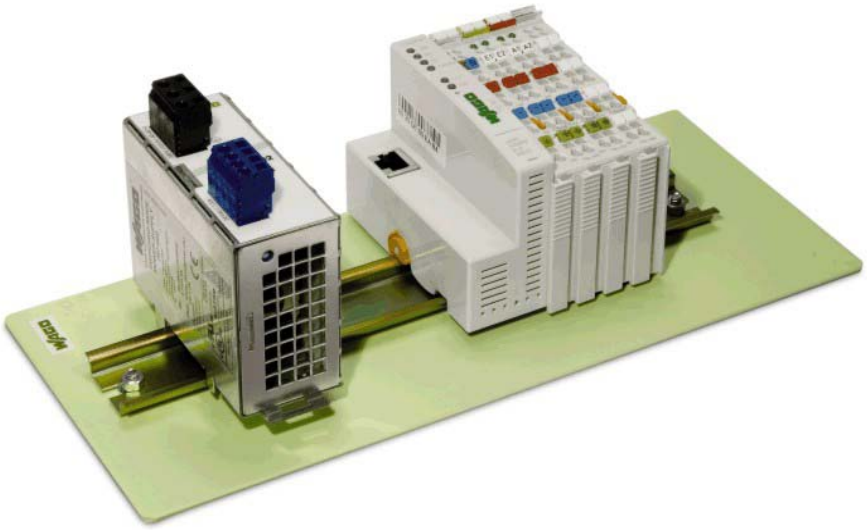
WAGO I/O SYSTEM 750

# Modular I/O System

# ETHERNET StarterKit 2
# 510 376 07



# Quick Start
# ETHERNET Fieldbus Controller
# 750-841

Version 1.0.5

WAGO®
INNOVATIVE CONNECTIONS

**WAGO Kontakttechnik GmbH & Co. KG**
Hansastraße 27
D-32423 Minden

Phone.: +49 (0) 571/8 87 – 0
Fax: +49 (0) 571/8 87 – 1 69

E-Mail: info@wago.com

Web: http://www.wago.com

**Technical Support**
Phone: +49 (0) 571/8 87 – 5 55
Fax: +49 (0) 571/8 87 – 85 55

E-Mail: support@wago.com

Every conceivable measure has been taken to ensure the correctness and completeness of this documentation. However, as errors can never be fully excluded, we would appreciate any information or ideas at any time.

E-Mail: documentation@wago.com

We wish to point out that the software and hardware terms as well as the trademarks of companies used and/or mentioned in the present manual are generally trademark or patent protected.

# CONTENT

# 1 Important Comments

To ensure fast installation and start-up of the units described in this manual, we strongly recommend that the following information and explanations are carefully read and abided by.

## 1.1 Legal Principles

### 1.1.1 Copyright

This manual is copyrighted, together with all figures and illustrations contained therein. Any use of this manual which infringes the copyright provisions stipulated herein, is not permitted. Reproduction, translation and electronic and photo-technical archiving and amendments require the written consent of WAGO Kontakttechnik GmbH & Co. KG. Non-observance will entail the right of claims for damages.

WAGO Kontakttechnik GmbH & Co. KG reserves the right to perform modifications allowed by technical progress. In case of grant of a patent or legal protection of utility patents all rights are reserved by WAGO Kontakttechnik GmbH & Co. KG. Products of other manufacturers are always named without referring to patent rights. The existence of such rights can therefore not be ruled out.

### 1.1.2 Personnel Qualification

The use of the product detailed in this manual is exclusively geared to specialists having qualifications in PLC programming, electrical specialists or persons instructed by electrical specialists who are also familiar with the valid standards. WAGO Kontakttechnik GmbH & Co. KG declines all liability resulting from improper action and damage to WAGO products and third party products due to non-observance of the information contained in this manual.

### 1.1.3 Intended Use

For each individual application, the components supplied are to work with a dedicated hardware and software configuration. Modifications are only permitted within the framework of the possibilities documented in the manuals. All other changes to the hardware and/or software and the non-conforming use of the components entail the exclusion of liability on part of WAGO Kontakttechnik GmbH & Co. KG.

Please direct any requirements pertaining to a modified and/or new hardware or software configuration directly to WAGO Kontakttechnik GmbH & Co. KG.

## 1.2 Symbols

**Danger**
Always abide by this information to protect persons from injury.

**Warning**
Always abide by this information to prevent damage to the device.

**Attention**
Marginal conditions must always be observed to ensure smooth operation.

**ESD** (Electrostatic Discharge)
Warning of damage to the components by electrostatic discharge. Observe the precautionary measure for handling components at risk.

**Note**
Routines or advice for efficient use of the device and software optimization.

**More information**
References on additional literature, manuals, data sheets and INTERNET pages

## 1.3 Number Notation

| Number Code | Example | Note |
|---|---|---|
| Decimal | 100 | normal notation |
| Hexadecimal | 0x64 | C notation |
| Binary | '100'<br>'0110.0100' | Within ',<br>Nibble separated with dots |

## 1.4 Safety Notes

**Attention**
Switch off the system prior to working on bus modules!

In the event of deformed contacts, the module in question is to be replaced, as its functionality can no longer be ensured on a long-term basis.

The components are not resistant against materials having seeping and insulating properties. Belonging to this group of materials is: e.g. aerosols, silicones, triglycerides (found in some hand creams).

If it cannot be ruled out that these materials appear in the component environment, then additional measures are to be taken:
- installation of the components into an appropriate enclosure
- handling of the components only with clean tools and materials.

**Attention**
Cleaning of soiled contacts may only be done with ethyl alcohol and leather cloths. Thereby, the ESD information is to be regarded.

Do not use any contact spray. The spray may impair the functioning of the contact area.

The WAGO-I/O-SYSTEM 750 and its components are an open system. It must only be assembled in housings, cabinets or in electrical operation rooms. Access must only be given via a key or tool to authorized qualified personnel.

The relevant valid and applicable standards and guidelines concerning the installation of switch boxes are to be observed.

**ESD** (Electrostatic Discharge)
The modules are equipped with electronic components that may be destroyed by electrostatic discharge. When handling the modules, ensure that the environment (persons, workplace and packing) is well grounded. Avoid touching conductive components, e.g. gold contacts.

## 1.5 Scope

These quick start instructions describe the ETHERNET StarterKit 2 with the Ethernet fieldbus controller 750-841 and some basic components of the WAGO-I/O-SYSTEM 750.

You will find detailed information about operation, assembly and start-up in the manuals “Ethernet TCP/IP 750-841“ and “WAGO-I/O-PRO CAA“. This documentation is hence only valid in connection with the appropriate manuals.

**Further information**
You can also find the manuals “Ethernet TCP/IP 750-841“ and “WAGO-I/O-PRO CAA“ on our “WAGO-I/O-PRO CAA“ CD-ROM (759-911) and in the Internet under www.wago.com.

# 2 Quick Start

## 2.1 Node Assembly

Build up the fieldbus node as follows (from left to right):

750-841; 750-400 ; 750-501 ; 750-600.

Connect the supplies for the controller and for the power jumper contacts to the DC 24 V power supply unit 787-602 (see Fig.1).

In this example it would be sufficient to bridge “24V“ and “+“ or “0V“ and “-“.

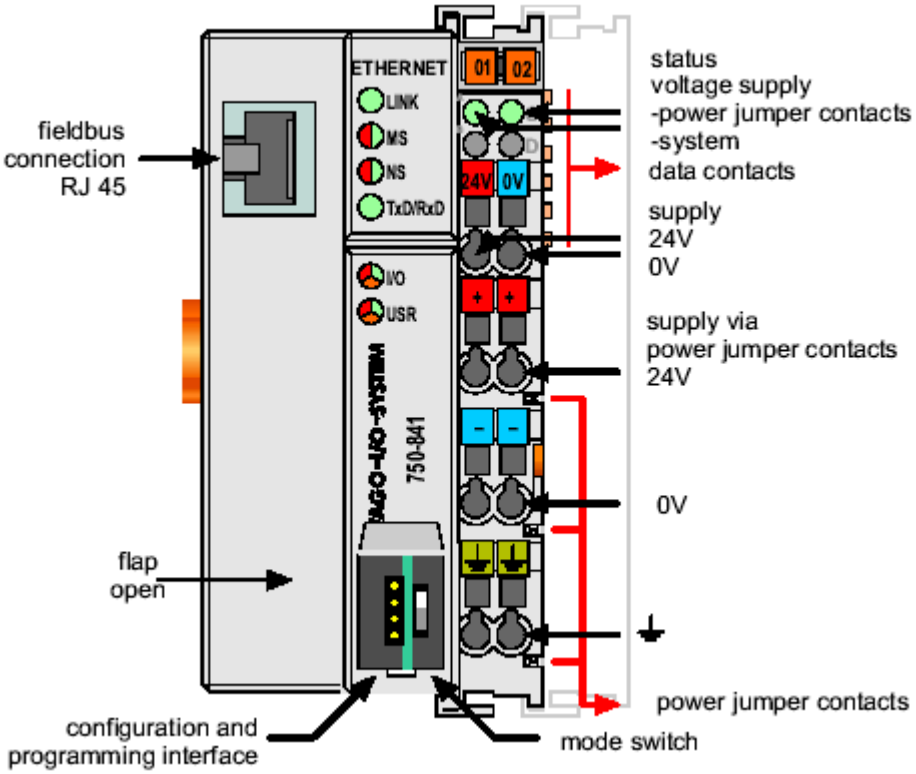



Fig.1: 10/100 MBit ETHERNET fieldbus controller 750-841

## 2.2 Ethernet Network Start-Up

Allocating the IP address is done very conveniently using both the included serial programming cable and the “WAGO Ethernet Settings“ program.
Connect your PC with the configuration and programming interface of the node.
Install the “WAGO Ethernet Settings“ program, which you can find either on the “Tools & Docs“ CD-ROM or on the WAGO Web pages under Service → Downloads → Software.
After starting, the program identifies the current settings of the node.

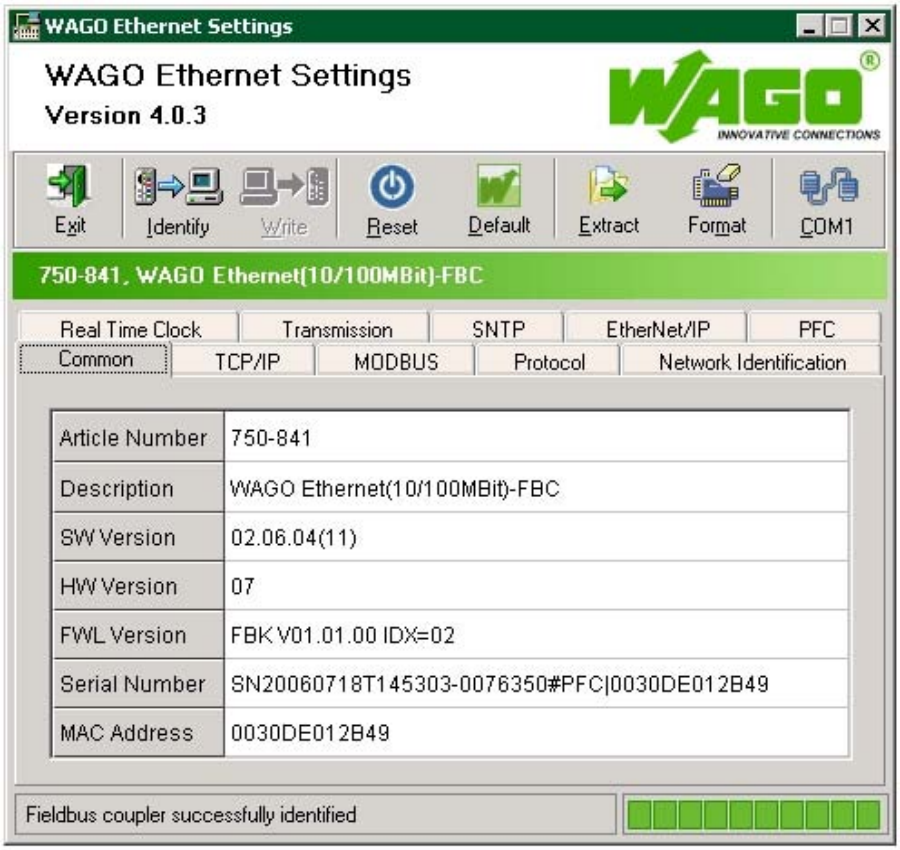
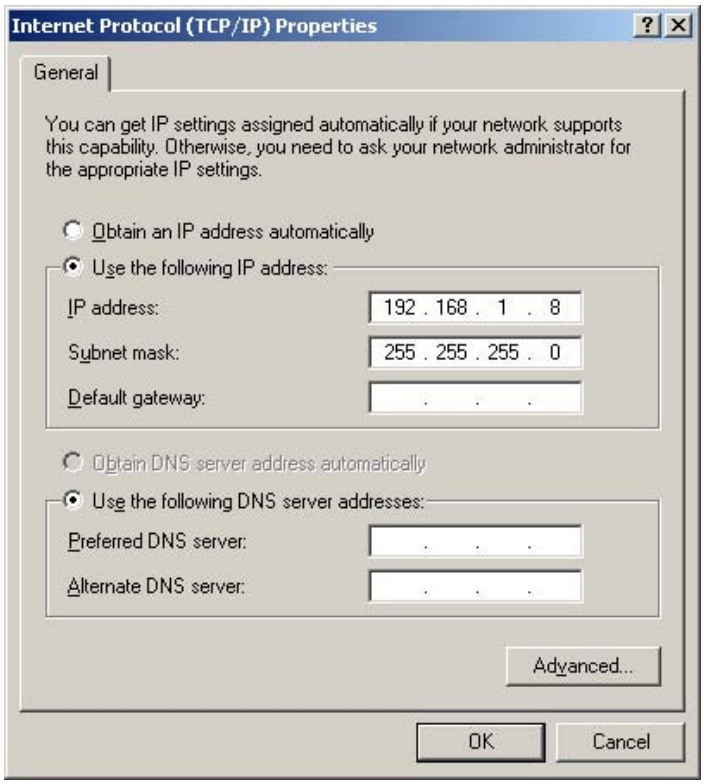



Fig. 2: WAGO Ethernet Settings

If a connection timeout occurs, you can retry the connection by using the “Identify“ button. If the problem persists, check the COM port settings and cabling.

An IP address always consists of both network and computer addresses. The network mask determines which bits belong to the network address and which do not.




```
IP-Address    : 192.168.  1.8
SubNetMask    : 255.255.255.0

Network-Adr. : 192.168.  1
Computer-Adr.:             8
```

Fig. 3: Settings of the PC network card

Select a free computer address from the local network. You can find out the network address of your PC under Start → Settings → Network and Dial-up connections →LAN connection x → Properties. To do so, select the Internet protocol (TCP/IP) from the selection list and press the ”Properties“ button.

**Attention**
Assigning IP addresses twice can lead to serious network failures.

Click on the “TCP/IP“ tab and enter an IP address for the node. Then, press the ”Write“ button to write the change to the node.

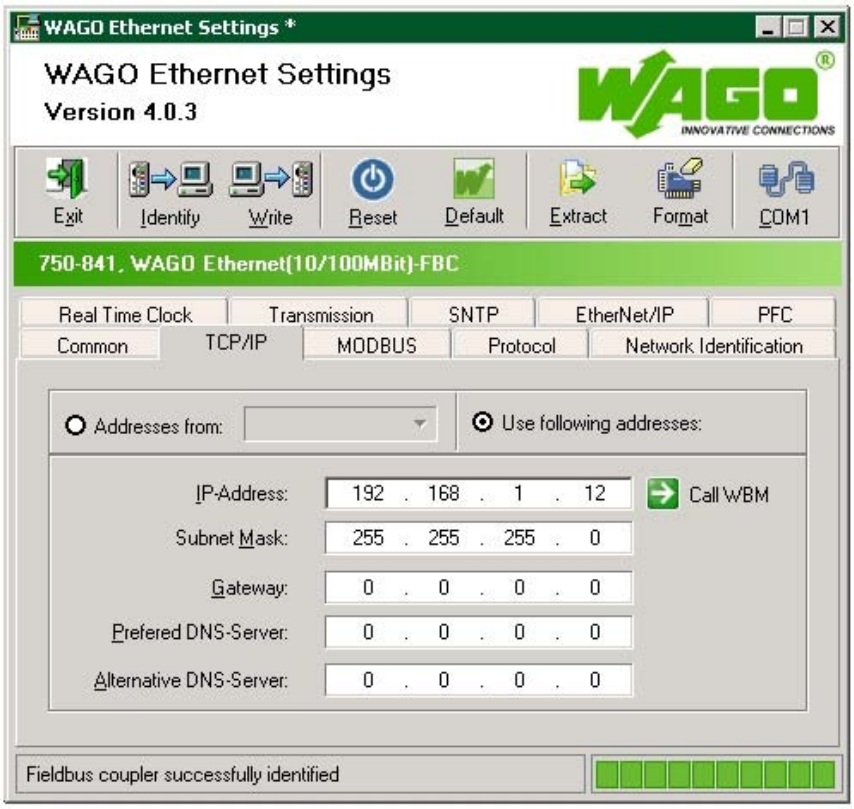



Fig. 4: Settings of the 750-841 Controller IP address

Now connect the node to the Ethernet network either directly to a PC using a crossover cable or to a Hub or Switch using a 1-to-1 patch cable.

**Attention**
The IP address of the controller can also be assigned using the ”WAGO BootP Server“ instead of the ”WAGO Ethernet Settings“. In this case, the allocation is done via the Ethernet interface of the device so that no separate serial cable is required. However, the assigned IP address will only be used temporarily. In the Web Base Management (WBM), the BootP protocol must be deactivated subsequently in order to save the IP address permanently.

To test the connection, open the DOS shell via Start → Programs → Accessories→ Command prompt, and send a PING to the IP address of the node.

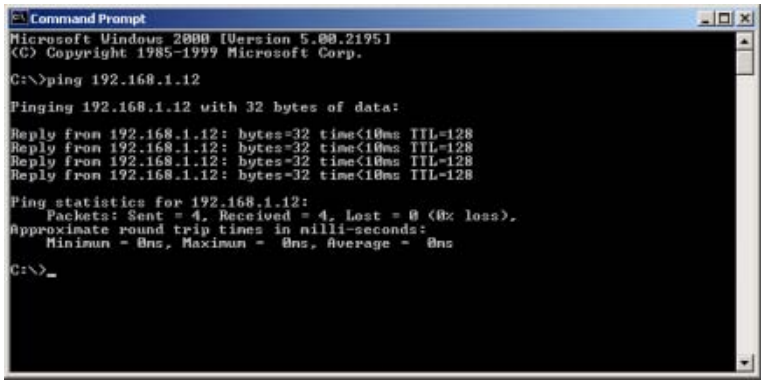



Fig. 5: Testing the connection to the 750-841 Controller with ”Ping“

If you get no response from the node, contact your network administrator.

All settings can also be viewed and modified via the **W**eb**B**ased**M**anagement (WBM).
To do so, start your browser and enter http:// followed by the IP address of the node (e.g. http://192.168.1.12).

The integrated Web server generates the following Web page:

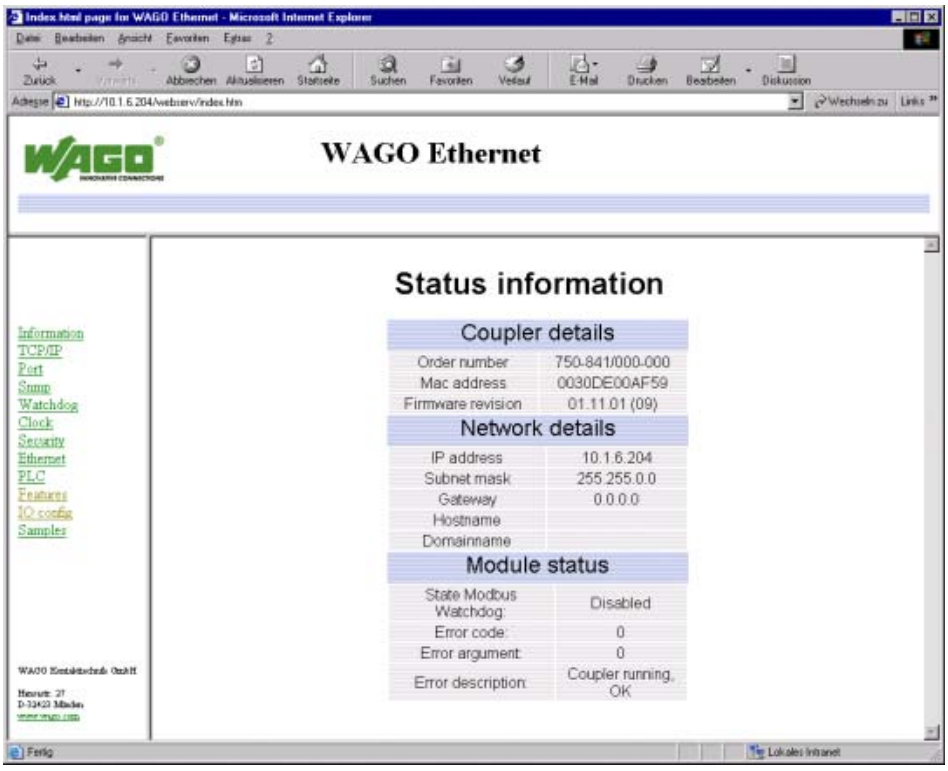



Fig. 6: Home page of the 750-841 Controller’s integrated WEB server

When using a proxy server, the proxy server must be bypassed for local addresses. Information on how to bypass the proxy server for local addresses can be found in the help section of your browser under “Proxy server” or “LAN settings”.

A login is required to access the configuration pages via hyperlinks.
The following users are created by default:

| User | Password |
|---|---|
| admin | wago |
| user | user |
| guest | guest |

Fig. 7: Users and passwords of the 750-841 Controller’s WEB server

Using firmware version (09) and above, the process data can be displayed in the right window of the ”IO config“ configuration page. Access to process data is based on the “GenIoConf.xml“ file in the “etc“ folder of the node’s file system. Data generation is disabled by default and can be activated on the “Features“ configuration page.

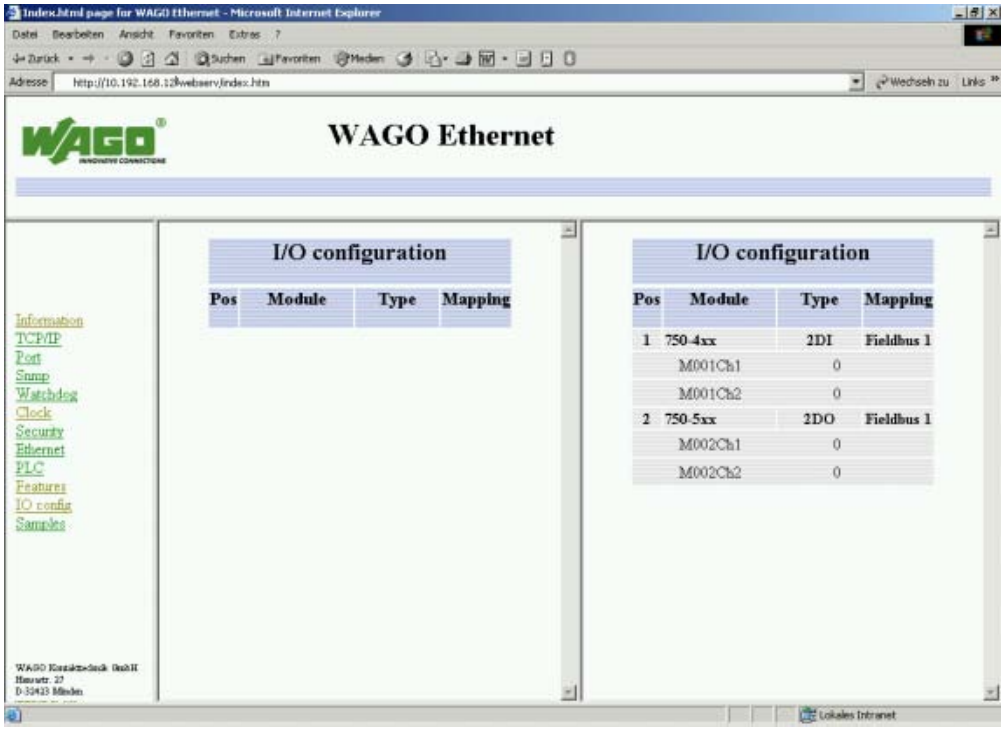



Fig. 8: “IO config“ page of the integrated WEB server

The left-side window of the “IO config“ configuration page shows the hardware configuration that has been created by CoDeSys as usual.
Write permission to the outputs is assigned by the hardware configuration either to the “PLC“, the fieldbus 1(Modbus) or the fieldbus 2 (Ethernet_IP).

## 2.3 The First Program

**Prerequisite:**
The WAGO-I/O-PRO CAA software (CoDeSys) has to be installed correctly and started.

You can start a new project via → File → New.

Example:

Before programming the 750-841 controller, select the controller in the “Target Settings“ window.


Fig. 9: Target settings

Please select the “WAGO_750-841“ target in this window.

In use of WAGO-I/O-PRO CAA, starting from version 2.3.9.3, you select the target system "WAGO_750-841_(FW12 -...)", if the WAGO Ethernet Controller 750-841 has already the firmware version 13.
This is to recognize on the basis of the manufacturing number, look at the 5th and 6th number.

**Further information**
Detailed Information about the manufacturing number, please refer to the manual „Ethernet TCP/IP 750-841”.
You can find this manual on the CD-ROM „WAGO-I/O-PRO CAA“ (759-911), on the CD-ROM „ELECTRONICC Tools & Docs“ (Item.-No.: 0888-0412/0001-0101) and also on the Internet under:
www.wago.com

Then confirm the selected target with “OK“ in the dialog box above.

A POU can now be created in the following dialog box. In this example, a new “PLC_PRG“ is created in “ST“ programming language (Fig. 10).

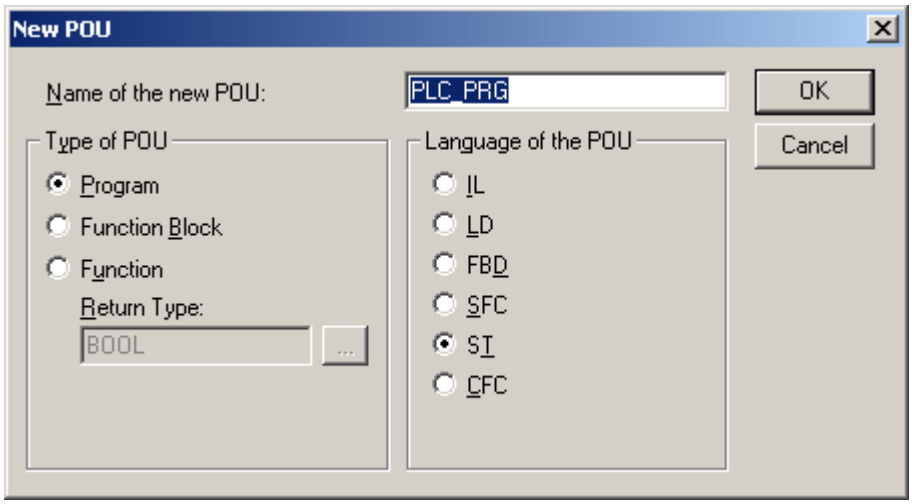



Fig. 10: WAGO-I/O-PRO CAA, creating a new POU

Before starting to create a POU, a PLC configuration must be performed.

To do so, change to PLC configuration in the “Resources“ register.

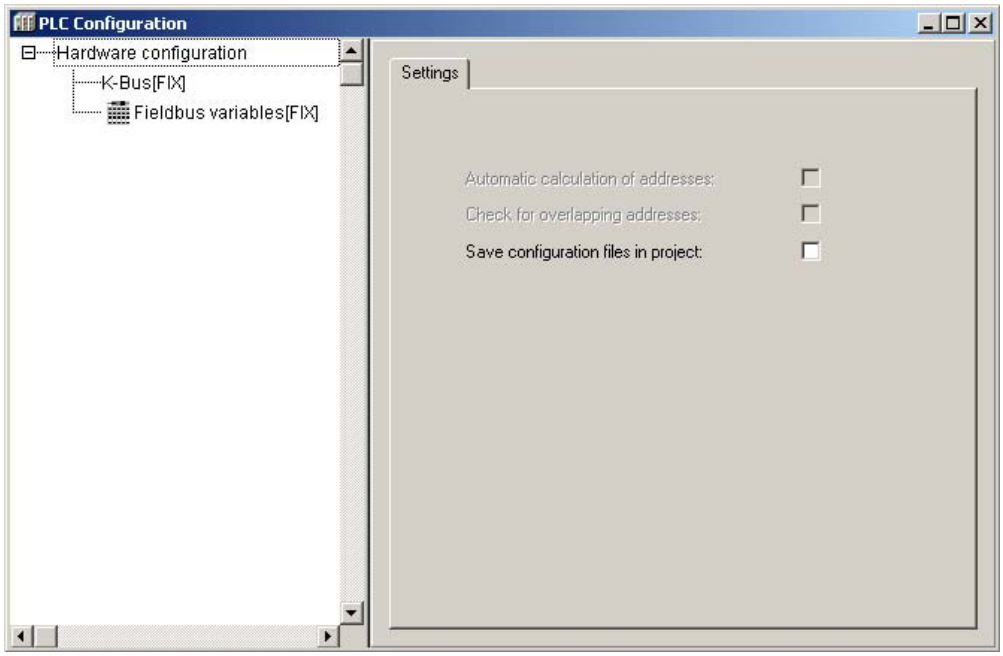



Fig. 11: WAGO-I/O-PRO CAA, PLC configuration

Select „K-Bus[FIX]“ in the PLC configuration and then right-click the mouse to open the context menu.

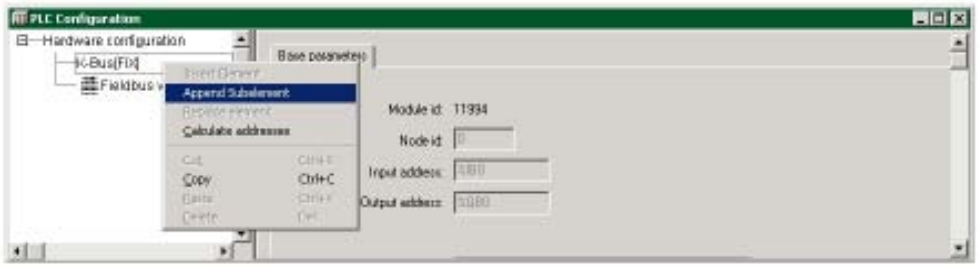



Fig. 12: WAGO-I/O-PRO CAA, I/O configuration, context menu

Select ”Append sub-element“. This will open the I/O configuration window with the I/O module catalog.

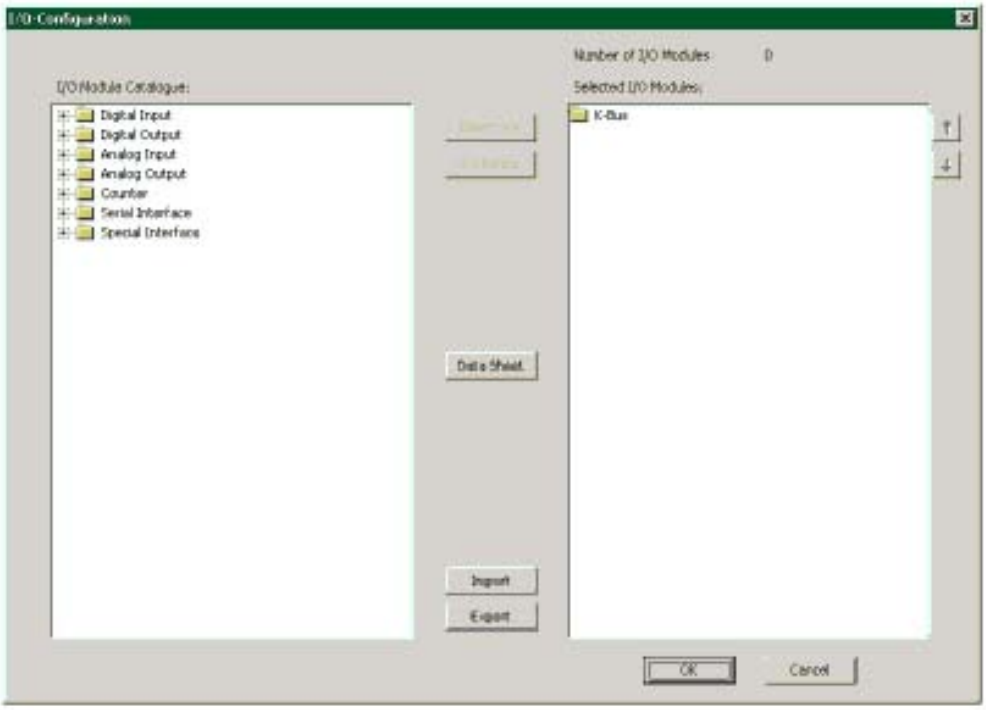



Fig. 13: WAGO-I/O-PRO CAA, with I/O configuration dialog

.In the application example shown in section 2.1, a single 750-400 digital input module has been connected to the 750-841 controller, followed by a 750-501 digital output module. The 750-600 end module is arranged on the very right hand side.

In the I/O module catalog on the left, first click on “Digital Input” to select the 750-400 module and then move this module to the right into your selection using the “Insert“ button.

Then, select the 750-501 module from the I/O module catalog on the left by clicking on “Digital Output” and move it to the right into your selection using the “Insert“ button.

If you wish to delete some modules from your selection, select the appropriate module on the right-hand side and click the ”Delete“ button.

You can change the position of a module in your selection by selecting and moving the module using the up and down arrow buttons on the right-hand side of the window.

The 750-600 end module is not shown in the I/O configuration window.

Then confirm your selection with “OK“.

For each I/O module, you can now determine via → Module parameters if the outputs of a module will be controlled by the controller (PLC) or by a particular Ethernet protocol (fieldbus 1 = Modbus/TCP or fieldbus 2 = Ethernet/IP) (Fig. 14).

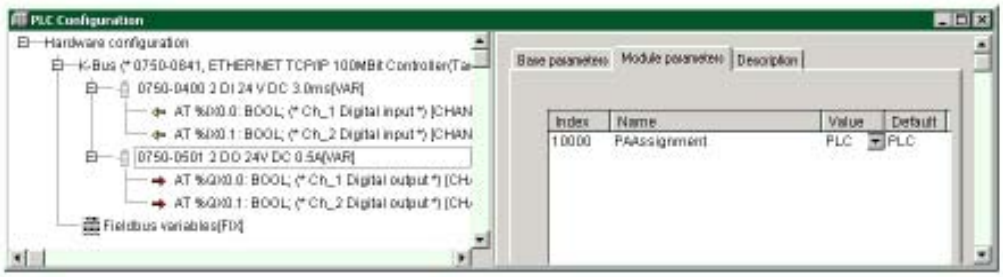



Fig. 14: WAGO-I/O-PRO CAA, PLC configuration with 750-400 and -501

In this example, we first used the “PLC“ value.

The I/O configuration window now shows the address of each channel, and thus each individual bit on the module.

Left clicking on the arrow next to “AT“ will open a text box in which you can enter a variable name (Fig. 15).

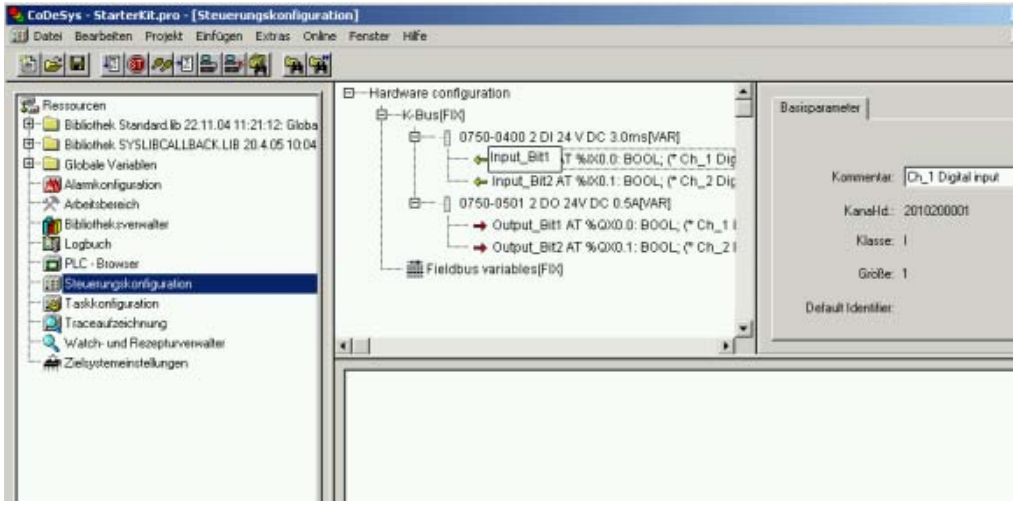



Fig. 15: WAGO-I/O-PRO, Declaration of variables with I/O configuration dialog

Now enter the following variable names:

Input_Bit1 (for the %IX0.0 address)
Input_Bit2 (for the %IX0.1 address)
Output_Bit1 (for the %QX0.0 address)
Output_Bit2 (for the %QX0.1 address)

Then, open the “PLC_PRG“ program from the “POUs“ register and complete a simple assignment within the statement part of the program:
Output_Bit1 := Input_Bit1;
Output_Bit2 := Input_Bit2;

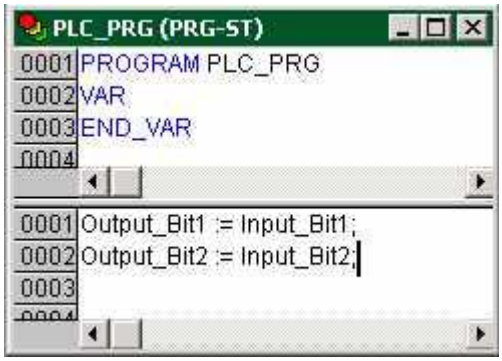



Fig. 16: WAGO-I/O-PRO CAA, Declaration and statement part of the PLC-PRG program

This way, both the first two inputs and first two outputs are assigned to each other (Fig. 16).

The program can now be loaded if the test program has been successfully compiled.

Compilation is initiated via → Project → Rebuild all.

## 2.4 Program Download to the Controller 750-841

Click → Communication Parameters in the online menu and create a new communication channel.
Choose EthernetTCP/IP (3S TCP/IP driver).

Enter the IP address of the coupler under “Address“ (e.g. 192.168.1.12).
Enter port number 2455.

Please note that *the simulation is deactivated*.

You can now download the program via → Online → Login.

Start the program via → Online → Run.

**Note**
If the program should be permanently stored and be available without renewed download on the Controller after a voltage failure, the menu “Online” -> “Create Boot Project” is selected in the logged in condition. Thus the compiled project is stored zero potential protected in the Flash memory. So that the Controller starts the program automatically after voltage return, the mode switch (fig. 1) is to switch into the upper position „RUN “.

## 2.5 Alternative to the Variable Declaration in the PLC Configuration

Instead of addressing bits individually and globally in the PLC configuration, a whole byte can also be addressed locally in the declaration part of the “PLC-PRG“ program.

Open the “PLC_PRG“ program under the “POUs“ register and declare the local process image as follows:

| | | | |
|---|---|---|---|
| Input_Byte | AT | %IB0: | BYTE; |
| Output_Byte | AT | %QB0: | BYTE; |

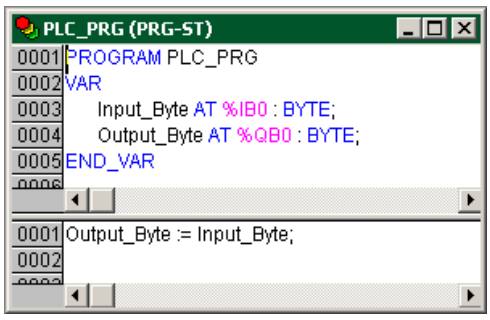



Fig. 17: WAGO-I/O-PRO CAA, Declaration and statement part of the PLC-PRG program

A simple assignment in the statement part of the program could be:
Output_Byte := Input_Byte;

This way, the inputs are assigned to the outputs (Fig. 17).

The program can now be loaded if the test program has been successfully compiled.

Compilation is activated via → Project → Rebuild all.
(Continue in section 2.4 “Program Download to the 750-841 Controller”)

## 2.6 Modbus/TCP DLL Application Examples

Also included is the “WAGO DLL Modbus/TCP“ (759-312) CD-ROM. The **D**ynamic**L**ink**L**ibrary (DLL) encapsulates the functionality of a Modbus master hence facilitating the creation of your own Windows-based applications. The procedural DLL can be used in almost any programming language. On the CD-Rom, you will find examples for Excel VBA, ”Visual Basic 6“, ”vb.net“, ”Delphi 5“, ”MS Visual C++ 6“, ”C#“ and ”LabVIEW 7”.

In addition, you will find the two ActiveX components “ActiveMBT“(Master) and “ActiveMBTSlave“ that are used in Visual Basic.
Registration is provided by a setup program that will also install the “Modbus Monitor“ program shown below.

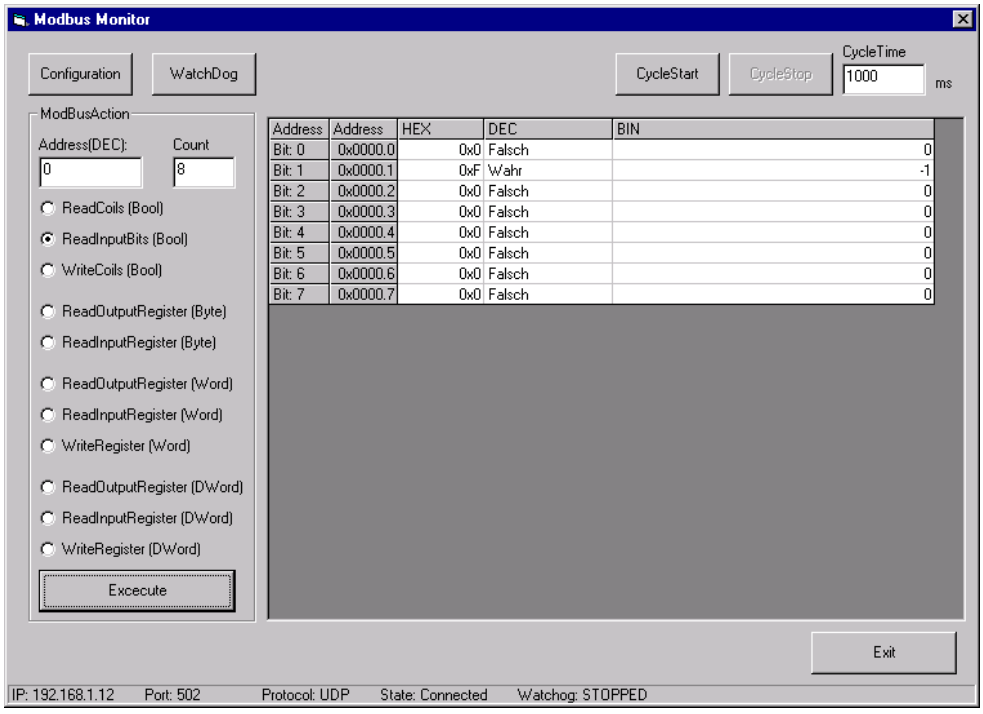



Fig. 18: Modbus/TCP DLL, Modbus Monitor

The “Modbus Monitor“ provides a graphical user interface for all functionalities of the “ActiveMBT“ ActiveX code component and allows reading and writing of data via the Modbus protocol.

Further information and examples can be found on the ”WAGO DLL Modbus/TCP“ (759-312) CD-ROM.

## 2.7 Information for Programming and Web Visualization

**Further information**
Detailed Information about the programming and visualization, especial the Web Visualization, please refer to the manuals „WAGO-I/O-PRO CAA (CoDeSys)“.
You can find these manuals on the CD-ROM „WAGO-I/O-PRO CAA“ (759-911), on the CD-ROM „ELECTRONICC Tools & Docs“ (Item.-No.: 0888-0412/0001-0101) and also on the Internet under: www.wago.com

# 3 Addressing

## 3.1 Comparison of Modbus and IEC1131 Addresses

In this section, you will find a list of memory ranges for both the Modbus addressable 750-841 Controller and 750-341 Coupler.

The following address spaces are available for Modbus communication using the 750-841 Controller from WAGO.

| Address space | Modbus access | PLC access | Description |
|---|---|---|---|
| Phys. inputs (1) | read | read | Physical inputs 1(%IW0 ... %IW255) |
| Phys. outputs (1) | read/write | read/write | Physical outputs 1 (%QW0 ... %QW255) |
| PFC-IN variables | read/write | read | Volatile PLC input variables (%IW256 ... %IW511) |
| PFC-OUT variables | read | read/write | Volatile PLC output variables (%QW256 ... %QW511) |
| Configuration register | read/write | --- | See manual |
| Firmware register | read | --- | See manual |
| RETAIN variables | read/write | read/write | Remanent memory (%MW0 ... %MW12288) |
| Phys. inputs (2) | read | read | Physical inputs 2(%IW512...%IW1275) |
| Phys. outputs (2) | read/write | read/write | Physical outputs 2 (%QW512 ... %QW1275) |

In contrast to the WAGO 750-841 Controller, only the following address spaces are available when communicating via the WAGO 750-341 Coupler:

| Address space | Modbus access | Description |
|---|---|---|
| Phys. inputs (1) | read | Physical inputs 1(%IW0 ... %IW255) |
| Phys. outputs (1) | read/write | Physical outputs 1(%QW0 ... %QW255) |
| Configuration register | read/write | See manual |
| Firmware register | read | See manual |
| Phys. inputs (2) | read | Physical inputs 1(%IW512 ... %IW1275) |
| Phys. outputs (2) | read/write | Physical outputs 1(%QW512 ... %QW1275) |

Modbus addresses are merely a numerical value from 0 to 65535 which is used to determine the start address of an operation.
The type of operation like writing and/or reading word or digital signals is defined by the “FunctionCode“.
In Modbus, a register represents a 16-bit value (WORD) and a “coil“ is a digital output (BIT).

Further information about Modbus can be found in the manual.

## 3.1.1 WORD Access

The following tables display the MODBUS addressing and the corresponding IEC61131 addressing for the process image, the PFC variables, the NOVRAM data, and the internal variables is represented.
Via the register services the states of the complex and digital I/O modules can be determined or changed.

**Register (Word) Access Reading (with FC3, FC4 and FC23):**

| Modbus-Address [dec] | Modbus-Address [hex] | IEC61131 Address | Memory Range |
|---|---|---|---|
| 0<br>... 255 | 0x0000<br>... 0x00FF | %IW0<br>... %IW255 | Physical Input Area (1)<br>First 256 Words of physical input data |
| 256<br>... 511 | 0x0100<br>... 0x01FF | %QW256<br>... %QW511 | PFC-OUT-Area<br>Volatile PFC Output variables |
| 512<br>... 767 | 0x0200<br>... 0x02FF | %QW0<br>... %QW255 | Physical Output Area (1)<br>First 256 Words of physical output data |
| 768<br>... 1023 | 0x0300<br>... 0x03FF | %IW256<br>... %IW511 | PFC-IN-Area<br>Volatile PFC Input variables |
| 1024<br>... 4095 | 0x0400<br>... 0x0FFF | - | Modbus Exception:<br>“Illegal data address” |
| 4096<br>... 12287 | 0x1000<br>... 0x2FFF | - | Configuration Register |
| 12288<br>... 24575 | 0x3000<br>... 0x5FFF | %MW0<br>... %MW12287 | NOVRAM<br>8kB retain memory (max. 24kB) |
| 24576<br>... 25340 | 0x6000<br>... 0x62FC | %IW512<br>... %IW1275 | Physical Input Area (2)<br>Additional 764 Words physical input data |
| 25341<br>... 28671 | 0x62FD<br>... 0x6FFF | - | Modbus Exception: “<br>Illegal data address” |
| 28672<br>... 29436 | 0x7000<br>... 0x72FC | %QW512<br>... %QW1275 | Physical Output Area (2)<br>Additional 764 Words physical output data |
| 29437<br>... 65535 | 0x72FD<br>... 0xFFFF | - | Modbus Exception: “<br>Illegal data address” |

## Register (Word) Access Writing (with FC6, FC16, FC22 and FC23):

| Modbus-Address [dec] | Modbus-Address [hex] | IEC61131 Address | Memory Range |
|---|---|---|---|
| 0<br>... 255 | 0x0000<br>... 0x00FF | %QW0<br>... %QW255 | Physical Output Area (1)<br>First 256 Words of physical output data |
| 256<br>... 511 | 0x0100<br>... 0x01FF | %IW256<br>... %IW511 | PFC-IN-Area<br>Volatile PFC Input variables |
| 512<br>... 767 | 0x0200<br>... 0x02FF | %QW0<br>... %QW255 | Physical Output Area (1)<br>First 256 Words of physical output data |
| 768<br>... 1023 | 0x0300<br>... 0x03FF | %IW256<br>... %IW511 | PFC-IN-Area<br>Volatile PFC Input variables |
| 1024<br>... 4095 | 0x0400<br>... 0x0FFF | - | Modbus Exception:<br>“Illegal data address” |
| 4096<br>... 12287 | 0x1000<br>... 0x2FFF | - | Configuration Register |
| 12288<br>... 24575 | 0x3000<br>... 0x5FFF | %MW0<br>... %MW12287 | NOVRAM<br>8kB retain memory (max. 24kB) |
| 24576<br>... 25340 | 0x6000<br>... 0x62FC | %QW512<br>... %QW1275 | Physical Output Area (2)<br>Additional 764 Words physical output data |
| 25341<br>... 28671 | 0x62FD<br>... 0x6FFF | - | Modbus Exception: “<br>Illegal data address” |
| 28672<br>... 29436 | 0x7000<br>... 0x72FC | %QW512<br>... %QW1275 | Physical Output Area (2)<br>Additional 764 Words physical output data |
| 29437<br>... 65535 | 0x72FD<br>... 0xFFFF | - | Modbus Exception: “<br>Illegal data address” |

## 3.1.2 Bit Access

The digital Modbus services (coil services) are Bit accesses, with which only the states of digital I/O modules can be determined or changed. Complex I/O modules are not attainable with these services and so they are ignored. Because of this the addressing of the digital channels begins again with 0, so that the MODBUS address is always identical to the channel number, (i.e. the digital input no. 47 has the MODBUS address "46").

**Bit Access Reading (with FC1 and FC2):**

| Modbus-Address [dec] | Modbus-Address [hex] | Memory Range | Description |
|---|---|---|---|
| 0<br>... 511 | 0x0000<br>... 0x01FF | Physical Input Area (1) | First 512 digital inputs |
| 512<br>... 1023 | 0x0200<br>... 0x03FF | Physical Output Area (1) | First 512 digital outputs |
| 1024<br>... 4095 | 0x0400<br>... 0x0FFF | - | Modbus Exception:<br>“Illegal data address” |
| 4096<br>... 8191 | 0x1000<br>... 0x1FFF | %QX256.0<br>...%QX511.15 | PFC-OUT-Area<br>Volatile PFC Output variables |
| 8192<br>... 12287 | 0x2000<br>... 0x2FFF | %IX256.0<br>...%IX511.15 | PFC-IN-Area<br>Volatile PFC Input variables |
| 12288<br>... 32767 | 0x3000<br>... 0x7FFF | %MX0<br>... %MX1279.15 | NOVRAM<br>8kB retain memory (max. 24kB) |
| 32768<br>... 34295 | 0x8000<br>... 0x85F7 | Physical Input Area (2) | Starts with the 513$^{th}$ and ends with the 2039$^{th}$ digital input |
| 34296<br>... 36863 | 0x85F8<br>... 0x8FFF | | Modbus Exception:<br>“Illegal data address” |
| 36864<br>... 38391 | 0x9000<br>... 0x95F7 | Physical Output Area (2) | Starts with the 513$^{th}$ and ends with the 2039$^{th}$ digital output |
| 38392<br>... 65535 | 0x95F8<br>... 0xFFFF | | Modbus Exception:<br>“Illegal data address” |

### Bit Access Writing (with FC5 and FC15):

| Modbus-Address [dec] | Modbus-Address [hex] | Memory Range | Description |
|---|---|---|---|
| 0<br>... 511 | 0x0000<br>... 0x01FF | Physical Output Area (1) | First 512 digital outputs |
| 512<br>... 1023 | 0x0200<br>... 0x03FF | Physical Output Area (1) | First 512 digital outputs |
| 1024<br>... 4095 | 0x0400<br>... 0x0FFF | - | Modbus Exception:<br>“Illegal data address” |
| 4096<br>... 8191 | 0x1000<br>... 0x1FFF | %IX256.0<br>...%IX511.15 | PFC-IN-Area<br>Volatile PFC Input variables |
| 8192<br>... 12287 | 0x2000<br>... 0x2FFF | %IX256.0<br>...%IX511.15 | PFC-IN-Area<br>Volatile PFC Input variables |
| 12288<br>... 32767 | 0x3000<br>... 0x7FFF | %MX0<br>... %MX1279.15 | NOVRAM<br>8kB retain memory (max. 24kB) |
| 32768<br>... 34295 | 0x8000<br>... 0x85F7 | Physical Output Area (2) | Starts with the 513$^{th}$ and ends with the 2039$^{th}$ digital output |
| 34296<br>... 36863 | 0x85F8<br>... 0x8FFF | | Modbus Exception:<br>“Illegal data address” |
| 36864<br>... 38391 | 0x9000<br>... 0x95F7 | Physical Output Area (2) | Starts with the 513$^{th}$ and ends with the 2039$^{th}$ digital output |
| 38392<br>... 65535 | 0x95F8<br>... 0xFFFF | | Modbus Exception:<br>“Illegal data address” |

**Note**

The “DigitalOffSet“ of the digital inputs is 0, when no analog input modules / specialty modules and no ”SET_DIGITAL_INPUT_OFFSET“ POU (from the ”mod_com.lib“ library) are used.
The “DigitalOffSet“ of the digital outputs is 0, when no analog output modules / specialty modules and no “SET_DIGITAL_OUTPUT_OFFSET“ POU (from the “mod_com.lib“ library) are used.

WAGO Kontakttechnik GmbH & Co. KG
Postfach 2880 • D-32385 Minden
Hansastraße 27 • D-32423 Minden
Phone: 05 71/8 87 – 0
Fax: 05 71/8 87 – 1 69
E-Mail: info@wago.com

Internet: http://www.wago.com