175
12592	12848	user-defined	%IW4.176	%QW4.176
12593	12849	user-defined	%IW4.177	%QW4.177
12594	12850	user-defined	%IW4.178	%QW4.178
12595	12851	user-defined	%IW4.179	%QW4.179
12596	12852	user-defined	%IW4.180	%QW4.180
12597	12853	user-defined	%IW4.181	%QW4.181
12598	12854	user-defined	%IW4.182	%QW4.182
12599	12855	user-defined	%IW4.183	%QW4.183
12600	12856	user-defined	%IW4.184	%QW4.184
12601	12857	user-defined	%IW4.185	%QW4.185
12602	12858	user-defined	%IW4.186	%QW4.186
12603	12859	user-defined	%IW4.187	%QW4.187
12604	12860	user-defined	%IW4.188	%QW4.188
12605	12861	user-defined	%IW4.189	%QW4.189
12606	12862	user-defined	%IW4.190	%QW4.190
12607	12863	user-defined	%IW4.191	%QW4.191
12608	12864	user-defined	%IW4.192	%QW4.192
12609	12865	user-defined	%IW4.193	%QW4.193
12610	12866	user-defined	%IW4.194	%QW4.194
12611	12867	user-defined	%IW4.195	%QW4.195
12612	12868	user-defined	%IW4.196	%QW4.196
12613	12869	user-defined	%IW4.197	%QW4.197
12614	12870	user-defined	%IW4.198	%QW4.198
12615	12871	user-defined	%IW4.199	%QW4.199
12616	12872	user-defined	%IW4.200	%QW4.200
12617	12873	user-defined	%IW4.201	%QW4.201
12618	12874	user-defined	%IW4.202	%QW4.202
12619	12875	user-defined	%IW4.203	%QW4.203
12620	12876	user-defined	%IW4.204	%QW4.204
12621	12877	user-defined	%IW4.205	%QW4.205
12622	12878	user-defined	%IW4.206	%QW4.206
12623	12879	user-defined	%IW4.207	%QW4.207
12624	12880	user-defined	%IW4.208	%QW4.208

Modbus addresses		Data content (16 bits = 1 word)	IEC addresses (PLC in controller$_e$)	
Data to the PLC	Data from the PLC		Data to the PLC	Data from the PLC
12625	12881	user-defined	%IW4.209	%QW4.209
12626	12882	user-defined	%IW4.210	%QW4.210
12627	12883	user-defined	%IW4.211	%QW4.211
12628	12884	user-defined	%IW4.212	%QW4.212
12629	12885	user-defined	%IW4.213	%QW4.213
12630	12886	user-defined	%IW4.214	%QW4.214
12631	12887	user-defined	%IW4.215	%QW4.215
12632	12888	user-defined	%IW4.216	%QW4.216
12633	12889	user-defined	%IW4.217	%QW4.217
12634	12890	user-defined	%IW4.218	%QW4.218
12635	12891	user-defined	%IW4.219	%QW4.219
12636	12892	user-defined	%IW4.220	%QW4.220
12637	12893	user-defined	%IW4.221	%QW4.221
12638	12894	user-defined	%IW4.222	%QW4.222
12639	12895	user-defined	%IW4.223	%QW4.223
12640	12896	user-defined	%IW4.224	%QW4.224
12641	12897	user-defined	%IW4.225	%QW4.225
12642	12898	user-defined	%IW4.226	%QW4.226
12643	12899	user-defined	%IW4.227	%QW4.227
12644	12900	user-defined	%IW4.228	%QW4.228
12645	12901	user-defined	%IW4.229	%QW4.229
12646	12902	user-defined	%IW4.230	%QW4.230
12647	12903	user-defined	%IW4.231	%QW4.231
12648	12904	user-defined	%IW4.232	%QW4.232
12649	12905	user-defined	%IW4.233	%QW4.233
12650	12906	user-defined	%IW4.234	%QW4.234
12651	12907	user-defined	%IW4.235	%QW4.235
12652	12908	user-defined	%IW4.236	%QW4.236
12653	12909	user-defined	%IW4.237	%QW4.237
12654	12910	user-defined	%IW4.238	%QW4.238
12655	12911	user-defined	%IW4.239	%QW4.239
12656	12912	user-defined	%IW4.240	%QW4.240
12657	12913	user-defined	%IW4.241	%QW4.241
12658	12914	user-defined	%IW4.242	%QW4.242
12659	12915	user-defined	%IW4.243	%QW4.243
12660	12916	user-defined	%IW4.244	%QW4.244
12661	12917	user-defined	%IW4.245	%QW4.245
12662	12918	user-defined	%IW4.246	%QW4.246
12663	12919	user-defined	%IW4.247	%QW4.247
12664	12920	user-defined	%IW4.248	%QW4.248
12665	12921	user-defined	%IW4.249	%QW4.249
12666	12922	user-defined	%IW4.250	%QW4.250

Modbus addresses		Data content (16 bits = 1 word)	IEC addresses (PLC in controller$_e$)	
Data to the PLC	**Data from the PLC**		**Data to the PLC**	**Data from the PLC**
12667	12923	user-defined	%IW4.251	%QW4.251
12668	12924	user-defined	%IW4.252	%QW4.252
12669	12925	user-defined	%IW4.253	%QW4.253
12670	12926	user-defined	%IW4.254	%QW4.254
12671	12927	user-defined	%IW4.255	%QW4.255

IEC addresses in the PLC of the controller$_e$ for the extended data from/to the PLC of the controller$_e$



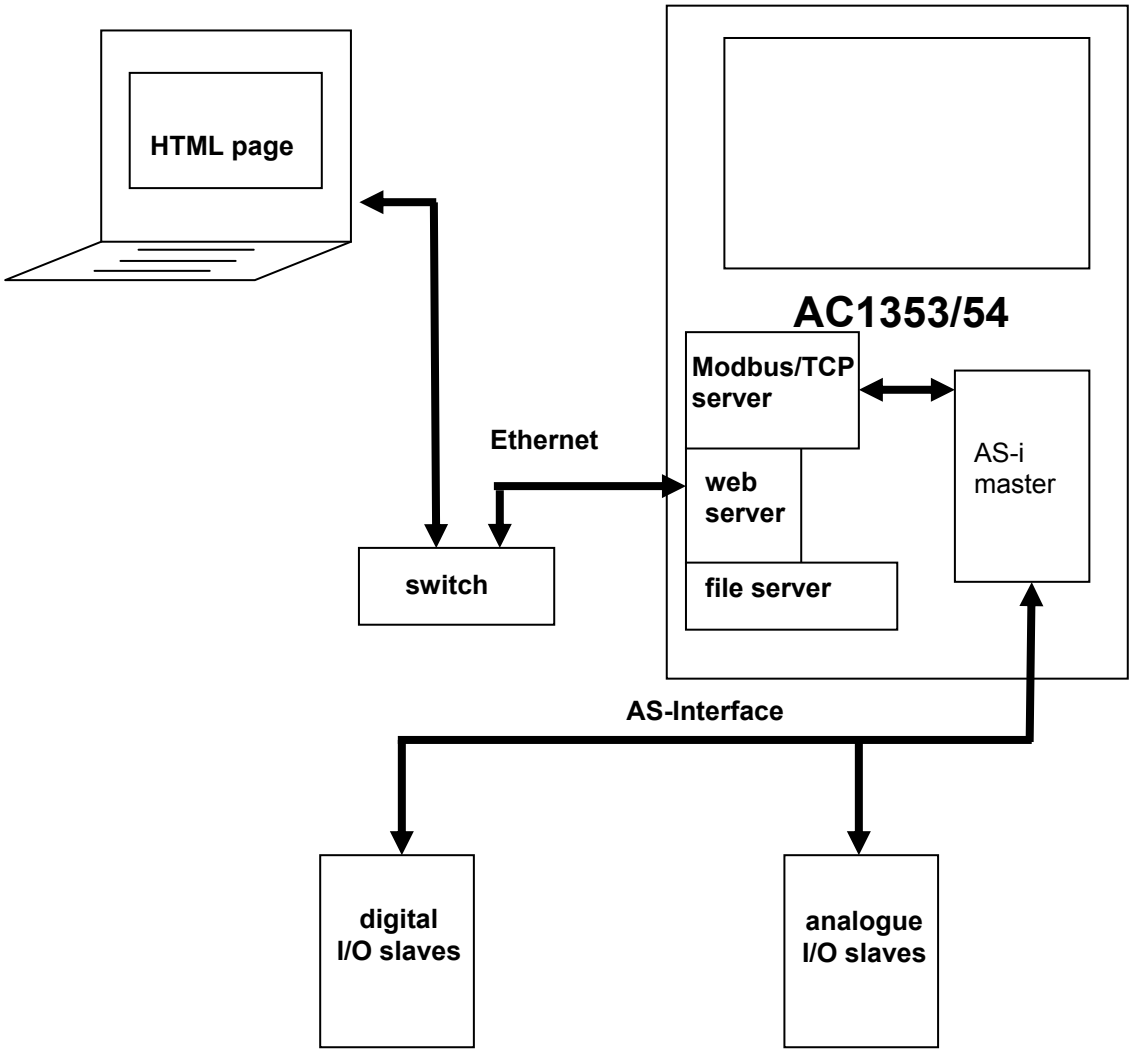
Data area	Type of access	Identification	IEC address index	—
I = input Q = output	W = word	4 (fixed value)	0…255 = word x of the data field	not used

4.5.4 Data exchange HTML page – controllere

Connection between		via	→ page
controllere	PC	point-to-point connection	4-4
controllere	controllere	network connection	4-18
controllere	client	MODBUS/TCP server / client	4-33
controllere HTML page	PC	HTML data exchange	here

Overview HTML data exchange

With an integrated web server HTML pages of the controllere can be represented on a PC via an Ethernet network by means of a standard browser. By integrating a Java applet in the HTML page data can dynamically be exchanged with the controllere. To do so, the applet uses the Modbus/TCP protocol.



Setting up an own web page

An own web page is to be realised on the controller. For this the following steps are required:

Step 1 ► Connect the devices via Ethernet (→ page 4-75)

Step 2 ► Set IP addresses and subnet mask (→ page 4-75)

Step 3 ► Open the HTML page in the browser (→ page 4-76)

Step 4 ► Address the file server via FTP (→ page 4-77)

Step 5 ► Edit the web page (→ page 4-79)

Step 6 ► Load and test the modified web page (→ page 4-81)

Here is the detailed description of the steps:

Step 1: Connect the devices via Ethernet

► Set up the Ethernet network by connecting the PC and the controller to the hub (or switch).

► To do so, use common CAT5 Ethernet patch cables with an RJ45 connector on both sides.

Step 2: Set IP addresses and subnet mask

► Set appropriate IP addresses and subnet masks on the controller and your PC (methode → page 4-5).

> **NOTE**
>
> In a local network the participants can only communicate if their IP addresses are from the same "family".
>
> Example: Subnet mask = 255.255.255.**0**
> Then the IP addresses of the first 3 address groups (where "255" is) must be identical for all participants. The IP address may (and must) only be different in the last block (where "0" is) (permitted values): 0…254).
>
> Ask the network administrator for the specifications!

In our example we assume the following values:
Subnet mask = 255.255.255.0
IP address of the controller = 192.168.10.11
IP address of the PC = 192.168.10.20

Set up the Ethernet network by connecting the PC and the controller to the switch.

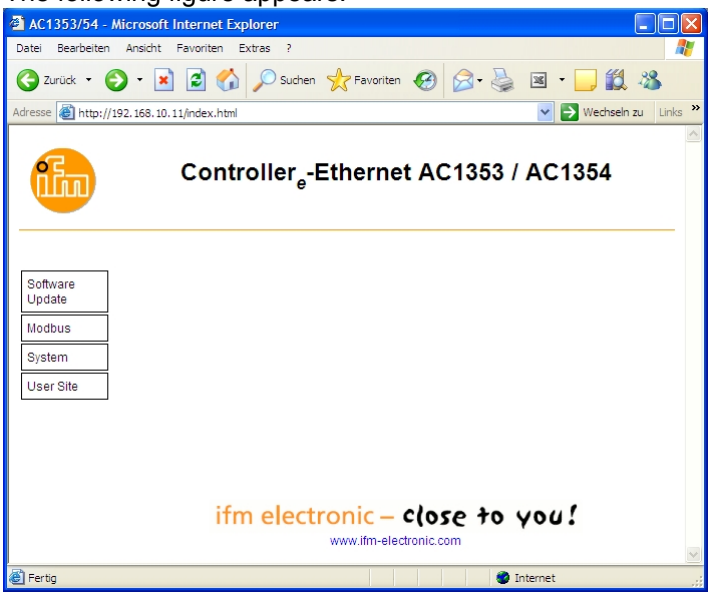
Step 3: Open the HTML page in the browser

On delivery, a start page is stored in the web server of the controller$_e$. This page is displayed when you access the IP address of the controller$_e$ with the HTTP protocol in a browser.

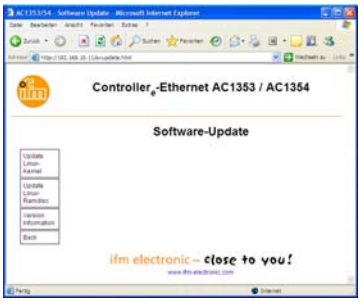
An **example** of calling the start page of the controller$_e$ with the IP address 192.168.10.11 is shown in the following figure.

► Enter "http://" and then the IP address of the controller$_e$ in the address bar.

> The following figure appears:

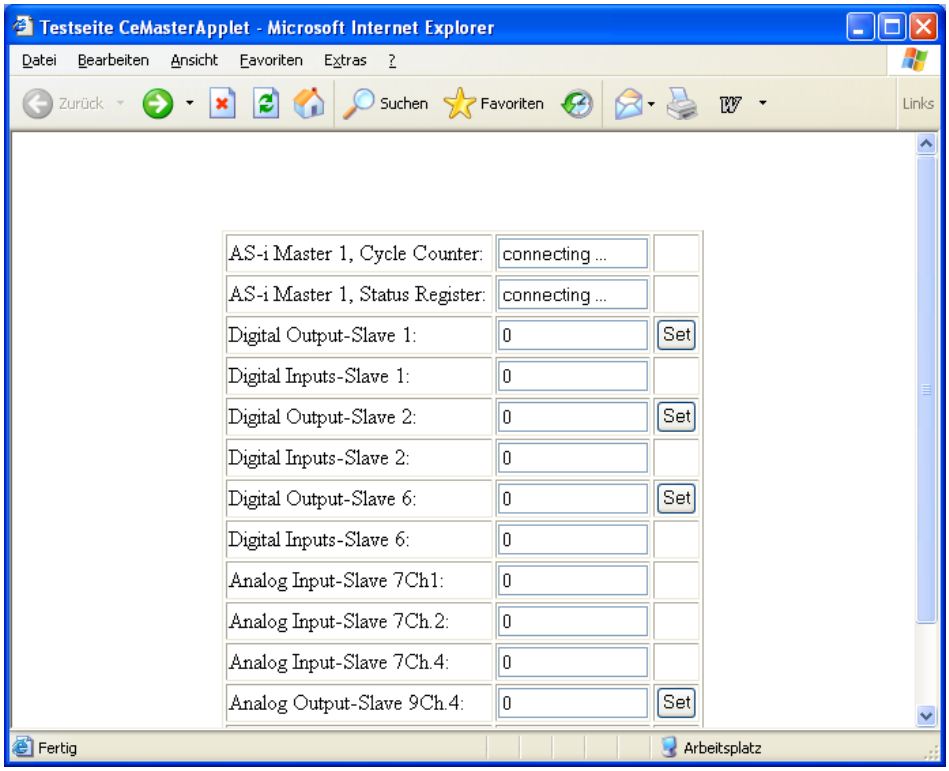


In this window:

A click on ...	enables ...	...in this window:
Software update	Update of the Ethernet driver software	
Modbus	Settings of the Modbus/TCP server	
User Site	Calling the HTML page In the following we will describe the adaptation of the contents:	

NOTE

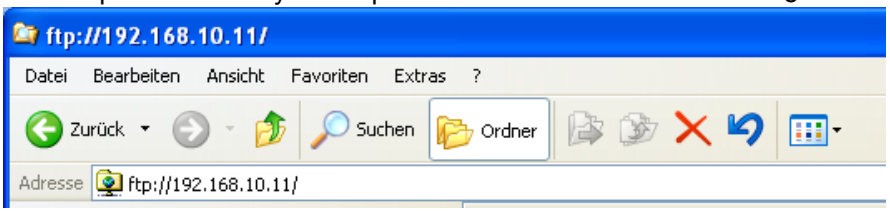
To allow the web page to be updated it must be possible to run Java applets in the browser (e.g. by Java 2 Runtime Environment 5.0).



Step 4: Address the file server via FTP

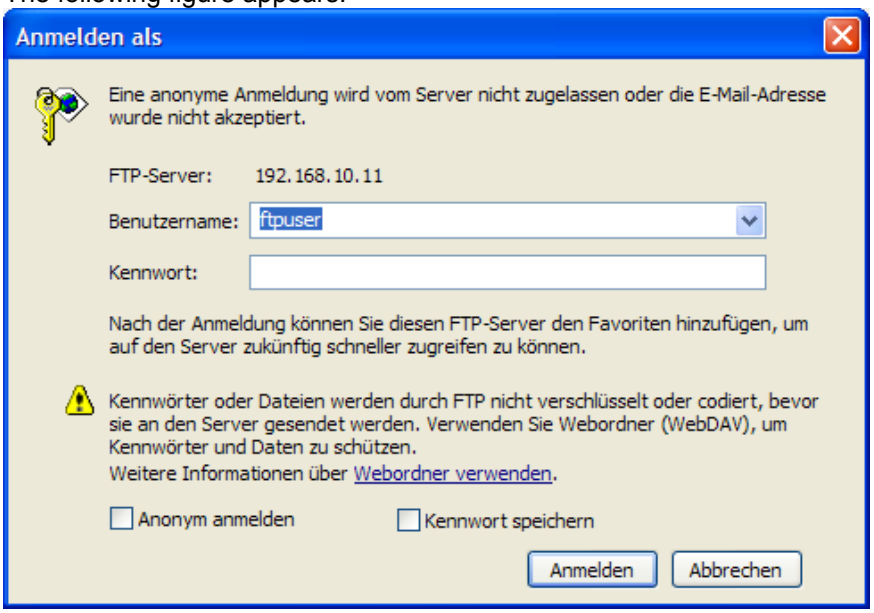
As shown in the following example the file server in the controller can be called via the browser / Explorer.

► Enter "ftp://" followed by the requested IP address of the controller as search address:



If RAM disk in the controllere <10.120:

> The following figure appears:



► User name = ftpuser
Do not enter a password

► Click [Login]

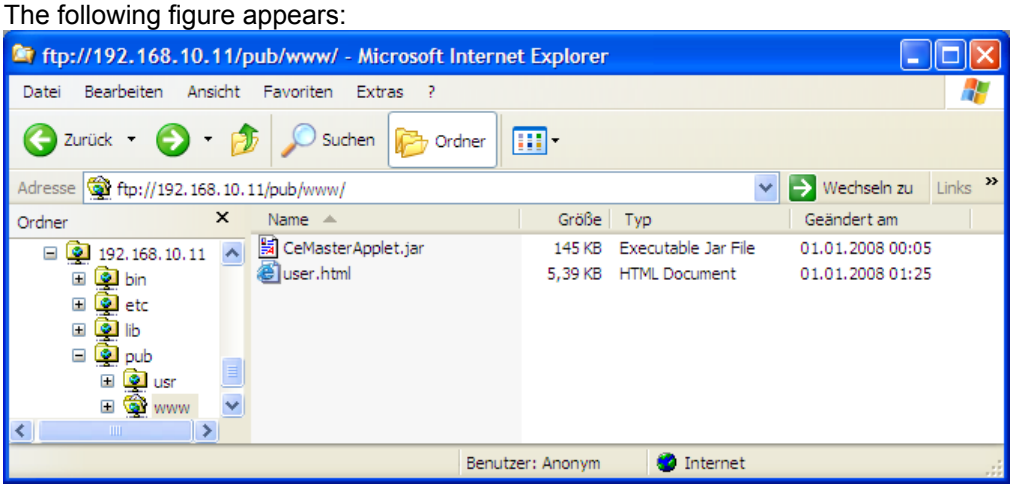
To all controllere devices the following applies:

> The browser opens a kind of Windows Explorer

► Open the directory `pub` under the requested IP address of the controllere

Then open the subdirectory `WWW`

> The following figure appears:



The file `user.html` contains the source code of the page accessed as example in step 3.
In the following you can adapt this page to your application.

Step 5: Edit the web page

► Right-click the file name `user.html`

► Select [Edit the source code]
In the following we describe the device-specific particularities of the HTML program code. Please find a description of the HTML orders in the appropriate technical literature.

> The following figure appears (detail):

```
<html>
<head>
       <title>Testseite CeMasterApplet</title>
</head>
<body  onLoad="LoadFct()">

<applet archive="CeMasterApplet.jar" code="CeMasterApplet" name="CeM" width="0"
height="0"><param name="DEBUG" value="0"><param name="UNITID" value="1"></applet>

<script LANGUAGE=javascript>
function LoadFct()
{
setTimeout("Timer()", 500);
}
```

etc.

In the HTML file an HTML tag <applet> must be indicated.
Detail from the above example:

```
<applet archive="CeMasterApplet.jar" code="CeMasterApplet"
name="CeM" width="0" height="0"> <param name="DEBUG" value="0">
<param name="UNITID" value="1">
</applet>
```

Here, the applet which is saved and available in the Java archive `CeMasterApplet.jar` is integrated in a web page.

`name="CeM"`	The object created by the applet is assigned the name "CeM"
`param name="UNITID"`	The parameter UNITID is passed to the applet
`value="1"`	UNITID is assigned the value 1

With this applet the user can read or write all registers of the Modbus register model (→ page 4-34) to display data on a web page or to enter values via the web page in the Modbus registers of the controllere.

Functions available in the applet

- `public int getUnitID()`
  With this function the `UnitID` of the Java applets can be read. The applet tries to address the controllere with this `UnitID`. If the `UnitID` of the applet and the controllere do not match, a Modbus connection is not set up.
- `public void setUnitID(int id)`
  `setUnitID` is used to change the `UnitID` of the applet.
- `public int readInputRegister(int ref)`
  `readInputRegister` reads the register "ref" of the Modbus register model of the controllere. In

case of a fault the value "-1" is returned. The content of the Modbus register is returned in the value range 0…65535.

- `public void writeSingleRegister(int ref, int value)`
  `writeSingleRegister` is used to write a register of a Modbus register model. With the parameter "`value`" the value to be written is transmitted. If the value is greater than 65535, the bits which are more significant are ignored. With the parameter "`ref`" the Modbus register to be written is identified.
- `public int readDigitalInputSlave(int slave)`
  `readDigitalInputSlave` reads the data of a digital input slave. This is a comfort function which makes it unnecessary for the user to extract the slave data from the register value. The 4-bit value of the corresponding input slave is directly delivered. The parameter "`slave`" must contain a slave address in the range of 0...62. The slave addresses 32...62 are used for B slaves.
- `public int writeDigitalOutputSlave(int slave, int value)`
  `writeDigitalOutputSlave` enables to write the outputs of a digital output slave. With the parameter "`slave`" a slave address in the range of 0...62 is transmitted. The addresses 32...62 are used to address B slaves. In the parameter "`value`" the value to be written is transmitted.
- `public void setDebugMode(int level)`
  With the `setDebugMode` the output of debug messages of the Java applet to the Java console can be controlled. For the parameter "`level`" the values 0...9 are possible. Outputs are activated by setting a bit in the parameter "`level`".
- `public int getDebugMode()`
  The function `getDebugMode` delivers the currently set debug level value.

Call the function cyclically

Function calls in the script are normally only processed when a page is loaded. To obtain a cyclic update of the data the function "`Timer`" recalls itself always at the end with a defined delay (here: 250 ms):

```
function LoadFct()
       {
              setTimeout("Timer()", 500); // Initial call of Timer()
       }

function Timer()             // List of all textboxes whose values shall be updated regularly
       {
       ... // Update of the data

       setTimeout("Timer()", 250);          //Restarts Timer() in 250 ms
       }
```

Step 6: Loading and testing the modified web page

There are two possibilities to open an application-specific web page:

- The page shall be accessible via the link [User-Site] from the supplied start page:
  ➔ A file `user.html` must be saved in the directory `/pub/www`.
- The web page shall replace the supplied start page:
  ➔ It must be saved as file `index.html` in the directoy `/pub/www`.

The page cannot be saved from the editor directly in the controllere. Therefore:

► First save the new page on the hard disk of the PC.

► Carry out the following steps to copy the user web page to the controllere:

- Login to the controllere by means of ftp → Step 4, page 4-77).
- Change to the directory `pub`.
- If it does not yet exist: create directory `www`
  Change to the directory `www`
- As required:
  Copy the file `user.html` or `index.html` from the PC to this directory.
- If `index.html` changed:
  Power the controllere off and on again

> **NOTE**
>
> After switching on the controllere it is checked whether a file `user.html` or `index.html` exists. If yes, a link to this file is created.
>
> If none of these both files exists, a link to a default page is created.

5 Menu

NOTE

In this manual the menu texts are all indicated in English.

Basic functions → separate basic instructions of the device manual

5.1 Menu "Ethernet Setup"“

Quick setting of the Ethernet programming interface, reading of the parameter data (password level 1 required).

Menu tree	Explanation
System Setup Ethernet Setup	> Display of the menu [Ethernet Setup] ► Menu selection with [▲] or [▼] and [OK] ► (Cancel with [ESC])
System Setup Ethernet Setup IP Address	> Display of the current IP address ► Setting of the IP addess block by block with [▲] or [▼] (only possible when DHCP = OFF) ► Confirm with [OK] ► (Cancel with [ESC])
System Setup Ethernet Setup Subnet Mask	> Display of the current subnet mask ► Setting of the subnet mask block by block [▲] or [▼] (only possible when DHCP = OFF) ► Confirm with [OK] ► (Cancel with [ESC])
System Setup Ethernet Setup Gateway Address	> Display of the current gateway address ► Setting of the gateway addess block by block with [▲] or [▼] ► Confirm with [OK] ► (Cancel with [ESC])
System Setup Ethernet Setup Baud rate	> Display of the current baud rate of the Ethernet interface ► Selection of the requested baud rate with [▲] or [▼]: ▪ 100MBd duplex (default setting) ▪ 100MBd simplex ▪ 10MBd duplex ▪ 10MBd simplex ► Confirm with [OK] ► (Cancel with [ESC])

Menu tree	Explanation
System Setup Ethernet Setup Auto Negotiation	> Display: automatic negotiation of the network connection parameters ► Selection: Use [▲] or [▼] to switch the function on or off. ► Confirm with [OK] ► (Cancel with [ESC])
System Setup Ethernet Setup DHCP Setup	> Display: assignment of the IP address by the host ► Selection: Use [▲] or [▼] to switch the function on or off. ► Confirm with [OK] ► (Cancel with [ESC])
System Setup Ethernet Setup MAC ID	> Display: manufacturer's identifier of the Ethernet interface ► Back with [ESC]

6 Operation

6.1 The Modbus command channel

In the Modbus address space a command channel with a length of 19 words is defined for each AS-i master. A Modbus TCP client operates as host system.

Modbus addresses			Contents	Access r=read w=write	Size [words]
Start		End			
dec.	hex.	dec.			
4794	12BA	4812	master 1 command channel request	r/w	19
4813	12CD	4831	master 1 command channel response	r	19
8890	22BA	8908	master 2 command channel request	r/w	19
8909	22CD	8927	master 2 command channel response	r	19

The commands are always triggered by the host by means of a corresponding entry in his output data area. The controllere responds then in the input data area of the host system.

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request							
2	reserved for string transfers								command number							
3...18	command data															
19	0#00															

Response from controllere:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	reflected user ID								command status							
2	reserved for string transfers								reflected command number							
3...18	command data															
19	0#00															

NOTE

If a command is to be executed, the value 0#65 must be entered in the command request. Changing the command number does not start the execution. If a command is to be executed several times, the user ID must be changed accordingly, e.g. by counting up. Before starting a command it should be verified in the command status whether the previous command has been completed.

The command status indicates the status of the command channel:

0#65 command request by the host

0#6A command is being processed

0#6B command aborted due to an error

0#6C abort after timeout during command processing

0#6D command completed, but response data not yet consistent

0#6E unknown command

0#6F command completed, response buffer is valid

6.1.1 Overview of the commands in the Modbus command channel

Command number		Description	→ page
Decimal	Hexadecimal		
0	0#00	no execution of a command	6-4
1	0#01	write parameters to a connected AS-i slave	6-5
3	0#03	adopt and save currently connected AS-i slaves in the configuration	6-7
4	0#04	change the list of the projected AS-i slaves (LPS)	6-9
5	0#05	set the operating mode of the AS-i master	6-11
6	0#06	readdress a connected AS-i slave	6-12
7	0#07	set the auto addressing mode of the AS-i master	6-14
9	0#09	change the extended ID code 1 in the connected AS-i slave	6-15
10...20	0#0A...0#14	force analogue data transmission directly to / from 3 AS-i slaves in each case	6-17
28	0#1C	deactivation of the slave reset when changing to the protected operation	6-24
31	0#1F	one-time execution of the "Extended safety monitor protocol" in the "Safety at work" monitor	6-25
21	0#15	read the ID string of an AS-i slave with profile S-7.4	6-21
33	0#21	read the diagnosis string of an AS-i slave with profile S-7.4	6-29
34	0#22	read the parameter string of an AS-i slave with profile S-7.4	6-31
35	0#23	write the parameter string of an AS-i slave with profile S-7.4	6-33
36	0#24	acyclic standard read call to an AS-i slave with CTT2 profile (S-7.5.5, S-7.A.5 or S-B.A.5) *– available from master profile M4 onwards -*	6-35
37	0#25	acyclic standard write call to an AS-i slave with CTT2 profile (S-7.5.5, S-7.A.5 or S-B.A.5) *- available from master profile M4 onwards -*	6-39
38	0#26	acyclic manufacturer-specific read call to an AS-i slave with CTT2 profile (S-7.5.5, S-7.A.5 or S-B.A.5) *- available from master profile M4 onwards -*	6-43

Command number		Description	→ page
Decimal	**Hexadecimal**		
39	0#27	acyclic manufacturer-specific read call to an AS-i slave with CTT2 profile (S-7.5.5, S-7.A.5 or S-B.A.5) *- available from master profile M4 onwards -*	6-47
50	0#32	read current configuration of AS-i slaves 0(A)...15(A)	6-51
51	0#33	read current configuration of AS-i slaves 16(A)...31(A)	
52	0#34	read current configuration of AS-i slaves (0)1B...15B	
53	0#35	read current configuration of AS-i slaves 16B...31B	
54	0#36	read current parameters of a connected AS-i slave	6-52
55	0#37	read current AS-i slaves lists	6-54
56	0#38	read projected configuration of the AS-i slaves 1(A)...15(A)	6-56
57	0#39	read projected configuration of the AS-i slaves 16(A)...31(A)	
58	0#3A	read projected configuration of the AS-i slaves (0)1B...15B	
59	0#3B	read projected configuration of the AS-i slaves 16B...31B	
96	0#60	save data non-volatilely in the flash memory of the controllere	6-57
97	0#61	carry out various settings in the controllere	6-58
102	0#66	retrieve the status of the controllere display	6-59
105	0#69	read the device properties of the controllere	6-61

Syntax and examples (values in hexadecimal representation) on the following pages.

6.1.2 Command 0 (0#00): no execution of a command

Request from host:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command request = 0#65							
2	0#00								command number = 0#00							
3…19	ignored								ignored							

Example:

1	0#0365	user ID changes to 0#03, command request with 0#65
2	0#0000	0#00 = command number 0
3…18	0#0000	not used

Response from controller:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command status = 0#6F							
2	reserved								reflected command number = 0#00							
3…19	ignored								ignored							

Example:

1	0#036F	user ID changes to 0#03, command status is "completed" = 0#6F (no error)
2	0#0300	0#00 = reflected command number 0
3…18	0#0000	not changed

6.1.3 Command 1 (0#01): Write parameters to a connected AS-i slave

Request from host:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command request = 0#65							
2	0#00								command number = 0#01							
3	ignored										A/B	AS-i slave address				
4	ignored												parameter value to be written			
5…19	ignored															

Legend:

A/B	Bit for addressing A or B slaves Length: 1 bit Permitted values: 0/1 Meaning: 0 = standard slave or A slave 1 = B slave (addition of 20_h or 32_d to the slave address)

Example:

1	0#0965	user ID changes to 0#09, command request with 0#65
2	0#0001	0#01 = command number 1
3	0#0024	slave address 4B
4	0#0003	parameter value to be written

Response from controller in the normal case:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#01							
3	ignored												parameter value read back			
4…17	ignored															
18…19	reserved															

Example:

1	0#096F	user ID changes to 0#09, command status is "completed" = 0#6F (no error)
2	0#0001	0#01 = reflected command number 1
3	0#0003	parameter value read back; might differ from the value to be written

Response from controller in the case of an error:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command status = 0#6B							
2	ignored								reflected command number = 0#01							
3	0#00								error code							

Possible error codes:

0#01	NOK	no slave response or master is in the offline mode when calling the command
0#0A	NA	slave is not in LAS
0#0B	ID	parameter or address invalid
0#14	IC	master is not in the normal mode

Example:

1	0#096B	user ID changes to 0#09, 0#6B = error during command execution
2	0#0001	0#01 = reflected command number 1
3	0#000A	error code 0#0A → slave is not in LAS

6.1.4 Command 3 (0#03): Adopt and save currently connected AS-i slaves in the configuration

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	ignored								command number = 0#03							
3…19	ignored															

Example:

1	0#0C65	user ID changes to 0#0C, command request with 0#65
2	0#0003	0#03 = command number 3
3…18	0#0000	not used

Response from controllere in the normal case:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#03							
3…19	ignored															

Example:

1	0#0C6F	user ID changes to 0#0C, command status is "completed" = 0#6F (no error)
2	0#0003	0#03 = reflected command number 3
3…18	0#0000	not changed

Response from controller in the case of an error:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command status = 0#6B							
2	0#00								reflected command number = 0#03							
3	0#00								error code							
4…19	ignored															

Possible error codes:

0#14	IC	master is not in the normal mode

Example:

1	0#0C6B	user ID changes to 0#0C, 0#6B = error during command execution
2	0#0003	0#03 = reflected command number 3
3	0#0014	error code 0#14 → master is not in the normal mode
4…18	0#0000	not changed

6.1.5 Command 4 0#04): List of the projected AS-i slaves (LPS)

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 0#04							
3	15(A)	14(A)	13(A)	12(A)	11(A)	10(A)	9(A)	8(A)	7(A)	6(A)	5(A)	4(A)	3(A)	2(A)	1(A)	res
4	31(A)	30(A)	29(A)	28(A)	27(A)	26(A)	25(A)	24(A)	23(A)	22(A)	21(A)	20(A)	19(A)	18(A)	17(A)	16(A)
5	15B	14B	13B	12B	11B	10B	9B	8B	7B	6B	5B	4B	3B	2B	1B	res
6	31B	30B	29B	28B	27B	26B	25B	24B	23B	22B	21B	20B	19B	18B	17B	16B
7…17	ignored															
18…19	reserved															

Example:

1	0#0265	user ID changes to 0#02, command request with 0#65
2	0#0004	0#04 = command number 4
3	0#003E	slaves 1 to 5 are to be projected
4	0#8000	slave 31(A) is to be projected
5	0#0002	slave 1B is to be projected
6	0#0001	slave 16B is to be projected

Response from controller in the normal case:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#04							

Example:

1	0#026F	user ID changes to 0#02, command status is "completed" = 0#6F (no error)
2	0#0004	0#04 = reflected command number 4

Response from controller in the case of an error:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command status = 0#6B							
2	0#00								reflected command number = 0#04							
3	ignored								error code							

Possible error codes:

0#14	IC	master is not in the configuration mode

Example:

1	0#026B	user ID changes to 0#02, 0#6B = error during command execution
2	0#0004	0#04 = reflected command number 4
3	0#0014	error code 0#14 → master not in the configuration mode

6.1.6 Command 5 (0#05): Set the operating mode of the AS-i master

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 0#05							
3	ignored								operating mode							
4…17	ignored															
18…19	reserved															

Example:

1	0#0165	user ID changes to 0#01, command request with 0#65
2	0#0005	0#05 = command number 5
3	0#0001	0#00 = activate the protected operation 0#01 = activate the configuration mode

Response from controller in the normal case:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#05							

Example:

1	0#016F	user ID changes to 0#01, command status is "completed" = 0#6F (no error)
2	0#0005	0#05 = reflected command number 5

Response from controller in the case of an error:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6B							
2	0#00								reflected command number = 0#05							
3	ignored								error code							

Possible error codes:

0#03	SD0	slave with address 0 connected

Example:

1	0#016B	user ID changes to 0#01, 0#6B = error during command execution
2	0#0005	0#05 = reflected command number 5
3	0#0003	error code 0#03 → slave with address 0 connected

6.1.7 Command 6 (0#06): Readdress a connected AS-i slave

Request from host:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command request = 0#65							
2	0#00								command number = 0#06							
3	ignored										A/B	old slave address				
4	ignored										A/B	new slave address				
5…17	ignored															
18…19	reserved															

Legend:

A/B	Bit for addressing A or B slaves Length: 1 bit Permitted values: 0/1 Meaning: 0 = standard slave or A slave 1 = B slave (addition of 20_h or 32_d to the slave address)

Example:

1	0#0865	user ID changes to 0#08, command request with 0#65
2	0#0006	0#06 = command number 6
3	0#0029	old slave address 9B
4	0#000B	new slave address 11A

Response from controller in the normal case:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#06							

Example:

1	0#086F	user ID changes to 0#08, command status is "completed" = 0#6F (no error)
2	0#0006	0#06 = reflected command number 6

Response from controller in the case of an error:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6B							
2	0#00								reflected command number = 0#06							
3	ignored								error code							

Possible error codes:

0#01	NOK	no slave response or master is in the offline mode when calling the command
0#02	SND	no slave with the old address found
0#03	SD0	slave with address 0 is connected
0#04	SD2	no slave with the new address found
0#05	DE	error when deleting the old address
0#06	RE	error when reading the IO configuration
0#07	SE	error when writing the new address or the extended ID code 1
0#08	AT	new address could only temporarily be saved
0#09	ET	extended ID code 1 could only be saved temporarily
0#0B	ID	parameter or address invalid
0#14	IC	master is not in the normal mode

Example:

1	0#086B	user ID changes to 0#08, 0#6B = error during command execution
2	0#0006	0#06 = reflected command number 6
3	0#0003	error code 0#03 → slave with address 0 connected

6.1.8 Command 7 (0#07): Set the auto address mode of the AS-i master

Request from host:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command request = 0#65							
2	0#00								command number = 0#07							
3	ignored								automatic addressing							
4…17	ignored															
18…19	reserved															

Example:

1	0#0465	user ID changes to 0#04, command request with 0#65
2	0#0007	0#07 = command number 7
3	0#0001	0#00 = automatic addressing is deactivated 0#01 = automatic addressing is possible

Response from controller:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#07							

Example:

1	0#046F	user ID changes to 0#04, command status is "completed" = 0#6F (no error)
2	0#0007	0#07 = reflected command number 7

6.1.9 Command 9 (0#09): Change the extended ID code 1 in the connected AS-i slave

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 0#09							
3	ignored										A/B	slave address				
4	ignored								new "extended ID code 1"							
5…17	ignored															
18…19	reserved															

Legend:

A/B	Bit for addressing A or B slaves Length: 1 bit Permitted values: 0/1 Meaning: 0 = standard slave or A slave 1 = B slave (addition of 20_h or 32_d to the slave address)

Example:

1	0#0F65	user ID changes to 0#0F, command request with 0#65
2	0#0009	0#09 = command number 9
3	0#0011	0#11 = slave address 17(A)
4	0#0008	new "extended ID code 1" = 8

Response from controller in the normal case:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#09							

Example:

1	0#0F6F	user ID changes to 0#0F, command status is "completed" = 0#6F (no error)
2	0#0009	0#09 = reflected command number 9

Response from controller in the case of an error:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6B							
2	0#00								reflected command number = 0#09							
3	0#00								error code							

Possible error codes:

0#01	NOK	no slave response or master is in the offline mode when calling the command
0#02	SND	no slave with the address found
0#03	SD0	slave with address 0 is connected
0#07	SE	error when writing the extended ID code 1
0#09	ET	extended ID code 1 could only be saved temporarily
0#0B	IA	address is invalid or: 2 slaves with address 0 detected

Example:

1	0#0F6B	user ID changes to 0#0F, 0#6B = error during command execution
2	0#0009	0#09 = reflected command number 9
3	0#0007	error code 0#07 → slave does not support the extended ID code

6.1.10 Command 10...20 (0#0A...0#14): Force analogue data transmission directly to / from 3 AS-i slaves in each case

Request from host:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command request = 0#65							
2	0#00								command number = 0#0A…0#14							
3	output data AS-i slave 1(A), channel 0															
4	output data AS-i slave 1(A), channel 1															
5	output data AS-i slave 1, channel 2 or output data AS-i slave 1B, channel 0															
6	output data AS-i slave 1, channel 3 or output data AS-i slave 1B, channel 1															
7	0#00								O3	V3	O2	V2	O1	V1	O0	V0
8	output data AS-i slave 2(A), channel 0															
9	output data AS-i slave 2(A), channel 1															
10	output data AS-i slave 2, channel 2 or output data AS-i slave 2B, channel 0															
11	output data AS-i slave 2, channel 3 or output data AS-i slave 2B, channel 1															
12	0#00								O3	V3	O2	V2	O1	V1	O0	V0
13	output data AS-i slave 3(A), channel 0															
14	output data AS-i slave 3(A), channel 1															
15	output data AS-i slave 3, channel 2 or output data AS-i slave 3B), channel 0															
16	output data AS-i slave 3, channel 3 or output data AS-i slave 3B, channel 1															
17	0#00								O3	V3	O2	V2	O1	V1	O0	V0
18…19	reserved															

Legend:

V0…V3	Valid: 0 = data invalid 1 = data valid Output data must be valid (V=1) to be enabled in the AS-i slave!
O0...O3	Overflow 0 = data is in the valid range 1 = data is in the invalid range (especially in case of input modules when the measuring range is not reached or exceeded)

Example:

1	0#0165	user ID changes to 0#01, command request with 0#65
2	0#000A	0#0A = command number 10
3	0#0169	output data AS-i slave 1, channel 0
4	0#0202	output data AS-i slave 1, channel 1
5	0#0395	output data AS-i slave 1, channel 2
6	0#1033	output data AS-i slave 1, channel 3
7	0#0055	overflow and valid bits for AS-i slave 1: 55_h = 0101 0101_b O3 = 0, V3 = 1, O2 = 0, V2 = 1, O1 = 0, V1 = 1, O0 = 0, V0 = 1
8	0#2009	output data AS-i slave 2, channel 0
9	0#2202	output data AS-i slave 2, channel 1
10	0#0195	output data AS-i slave 2, channel 2
11	0#1022	output data AS-i slave 2, channel 3
12	0#0055	overflow and valid bits for AS-i slave 2: 55_h = 0101 0101_b O3 = 0, V3 = 1, O2 = 0, V2 = 1, O1 = 0, V1 = 1, O0 = 0, V0 = 1
13	0#3339	output data AS-i slave 3, channel 0
14	0#1102	output data AS-i slave 3, channel 1
15	0#1953	output data AS-i slave 3, channel 2
16	0#1234	output data AS-i slave 3, channel 3
17	0#0055	overflow and valid bits for AS-i slave 3: 55_h = 0101 0101_b O3 = 0, V3 = 1, O2 = 0, V2 = 1, O1 = 0, V1 = 1, O0 = 0, V0 = 1

Response from controller:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#0A...0#14							
3	input data or reflected output data AS-i slave 1(A), channel 0															
4	input data or reflected output data AS-i slave 1(A), channel 1															
5	input data or reflected output data AS-i slave 1, channel 2 or input data or reflected output data AS-i slave 1B, channel 0															
6	input data or reflected output data AS-i slave 1, channel 3 or input data or reflected output data AS-i slave 1B, channel 1															
7	TIB	TOB	TIA	TOA	TVB	OVB	TVA	OVA	O3	V3	O2	V2	O1	V1	O0	V0
8	input data or reflected output data AS-i slave 2(A), channel 0															
9	input data or reflected output data AS-i slave 2(A), channel 1															
10	input data or reflected output data AS-i slave 2, channel 2 or input data or reflected output data AS-i slave 2B, channel 0															
11	input data or reflected output data AS-i slave 2, channel 3 or input data or reflected output data AS-i slave 2B, channel 1															
12	TIB	TOB	TIA	TOA	TVB	OVB	TVA	OVA	O3	V3	O2	V2	O1	V1	O0	V0
13	input data or reflected output data AS-i slave 3(A), channel 0															
14	input data or reflected output data AS-i slave 3(A), channel 1															
15	input data or reflected output data AS-i slave 3, channel 2 or input data or reflected output data AS-i slave 3B, channel 0															
16	input data or reflected output data AS-i slave 3, channel 3 or input data or reflected output data AS-i slave 3B, channel 1															
17	TIB	TOB	TIA	TOA	TVB	OVB	TVA	OVA	O3	V3	O2	V2	O1	V1	O0	V0

Legend:

OVA
Channel-independent data valid flag of the A slave / standard slave:
1 = within max. 3 seconds the slave requests new data (CTT1) or:
the slave has received new output values (CTT2...5)
0 = the last valid data transmission took place more than 3.5 s ago (TT1) or:
the slave has not received new output values (CTT2...5)

OVB
Channel-independent data valid flag of the B slave (from master profile M4 onwards):
1 = slave has received new output values
0 = the slave has not received new output values

Note: valid only for reflected output data

TVA
Channel-independent transmission valid flag of the A slave / standard slave:
1 = analogue data transfer is running
0 = transmission error or timeout occurred

TVB
Channel-independent transmission valid flag of the B slave (from master profile M4 onwards):
1 = analogue data transfer is running
0 = transmission error or timeout occurred

Note: since this flag evaluates the data transmission cycle which was last connected, the response is delayed by up to 140 ms.

From master profile M4 onwards:

TIA 1 = slave sends input data as bit pattern (16-bit length, without sign)

TIB 0 = slave sends input data as value (15-bit length, with sign)

TOA 1 = slave receives output data as bit pattern (16-bit length, without sign)

TOB 0 = slave receives output data as value (15-bit length, with sign)

Command number		Slaves		
Decimal	**Hexadecimal**			
10	0#0A	1	2	3
11	0#0B	4	5	6
12	0#0C	7	8	9
13	0#0D	10	11	12
14	0#0E	13	14	15
15	0#0F	16	17	18
16	0#10	19	20	21
17	0#11	22	23	24
18	0#12	25	26	27
19	0#13	28	29	30
20	0#14	31	–	–

Example:

1	0#016F	user ID changes to 0#01, command status is "completed" = 0#6F (no error)
2	0#000A	0#0A = reflected command number 10
3	0#3169	slave 1 is a 4-channel input slave: input data AS-i slave 1, channel 0
4	0#2202	input data AS-i slave 1, channel 1
5	0#1395	input data AS-i slave 1, channel 2
6	0#0033	input data AS-i slave 1, channel 3
7	0#0255	overflow and valid bits for AS-i slave 1: 0255_h = 0000 0010 0101 0101_b TVA = 1, OVA = 0, O3 = 0, V3 = 1, O2 = 0, V2 = 1, O1 = 0, V1 = 1, O0 = 0, V0 = 1
8	0#2229	slave 2 is a 2-channel input slave: input data AS-i slave 2, channel 0
9	0#2332	input data AS-i slave 2, channel 1
10	0#7FFF	no valid value for channel 2
11	0#7FFF	no valid value for channel 3
12	0#0205	overflow and valid bits for AS-i slave 2: 0205_h = 0000 0010 0000 0101_b TVA = 1, OVA = 0, O3 = 0, V3 = 0, O2 = 0, V2 = 0, O1 = 0, V1 = 1, O0 = 0, V0 = 1
13	0#3339	slave 3 is a 4-channel input slave: output data AS-i slave 3, channel 0
14	0#1102	output data AS-i slave 3, channel 1
15	0#1953	output data AS-i slave 3, channel 2
16	0#1234	output data AS-i slave 3, channel 3
17	0#0255	overflow and valid bits for AS-i slave 3: 0255_h = 0000 0010 0101 0101_b TVA = 1, OVA = 0, O3 = 0, V3 = 1, O2 = 0, V2 = 1, O1 = 0, V1 = 1, O0 = 0, V0 = 1

6.1.11 Command 21 (0#15): Read the ID string of an AS-i slave with profile S-7.4

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0	0	0	AS-i slave address					command number = 21 (0#15)							
3…19	ignored															

Example:

1	0#0265	user ID changes to 0#02, command request with 0#65
2	0#0315	slave address = 3, 0#15 = command number 21

Response from controllere in the normal case:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	TG	S	AS-i slave address					F	reflected command number = 0#15							
3	I/O	2D	DT-Start			DT-Count			Mux field			E type				
4	number of parameters to be read								EDT Read			reserved		diag	reserved	
5	EDT Write		reserved						number of parameters to be written							
6	device-specific information								manufacturer identification							
7…16	device-specific information								device-specific information							
18	reserved								number of bytes received							
19	reserved															

Legend:

S	Sequence bit Length: 1 bit Permitted values: 0/1 Meaning: 0 = data transmission completed. 1 = data transmission not yet completed, at least one more packet follows.
TG	Toggle Length: 1 bit Permitted values: 0/1 Meaning: 1 = value changes for each execution of the command
F	Error bit Length: 1 bit Permitted values: 0/1 Meaning: 0 = execution was error free 1 = an error occurred during execution, e.g. slave does not have the profile S-7.4
Mux field	Number of multiplexed data words Length: 3 bit Permitted values: 0...3 Meaning: number = Mux field + 1

E type	Characterises the slave concerning functionality and data structure Length: 5 bits Permitted values: 0...31 Meaning: 0 = reserved 1 = transmitted values are measured values 2 = transmitted values are 16 digital bit values 3 = normal operation in 4-bit mode (4E/4A) 4...31 = reserved
I/O	Direction of data for the devices with E type <> 3 Length: 1 bit Permitted values: 0/1 Meaning: 0 = input 1 = output
Number of parameters to be read	Number of bytes which can be read as parameter string Length: 8 bits Permitted values: 0...219 Meaning: 0 = no parameter string readable 1...219 = number of bytes
Number of parameters to be written	Number of bytes which can be written as parameter string Length: 8 bits Permitted values: 0…219 Meaning: 0 = no parameter string readable 1…219 = number of bytes
2D	Double data transfer possible (→ redundancy) Length: 1 bit Permitted values: 0/1 Meaning: 0 = simple data transfer
DT-Start	Start triple (information for the driver in the master)
DT-Count	Number of data triples (information for the driver in the master)
EDT Read	Reserved for later profiles
EDT Write	Reserved for later profiles
Diag	Slave supports the 7.4 diagnosis string Length: 1 bit Permitted values: 0/1 Meaning: 0 = diagnosis string is not supported 1 = diagnosis string is supported
Manufacturer identification	Defined manufacturer number assigned by AS-International
Device-specific information	As an option more bytes for the manufacturer-specific device description

Example:

1	0#026F	user ID changes to 0#02, command status is "completed" = 0#6F (no error)
2	0#0615 or 0#8615	06_h = 00**00011**0_b → slave address = 3 0#15 = reflected command number 21 the most significant bit changes after each execution
3	0#2D01	1st word of the ID string of slave 3
4	0#0203	2nd word of the ID string of slave 3
…	…	…
17	0#0008	in this case the device sends an ID string of 8-byte length

Response from controller in the case of an error:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6B							
2	0#00								reflected command number = 0#15							
3	0#00								error code							

Possible error codes:

0#0C	faulty S-7.4 protocol sequence
0#0D	S-7.4 protocol aborted (timeout)
0#0E	invalid AS-i slave address for the S-7.4 protocol (e.g. B slaves)
0#0F	AS-i slave has completed the S-7.4 string
0#10	AS-i S-7.4 no longer connected (no longer in LAS)
0#11	to this AS-i slave another S-7.4 transfer is already active
0#12	previous segmented S-7.4 transfer not yet completed
0#13	invalid S-7.4 data length
0#14	invalid S-7.4 command

Example:

1	0#026B	user ID changes to 0#02, 0#6B = error during command execution
2	0#0015	0#15 = reflected command number 21
3	0#0014	error code 0#14 → master is not in the normal mode

6.1.12 Command 28 (0#1C): Deactivate the slave reset when changing to the protected mode

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 28 (0#1C)							
3	ignored								with / without offline phase							
4…19	ignored															

Example:

1	0#0465	user ID changes to 0#04, command request with 0#65
2	0#001C	0#1C = command number 28
3	0#0001	0#00 = offline phase when changing to the protected mode 0#01 = no offline phase when changing to the protected mode

Response from controller:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#1							

Example:

1	0#046F	user ID changes to 0#04, command status is "completed" = 0#6F (no error)
2	0#001C	0#1C = reflected command number 28

6.1.13 Command 31 (0#1F): One-time execution of the "Extended Safety Monitor Protocol" in the "Safety at work" monitor

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 31 (0#1F)							
3	sub command								0	0	0	AS-i slave address ($1...31_{10}$)				
4…17	see sub command															
18	field number (0#00 / 0#01)								data length = 0#00							

Example:

1	0#0765	user ID changes to 0#07, command request with 0#65
2	0#001F	0#1F = command number 31
3	0#001E	sub command 0#00 = one-time execution of the "Extended safety monitor protocol" in the "Safety at work" monitor with the address 30 (0#1E)

Response from controller in the normal case:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#1F							
3	sub command = 0#00								0	0	0	AS-i slave address				
4	LEDs OSSD 2				LEDs OSSD 1				data call 1				data call 0			
5	OSSD2 not green								OSSD1 not green							
6	1st colour output circuit 1								1st module address output circuit 1							
7	2nd colour output circuit 1								2nd module address output circuit 1							
8	3rd colour output circuit 1								3rd module address output circuit 1							
9	4th colour output circuit 1								4th module address output circuit 1							
10	5th colour output circuit 1								5th module address output circuit 1							
11	6th colour output circuit 1								6th module address output circuit 1							
12	1st colour output circuit 2								1st module address output circuit 2							
13	2nd colour output circuit 2								2nd module address output circuit 2							
14	3rd colour output circuit 2								3rd module address output circuit 2							
15	4th colour output circuit 2								4th module address output circuit 2							
16	5th colour output circuit 2								5th module address output circuit 2							
17	6th colour output circuit 2								6th module address output circuit 2							
18	field number = 0/1								0#00							

Description of the different fields:

Word no. 4:

LEDs OSSD 1				LEDs OSSD 2				Meaning
15	14	13	12	11	10	9	8	
0	0	0	0	0	0	0	0	green: contacts of the output circuits closed
0	0	0	1	0	0	0	1	yellow: startup / restart disable active
0	0	1	0	0	0	1	0	yellow flashing or red: contacts of the output circuits open
0	0	1	1	0	0	1	1	red flashing: error on the level of the monitored AS-i components
0	1	x	x	0	1	x	x	reserved (x = any value)

data call 1				data call 0				Meaning
7	6	5	4	3	2	1	0	
0	0	0	0	0	0	0	0	protective operation; everything OK (output circuits which are not available, not configured or dependent are indicated as OK)
0	0	0	1	0	0	0	1	protective operation, output circuit 1 off
0	0	1	0	0	0	1	0	protective operation, output circuit 2 off
0	0	1	1	0	0	1	1	protective operation, both output circuits off
0	1	0	0	0	1	0	0	configuration operation: power on
0	1	0	1	0	1	0	1	configuration operation
0	1	1	0	0	1	1	0	not reserved / not defined
0	1	1	1	0	1	1	1	configuration operation: fatal device error, RESET or replacement of devices required
1	x	x	x	1	x	x	x	No current diagnosis information available, please wait.

Word no. 5:

OSSD2 not green			OSSD1 not green			Meaning
15..12	11	10...8	7…4	3	2...0	
reserved	0	0	reserved	0	0	no modules - responses of the data calls in the words 6..17 are not relevant
reserved	0	1...6	reserved	0	1...6	number of modules which are not green
reserved	0	7	reserved	0	7	more than 6 modules are not green

Word no. 6...17:

1st to 6th module address output circuit 1/2:
Indicates the index of the module of the configuration. The module address which was defined in the program ASIMON is indicated.

1st to 6th colour output circuit 1/2:

3	2	1	0	Meaning
0	0	0	0	green, continuous
0	0	0	1	green, flashing
0	0	1	0	yellow, continuous
0	0	1	1	yellow, flashing
0	1	0	0	red, continuous
0	1	0	1	red, flashing
0	1	1	0	grey, off

Example („Safety at work“ monitor has not switched):

1	0#076F	user ID changes to 0#07, command status is "completed" = 0#6F (no error)
2	0#001F	0#1F = reflected command number 31
3	0#001E	0#00 = reflected sub command 0; 0#1E = AS-i slave address 30
4	0#0000	green: contacts of the output circuits closed
5	0#0000	both output circuits green
6…17	0#xxxx	not relevant because 5th word = 0#0000
18	0#0100	field number = 1

Example ("Safety at work" monitor has switched):

1	0#076F	user ID changes to 0#07, command status is "completed" = 0#6F (no error)
2	0#001F	0#1F = reflected command number 31
3	0#001E	0#00 = reflected sub command 0 0#1E = AS-i slave address 30
4	0#0211	0#0xxx = output circuit 2 green 0#x2xx = output circuit 1 red 0#xx11 = protective operation, output circuit 1 off (in both data calls)
5	0#0003	result from 4th word = OSSD2 green; OSSD1 not green 0#03 = delivers 3 modules which are not green
6	0#0421	module 33 (0#21) red, continuous (0#04)
7	0#0422	module 34 (0#22) red, continuous (0#04)
8	0#0423	module 35 (0#23) red, continuous (0#04)
9…11	0#xxxx	not relevant because low byte from 5th word = 0#03 → 3 modules relevant
12…17	0#xxxx	not relevant because high byte from 5th word = 0#00: green → no module relevant
18	0#0100	field number = 1

Response from controller in the case of an error:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6B							
2	0#00								reflected command number = 0#1F							
3	0#00								error code							

Possible error codes:

0#00… 0#02	general errors during command processing
0#0A… 0#-C	internal protocol error
0#10	sub command invalid
0#11	no slave with the profile S-7.F.F on the slave address
0#16	the protocol mode of the monitor at the address was changed
0#20	it was not possible to process the command within the specified time
0#EE	fatal error during command execution

Example:

1	0#076B	user ID changes to 0#07, error during command execution
2	0#001F	0#1F = reflected command number 31
3	0#0011	error code 0#11 → no slave with the profile S-7.F.F

6.1.14 Command 33 (0#21): Read the diagnosis string of an AS-i slave with profile S-7.4

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	reserved = 0			AS-i slave address					command number = 33 (0#21)							
3…17	ignored															
18	field number (0#00 / 0#01)								number of bytes to be read							

Example:

1	0#0765	user ID changes to 0#07, command request with 0#65
2	0#0321	slave address = 3(A), 0#21 = command number 33

Response from controllere:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	TG	S	AS-i slave address					F	reflected command number = 0#21							
3	diagnosis string 1								diagnosis string 0							
4…16	diagnosis string 2...27															
17	diagnosis string 29								diagnosis string 28							
18	0#00								number of bytes received							

Legend:

S	Sequence bit Length: 1 bit Permitted values: 0/1 Meaning: 0 = data transmission completed 1 = data transmission not yet completed, at least one more packet follows.
TG	Toggle Length: 1 bit Permitted values: 0/1 Meaning: value changes for each execution of the command
F	Error bit Length: 1 bit Permitted values: 0/1 Meaning: 0 = execution was error free 1 = an error occurred during execution, e.g.: slave has not the profile S-7.4

NOTE

The control bytes defined in profile 7.4 with follow byte and valid byte are filtered out by the system.

Example:

1	0#076F	user ID changes to 0#07, command status is "completed" = 0#6F (no error)
2	0#0621 or 0#8621	S = 0: last sequence, 06_h = 00**00011**0_b → slave address = 3(A), 0#21= reflected command number 33 the most significant bit changes after each execution
3	0#2D01	1st word of the diagnosis data of slave 3(A)
4	0#0203	2nd word of the diagnosis data of slave 3(A)
5	0#1122	3rd word of the diagnosis data of slave 3(A)
6	0#3344	4th word of the diagnosis data of slave 3(A)
…	…	…
18	0#0008	8 bytes diagnosis data

6.1.15 Command 34 (0#22): Read the parameter string of an AS-i slave with profile S-7.4

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	reserved = 0			AS-i slave address					command number = 34 (0#22)							
3…17	ignored															
18	field number (0#00 / 0#01)								number of bytes to be read							

Example:

1	0#0865	user ID changes to 0#08, command request with 0#65
2	0#0322	slave address = 3, 0#22 = command number 34

Response from controllere:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	TG	S	AS-i slave address					F	reflected command number = 0#22							
3	parameter string 1								parameter string 0							
4…16	parameter strings 2…27															
17	parameter string 29								parameter string 28							
18	0#00								number of bytes received							

Legend:

S	Sequence bit Length: 1 bit Permitted values: 0/1 Meaning: 0 = data transmission completed 1 = data transmission not yet completed, at least one more packet follows.
TG	Toggle Length: 1 bit Permitted values: 0/1 Meaning: value changes for each execution of the command
F	Error bit Length: 1 bit Permitted values: 0/1 Meaning: 0 = execution was error free 1 = an error occurred during execution, e.g.: slave has not the profile S-7.4

NOTE

The control bytes defined in profile 7.4 with follow byte and valid byte are filtered out by the system.

Example:

1	0#086F	user ID changes to 0#08, command status is "completed" = 0#6F (no error)
2	0#0622 or 0#8622	06_h = 00**00011**0_b → slave address = 3(A), 0#22 = reflected command number 34 the most significant bit changes after each execution
3	0#1234	1st word of the parameter string of slave 3(A)
4	0#5678	2nd word of the parameter string of slave 3(A)
…	…	…
18	0#0004	4-byte parameter string was read

6.1.16 Command 35 (0#23): Write parameter string of an AS-i slave with the profile S-7.4

Request from host:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command request = 0#65							
2	R	S	R	AS-i slave address					command number = 35 (0#23)							
3	parameter string 1								parameter string 0							
4…16	parameter strings 2…27															
17	parameter string 29								parameter string 28							
18	field number (0#00 / 0#01)								number of bytes to be sent (rest is ignored)							

Legend:

R	Reserved; in request = "0"
S	Sequence bit Length: 1 bit Permitted values: 0/1 Meaning: 0 = data transmission completed 1 = data transmission not yet completed, at least one more packet follows.

Example:

1	0#0965	user ID changes to 0#09, command request with 0#65
2	0#0323	slave address = 3(A), 0#23 = command number 35
3	0#1AF4	1st word of the parameter string for slave 3(A)
4	0#5BB8	2nd word of the parameter string for slave 3(A)
…	…	…
18	0#0004	4-byte parameter string to be sent

Response from controller:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	TG	S	AS-i slave address					F	reflected command number = 0#23							
3...18	0#00								0#00							

Legend:

S
Sequence bit
Length: 1 bit
Permitted values: 0/1
Meaning:
0 = data transmission completed
1 = data transmission not yet completed, at least one more packet follows.

TG
Toggle
Length: 1 bit
Permitted values: 0/1
Meaning: value changes for each execution of the command

F
Error bit
Length: 1 bit
Permitted values: 0/1
Meaning:
0 = execution was error free
1 = an error occurred during execution, e.g.: slave has not the profile S-7.4

NOTE

The number of the bytes to be sent must be divisible by 2 since the system always transmits only multiples of 2 bytes in the S7.4 protocol.

The control bytes defined in profile 7.4 with follow bit and valid bit are automatically added by the system. Therefore, without segmentation, this command is limited to 20 bytes of parameter data. Larger data volumes must be divided into segments.

Example:

1	0#096F	user ID changes to 0#09, command status is "completed" = 0#6F (no error)
2	0#0623 or 0#8623	$x6_h$ = xx**00011**0_b → slave address = 3(A), 0#23 = reflected command number 35 the most significant bit changes after each execution

6.1.17 Command 36 (0#24): Acyclic standard read call to an AS-i slave with CTT2 profile (S-7.5.5, S-7.A.5 or S-B.A.5)

– available from master profile M4 onwards –

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0	0	A/B	AS-i slave address					command number = 36 (0#24)							
3	number of bytes to be read								Index							
4…17	ignored															
18…19	reserved															

Legend:

A/B	Bit for addressing A or B slaves Length: 1 bit Permitted values: 0/1 Meaning: 0 = A slave 1 = B slave (addition of 20_h or 32_d to the slave address)
Index	Pointer to the page to be read Length: 1 byte Permitted values: 0…255 Meaning: → data sheet of the addressed CTT2 slaves
Number of bytes to be read	Number of bytes to be read Length: 1 byte Permitted values: 1…32 Meaning: → data sheet of the addressed CTT2 slaves

Example:

1	0#0465	user ID changes to 0#04, command request with 0#65
2	0#0324	0#03 = slave address 3(A), 0#24 = command number 36
3	0#0409	in index 9, 4 parameter bytes are to be read

Response from controller in the normal case:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	TG	L32	reserved					F=0	reflected command number = 0#24							
3	parameter byte 0								parameter byte 1							
4…16	parameter bytes 2...27															
17	parameter byte 28								parameter byte 29							
18	parameter byte 30 or number of bytes read								parameter byte 31							

Legend:

L32	Number of parameter bytes = 32 Length: 1 bit Permitted values: 0/1 Meaning: 0 = number of bytes to be sent <32 1 = number of bytes to be sent =32
TG	Toggle Length: 1 bit Permitted values: 0/1 Meaning: value changes for each execution of the command
F	Error bit Length: 1 bit Permitted values: 0/1 Meaning: 0 = execution was error free 1 = an error occurred during execution

NOTE

The high byte in the 18th word contains the number of the parameter bytes read as long as the number is <32 (L32 = 0).
If the length is 32 (= maximum possible length), the bit L32 is set and the high byte in the 18th word contains the 32nd parameter byte.

Example:

1	0#046F	user ID changes to 0#04, command status is "completed" = 0#6F (no error)
2	0#0024 or 0#8024	0#00 / 0#80 → L32 = → net length < 0#24 = reflected command number 36 the most significant bit changes after each execution
3	0#1234	1st and 2nd parameter byte from index 9 in slave 3(A)
4	0#5678	3rd and 4th parameter byte from index 9 in slave 3(A)
5…17	0#0000	invalid / not used
18	0#0400	4-byte parameter string was read

Response from controller in the case of an error (error detected by AS-i master):

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6B							
2	TG	reserved							reflected command number = 0#24							
3	0#00								error code							

Legend:

TG	Toggle Length: 1 bit Permitted values: 0/1 Meaning: value changes for each execution of the command

Possible error codes:

0#16	timeout during command processing
0#17	wrong slave profile or slave not in LAS or master not in the normal mode
0#E0... 0#EF	error detected by AS-i slave; see error code CTT2 (see below)
0#F0	invalid CTT2 command
0#F1	invalid CTT2 response
0#F2	7.5 data length longer than 30 bytes

Example:

1	0#046B	user ID changes to 0#04, 0#6B = error during command execution
2	0#0024 or 0#8024	reflected command number = 0#24 the most significant bit changes after each execution
3	0#0016	error code 0#16 → timeout during command processing

Response from controller in the case of an error (error detected by AS-i slave):

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6B							
2	TG	0	reserved					F=1	reflected command number = 0#24							
3	CTT2 error code								error code = 0#E1							

Legend: :

TG
Toggle
Length: 1 bit
Permitted values: 0/1
Meaning: 1 = value changes for each execution of the command

F
Error bit
Length: 1 bit
Permitted values: 0/1
Meaning:
0 = execution was error free
1 = an error occurred during execution

Possible CTT2 error codes:

0#00	no error
0#01	invalid index
0#02	invalid length
0#03	command not implemented
0#04	used; it was not possible to complete the command in the specified time
0#05	command was not acknowledged

Example:

1	0#046B	user ID changes to 0#04, 0#6B = error during command execution
2	0#0124 or 0#8124	0#x1 = error during command execution 0#24 = reflected command number 36 the most significant bit changes after each execution
3	0#01E1	error code 0#01 = invalid index, → data sheet of the AS-i slave Error code 0#E1 = error detected by AS-i slave CTT2 error

6.1.18 Command 37 (0#25): Acyclic standard write call for an AS-i slave with CTT2 profile (S-7.5.5, S-7.A.5 or S-B.A.5)

– available from master profile M4 onwards –

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0	0	A/B	AS-i slave address					command number = 37 (0#25)							
3	number of bytes to be sent								index							
4	parameter byte 0								parameter byte 1							
5…17	parameter bytes 2...27															
18	parameter byte 28								parameter byte 29							
19	reserved															

Legend:

A/B	Bit for addressing A or B slaves Length: 1 bit Permitted values: 0/1 Meaning: 0 = A slave 1 = B slave (addition of 20_h or 32_d to the slave address)
Index	Pointer to the page to be read Length: 1 byte Permitted values: 0…255 Meaning: → data sheet of the addressed CTT2 slaves
number of bytes to be sent	Number of bytes to be sent Length: 1 byte Permitted values: 1…30 Meaning: → data sheet of the addressed CTT2 slaves

Example:

1	0#0565	user ID changes to 0#05, command request with 0#65
2	0#0325	0#03 = slave address 3(A), 0#25 = command number 37
3	0#0207	in index 7, 2 parameter bytes are to be written
4	0#1AF4	both parameter bytes for slave 3(A)

Response from controller in the normal case:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command status = 0#6F							
2	TG	0	reserved					F=0	reflected command number = 0#25							

Legend:

TG	Toggle Length: 1 bit Permitted values: 0/1 Meaning: value changes for each execution of the command
F	Error bit Length: 1 bit Permitted values: 0/1 Meaning: 0 = execution was error free 1 = an error occurred during execution

Example:

1	0#056F	user ID changes to 0#05, command status is "completed" = 0#6F (no error)
2	0#0025 or 0#8025	0#25 = reflected command number 37 the most significant bit changes after each execution

Response from controller in the case of an error (error detected by AS-i master):

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6B							
2	TG	reserved							reflected command number = 0#25							
3	0#00								error code							

Possible error codes:

0#16	timeout during command processing
0#17	wrong slave profile or slave not in LAS or master not in the normal mode
0#E... 0#EF	error detected by AS-i slave; see error code CTT2 (see below)
0#F0	invalid CTT2 command
0#F1	invalid CTT2 response
0#F2	7.5 data length longer than 30 bytes

Example:

1	0#056B	user ID changes to 0#05, 0#6B = error during command execution
2	0#0025 or 0#8025	0#25 = reflected command number 37 the most significant bit changes after each execution
3	0#0016	error code 0#16 → timeout during command processing

Response from controller in the case of an error (error detected by AS-i slave):

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6B							
2	TG	0	reserved					F=1	reflected command number = 0#25							
3	CTT2 error code								error code = 0#E1							

Possible CTT2 error codes:

0#00	no error
0#01	invalid index
0#02	invalid length
0#03	command not implemented
0#04	used; it was not possible to complete the command in the specified time
0#05	command was not acknowledged

Example:

1	0#056B	user ID changes to 0#05, 0#6B = error during command execution
2	0#0125 or 0#8125	0#x1 = error during command execution 0#25 = reflected command number 37 the most significant bit changes after each execution
3	0#01E1	error code 0#01 = invalid index, → data sheet of the AS-i slave error code 0#E1 = error detected by AS-i slave; CTT2 error

6.1.19 Command 38 (0#26): Acyclic manufacturer-specific read call to an AS-i slave with CTT2 profile (S-7.5.5, S-.7.A.5 or S-B.A.5)

– available from master profile M4 onwards –

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	reserved = 0		A/B	AS-i slave address					command number = 38 (0#26)							
3	number of bytes to be read								Index							
4…17	ignored															
18…19	reserved															

Legend:

A/B	Bit for addressing A or B slaves Length: 1 bit Permitted values: 0/1 Meaning: 0 = A slave 1 = B slave (addition of 20_h or 32_d to the slave address)
Index	Pointer to the page to be read Length: 1 byte Permitted values: 0…255 Meaning: → data sheet of the addressed CTT2 slaves
Number of bytes to be read	Number of bytes to be read Length: 1 byte Permitted values: 1…32 Meaning: → data sheet of the addressed CTT2 slaves

Example:

1	0#0665	user ID changes to 0#06, command request with 0#65
2	0#0326	0#03 = slave address 3(A), 0#26 = command number 38
3	0#0409	in index 9, 4 parameter bytes are to be read

Response from controller in the normal case:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	TG	L32	reserved					F=0	reflected command number = 0#26							
3	parameter byte 0								parameter byte 1							
4…16	parameter bytes 2...27															
17	parameter byte 28								parameter byte 29							
18	parameter byte 30 or number of bytes read								parameter byte 31							

Legend:

L32	Number of parameter bytes = 32 Length: 1 bit Permitted values: 0/1 Meaning: 0 = number of bytes to be sent <32 1 = number of bytes to be sent =32
TG	Toggle Length: 1 bit Permitted values: 0/1 Meaning: value changes for each execution of the command
F	Error bit Length: 1 bit Permitted values: 0/1 Meaning: 0 = execution was error free 1 = an error occurred during execution

NOTE

The high byte in the 18th word contains the number of the parameter bytes read as long as the number is <32 (L32 = 0).
If the length is 32 (= maximum possible length), the bit L32 is set and the high byte in the 18th word contains the 32nd parameter byte.

Example:

1	0#066F	user ID changes to 0#06, command status is "completed" = 0#6F (no error)
2	0#0026 or 0#8026	0#0x / 0#8x → L32 = → number of parameter bytes <32 0#26 = reflected command number 38 the most significant bit changes after each execution
3	0#1234	1st and 2nd parameter byte from index 9 in slave 4
4	0#5678	3rd and 4th parameter byte of index 9 in slave 4
5…17	0#0000	invalid / not used
18	0#0400	4-byte parameter string was read

Response from controller in the case of an error (error detected by AS-i master):

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command status = 0#6B							
2	TG	reserved							reflected command number = 0#26							
3	0#00								error code							

Possible error codes:

0#16	timeout during command processing
0#17	wrong slave profile or slave not in LAS or master not in the normal mode
0#E0... 0#EF	error detected by AS-i slave; see error code CTT2 (see below)
0#F0	invalid CTT2 command
0#F1	invalid CTT2 response
0#F2	7.5 data length longer than 30 bytes

Example:

1	0#066B	user ID changes to 0#06, 0#6B = error during command execution
2	0#0026 or 0#8026	0#26 = reflected command number 38 the most significant bit changes after each execution
3	0#0016	error code 0#16 → timeout during command processing

Response from controller in the case of an error (error detected by AS-i slave):

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6B							
2	TG	0	reserved					F=1	reflected command number = 0#26							
3	CTT2 error code								error code = 0#E1							

Possible CTT2 error codes:

0#00	no error
0#01	invalid index
0#02	invalid length
0#03	command not implemented
0#04	used; it was not possible to complete the command in the specified time
0#05	command was not acknowledged

Example:

1	0#066B	user ID changes to 0#06, 0#6B = error during command execution
2	0#0126 or 0#8126	0#x1 = error during command execution reflected command number = 0#26 the most significant bit changes after each execution
3	0#01E1	error code 0#01 = invalid index, → data sheet of the AS-i slave error code 0#E1 = error detected by AS-i slave; CTT2 error

6.1.20 Command 39 (0#27): Acyclic standard manufacturer-specific write call to an AS-i slave with CTT2 profile (S-7.5.5, S-7.A.5 or S-B.A.5)

– available from master profile M4 onwards –

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0	0	A/B	AS-i slave address					command number = 39 (0#27)							
3	number of bytes to be sent								index							
4	parameter byte 0								parameter byte 1							
5…17	parameter bytes 2…27															
18	parameter byte 28								parameter byte 29							
19	reserved															

Legend:

A/B	Bit for addressing A or B slaves Length: 1 bit Permitted values: 0/1 Meaning: 0 = A slave 1 = B slave (addition of 20_h or 32_d to the slave address)
Index	Pointer to the page to be read Length: 1 byte Permitted values: 0…255 Meaning: → data sheet of the addressed CTT2 slaves
Number of bytes to be sent	Number of bytes to be sent Length: 1 byte Permitted values: 1…30 Meaning: → data sheet of the addressed CTT2 slaves

Example:

1	0#0765	user ID changes to 0#07, command request with 0#65
2	0#0327	0#03 = slave address 3(A), 0#27 = command number 39
3	0#0207	in index 7, 2 parameter bytes are to be written
4	0#1AF4	both parameter bytes for slave 3(A)

Response from controller in the normal case:

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command status = 0#6F							
2	TG	0	reserved					F=0	reflected command number = 0#27							

Legend:

TG	Toggle Length: 1 bit Permitted values: 0/1 Meaning: value changes for each command execution
F	Error bit Length: 1 bit Permitted values: 0/1 Meaning: 0 = execution was error free 1 = an error occurred during execution

Example:

1	0#076F	user ID changes to 0#07, command status is "completed" = 0#6F (no error)
2	0#0027 or 0#8027	0#27 = reflected command number 39 the most significant bit changes after each execution

Response from controller in the case of an error (error detected by AS-i master):

Word no.	Bit															
	15	**14**	**13**	**12**	**11**	**10**	**9**	**8**	**7**	**6**	**5**	**4**	**3**	**2**	**1**	**0**
1	user ID								command status = 0#6B							
2	TG	reserved							reflected command number = 0#27							
3	0#00								error code							

Possible error codes:

0#16	timeout during command processing
0#17	wrong slave profile or slave not in LAS or master not in the normal mode
0#E0... 0#EF	error detected by AS-i slave; note error code CTT2 (see below)
0#F0	invalid CTT2 command
0#F1	invalid CTT2 response
0#F2	7.5 data length longer than 30 bytes

Example:

1	0#076B	user ID changes to 0#07, 0#6B = error during command execution
2	0#0027 or 0#8027	0#27 = reflected command number 39 the most significant bit changes after each execution
3	0#0016	error code 0#16 → timeout during command processing

Response from controller in the case of an error (error detected by AS-i slave):

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6B							
2	TG	0	reserved					F=1	reflected command number = 0#27							
3	CTT2 error code								error code = 0#E1							

Possible CTT2 error codes:

0#00	no error
0#01	invalid index
0#02	invalid length
0#03	command not implemented
0#04	used; it was not possible to complete the command in the specified time
0#05	command was not acknowledged

Example:

1	0#076B	user ID changes to 0#07, 0#6B = error during command execution
2	0#0127 or 0#8127	0#x1 = error during command execution 0#27 = reflected command number the most significant bit changes after each execution
3	0#01E1	error code 0#01 = invalid index, → data sheet of the AS-i slave Error code 0#E1 = error detected by AS-i slave CTT2 error

6.1.21 Command 50 (0#32): Read current configuration of AS-i slaves 0(A)...15(A)

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 50 (0#32)							
3…17	ignored															
18…19	reserved															

Example:

1	0#0265	user ID changes to 0#02, command request with 0#65
2	0#0032	0#32 = command number 50

Response from controller:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#32							
3	slave0, ID2				slave0, ID1				slave0, ID code				slave 0, IO conf.			
4	slave1(A), ID2				slave 1 (A), ID1				slave 1(A), ID code				slave 1(A), IO conf.			
5...17	...				...				...				...			
18	slave15(A), ID2				slave15(A), ID1				slave15(A), ID code				slave15(A), IO conf.			

Example:

1	0#026F	user ID changes to 0#02, command status is "completed" = 0#6F (no error)
2	0#0032	0#32 = reflected command number 50
3	0#FFFF	current configuration of slave 0: ID2 =F, ID1=F, ID=F and IO=F
4	0#EF03	current configuration of slave 1(A) ID2 =E, ID1=F, ID=0 and IO=3
…	…	…
18	0#EF37	current configuration of slave 15(A): ID2 =E, ID1=F, ID=3 and IO=7

Command 51 (0#33): read current configuration AS-i slaves 16(A)....31(A)
Command 52 (0#34): read current configuration of AS-i slaves (0)1B...15B
Command 53 (0#35): read current configuration of AS-i slaves 16B...31B
→ Command 50 (0#32)

6.1.22 Command 54 (0#36): Read current parameters of a connected AS-i slave

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 54 (0#36)							
3…17	ignored															
18…19	reserved															

Example:

1	0#0665	user ID changes to 0#06, command request with 0#65
2	0#0036	0#36 = command number 54

Response from controller:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#36							
3	param. slave4(A)				param. slave3(A)				param. slave2(A)				param. slave1(A)			
4	param. slave8(A)				param. slave7(A)				param. slave6(A)				param. slave5(A)			
5	param. slave12(A)				param. slave11(A)				param. slave10(A)				param. slave9(A)			
6	param. slave16(A)				param. slave15(A)				param. slave14(A)				param. slave13(A)			
7	param. slave20(A)				param. slave19(A)				param. slave18(A)				param. slave17(A)			
8	param. slave24(A)				param. slave23(A)				param. slave22(A)				param. slave21(A)			
9	param. slave28(A)				param. slave27(A)				param. slave26(A)				param. slave25(A)			
10	param. slave1B				param. slave31(A)				param. slave30(A)				param. slave29(A)			
11	param. slave5B				param. slave4B				param. slave3B				param. slave2B			
12	param. slave9B				param. slave8B				param. slave7B				param. slave6B			
13	param. slave13B				param. slave12B				param. slave11B				param. slave10B			
14	param. slave17B				param. slave16B				param. slave15B				param. slave14B			
15	param. slave21B				param. slave20B				param. slave19B				param. slave18B			
16	param. slave25B				param. slave24B				param. slave23B				param. slave22B			
17	param. slave29B				param. slave28B				param. slave27B				param. slave26B			
18	not used				not used				param. slave31B				param. slave30B			

Example:

1	0#066F	user ID changes to 0#06, command status is "completed" = 0#6F (no error)
2	0#0036	0#36 = reflected command number 54
3	0#4321	parameters of slave 1 [value = 1] to slave 4 [value = 4]
4	0#8765	parameters of slave 5 [value = 5] to slave 8 [value = 8]
…	…	…
9	0#6543	slave 29(A) [value = 3], slave 30(A) [value = 4], slave 31(A) [value = 5], slave 1B [value = 6]
…	…	…
17	0#FE98	parameters of slave 26B [value = 8] to slave 29B [value = F]
18	0#0098	parameters of slave 30B [value = 8] and slave 31B [value = 9]

6.1.23 Command 55 (0#37): Read current AS-i slaves

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 55 (0#37)							
3…17	ignored															
18…19	reserved															

Example:

1	0#0765	user ID changes to 0#07, command request with 0#65
2	0#0037	0#37 = command number 55

Response from controller:

Word no.	Bit															
	—	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1		user ID								command status = 0#6F							
2		0#00								reflected command number = 0#37							
3	LAS	15A	14A	13A	12A	11A	10A	9A	8A	7A	6A	5A	4A	3A	2A	1A	–
4		31A	30A	29A	28A	27A	26A	25A	24A	23A	22A	21A	20A	19A	18A	17A	16A
5		15B	14B	13B	12B	11B	10B	9B	8B	7B	6B	5B	4B	3B	2B	1B	–
6		31B	30B	29B	28B	27B	26B	25B	24B	23B	22B	21B	20B	19B	18B	17B	16B
7	LDS	15A	14A	13A	12A	11A	10A	9A	8A	7A	6A	5A	4A	3A	2A	1A	0
8		31A	30A	29A	28A	27A	26A	25A	24A	23A	22A	21A	20A	19A	18A	17A	16A
9		15B	14B	13B	12B	11B	10B	9B	8B	7B	6B	5B	4B	3B	2B	1B	–
10		31B	30B	29B	28B	27B	26B	25B	24B	23B	22B	21B	20B	19B	18B	17B	16B
11	LPF	15A	14A	13A	12A	11A	10A	9A	8A	7A	6A	5A	4A	3A	2A	1A	–
12		31A	30A	29A	28A	27A	26A	25A	24A	23A	22A	21A	20A	19A	18A	17A	16A
13		15B	14B	13B	12B	11B	10B	9B	8B	7B	6B	5B	4B	3B	2B	1B	–
14		31B	30B	29B	28B	27B	26B	25B	24B	23B	22B	21B	20B	19B	18B	17B	16B
15	LPS	15A	14A	13A	12A	11A	10A	9A	8A	7A	6A	5A	4A	3A	2A	1A	–
16		31A	30A	29A	28A	27A	26A	25A	24A	23A	22A	21A	20A	19A	18A	17A	16A
17		15B	14B	13B	12B	11B	10B	9B	8B	7B	6B	5B	4B	3B	2B	1B	–
18		31B	30B	29B	28B	27B	26B	25B	24B	23B	22B	21B	20B	19B	18B	17B	16B

Example:

1	0#076F	user ID changes to 0#07, command status is "completed" = 0#6F (no error)
2	0#0037	0#37 = reflected command number 55
3	0#0102	LAS slaves 1(A) to 15(A): 0102_h = 0000 0001 0000 0010_b slaves 1 and 8 are active
4	0#8001	LAS slaves 16(A) to 31(A): 8001_h = 1000 0000 0000 0001_b slaves 16(A) and 31(A) are active
5	0#0102	LAS slaves 1B to 15B: 0102_h = 0000 0001 0000 0010_b slaves 1B and 8B are active
6	0#8001	LAS slaves 16B to 31B: 8001_h = 1000 0000 0000 0001_b slaves 16B and 31B are active
7	0#0102	LDS slaves 0 to 15(A): 0102_h = 0000 0001 0000 0010_b slaves 1(A) and 8(A) are detected
8	0#8001	LDS slaves 16(A) to 31(A): 8001_h = 1000 0000 0000 0001_b slaves 16(A) and 31(A) are detected
9	0#0102	LDS slaves 1B to 15B: 0102_h = 0000 0001 0000 0010_b slaves 1B and 8B are detected
10	0#8001	LDS slaves 16B to 31B: 8001_h = 1000 0000 0000 0001_b slaves 16B and 31B are detected
11	0#0100	LPF slaves 0 to 15(A): 0100_h = 0000 0001 0000 0000_b peripheral fault on slave 8(A) signalled
12	0#0001	LPF slaves 16(A) to 31(A): peripheral fault on slave 16(A) signalled
13	0#0002	LPF slaves 1B to 15B: peripheral fault on slave 1B signalled
14	0#8000	LPF slaves 16B to 31B: 8000_h = 1000 0000 0000 0000_b peripheral fault on slave 31B signalled
15	0#0102	LPS slaves 1(A) to 15(A): 0102_h = 0000 0001 0000 0010_b slaves 1(A) and 8(A) are projected
16	0#8001	LPS slaves 16(A) to 31(A): 8001_h = 1000 0000 0000 0001_b slaves 16(A) and 31(A) are projected
17	0#0102	LPS slaves 1B to 15B: 0102_h = 0000 0001 0000 0010_b slaves 1B and 8B are projected
18	0#8001	LPS slaves 16B to 31B: 8001_h = 1000 0000 0000 0001_b slaves 16B and 31B are projected

6.1.24 Command 56 (0#38): Read projected configuration of the AS-i slaves 1(A)...15(A)

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 56 (0#38)							
3…17	ignored															
18…19	reserved															

Example:

1	0#0265	user ID changes to 0#02, command request with 0#65
2	0#0038	0#38 = command number 56

Response from controller:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#38							
3	slave0, ID2				slave0, ID1				slave 0, ID code				slave 0, IO conf.			
4	slave1(A), ID2				slave 1 (A), ID1				slave 1(A), ID code				slave 1(A), IO conf.			
5...17	...				...				...				...			
18	slave15(A), ID2				slave15(A), ID1				slave15(A), ID code				slave15(A), IO conf.			

Example:

1	0#026F	user ID changes to 0#02, command status is "completed" = 0#6F (no error)
2	0#0038	0#38 = reflected command number 56
3	0#FFFF	here not used since slave 0 cannot be projected
4	0#EF03	projected configuration for slave 1(A): ID2 =E, ID1=F, ID=0 and IO=3
…	…	…
18	0#EF37	projected configuration for slave 15(A): ID2 =E, ID1=F, ID=3 and IO=7

Command 57 (0#39): read projected configuration of the AS-i slaves 16(A)...31(A)
Command 58 (0#3A): read projected configuration of the AS-i slaves (0)1B...15B
Command 59 (0#3B): read projected configuration of the AS-i slaves 16B...31B
→ Command 56 (0#38)

6.1.25 Command 96 (0#60): Save data non-volatilely in the flash memory of the controller

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 96 (0#60)							
3	0#00								area number							
4…19	ignored															

Example:

1	0#0965	user ID changes to 0#09, command request with 0#65
2	0#0060	0#60 = command number 96
3	0#0002	area number 0#02 = non-volatilely save the configuration of AS-i master 1 0#03 = non-volatilely save the configuration of AS-i master 2

Response from controller:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#60							
3	0#00								area number							

Example:

1	0#096F	user ID changes to 0#09, command status is "completed" = 0#6F (no error)
2	0#0060	0#60 = reflected command number 96
3	0#0002	reflected area number 0#02 = non-volatilely save the configuration of AS-i master 1

6.1.26 Command 97 (0#61): Carry out various settings in the controller

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 97 (0#61)							
3	0#00								command number							
4…19	parameters 1…16															

Example:

1	0#0865	user ID changes to 0#08, command request with 0#65
2	0#0061	0#61 = command number 97
3	0#0010	command number: 0#10 = changes the operating mode of the PLC (corresponding parameters → word 4) further command numbers: 0#12 = reset all slave error counters 0#13 = reset the configuration error counter 0#14 = reset AS-i cycle error counter
4	0#0002	parameters, here for command number 0#10: 0#0000 = activates the gateway mode 0#0001 = stops the PLC 0#0002 = sets the operation mode of the PLC to RUN

Response from controller:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#61							
3...18	0#00								0#00							

Example:

1	0#086F	user ID changes to 0#08, command status is "completed" = 0#6F (no error)
2	0#0061	0#61 = reflected command number 97

6.1.27 Command 102 (0#66): Retrieve the status of the controllere display

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 102 (0#66)							
3	0#00								command number							
4…n	parameter(s) (according to command number)															

Example:

1	0#0765	user ID changes to 0#07, command request with 0#65
2	0#0066	0#66 = command number 102
3	0#0001	command number, here: 0#01 = enquires the display status further command numbers: 0#02 = change to menu screen 0 0#03 = change to user menu screen 0#A1

Response from controllere:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#66							
3	0#00								reflected command number here: 0#01							
4	buttons pressed															
5	activated menu area															
6	process error occurred															
7	currently displayed menu screen															
8	activated system language															
9…18	reserved															

Legend:

Buttons pressed	0#0001	left button pressed
	0#0002	button [▲] is pressed
	0#0004	button [▲] is pressed
	0#0008	right button is pressed
Active menu area	0#00A0	system menu is active
	0#00A1	user menu is active
	0#00AE	process error display is active (E10...E30)
	0#00AF	system error display is active (acknowledgement required)
Process	0#0000	no process error

error occurred	0#0001	process error occurred
Currently displayed menu screen	0#xxxx	number of the current menu screen
Activated system language	0#0000	display of menus in English
	0#0001	display of menus in the second system language (e.g. German)

Example:

1	0#076F	user ID changes to 0#07, command status is "completed" = 0#6F (no error)
2	0#0066	0#66 = reflected command number 102
3	0#0001	0#01 = reflected command number
4	0#0008	right button is pressed
5	0#00A0	system menu is active
6	0#0001	process error occurred
7	0#001B	menu screen 27 "Quick Setup" is displayed
8	0#0000	display of menus in English

6.1.28 Command 105 (0#69): Read the device properties of the controller

Request from host:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command request = 0#65							
2	0#00								command number = 105 (0#69)							
3…17	ignored															
18…19	reserved															

Example:

1	0#0665	user ID changes to 0#06, command request with 0#65
2	0#0069	0#69 = command number 105

Response from controller:

Word no.	Bit															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
1	user ID								command status = 0#6F							
2	0#00								reflected command number = 0#69							
3	2M	DP	EN	reserved					PLC mode							
4	0#00								fieldbus type							
5	0#00								flash memory type							
6	hardware version															
7	RTS firmware version number															
8	RTS firmware release number															
9	AS-i master 1 firmware version number															
10	AS-i master 1 firmware release number															
11	AS-i master 2 firmware version number															
12	AS-i master 2 firmware release number															
13	Linux kernel version															
14	Linux ramdisk version															
15…18	0#00															

Legend:

2M	0	device with 1 AS-i master
	1	device with 2 AS-i master
DP	0	fieldbus interface Profibus DP(V1) not available
	1	fieldbus interface Profibus DP is available
EN	0	device without Ethernet programming interface
	1	device with Ethernet programming interface

PLC mode	0#01	PLC is in the RUN mode
	0#02	PLC is in the STOP mode
	0#04	PLC stops at the breakpoint
	0#08	gateway mode
fieldbus type	0#01	Anybus Profibus DP
	0#04	Anybus CANopen
	0#05	Anybus DeviceNet
	0#09	Anybus Ethernet IT
	0#0A	Anybus Ethernet/IP
	0#0B	ifm Profibus DP
	0#0C	no fieldbus module detected

Example:

1	0#066F	user ID changes to 0#06, command status is "completed" = 0#6F (no error)
2	0#0069	0#69 = reflected command number 105
3	0#4008	40_h = 0100 0000_b 2M = 0 → with one AS-i master, DP = 1 → Profibus DP controller EN = 0 → without Ethernet programming interface, PLC mode 0#08 = gateway; signal preprocessing is not used
4	0#000B	fieldbus interface "ifm Profibus DP" used
5	0#0002	flash memory type
6	0#1000	hardware version
7	0#0002	1st part of the RTS firmware number 02.218B: RTS firmware version number = 02
8	0#218B	2nd part of the RTS firmware number 02.218B: RTS firmware release number = 218B
9	0#0000	1st part of the firmware number 0.238A for AS-i master 1: AS-i master 1 firmware version number = 0
10	0#238A	2nd part of the firmware number 0.238A for AS-i master 1: AS-i master 1 firmware release number = 238A
11	0#0000	1st part of the firmware number 0.238A for AS-i master 2: AS-i master 2 firmware version number = 0
12	0#238A	2nd part of the firmware number 0.238A for AS-i master 2: AS-i master 2 firmware release number = 238A
13	0#0196	Linux kernel version: 406_d = 0#0196
14	0#0A6E	Linux ramdisk version: 10.110_d = 0#0A.0#6E

7 Terms, abbreviations

A/B slave	→Slave with an A or B being appended to its address number and which may therefore be present twice on the →master.
Address	This is the "name" of the bus participant. All participants need a unique defined address so that the signals can be exchanged without problem.
AS-i	AS-i = **A**ctuator-**S**ensor-**I**nterface Bus system for the first binary field level.
Auto-negotiation	Auto-negotiation designates a procedure which allows network cards or host bus adapters to independently detect the correct transmission speed and the duplex mode of the network interface to which they are connected, and to configure themselves accordingly.
Baud	Baud, abbrev.: Bd = unit for the data transmission speed. Do not confuse baud with "bits per second" (bps, bits/s). Baud indicates the number of changes of state (steps, cycles) per second over a transmission length. But it is not defined how many bits per step are transmitted. The name baud can be traced back to the French inventor J. M. Baudot whose code was used for telex machines. 1 MBd = 1024 x 1024 Bd = 1 048 576 Bd
Bus	Serial data transmission of several participants on the same cable.
CAN	CAN = **C**ontroller **A**rea **N**etwork CAN is a priority controlled fieldbus system for larger data volumes. It is available in different variants, e.g. CANopen, CAN in Automation (CiA) or →DeviceNet. CAN can be used e.g. as a supplier for AS-i over larger distances. Corresponding →gateways are available.
COB ID	COB = **C**ommunication **Ob**ject ID = **Id**entifier Each communication object has a unique COB ID in the network. The COB ID consists of 32-bit values; the first two bits have each an object-specific meaning.
CoDeSys®	CoDeSys for Automation Alliance associates companies of the automation industry whose hardware devices are all programmed with the widely used IEC 61131-3 development tool CoDeSys®. CoDeSys® is a registered trademark of 3S – Smart Software Solutions GmbH, Germany → http://www.3s-software.com
controllere	Master in the AS-i bus system of the generation E
Cycle time	This is the time for a cycle. The following process happens: • PLC cycle: the PLC program performs one complete run. • AS-i cycle: all AS-i slaves are updated (5...10 ms).
DeviceNet	Fieldbus system for larger data volumes based on →CAN technology, requires special cables, complex connection technology. Can be used e.g. as a supplier for AS-i over longer distances. Corresponding →gateways are available.

DHCP	DHCP = **D**ynamic **H**ost **C**onfiguration **P**rotocol = protocol for the dynamic configuration by the →host DHCP is a protocol which offers the dynamic configuration of IP addresses and thus coherent information. The protocol supports the further use of IP-addresses which are only available in a limited number by a centralised management of the address assignment. At first power on of a participant in a network the participant logs on a server using this service. The server assigns a local free →IP address to the participant.
EMC	EMC = **E**lectro **M**agnetic **C**ompatibility According to the EC directive (89/336 EEC) concerning electromagnetic compatibility (in short EMC directive) requirements are made for electrical and electronic apparatus, equipment, systems or components to operate satisfactorily in the existing electromagnetic environment. The devices must not interfere with their environment and must not be adversely influenced by external electromagnetic interference.
Ethernet	Ethernet is a widely used, manufacturer-independent technology which enables data transmission in the network at a speed of 10 or 100 million bits per second (Mbps). Ethernet belongs to the family of so-called "optimum data transmission" on a non exclusive transmission medium. The concept was developed in 1972 and specified as IEEE 802.3 in 1985.
FE	FE = **F**unctional **E**arth Functional earth is a reference potential which is not connected to protective earth or only connected when special measures are taken. The functional earth serves as equalisation of potential for an ungrounded installation (e.g. →SELV).
Fieldbus	A →bus for industrial applications: mechanically extremely robust and excellent data protection
Firmware	Basic program in the device, virtually the operating system The firmware establishes the connection between the hardware of the device and the user software.
Gateway	Access, coupler Gateways enable connection of completely different systems. Gateways are used when two incompatible network types are to be connected by converting the protocol of one system to the protocol of the other system. Here: connection between AS-i and higher-level fieldbus systems such as →Profibus-DP, →DeviceNet, Interbus-S or other interfaces, e.g. RS-485. The device includes an AS-i master which is directly coupled to the →host interface (e.g. →Profibus-DP slave).
GSD	Device master file Describes the interface to the device to be connected to the fieldbus. File → www.ifm.com> Select country/language > [Service] > [Download] > [Bus system AS-Interface]
Host	The controller in the hierarchy above the AS-i master, e.g. a PLC or a processor.
ID	ID = **Id**entifier Name to differentiate the devices / participants connected to a system.
IP address	IP = **I**nternet **P**rotocol The IP address is a number which is necessary to clearly identify an internet participant. For the sake of clarity the number is written in 4 decimal values, e.g. 127.215.205.156.

Jitter	By jitter is understood a slight fluctuation in accuracy in the transmission cycle when transmitting digital signals. In general jitter is an abrupt and undesired change of the signal characteristics in transmission technology.
LAS	**L**ist of **A**ctive **S**laves In this slave list the controller enters the slaves detected as active for this AS-i master.
LDS	**L**ist of **D**etected **S**laves In this slave list the controller enters the slaves detected as present for this AS-i master.
LED	LED = **L**ight **E**mitting **D**iode Light emitting diode, also called luminescent diode, an electronic element of high coloured luminosity at small volume with negligible power loss.
LFS	**L**ist of **F**ailed **S**laves In this slave list the controller enters the slaves with a configuration error on this AS-i master.
LPS	**L**ist of **P**rojected **S**laves In this slave list the controller enters the slaves projected for this AS-i master.
MAC ID	MAC = **M**anufacturer‘s **A**ddress **C**ode →ID = **Id**entifier Every network card has a MAC address, a clearly defined worldwide unique numerical code, more or less a kind of serial number. Such a MAC address is a sequence of 6 hexadecimal numbers, e.g. "00-0C-0E-D0-02-3F".
Master	Handles the complete organisation on the bus. The master decides on the bus access time and polls the →slaves cyclically.
Master-slave communication	AS-i strictly operates to the master-slave principle. The master polls all slaves one after the other in always the same order. Only one master per network line is allowed (→cyclical polling).
MBd	→Baud
Modbus	The Modbus protocol is a communication protocol based on a →master/slave architecture and was generated by Modicon* in 1979 for communication with its PLCs. In industry Modbus is the de-facto standard. Modbus/TCP is based on →Ethernet-TCP/IP. Modbus/TCP ports the protocol defined for the serial interface to TCP. The →IP address clearly defines every device in a network. Therefore the slave address was used to identify one of several logical units (unit IDs) in a physical device. To do so, extended IP addressing is used. Example: 192.168.83.28.1 means unit ID 1 on IP address 192.168.83.28. *) Modicon passed from AEG to the group Schneider in 1994.
Operating system	Basic program in the device, establishes the connection between the hardware of the device and the user software.
OSSD	OSSD = **O**utput **S**ignal **S**witching **D**evice = output signal of a switching device, here: output signal of an AS-i safety monitor

Password	In the menu [System Setup] in the menu item [Password] the operation can be restricted or enabled. When delivered, the device is in the user mode. By entering an invalid password (e.g. 1000) all menu items which can change settings are blocked. → separate basic instructions of the device manual
PELV	PELV = **P**rotective **E**xtra **L**ow **V**oltage Functional extra low voltage with safe separation, grounded variant of SELV. Extra low voltage with safe separation (grounded variant of SELV). The specification as PELV system to IEC364-4-41 (initially DIN VDE 0100-410:1997-01) covers a measure to protect against direct and indirect contact with dangerous voltages by a "safe separation" between primary and secondary side in the device (e.g. power supply to PELV specification). For this reason no separate PE conductor is required in a PELV system. It is allowed to ground circuits and / or bodies in a PELV system.
Pictograms	Image symbols which convey information by a simplified graphic representation. → page 1-1 chapter: What do the symbols and formats stand for?
Polling	to poll = to count votes The controller master fetches the data from every participant in the system successively: 1. Master calls participant 1 2. Participant 1 replies with its current data (actual values) 3. Master transmits more data (preset values) to participant 1 if needed 4. Participant 1 acknowledges receipt of the data etc., the same procedure for all other participants. Cyclical polling: AS-i master cyclically polls the data of all →slaves in the bus (see above). The data is updated in the →master after max. 5 ms. If A/B slaves are used, the →cycle time can be 10 ms.
Profibus	Fieldbus system for larger data volumes, it requires special cables, complex connection technology. It is available in different variants as Profibus FMS, DP or PA. The Profibus DP can be used as a supplier for AS-i over longer distances. Corresponding →gateways are available. → http://www.profibus.com/
Profibus DP	Profibus DP (**D**ecentralised **P**eriphery) to trigger sensors and actuators by a central controller in production technology. In particular the numerous standard diagnostic options are important. More applications are the connection of "distributed intelligence", i.e. networking of several controllers among each (similar to →Profibus FMS). Data rates up to 12 Mbits/s on twisted two-wire cables and/or fibre optics are possible.
Profibus PA	Profibus PA (**P**rocess-**A**utomation) is used to control field devices by means of a process control system in process technology. This Profibus variant is suitable for hazardous areas (zones 0 and 1). Only a small current flows on the bus cables in an intrinsically safe circuit so that even in case of a problem no sparks are produced. The disadvantage of this variant is the slower data transmission rate.
Profibus-FMS	Profibus FMS (**F**ieldbus **M**essage **S**pecification) to network controllers – no longer standardised as from 2007

Remanent	Remanent data is protected against data loss in case of power failure. The operating system for example automatically copies the remanent data to a flash memory as soon as the voltage supply falls below a critical value. If the voltage supply is available again, the →operating system loads the remanent data back to the RAM memory. The data in the RAM memory of a controller, however, is volatile and normally lost in case of power failure.
RTS	RTS = **R**un **T**ime **S**ystem Run time systems are basic versions of applications. These minimum versions are supplied with certain products to meet the prerequisites for the execution of the actual product or to be able to look at or use results generated by this product on other processors: making available all routines required to execute a program in a programming language, e.g. interactions with the →operating system, memory requirements, error routines, inputs and outputs.
Run time system	→RTS
SELV	SELV = **S**afety **E**xtra **L**ow **V**oltage Active parts integrated in SELV circuits must not be connected to ground or protective conductors of other circuits. They must be safely separated from live parts of higher voltage. SELV circuit = secondary circuit (output voltage) which is rated and protected so that its voltages do not exceed a safe value in case of correct operation (of the power supply) or in case of a single fault (of the power supply). SELV circuits are separated from the input voltage (mains voltage) by double or enhanced insulation. The voltage value must not exceed 60 V DC (or 42.4 V AC).
Single slave	→Slave whose address number may only occur once on the →master.
Slave	Passive participant on the bus, only replies on request of the →master. Slaves have a clearly defined and unique →address in the bus. A distinction is made between: • single slaves whose address numbers may only occur once on the →master and • →A / B slaves with an A or B being appended to its address number which may therefore be present twice on the master.
Target	The target indicates the target system where the PLC program is to run. The target contains the files (drivers) required for programming and parameter setting.
UDP	UDP = **U**ser **D**atagram **P**rotocol UDP is a minimal connectionless network protocol belonging to the transport layer of the internet protocol family. The task of UDP is to ensure that data which is transmitted via the internet is passed to the right application.
Unit ID	→Modbus
Watchdog	In general the term watchdog is used for a component of a system which watches the function of other components. If a possible malfunction is detected, this is either signalled or suitable program branchings are activated. The signal or branchings serve as a trigger for other co-operating system components to solve the problem.

8 Index

nn-n	The indication of the page where you can find some information about the keyword is written in normal characters.
ii-i	The indication of the page where the keyword is *detailed* is written in *italics*.