BAUMÜLLER

be in motion be in motion

BM4-O-ECT-02

Ethernet with EtherCAT-Master for b maXX drive PLC

Application Manual

E	5.07002.02

Title	Application Manual
Product	Ethernet with EtherCAT-Master for b maXX drive PLC BM4-O-ECT-02
Last Revision:	January 30, 2008

Binding nature

This Application Manual is a part of the unit/machine. This Application Manual must always be available to operators and be legible. If the unit/machine is sold, the owner must pass on this Application Manual together with the unit/machine.
After selling the unit/machine you must pass on this original and all the copies that you made to the purchaser. After disposing of the machine in any way, you must destroy this original and all the copies that you made.

When you pass on this Application Manual, all earlier revisions of the corresponding Application Manuals are invalidated.
Note that all the data/numbers/information that are quoted are **current values at the time of printing.** This information is **not legally binding** for dimensioning, calculation and costing.
Within the scope of further-development of our products, Baumüller Nürnberg GmbH reserve the right to change the technical data and handling.

We cannot guarantee this Application Manual is completely error-free unless this is expressly indicated in our General Conditions of Business and Delivery.

Manufacturer

Baumüller Nürnberg GmbH
Ostendstr. 80 - 90
90482 Nuremberg
Germany
Tel. +49 9 11 54 32 - 0 Fax: +49 9 11 54 32 - 1 30
www.baumueller.de

TABLE OF CONTENTS

1

INTRODUCTION

This application manual is an important part of your b maXX 4400 device. It is important to read through this documentation fully, especially in the interest of your own safety.

This chapter describes the initial procedure.

1.1 Initial procedure

1 To program the Ethernet with EtherCAT master option module, you will require the following hardware:
- a b maXX 4400 basic unit,
- a b maXX drive PLC option module and
- an Ethernet with EtherCAT master for b maXX drive PLC option module

You must have installed the hardware in accordance with the Operating Instructions in each case and it must be ready for operation.

2 You will also need the following software:
- ProMaster for configuration of an EtherCAT network
- ProProg wt III for programming the b maXX drive PLC and the Ethernet with EtherCAT master for b maXX drive PLC option module.

NOTE

The ProProg wt III programming system is required to program the b maXX drive PLC BM4-O-PLC-0x in conjunction with ProMaster.

1.2 Terms Used

The terms "Ethernet plug-in module" and "EtherCAT master" or "EtherCAT master plug-in module" are used for the "Ethernet with EtherCAT master for b maXX drive PLC" (BM4-O-ECT-02) product.

We will also use the term "b maXX" for the "Basic Unit b maXX 4400" product. The controller in the basic unit is also referred to as the "b maXX controller".

A list of abbreviations used is found in ▷Appendix A - Abbreviations◁ from page 93 onward.

1.3 Conditions

This application manual builds on the application manual of the b maXX drive PLC and requires knowledge of the ProMaster engineering framework and its online help and the ProProg wt III programming system and the accompanying Online Help System.

2

BASIC SAFETY INSTRUCTIONS

We have designed and manufactured each Baumüller plug-in module in accordance with the strictest safety regulations. Despite this, working with the plug-in module can be dangerous for you.

In this chapter, we will describe the risks that can occur when working with a Baumüller plug-in module. Risks are illustrated by icons. All the symbols that are used in this documentation are listed and explained.

In this chapter, we cannot explain how you can protect yourself from specific risks in individual cases. This chapter contains only general protective measures. We will go into concrete protective measures in subsequent chapters directly after information about the individual risk.

2.1 Hazard information and instructions

WARNING

The following **may occur**, if you do not observe this warning information:

- serious personal injury
- death

The hazard information is showing you the hazards which can lead to injury or even to death.

Always observe the hazard information given in this documentation.

Hazards are always divided into three danger classifications. Each danger classification is identified by one of the following words:

DANGER

- Considerable damage to property
- Serious personal injury
- Death **will** occur

WARNING

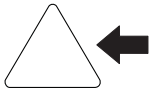
- Considerable damage to property
- Serious personal injury
- Death **can** occur

CAUTION

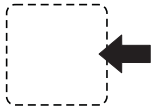
- Damage to property
- Slight to medium personal injury **can** occur

2.1.1 Structure of hazard information

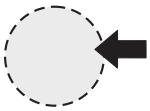
The following two examples show how hazard information is structured in principle. A triangle is used to warn you about danger to living things. If there is no triangle, the hazard information refers exclusively to damage to property.



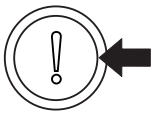
A triangle indicates that there is danger to living things.
The color of the border shows how severe the hazard is: the darker the color, the more severe the hazard is.



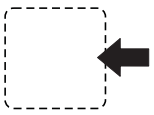
The icon in the rectangle represents the hazard.
The color of the border shows how severe the hazard is: the darker the color, the more severe the hazard is.



The icon in the circle represents an instruction. Users must follow this instruction.
(The circle is shown dashed, since an instruction is not available as an icon for each hazard advisory).



The circle shows that there is a risk of damage to property.



The icon in the rectangle represents the hazard.
The color of the border shows how severe the hazard is: the darker the color, the more severe the hazard is. (The rectangle is shown dashed, since the danger is not represented as an icon with every hazard advisory)

The text next to the icons is structured as follows:

THE SIGNAL WORD IS HERE THAT SHOWS THE DEGREE OF RISK

Here we indicate whether one or more of the results below occurs if you do not observe this warning.

- Here, we describe the possible results. The worst result is always at the extreme right.

Here, we describe the hazard.

Here, we describe what you can do to avoid the hazard.

2.1.2 Hazard advisories that are used

If a signal word is preceded by one of the following danger signs: ⚠ or ⚠ or ⚠, the safety information refers to injury to people.

If a signal word is preceded by a round danger sign: Ⓘ, the safety information refers to damage to property.

2.1.2.1 Hazard advisories about injuries to people

To be able to differentiate visually, we use a separate border for each class of hazard information with the triangular and rectangular pictograms.

For danger classification **DANGER,** we use the ⚠ danger sign. The following hazard information of this danger classification is used in this documentation.

DANGER

The following **will occur**, if you do not observe this danger information:

- serious personal injury
- death

*Danger from: **electricity.** The hazard may be described in more detail here.*

Here, we describe what you can do to avoid the hazard.

DANGER

The following **will occur**, if you do not observe this danger information:

- serious personal injury
- death

*Danger from: **mechanical effects.** The hazard may be described in more detail here.*

Here, we describe what you can do to avoid the hazard.

For danger classification **WARNING,** we use the ⚠ danger sign. The following hazard information of this danger classification is used in this documentation.

WARNING

The following **may occur**, if you do not observe this warning information:

- serious personal injury
- death

*Danger from: **electricity.**The hazard may be described in more detail here.*

Here, we describe what you can do to avoid the hazard.

For danger classification **CAUTION,** we use the ⚠ danger sign. The following hazard information of this danger classification is used in this documentation.

CAUTION

The following **may occur**, if you do not observe this caution information:

- minor to medium personal injury.

*Danger from: **sharp edges.** The hazard may be described in more detail here.*

Here, we describe what you can do to avoid the hazard.

CAUTION

The following **may occur**, if you do not observe this danger information:

- environmental pollution.

*Danger from: **incorrect disposal.** The hazard may be described in more detail here.*

Here, we describe what you can do to avoid the hazard.

2.1.2.2 Hazard advisories about damage to property

If a signal word is preceded by a round danger sign: ⓘ, the safety information refers to damage to property.

CAUTION

The following **may occur**, if you do not observe this caution information:

- property damage.

*Danger from: **electrostatic discharge.** The hazard may be described in more detail here.*

Here, we describe what you can do to avoid the hazard.

2.1.2.3 Instruction signs that are used

wear safety gloves

wear safety shoes

2.2 Information signs

NOTE

This indicates particularly important information.

2.3 Legal information

This documentation is intended for technically qualified personnel that has been specially trained and is completely familiar with all warnings and maintenance measures.

The equipment is manufactured to the state of the art and is safe in operation. It can be put into operation and function without problems if you ensure that the information in the documentation is complied with.

Operators are responsible for carrying out servicing and commissioning in accordance with the safety regulations, applicable standards and any and all other relevant national or local regulations with regard to cable rating and protection, grounding, isolators, overcurrent protection, etc.

Operators are legally responsible for any damage that occurs during assembly or connection.

2.4 Appropriate Use

You must always use the plug-in module appropriately. Some important information is listed below. The information below should give you an idea of what is meant by appropriate use of the plug-in module. The information below has no claim to being complete; always observe all the information that is given in these operating instructions.

- You must only install the plug-in module in series b maXX 4400 units.
- Configure the application such that the module is always operating within its specifications.
- Ensure that only qualified personnel works with this plug-in module.
- Mount the plug-in module only in the specified slot/slots.
- Install the plug-in module as specified in this documentation.
- Ensure that connections always comply with the stipulated specifications.
- Operate the plug-in module only when it is in technically perfect condition.
- Always operate the plug-in module in an environment that is specified in the technical data.
- Always operate the plug-in module in a standard condition.
  For safety reasons, you must not make any changes to the plug-in module.
- Observe all the information on this topic if you intend to store the plug-in module.

You will be using the plug-in module in an appropriate way if you observe all the comments and information in these operating instructions.

2.5 Inappropriate Use

Below, we will list some examples of inappropriate use. The information below should give you an idea of what is meant by inappropriate use of the plug-in module. We cannot, however, list all possible cases of inappropriate use here. Any and all applications in which you ignore the information in this documentation are inappropriate; particularly, in the following cases:

- You installed the plug-in module in units that are not Series b maXX 4400.
- You ignored information in the operating instructions of this module.
- You did not use the module as intended.
- You handled the module as follows
  - you mounted it incorrectly,
  - you connected it incorrectly,
  - you commissioned it incorrectly,
  - you operated it incorrectly,
  - you allowed non-qualified or insufficiently qualified personnel to mount the module, commission it and operate it,
  - you overloaded it,
  - You operated the module
    - with defective safety devices,
    - with incorrectly mounted guards or without guards at all,
    - with non-functional safety devices and guards
    - outside the specified environmental operating conditions
- You modified the module without written permission from Baumüller Nürnberg GmbH.
- You ignored the maintenance instructions in the component descriptions.
- You incorrectly combined the module with third-party products.
- You combined the drive system with faulty and/or incorrectly documented third-party products.
- Your self-written PLC software contains programming errors that lead to a malfunction.

Version 1.1 of Baumüller Nürnberg GmbH's General Conditions of Sale and Conditions of Delivery dated 2/15/02 or the respective latest version applies in all cases. These will have been available to you since the conclusion of the contract at the latest.

2.6 Protective equipment

In transit, the modules are protected by their packaging. Do not remove the module from its packaging until just before you intend to mount it.

The cover on the b maXX units' controller sections provides IP20 protection to the plugin modules from dirt and damage due to static discharges from contact. This means that you must replace the cover after successfully mounting the plug-in module.

2.7 Personnel training

WARNING

The following **may occur**, if you do not observe this warning information:

- serious personal injury
- death

Only qualified personnel are allowed to mount, install, operate and maintain equipment made by Baumüller Baumüller Nürnberg GmbH.

Qualified personnel (specialists) are defined as follows:

Qualified Personnel

Electrical engineers and electricians of the customer or of third parties who are authorized by Baumüller Nürnberg GmbH and who have been trained in installing and commissioning Baumüller drive systems and who are authorized to commission, ground and mark circuits and equipment in accordance with recognized safety standards.

Qualified personnel has been trained or instructed in accordance with recognized safety standards in the care and use of appropriate safety equipment.

Requirements of the operating staff

The drive system may only be operated by persons who have been trained and are authorized.

Only trained personnel are allowed to eliminate disturbances, carry out preventive maintenance, cleaning, maintenance and to replace parts. These persons must be familiar with the Operating Instructions and act in accordance with them.

Commissioning and instruction must only be carried out by qualified personnel.

2.8 Safety measures in normal operation

- At the unit's place of installation, observe the applicable safety regulations for the plant in which this unit is installed.
- Provide the unit with additional monitoring and protective equipment if the safety regulations demand this.
- Observe the safety measures for the unit in which the module is installed.

2.9 Responsibility and liability

To be able to work with this option module in accordance with the safety requirements, you must be familiar with and observe the hazard information and safety instructions in this documentation.

2.9.1 Observing the hazard information and safety instructions

In the operating instructions of this option module, we use visually consistent safety instructions that are intended to prevent injury to people or damage to property.

WARNING

The following **may occur**, if you do not observe this warning information:

- serious personal injury
- death

Any and all persons who work on and with Series b maXX units must always have available the Operating Instructions of this option module and must observe the instructions and information they contain – this applies in particular to the safety instructions.

Apart from this, any and all persons who work on this unit must be familiar with and observe all the rules and regulations that apply at the place of use.

2.9.2 Danger arising from using this module

The Ethernet with EtherCAT-Master option module for b maXX drive PLC has been developed and manufactured to the state of the art and complies with applicable guidelines and standards. It is still possible that hazards can arise during use. For an overview of possible hazards, refer to the chapter entitled ▹Basic Safety Instructions ◃ from page 7 onward.
We will also warn you of acute hazards at the appropriate locations in this documentation.

2.9.3 Warranty and Liability

All the information in this documentation is non-binding customer information; it is subject to ongoing further development and is updated on a continuous basis by our permanent change management system.

Warranty and liability claims against Baumüller Nürnberg GmbH are excluded; this applies in particular if one or more of the causes listed in ▹Inappropriate Use ◃ from page 12 onward or below caused the fault:

- Disaster due to the influence of foreign bodies or force majeure.

3

ETHERNET

In this chapter, you will find information about connecting the option module to an Ethernet.

3.1 General information on Ethernet and the Ethernet option module

Ethernet is an established and proven technology for data transfer in information technology and in office communication. In the Ethernet option module, a twisted pair cable is used as the transfer medium (for more information, see the option module Operating Instructions). This means that you can easily implement a link to an existing 10BaseT network or to a 100BaseTX network (IEEE 802.3 standard).

You can use the Ethernet option module on a network containing other components like PCs, hubs, switches and repeaters. Alternatively, you can make a direct connection to a PC's network adapter by means of a crosslink cable; in this case, you do not need any additional network components.

For communication, an Ethernet-based TCP/IP protocol stack and the

- PROPROG / OmegaOS communication

application protocol are implemented.

This protocol allows you to use the following functionality for ProProg wt III and OmegaOS:

- PLC control functions like Start and Stop
- Program management
- Online debugging, watch windows
- Logic Analyzer/oscilloscope
- OPC Server.

For further details and other functionality, refer to the Application Manual for the b maXX drive PLC and to the OPC server.

To use the functions stated above, you only need to set the communications port and the IP address in your ProMaster project. On the option module side, you must set the IP address, the subnet mask and the gateway or you must code them if you cannot use the default settings.

For more information on the individual steps that you have to carry out, refer to the Chapter entitled ▹Configuring Ethernet TCP/IP networks ◃ from page 30 onward.

The following chapters will give you an introduction to the structure of Ethernet-TCP/IP networks and the conditions for structuring communications connections between the Ethernet option module and other components on the network.

NOTE

The branching of Ethernet-TCP/IP networks can be very complex. There can also be a large number of components installed that are configured differently. The components influence the data traffic on the network. Apart from this, you must consider aspects of data security and unauthorized network access. This complexity means that the manual can only give you an overview in principle of using the Ethernet option module on Ethernet-TCP/IP networks. To get more details about your local area network (LAN), ask your network administrator and refer to the manuals of the network components.

3.2 Basics of Ethernet Networks and TCP/IP

3.2.1 Transfer standards

The Ethernet option module supports both of the most common physical models in the IEEE 802.3 standard:

10BaseT

Each network node is connected via its own twisted-pair cable to a (star) hub that forwards all the data packets equally to all the network nodes. Eight-pin RJ45 types are used as the plug-in connectors. The maximum length of a segment (the connection between the hub and the terminal equipment) is limited to 100 meters.

100BaseTX

The 100BaseTX transmission standard is similar to 10BaseT. Each network node is also connected via its own twisted-pair cable to a (star) hub that forwards all the data packets equally to all the network nodes. Eight-pin RJ45 types are also used as the plug-in connectors. Here too, the maximum length of a segment (the connection between the hub and the terminal equipment) is limited to 100 meters. However, all the components (cables, RJ45 outlets, etc.) must be rated for the higher transfer rate of 100 MHz.

The Ethernet option module automatically detects the transfer rate that is being used.

3.2.2 Structure of Ethernet networks with a hub and switch

There are various options for structuring a star topology network.

All the nodes are connected via one hub.

If a node sends a data packet, it is broadcast across the entire network and is available to each of the connected nodes. The message is further-processed in each case only by the node with the correct target address. Due to the large amounts of data, collisions can occur and the system must transfer messages again.

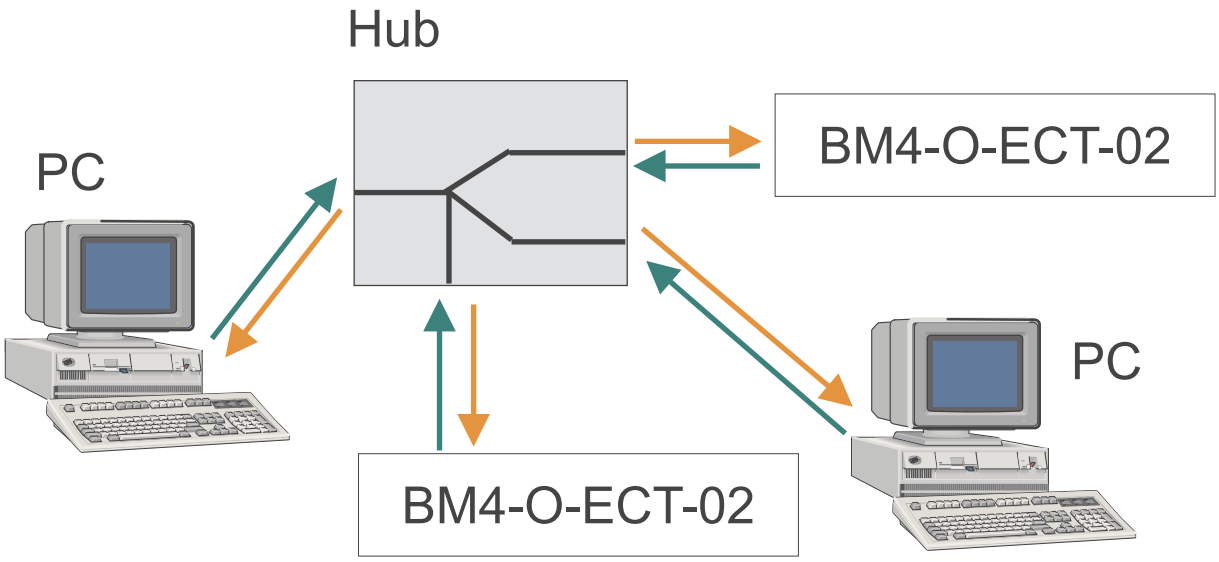


Figure 1: Principle of Shared Ethernet

All the nodes are connected via one switch.

In the case of a Switched Ethernet, a switch is used to connect several nodes. If data reaches the switch from a segment, the system checks the segment to which this data is to be transmitted. The system sends the data exclusively to the node with the correct target address. This reduces the amount of data and, with this, the risk of data collisions on the network.

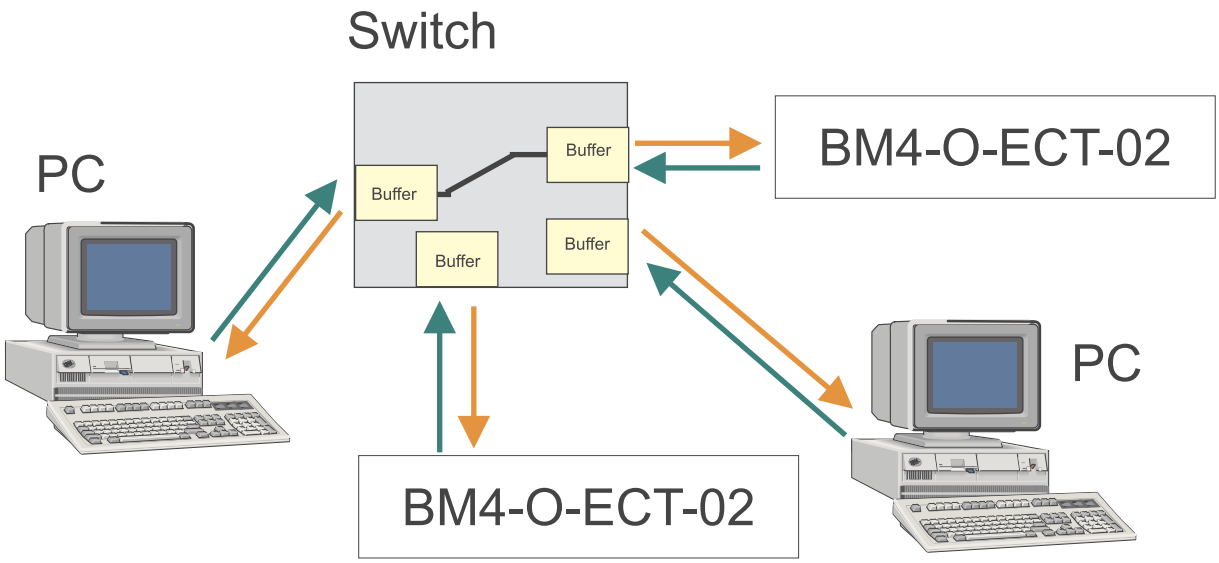


Figure 2: Principle of Switched Ethernet

By combining star structures, it is possible to form tree structures. Apart from this, it is possible to make the connection between the nodes at the TCP/IP level (see Chapter ▹TCP/IP ◃ from page 19 onward)

3.2.3 Ethernet address and MAC address

Each Ethernet option module has its own unique global Ethernet address. Ethernet addresses are also referred to as MAC addresses or MAC-IDs (Media Access Control Identity). The Ethernet address is permanently stored in the option module and you cannot change it. Note the difference between this address and the IP address, which has a different meaning and which you can change. The MAC address is used at the Ethernet level for addressing. It has a fixed length of 6 bytes (48 bits) and contains the manufacturer code and a consecutive manufacturer number.

3.2.4 Ethernet data packet

Data packets are transferred on a connection-free basis, i.e. the sender does not get a feedback message from the receiver. The user data is packed in a frame of address information. The structure of an Ethernet data packet of this type looks like this:

Preamble	Destination	Source	Type	Data	FCS

Figure 3: Ethernet data packet

Preamble:	A bit sequence for identifying the start of the packet
Destination:	The receiver's Ethernet address
Source:	The sender's Ethernet address
Type:	Indicates the higher level purpose (e.g. IP)
Data:	User data contains the higher level protocols, e.g. IP
FCS:	Checksum

The preamble is for synchronizing between the sending and receiving station.

The Ethernet header contains the MAC addresses of the sender and the receiver and a type field for identifying the subsequent protocol that is contained in the data area.

3.2.5 Bridges and Repeaters

Bridges also work at the level of the Ethernet (passing on data on the basis of the MAC-ID). Bridges connect subnets and decide on the basis of the MAC-ID the data packets that are to pass the bridge and the ones that are not to pass. By contrast with the switch, a bridge does not pass on broadcast messages (data packets to several nodes at the same time).

Repeaters are amplifiers for refreshing signals and they do not change the contents of the data packets. If a repeater detects an error on one of the connected segments, the system disconnects the connection to this segment. As soon as the error has been eliminated, the system establishes the connection again.

3.2.6 TCP/IP

Ethernet alone is not enough to connect several different networks for data transfer. Let us take a PC, for example, that is connected to an Internet Service Provider by means of an external modem and can use this connection to communicate with other servers on the Internet. In this case, there are other connections like the serial or USB connection to the modem and the phone line as transfer media in addition to the Ethernet. The TCP/IP protocol was developed to continue a connection outside the Ethernet too. TCP/IP is usually used as a common term, you must, however, differentiate between two different protocols.

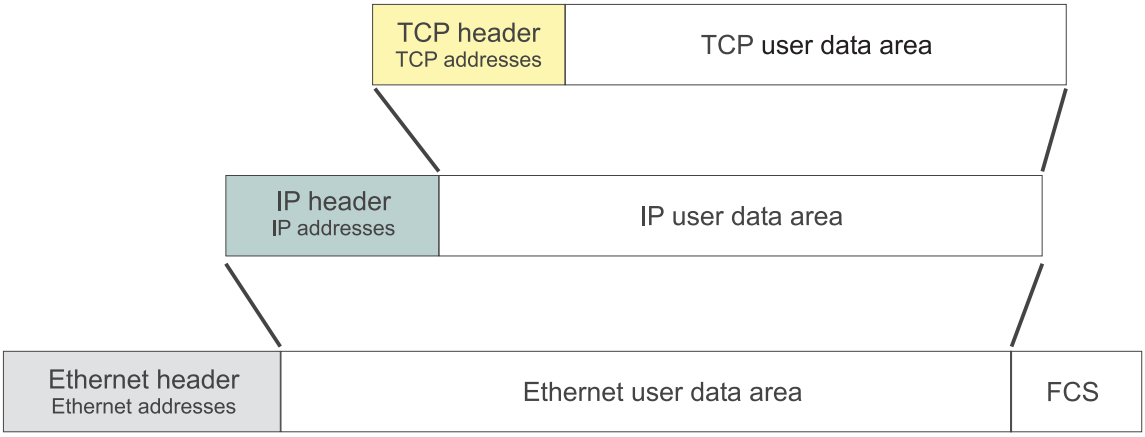


Figure 4: Structure of a TCP/IP-Ethernet data packet

3.2.6.1 IP protocol

Using **IP (Internet Protocol),** you go beyond the limits of a LAN (local area network). IP carries out correct addressing and delivery of data packet across a gateway to other networks. IP is in the Ethernet user data area. This means that IP packs the data that is received from higher levels into a separate frame. This packet is passed on to the Ethernet below it and it represents the user data of one or more Ethernet message frames.

Amongst other things, this IP frame contains a new form of addresses (Internet address, IP number). Under IP, each network node has its own unique Internet address (at least in a specific subnet). This is the basis of forwarding across the limits of an Ethernet segment and linking to non-Ethernet-based networks.

a) Router

You connect two or more different IP networks via routers. These routers decide on the basis of the IP address whether a data packet is to be forwarded to another network or not.

b) IP address

You must not confuse the IP address with the MAC-ID (or Ethernet address).

Users assign the IP address. As-delivered, the Ethernet option module has the IP address 192.168.1.1. For information on how to change this setting, refer to the chapter entitled ▷Configuring the Ethernet option module and checking settings ◁ from page 34 onward.

c) Structure of an IP address

The IP address consists of a total of 4 bytes that are normally shown in decimal form separated by periods, e.g. 192.168.1.1.

If you use the option module on a network whose nodes are only connected via a hub or switch and that does not have a connection to another network, you can assign virtually any IP address you like for each node. You must just ensure that you never set all the bits in any one byte equal to 0 or to 1 (byte = 0 or 255). These are reserved for special functions and you must not assign them. For example, you cannot use the address 194.11.0.13 due to the zero in the third byte.

If there is a connection to another network or to the Internet, you can, with the exceptions below, no longer freely assign the IP address:

For use on private networks, a corresponding IP standard reserves three address classes that may not be routed on the Internet, which means that they are not visible there. The addresses / address ranges are as follows:

- 10.xxx.xxx.xxx
- 172.16.xxx.xxx - 172.31.xxx.xxx
- 192.168.x.x

Routers on the LAN are generally set such that these address ranges are not routed. If it is not possible to communicate with an Ethernet option module, this may be due to the router on your LAN not routing the set IP address. The Ethernet option module's default IP address of 192.168.1.1 is an IP address of this type. Ask your network administrator which IP address range you may use if there is a connection to another network.

The address 127.0.0.1 is special, since it is always addressed as the local host. According to the standard, you are not allowed to use network address 127.x.x.x. This means that you can use address 127.0.0.1 to test the installation of your own equipment (e.g. a PC).

If you intend to connect network directly to the Internet, you can only use the unique IP addresses that are assigned by the IANA organization. Assignment depends on the country in which the network is operated.

NOTE

If the option module is in a network that has a connection to another network of if you want to make a direct Internet connection, you will need comprehensive knowledge of the entire LAN, of assigning IP addresses, of routing mechanisms and, in particular, of security requirements. This means that you should only make a connection to another network or establish an Internet connection together with an authorized network administrators.

d) Subnet mask and gateway

To make a connection to another network, you must address this network. The subnet mask is used for this. If a network node is to send a data packet, its own IP address is logically ANDed with the subnet mask and the destination IP address is logically ANDed with the subnet mask. If the network node gets the same result both times, it knows that the other node is on the same network. If the results are different, it is not possible to directly address the other network node. In the case, the node passes on the data packet to a gateway or router for transmission. A gateway differs from a router inasmuch as gateways can access non-TCP/IP networks. Since the receiver of the data packet must also carry out the logical operation with the subnet mask for the response, the subnet mask must also be set correctly on the receiver side.

NOTE

Since a router is only a special gateway, when configuring the IP address people often use the term "gateway" even though a router is physically present. This is particularly the case with the network configuration in Windows. With the Ethernet option module, too, we use the more general term "Gateway".

Users assign the subnet mask and the gateway or router IP address. For information on how to make these settings in the Ethernet option module, refer to the chapter entitled ▷Configuring the Ethernet option module and checking settings ◁ from page 34 onward.
The default settings are as follows:

- Subnet mask: 255.255.255.0
- Gateway: 0.0.0.0

The gateway IP address 0.0.0.0 with the Ethernet option module means that no gateway is used. This means that the data packet is also sent even if the system detects that due to the subnet mask link the receiver is not on the same network.

e) Subnet mask and network class

The InterNIC (International Network Information Center) divided the IP address into a "network section" and a "host section" to create what are known as address classes. The table below shows the different address classes, the assigned values of the most significant bit of the IP address and the division into the "network section" and the "host section".

Address class	Description	Address range of the network section	Possible number of hosts
Class A	The first byte of the IP address is for addressing the network section; the last three bytes address the host section	1.xxx.xxx.xxx to 126.xxx.xxx.xxx	Approximately 16 million
Class B	The first two bytes of the IP address are for addressing the network section; the last two bytes address the host section	128.0.xxx.xxx - 191.255.xxx.xxx	Approximately 65 thousand
Class C	The first three bytes of the IP address are for addressing the network section; the last byte addresses the host section	192.0.0.xxx to 223.255.255.xxx	254

In addition, there are Class D and Class E networks that are of no great significance in practical applications.

This subdivision also affects the subnet mask. Depending on the address class to which an IP address belongs, the subnet mask has a minimum value that is dependent on the permitted range of the IP address's "network section".

Address class	Minimum subnet mask
Class A	255.0.0.0
Class B	255.255.0.0
Class C	255.255.255.0

The Ethernet option module complies with these address class-dependent subnet mask, i.e. if the subnet mask that a user enters is less than the address class mask that belongs to the IP address, the system uses the address class mask.

Example:

A PC with ProMaster has an IP address of 192.075.191.188; an Ethernet option module has an IP address of 192.168.1.1. Both network nodes are located physically on the same network.

Since only the first bytes of both IP addresses are identical, you should choose 255.0.0.0 as the subnet mask on both systems to get the same result of logically ANDing the "IP address AND the subnet mask". However, since the IP addresses belong to a Class C network, the option module uses subnet mask 255.255.255.0. Logically ANDing the "IP address AND the subnet mask" yields different networks: 192.075.191.0 and 192.168.1.0. This means that communication is not possible, since the data packet are transferred to the set gateway.

You can avoid this problem and establish communication by deactivating the gateway and assigning gateway IP address 0.0.0.0.

f) Examples of IP networks

Example 1:

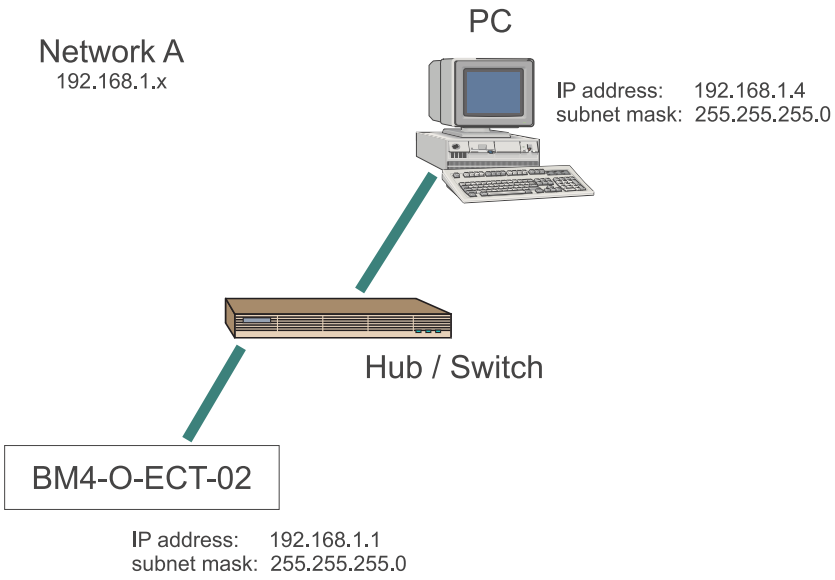


Figure 5: Example: Option module and PC on the same IP network

On network A, the components have the stated settings. Communication is possible between the PC and the option module, since logically ANDing both components "IP address AND subnet mask" yields the same network: 192.168.1.0

Example 2:

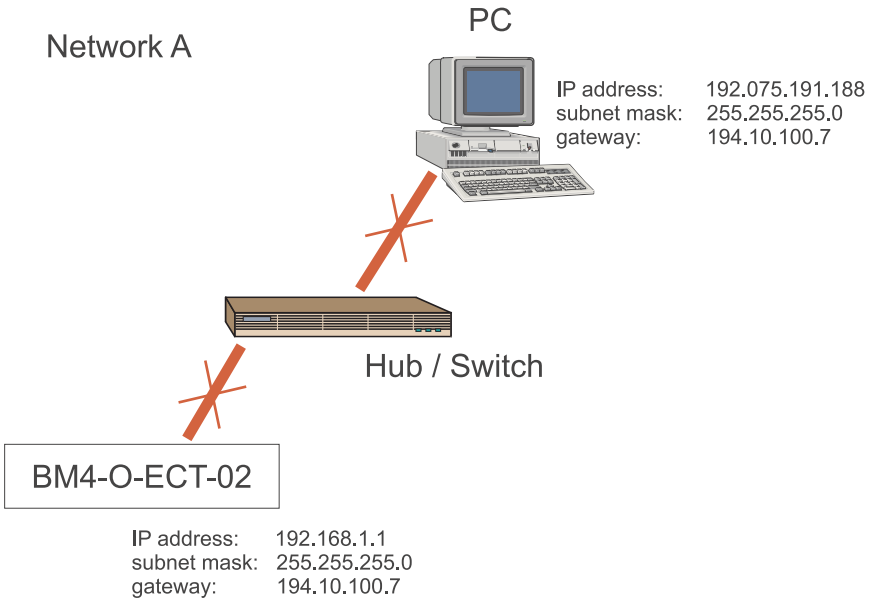


Figure 6: Example: Option module and PC on different IP networks

On network A, the components have the stated settings. Communication between the PC and the option module is not possible, since logically ANDing both components "IP address AND subnet mask" yields different networks. The PC and the option module would transfer the data packet to a gateway.

Example 3:

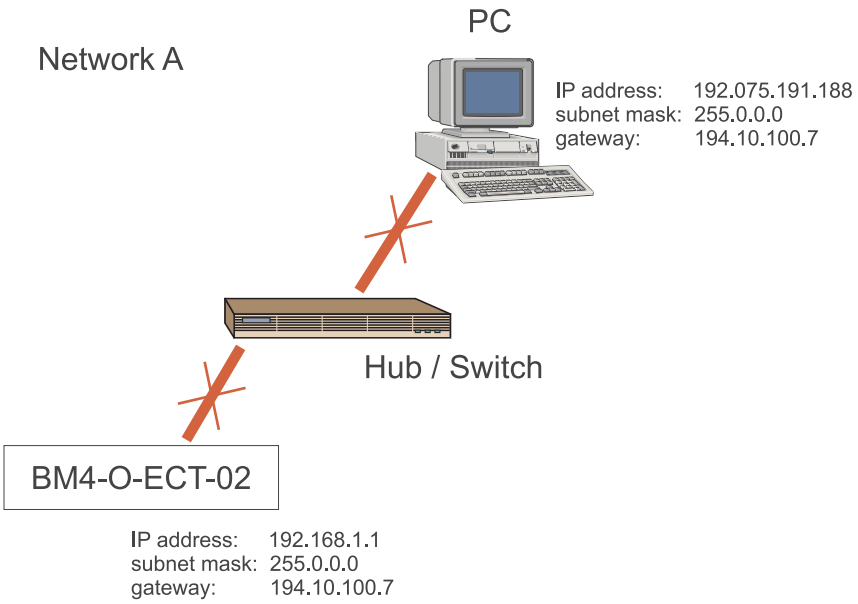


Figure 7: Example: Option module and PC with the wrong subnet mask

On network A, the components have the stated settings. Communication between the PC and the option module is not possible. Logically ANDing the "IP address AND the subnet mask" yields the same network 192.x.x.x. However, the option module detects a Class C network and uses subnet mask 255.255.255.0. and would transfer the data packets to a gateway. For a remedy, see example 4.

Example 4:

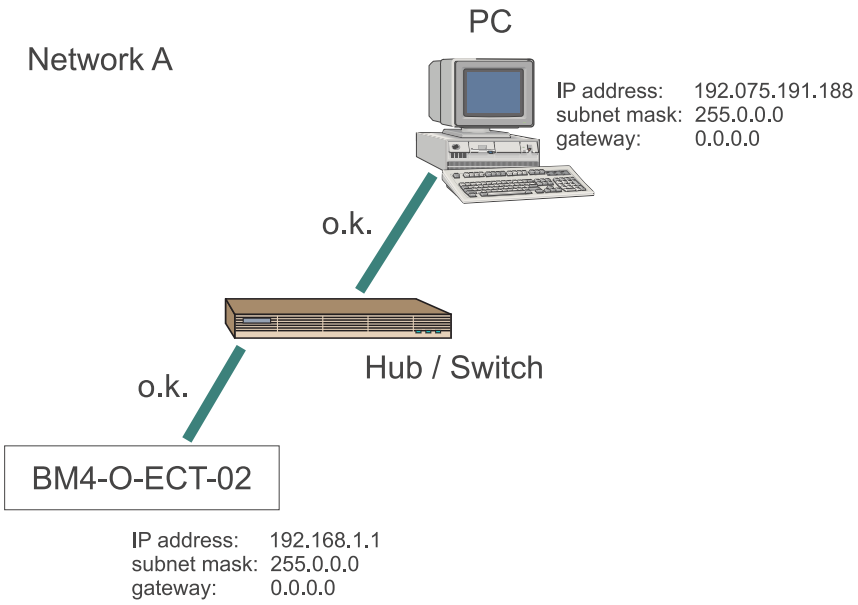


Figure 8: Example: Option module and PC without a gateway

On network A, the components have the stated settings. Communication between the PC and the option module is possible. Logically ANDing the "IP address AND the subnet mask" yields the network 192.x.x.x. However, the option module detects a Class C network and uses subnet mask 255.255.255.0. and would transfer the data packets to a gateway. However, since no gateway is set, the system does not transfer any data.

Example 5:

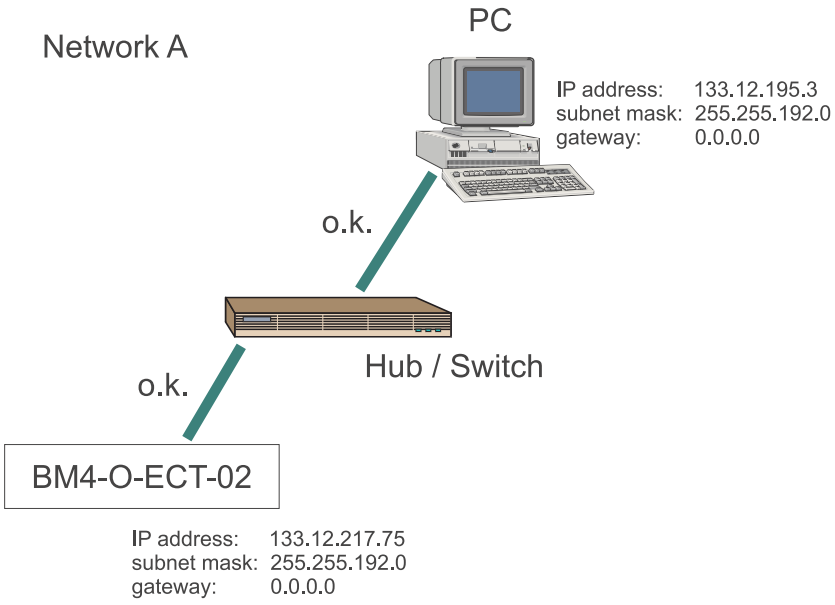


Figure 9: Example: Option module and PC on a Class-B IP network

On network A, the components have the stated settings. Communication is possible between the PC and the option module, since logically ANDing both components "IP address AND subnet mask" yields the same network: 133.12.192.0

Example 6:

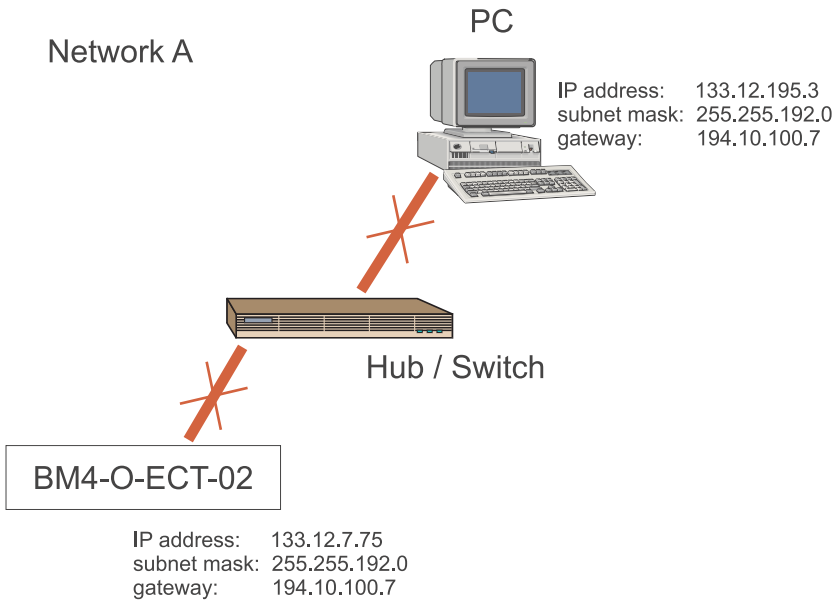


Figure 10: Example: Option module and PC on different Class-B IP networks

On network A, the components have the stated settings. Communication between the PC and the option module is not possible, since logically ANDing "the IP address AND the subnet mask" yields different networks: 133.12.192.0 and 133.12.0.0. The system would transfer the data packets to a gateway.

Example 7:

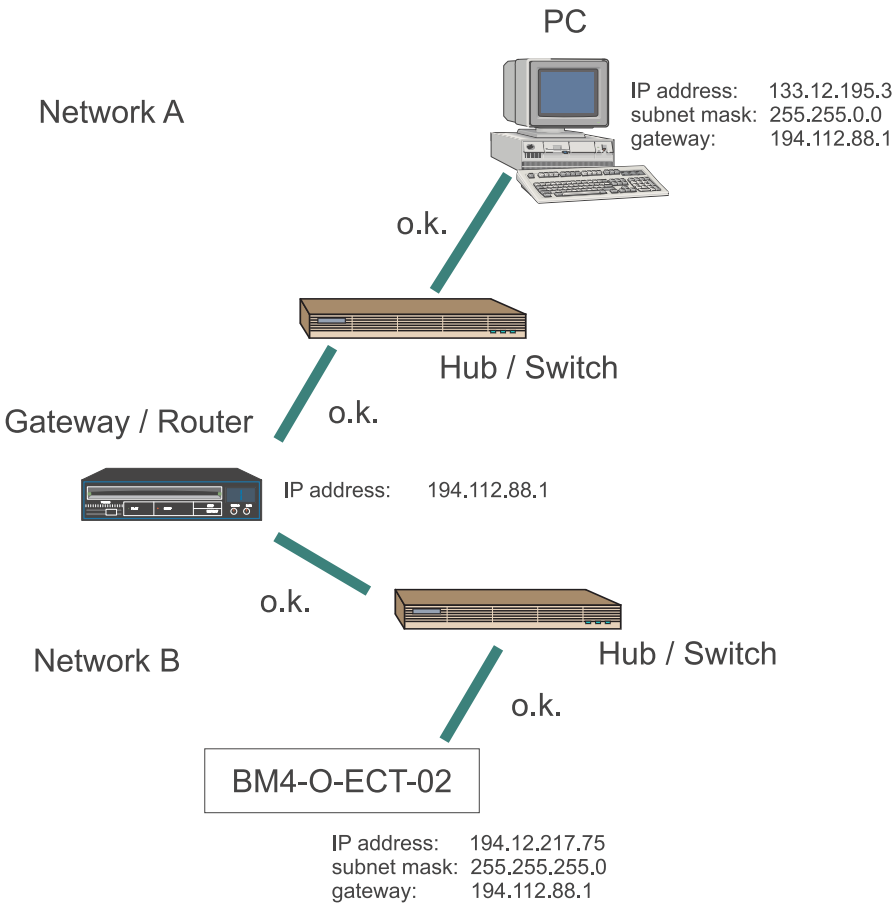


Figure 11: Example: Option module and PC with a gateway

On network A and network B, the components have the stated settings. The networks are connected via a gateway with IP address 194.112.88.1. Communication between the PC and the option module is possible, since logically ANDing both components with "IP address AND subnet mask" yields different networks – 133.12.0.0 and 194.12.217.0 – but both transfer data packets to the same gateway that carries out further- distribution.

3.2.6.2 TCP protocol

The **TCP (Transport Control Protocol)** is responsible for transporting and saving data and is located in the IP user data area. TCP works on the client server principle. For the duration of the data transfer, it makes a connection between two network nodes that is known as a session. Within any one session, the system divides up user data to individual numbered TCP packets and puts them back together again such that if necessary they can be transferred over different paths with different runtimes. Checksums make it possible to check and confirm correct reception. The system detects that individual TCP packets have been lost and it can request them again.

TCP packets are inserted in the user data area of an IP packet. Apart from this, TCP forwards the user data to the correct application program on the destination computer by assigning different port numbers to different applications (services).

The port number for die PROPROG communication is 0x5043 (= 20547 dec). The originating port is the port number of the sending application and at the same time the return sending port of the response.

As a user, you do not need to make any settings at the TCP level.

3.2.7 Interaction between Ethernet and TCP/IP, ARP

Up to now, addressing the nodes is only ensured by the IP addresses. At the Ethernet level, the network node only knows its own MAC-ID. To be able to deliver the IP packet in an Ethernet frame, however, the node must know the receiver's MAC-ID. The ARP protocol (Address Resolution Protocol) was developed to determine this. To find out the pair of Ethernet IP addresses of a network node, the ARP protocol sends an Ethernet broadcast containing the known IP address of the receiver. The receiver that has this IP address replies to the requesting system with a packet that contains the pair of Ethernet IP addresses. To avoid unnecessary ARP requests, the reply packet is stored on the requesting system in an ARP table. To keep this ARP table as small as possible, the system deletes the entries at specific, system-dependent time intervals. From the command line, you can read out the current ARP table for your PC:

```
arp -a
```

If communication has already taken place with the Ethernet option module, you will find the option module in the listing. To trigger communication, you can use the ping command from the command line.

```
ping 192.168.1.1
```

Assuming that your network is correctly configured (the IP addresses, subnet masks and the gateways), the option module with IP address 192.168.1.1 will reply in the following form:

```
Reply from 192.168.1.1: Bytes.....
```

If you do not get a reply, check your network settings.

If you do get a reply, the option module is now also listed in your PC's ARP table. It is not possible to read out the Ethernet option module's ARP table.

3.2.8 Proxy

A proxy is a computer that accepts requests from a network node and forwards them to the intended destination. The destination transfers the result of the request to the proxy, which, for its part, can return the result to the other network node. A proxy can work on different application levels and check the data to be transferred. If often has a large

amount of memory (a proxy cache) to buffer requested data and, in the case of a possible subsequent request from a different network node or from the same one, to make available the data to the actual destination on a direct basis, i.e. without needing an additional request. A firewall proxy is generally used if an internal network, e.g. an Intranet/LAN is intended to make a protected connection to another network (e.g. the Internet). In this connection, the proxy acts as a kind of gateway with IP and packet filters. On the basis of the origin, destination, port and packet type information that each packet contains, the proxy can inspect data transfer from one network to another one and if necessary it can limit it. Using appropriate software functions, the proxy can carry out further security tasks above the IP level.

3.2.9 Application layer ProProg wt III / OmegaOS

ProMaster / ProProg wt III / OmegaOS data packets are transferred in the user data area of TCP packets. Users do not need to make any settings here. You must only set the communications port in ProMaster (see the chapter entitled ▹Setting a connection via Ethernet-TCP/IP ◃ from page 41 onward). The Ethernet option module makes available two communications channels; i.e. using the option module you can operate in-parallel a maximum of

two ProMaster

or

one ProMaster and one ProProg wt III

or

two ProProg wt III

or

one ProMaster (or ProProg wt III) and one OPC application

or

two OPC applications

3.2.10 Ethernet over EtherCAT (EoE) - TCP/IP-Tunneling over EtherCAT

For the Ethernet communication to EtherCAT-Slaves (e.g. to a b maXX controller with EtherCAT-Slave) TCP packets within the EtherCAT packets are transmitted (tunneling). In this case an own IP address must be set at every EtherCAT-Slave. The EtherCAT-Slave is addressed as Ethernet partner via this IP address. However you set the IP address at EtherCAT-Slaves, see the respective Operating Instructions please.

3.3 Configuring Ethernet TCP/IP networks

3.3.1 Overview

To be able to use the Ethernet option module via Ethernet with TCP/IP using ProMaster or ProProg wt III or via OPC, you must carry out the following steps:

- physically commission the network (see the Operating Instructions of option module BM4-O-ECT-02 for b maXX drive PLC and the Operating Instructions of the network components)
- specify the network data: the IP addresses and subnet masks (see the previous chapter ▹TCP/IP ◃ from page 19 onward)
- set IP addresses, subnet masks and gateways in the network components (see the next chapter)
- set the communications port in ProMaster (see chapter entitled ▹Setting a connection via Ethernet TCP/IP in ProMaster ◃ from page 41 onward)

3.3.2 Configuring the Windows PC

Since ProMaster / ProProg wt III and the OPC server need a Windows operating system, we will only describe the settings that are necessary for Windows operating systems. The basic condition is that the TCP/IP protocol is installed on your Windows PC and that your computer has a configured network adapter. For details about this installation, refer to the Windows documentation.

3.3.2.1 Configuring TCP/IP under Windows XP

- Click on the *Start* pushbutton, and then choose My Network Places. Click on *Network Tasks* to View *Network Connections.*
- Double-click on the *Local Area Connection* icon and click on the *Properties* → pushbutton; the system displays a list of the components that use your network adapter.
- Check *Internet Protocol (TCP/IP)* and click on the *Properties* pushbutton.

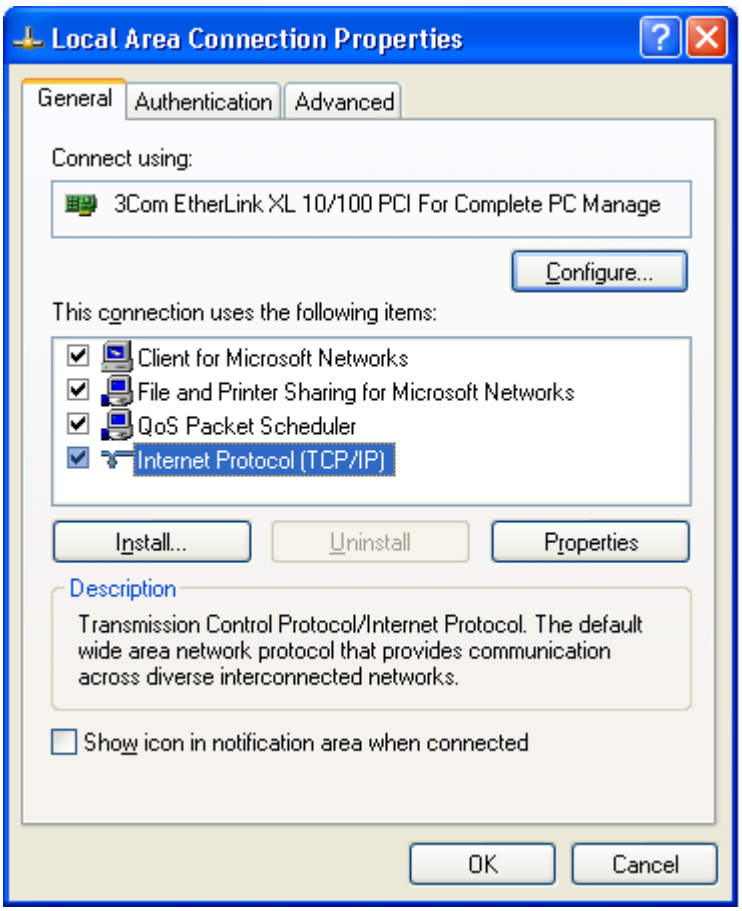


Figure 12: Setting up Windows XP with TCP/IP – overview of LAN connections

- You can now choose between *Obtain an IP address automatically* or *Use the following IP address*. If you obtain the IP address automatically, you can query the current setting either in field *Details* of *LAN connection* in the *Network Connections* window, or

Figure 13: Setting up Windows XP with TCP/IP – details of LAN connections

on the command line by entering

```
ipconfig
```

If the settings match the ones you specified before, you can end the dialog and configuration. Note, however, that the system makes these settings dynamically, i.e. the next time you start your PC, this may result in different settings.

Check Use the following IP address if you want to enter special values for the IP address, the subnet mask and the gateway:

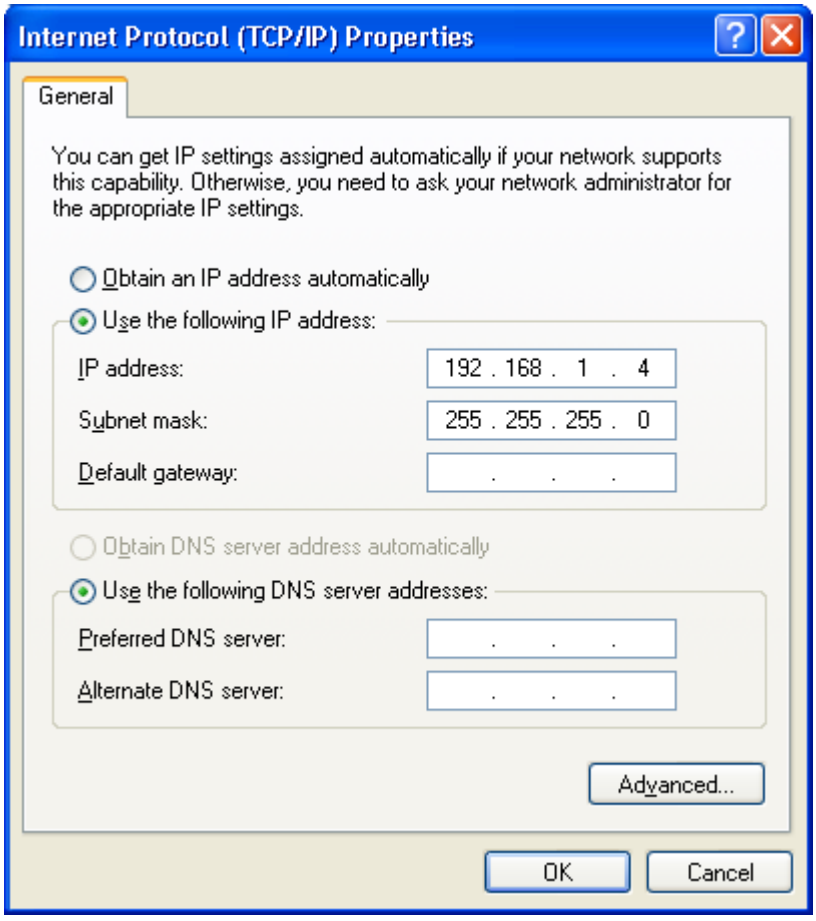


Figure 14: Setting up Windows XP with TCP/IP – TCP/IP properties

- Close the windows that you opened for configuration by clicking on the *OK* pushbutton.

This completes installation of TCP/IP support on your Windows XP system. Under Windows XP, you do not need to reboot your computer.

3.3.2.2 Configuring TCP/IP under Windows 2000

- Click on the *Start* pushbutton and under *Settings* open *Network and Dial-up Connections*
- Double-click on the *Local Area Connection* icon and click on the *Properties* → pushbutton; the system displays a list of the components that use your network adapter.
- Check *Internet Protocol (TCP/IP)* and click on the *Properties* pushbutton.

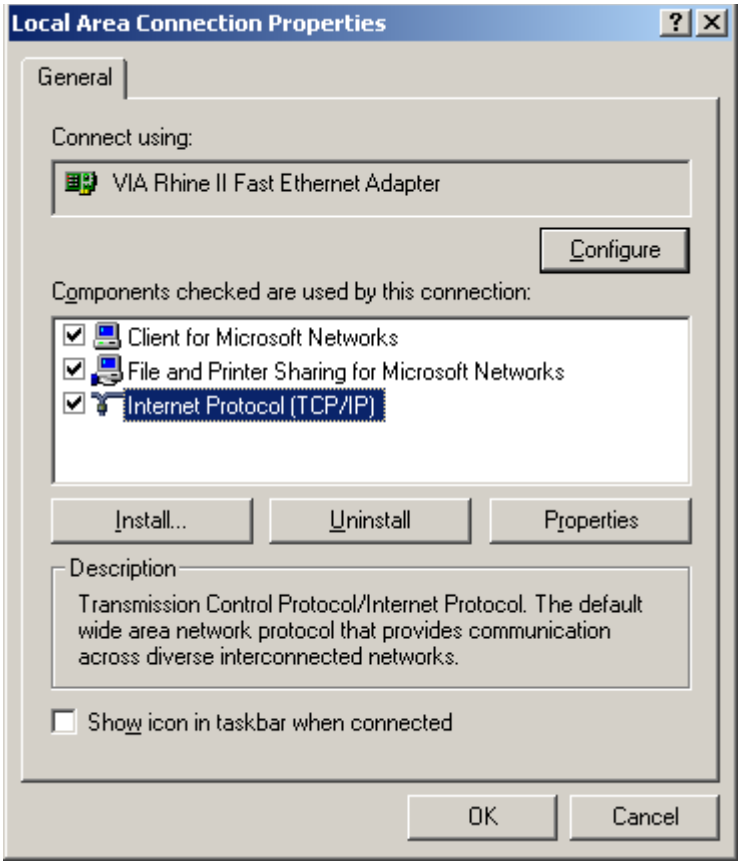


Figure 15: Setting up Windows 2000 with TCP/IP – overview of LAN connections

- You can now choose between *Obtain IP address automatically* or *Use the following IP address*. If you want to *Obtain IP address automatically,* you can now query the current setting on the command line by entering

```
ipconfig
```

If the settings match the ones you specified before, you can end the dialog and configuration. Note, however, that the system makes these settings dynamically, i.e. the next time you start your PC, this may result in different settings.

Check *Use the following IP address* if you want to enter special values for the IP address, the subnet mask and the gateway:

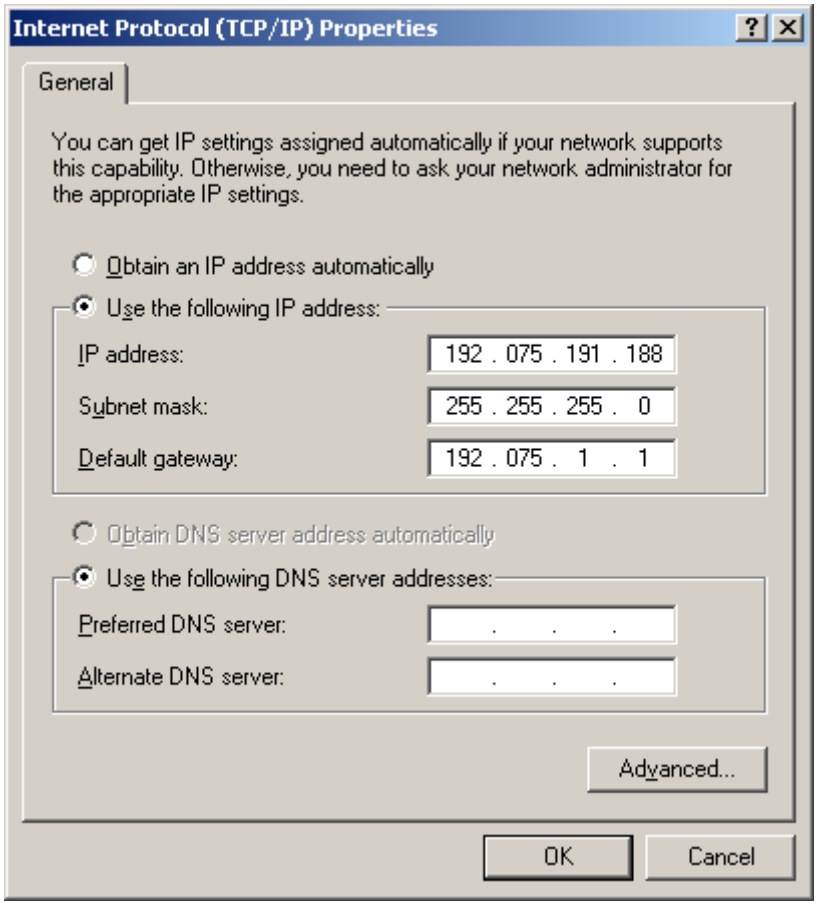


Figure 16: Setting up Windows 2000 with TCP/IP – TCP/IP properties

- Close the windows that you opened for configuration by clicking on the *OK* pushbutton.

This completes installation of TCP/IP support on your Windows 2000 system. Under Windows 2000, you do not need to reboot your computer.

3.3.3 Configuring the Ethernet option module and checking settings

You have two options for configuring the Ethernet option module for TCP/IP communication:

- by using the default settings
- by freely configuring using ProProg wt III

3.3.3.1 Default settings for TCP/IP

The default settings for TCP/IP communication of the Ethernet option module are:

IP address:	192.168.1.1+ "DIP switch"
Subnet mask:	255.255.255.0

Gateway: 0.0.0.0

Using DIP switches 1 - 5 (SW13100 on the option module), you can set IP addresses in the range 192.168.1.1 to 192.168.1.32. In order to set the DIP switches of the option module see ▹Operating Instructions Ethernet with EtherCAT-Master for b maXX drive PLC◃.

3.3.3.2 Freely configuring TCP/IP

a) Prepare for configuration using the ProProg wt III

You can also freely set the values for the IP address, the subnet mask and the gateway. To do this, proceed as follows:

1. Create a ProProg wt III project for the b maXX drive PLC if you have not already created a separate project for your application.
2. In this project, link library BM_TYPES_30bd01 (or higher) if it is not already present.
3. Create a POU that can later be called in a cold boot and warm restart task, assuming that one is not present.
4. Create a global variable of data type ETHERNET_PLC_CONFIG_BMSTRUCT. You must connect this global variable to the base address to the Ethernet configuration. The base address depends on the slot in which the Ethernet option module is fitted. The following addresses are possible:

Slot	Base address for Ethernet configuration
G	%MB3.2012288
H	%MB3.3012288
J	%MB3.4012288
K	%MB3.5012288
L	%MB3.6012288
M	%MB3.7012288

Example if the Ethernet option module is fitted in slot G:

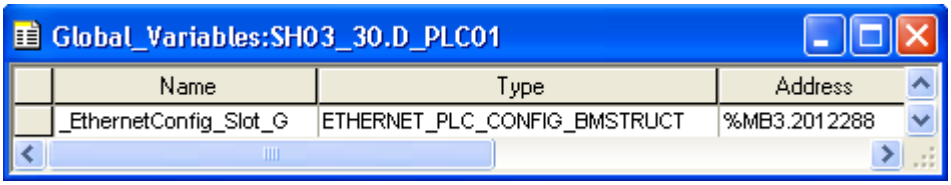


Figure 17: Example: Global variable of type ETHERNET_PLC_CONFIG_BMSTRUCT

Where:

`_EthernetConfig_Slot_G`	is the variable name with data type short designation "_" for STRUCT
`%MB3.2012288`	is the address for Ethernet configuration of slot G
`ETHERNET_PLC_CONFIG_BMSTRUCT`	is the data type of the variable

5 To set the IP address, subnet mask and gateway, you need structure elements a_IP_ADDRESS, a_IP_MASK and a_GATEWAY of the created global variable. All three elements consist of one field (array) containing four entries each of data type USINT. Each address starts in field index 0 with the network section. To activate the configuration, you additionally need structure element d_IP_CONFIG of data type DWORD. Create these elements in the POU for cold boot/warm restart.

Example:

```
(* Configuration of IP-Address *)
_EthernetConfig_MNr_01.a_IP_ADDRESS[0]
_EthernetConfig_MNr_01.a_IP_ADDRESS[1]
_EthernetConfig_MNr_01.a_IP_ADDRESS[2]
_EthernetConfig_MNr_01.a_IP_ADDRESS[3]

(* Configuration of Subnetmask *)
_EthernetConfig_MNr_01.a_IP_MASK[0]
_EthernetConfig_MNr_01.a_IP_MASK[1]
_EthernetConfig_MNr_01.a_IP_MASK[2]
_EthernetConfig_MNr_01.a_IP_MASK[3]

(* Configuration of Gateway *)
_EthernetConfig_MNr_01.a_GATEWAY[0]
_EthernetConfig_MNr_01.a_GATEWAY[1]
_EthernetConfig_MNr_01.a_GATEWAY[2]
_EthernetConfig_MNr_01.a_GATEWAY[3]

(* The configuration command *)
_EthernetConfig_MNr_01.d_IP_CONFIG
```

Figure 18: Example: Elements of a global variable of type ETHERNET_PLC_CONFIG_BMSTRUCT

6 There are two options for configuring the IP address:
A fixed IP address or a DIP switch-dependent IP address. In the first case, you set an IP address that is independent of the DIP switches. In the second case, you add the value of the DIP switches 1-5 to the configured IP address. The value of element d_IP_CONFIG decides which procedure is to be used.

b) Assigning a fixed IP address

To assign a fixed IP address that is independent of the DIP switches, you assign the structure elements of the global variable for Ethernet configuration with the required values.

Finally, you must write element d_IP_CONFIG with DWORD#16#12345678. When writing to the elements, you must absolutely comply with the following sequence

a_IP_ADDRESS → a_IP_MASK → a_GATEWAY → d_IP_CONFIG

Example:

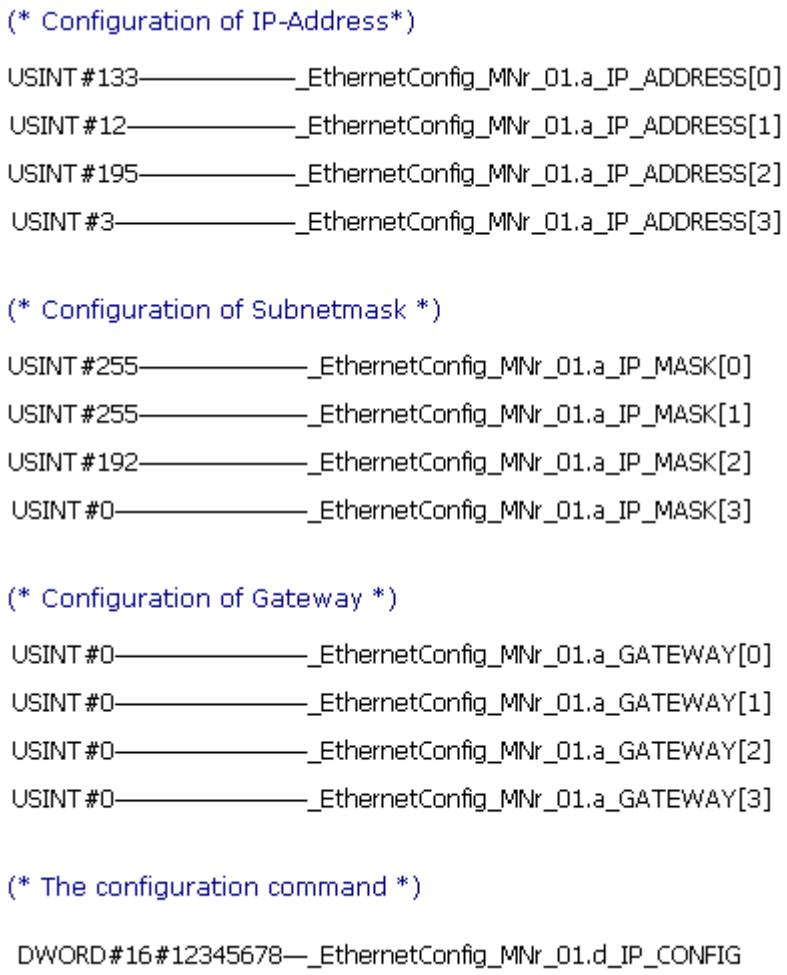


Figure 19: Example: Assigning a fixed IP address, gateway and subnet mask for the option module

For the Ethernet option module, this yields the IP address 133.12.195.3 and the subnet mask 255.255.192.0. The gateway has the address 0.0.0.0, i.e. no gateway is used.

c) Assigning a variable, DIP switch-dependent IP address

To assign a variable IP address that is dependent on the DIP switches, you assign the structure elements of the global variable for Ethernet configuration with the required values. Finally, you must write element d_IP_CONFIG with DWORD#16#12345600. When writing to the elements, you must absolutely comply with the following sequence

a_IP_ADDRESS → a_IP_MASK → a_GATEWAY → d_IP_CONFIG

Example:

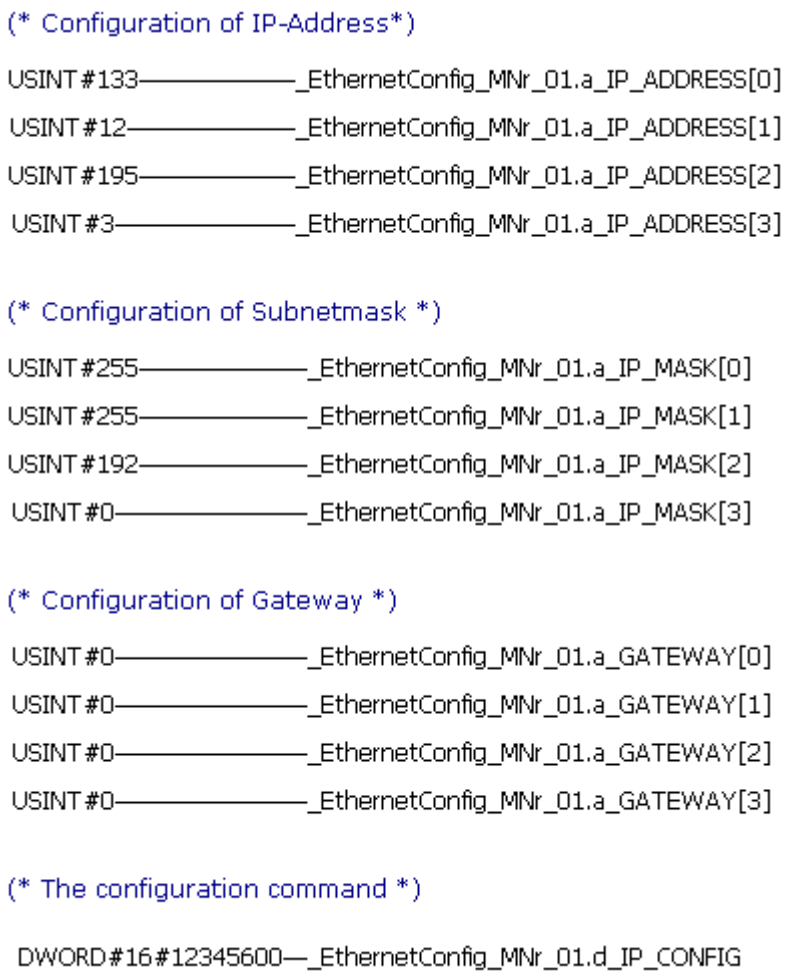


Figure 20: Example: Assigning a variable IP address, gateway and subnet mask for the option module

For the Ethernet option module, this yields the IP address 133.12.195.3 + DIP switch and the subnet mask 255.255.192.0. The gateway has the address 0.0.0.0, i.e. no gateway is used.

d) Activating the TCP/IP configuration

The POU containing the elements for TCP/IP configuration has been created. After this, carry out the following steps:

1 Now link the POU in a cold boot and a warm restart task. After this, compile the program and load it as a boot project on the b maXX drive PLC.

2 Switch the b maXX unit off and back on again twice at an interval of approximately 10 seconds. It is absolutely crucial to switch off and back on again twice.

The Ethernet option module now has the TCP/IP configuration that you set. You can remove the program code from the created POU. The TCP/IP configuration is retained. In the same way, you can load another project (even a boot project) on the b maXX drive PLC without losing the TCP/IP configuration. You can only change the TCP/IP configura-

tion of the Ethernet option module by implementing again the code lines that were described above.

e) Checking the TCP/IP configuration

You can use ProProg wt III to check the settings of the Ethernet option module's TCP/IP configuration.

To do this, proceed as follows:

1 Create a ProProg wt III project for the b maXX drive PLC if you have not already created a separate project for your application or by the configuration.

2 In this project, link library BM_TYPES_30bd01 (or higher) if it is not already present.

3 Create a global variable of data type ETHERNET_PLC_DIAG_BMSTRUCT. You must connect this global variable to the base address to the Ethernet diagnostics. In this connection, the base address depends on the slot in which the Ethernet option module is fitted. The following addresses are possible:

Slot	Base address for Ethernet diagnostics
G	%MB3.2012320
H	%MB3.3012320
J	%MB3.4012320
K	%MB3.5012320
L	%MB3.6012320
M	%MB3.7012320

Example if the Ethernet option module is fitted in slot G:

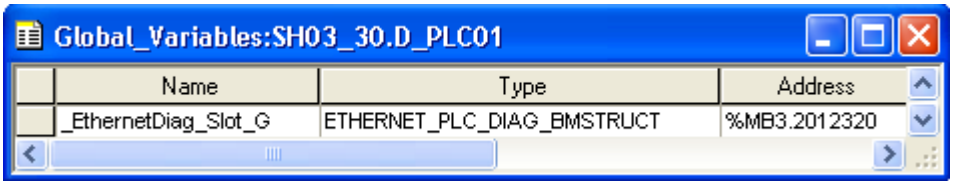


Figure 21: Example: Global variable for checking the TCP/IP configuration

Where:

`_ EthernetDiag_Slot_G`	is the variable name with data type short designation "_" for STRUCT
`%MB3.2012320`	is the address for Ethernet diagnostics of slot G.
`ETHERNET_PLC_DIAG_BMSTRUCT`	is the data type of the variable

4 Structure elements a_MAC_ADDRESS, a_IP_ADDRESS, a_IP_MASK, d_IP_CONFIG, d_DIP_SWITCH and a_GATEWAY of the created global variable are available for diagnostics. The individual elements have the following meanings:

Element	Meaning
a_MAC_ADDRESS	MAC address of the option module. Field (array) containing 8 entries of data type BYTE in each case. Field indexes 0 to 5 are valid. Indexes 0 to 2 designate the code for manufacturer identification; indexes 3 to 5 designate the manufacturer code for the respective device.
a_IP_ADDRESS, a_IP_MASK, a_GATEWAY	Active IP address, subnet mask and gateway of the option module. All three elements consist of one field (array) containing four entries each of data type USINT. Each address starts in field index 0 with the network section.
d_IP_CONFIG	Designates the type of evaluation of the IP address. 16#00000000 No evaluation 16#12345678 Fixed IP address 16#12345600 IP address + DIP switches 1-5
d_DIP_SWITCH	Settings of DIP switches 1-5

5 Compile the program and load it as a project (even a boot project) on the b maXX drive PLC. You do not need to create a special POU.

6 Start the program and switch to online mode. Add the created global variable to TCP/IP diagnostics in the watch window; display the watch window and open the relevant structure elements. You can now see the set data.

Example:

Watch Window

Variable	Value	Default value	Type
_EthernetDiag_MNr_1			ETHERNET_PLC_DIAG_BMSTRUCT
a_MAC_ADDRESS			BYTE_8_BMARRAY
[0]	16#00		BYTE
[1]	16#02		BYTE
[2]	16#FB		BYTE
[3]	16#FF		BYTE
[4]	16#FE		BYTE
[5]	16#28		BYTE
[6]	16#00		BYTE
[7]	16#00		BYTE
a_Reserved8			BYTE_8_BMARRAY
a_IP_ADDRESS			USINT_4_BMARRAY
[0]	133		USINT
[1]	12		USINT
[2]	195		USINT
[3]	7		USINT
a_IP_MASK			USINT_4_BMARRAY
[0]	255		USINT
[1]	255		USINT
[2]	192		USINT
[3]	0		USINT
d_IP_CONFIG	16#12345600		DWORD
d_GENERAL_STAT	16#00000001		DWORD
d_DIP_SWITCH	16#00000004		DWORD
a_GATEWAY			USINT_4_BMARRAY
[0]	0		USINT
[1]	0		USINT
[2]	0		USINT
[3]	0		USINT

Watch 1 | Watch 2 | Watch 3 | Watch 4

Figure 22: Example: Reading out the TCP/IP configuration in the watch window

3.4 Setting a connection via Ethernet-TCP/IP

3.4.1 Setting a connection via Ethernet TCP/IP in ProMaster

You set the uses of Ethernet-TCP/IP in ProMaster individually for each device.

In this chapter, we will describe how you set the uses of Ethernet TCP/IP in ProMaster for a b maXX drive PLC. To do this, proceed as follows:

- Open your complete configurated ProMaster project (in which the ProProg wt III project must be linked). Assign the device with b maXX drive PLC in ProMaster network window and select it via the context menu „Communication settings“.

The window „Port parameter“ will be opened.

- Activate the radio button „TCP/IP“ and write the TCP/IP address of the b maXX drive PLC resource in the edit box „Address“.

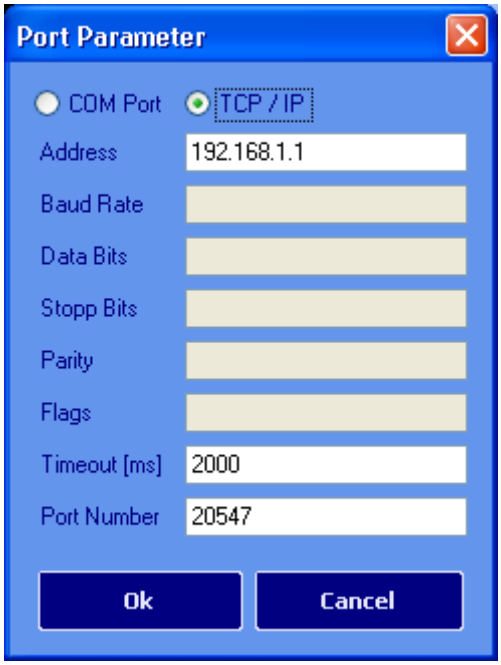


Figure 23: Setting the TCP/IP configuration in ProMaster for a b maXX drive PLC resource

To be able to access the b maXX drive PLC via Ethernet, an Ethernet compliant module (e.g. BM4-O-ECT-02) must be fitted in the b maXX system in addition to the b maXX drive PLC.

The TCP/IP address "192.168.1.1" corresponds to the address that is preinstalled on the Ethernet module when it leaves the factory, so that the user can contact the module first for the basic initialization to make the adjustments of its TCP/IP address for the TCP/IP network at the machine.

The communication setting of ProMaster will be accepted automatically in the linked ProProg wt III program.

3.4.2 Setting a connection via Ethernet TCP/IP in ProProg wt III (PLC)

NOTE

If you change the communication settings only in ProProg wt III project, these communication settings will be **not** accepted in ProMaster

In this chapter, we will describe how you set the uses of Ethernet TCP/IP in ProProg wt III for a b maXX drive PLC. To do this, proceed as follows:

- In ProProg wt III project open the dialog „Resource settings for SH03_30“ via the context menu of the resource b maXX drive PLC
- Set the port to „DLL“

Setting the Ethernet communication source by choosing or stating the TCP/IP address

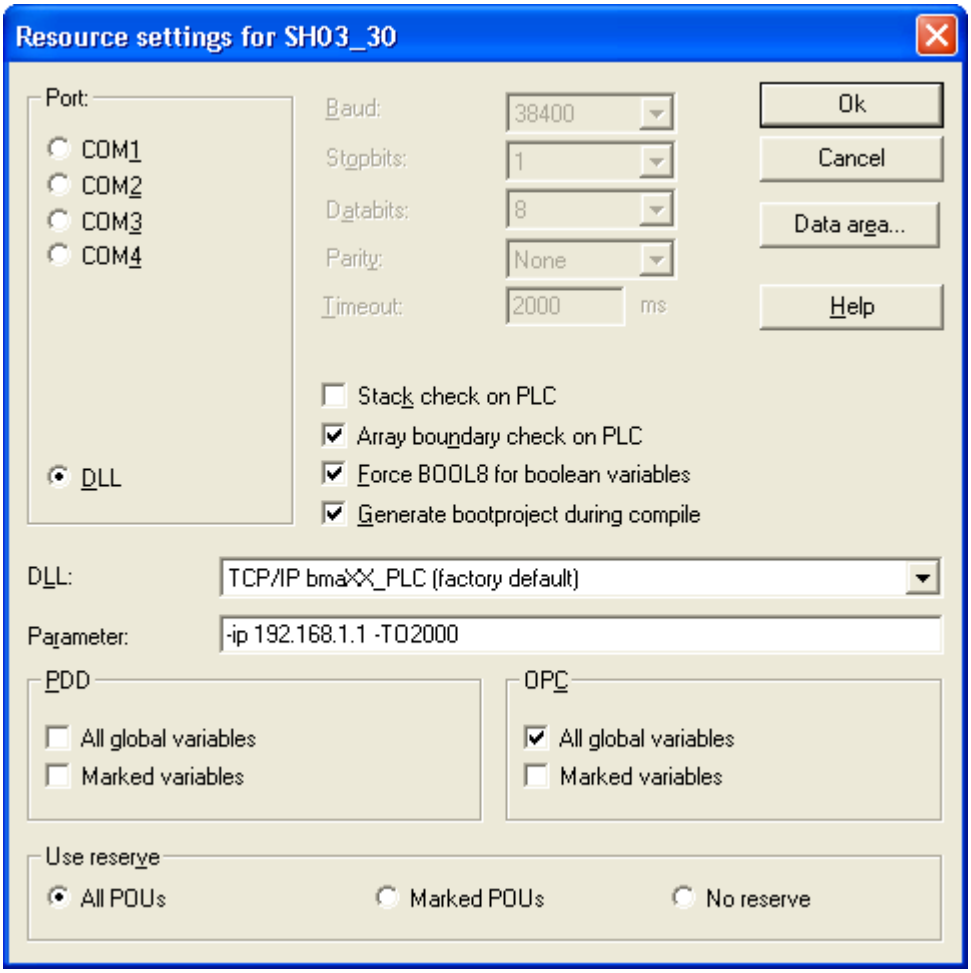


Figure 24: Setting TCP/IP in PROPROG wt II for one resource

After changing the port to "DLL", you can use the DLL menu to choose applications "Soft-PLC" (=TCP/IP local host (this PC)) and "b maXX drive PLC access via Ethernet" (=TCP/IP bmaXX_PLC (factory default)):

- Soft-PLC:
  You do not need to change the preset TCP/IP address in field "Parameter" if the Soft-PLC is installed on the same computer. If installing Soft-PLC on another computer, you must enter here the appropriate TCP/IP target address (or the appropriate network name) to be able to access Soft-PLC via TCP/IP from ProProg wt III.
- b maXX drive PLC:
  To be able to access the b maXX drive PLC via Ethernet, an Ethernet compliant option module (e.g. BM4-O-ECT-02) must be fitted in the unit b maXX 4400 in addition to the b maXX drive PLC. The TCP/IP address "192.168.1.1" that is preset in the parameter field corresponds to the IP address that is preinstalled on the Ethernet option module when it leaves the factory.

3.5 EoE - Ethernet over EtherCAT

3.5.1 General

As well as the exchange of process data and service data, the EtherCAT field bus system also allows the transmission of Ethernet data (TCP/IP) to the EtherCAT slaves.

This takes place using the Ethernet link between the PC (e. g. PC1) and the EtherCAT master option module (ECT-02) for b maXX drive PLC.

- The PC sends the Ethernet data (intended for a specific EtherCAT slave) to the EtherCAT master:

  PC1 → ECT-02, X4

- The EtherCAT master forwards the Ethernet data to the relevant EtherCAT slave (ECT-01):

  ECT-02, X3 → ECT-01, X1/X2

- The EtherCAT slave processes the Ethernet data (e. g. a read request) and provides the answer for collection by the EtherCAT master:

  ECT-01, X1/X2 → ECT-02, X3

- The EtherCAT master forwards the slave's answer to the PC:

  ECT-02, X4 → PC1

Using ProDrive (in ProMaster) it is therefore possible to establish connections with individual EtherCAT slaves via TCP/IP and parameterize, for instance, the drive in question.

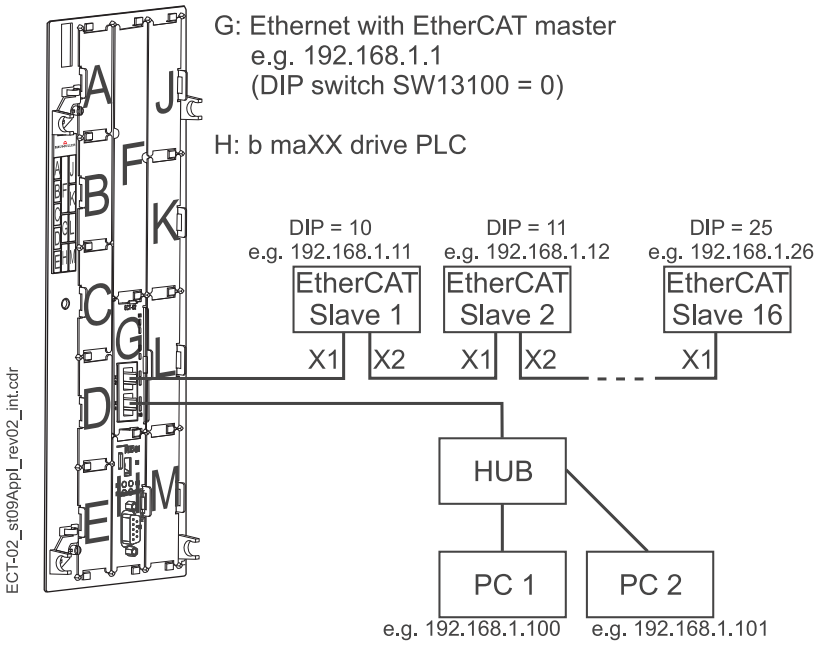


Figure 25: EoE - Ethernet over EtherCAT - overview

To configure the Ethernet TCP/IP network on your PC see ▹Configuring the Windows PC ◃ from page 30 onward.

To set the IP address on the EtherCAT master module for b maXX drive PLC see ▹Configuring the Ethernet option module and checking settings ◃ from page 34 onward.

To set the IP address on the optional b maXX controller EtherCAT slave module see ▹ b maXX controller EtherCAT slave module operating instructions ◃.

To set the Ethernet TCP/IP connection between ProMaster, the EtherCAT master and the b maXX drive PLC see ▹Setting a connection via Ethernet TCP/IP in ProMaster ◃ from page 41 onward.

To set the Ethernet TCP/IP connection between ProProg wt III and the b maXX drive PLC see ▹Setting a connection via Ethernet TCP/IP in ProProg wt III (PLC) ◃ from page 42 onward.

To set the Ethernet TCP/IP connection between ProMaster and the EtherCAT slave see ▹Setting an Ethernet TCP/IP connection in ProMaster (EtherCAT slave) ◃.

3.5.2 Setting an Ethernet TCP/IP connection in ProMaster (EtherCAT slave)

Once IP addresses have been set for the optional b maXX controller EtherCAT slave modules, the Ethernet TCP/IP connection is set between ProMaster and the EtherCAT slaves.

In ProMaster, Ethernet TCP/IP must be assigned individually for each module.

In this chapter we describe how to apply Ethernet TCP/IP in ProMaster for a single EtherCAT slave. Simply complete the following steps:

- Open your configured ProMaster project. In the ProMaster network view highlight the EtherCAT slave and select "Communication settings" from the context menu.

This opens the "Port parameter" dialog box.

- Select the "TCP/IP" radio button and write the EtherCAT slave's TCP/IP address in the "Address" edit box. Then click the "OK" button to close the "Port parameter" dialog box.

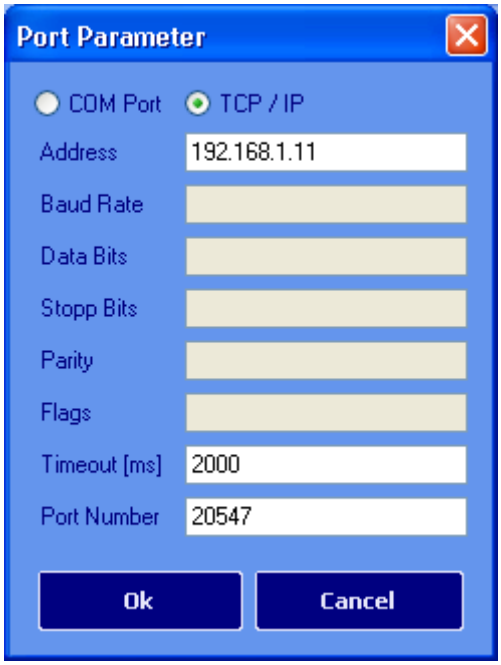


Figure 26: Setting the ProMaster TCP/IP configuration for an EtherCAT slave

The TCP/IP address "192.168.1.1 + 'dip switch' " is the default address for the optional b maXX controller EtherCAT slave module.

In ▹Figure 25 on page 45 the dip switch setting for EtherCAT slave 1 is 10, the TCP/IP address is therefore 192.168.1.1 + 10 = 192.168.1.11.

Alternatively you can set the Ethernet TCP/IP connection between ProMaster and the EtherCAT slaves during the configuration of EtherCAT slave communications. See ▹Network tab◃ in chapter "4.3.6.1 Configuring EtherCAT slave communication" from page 61.

You may also set the Ethernet TCP/IP connection between ProMaster and the EtherCAT slaves during the configuration of EtherCAT master communications. See ▹Motion Control tab◃ in chapter "4.3.6.2 Configuration of EtherCAT master communication" from page 72.

4

ETHERCAT

This chapter contains information on data exchange via EtherCAT.

4.1 General information on EtherCAT and the use of the EtherCAT master option module

EtherCAT is an Ethernet-based field bus system that combines the advantages of Ethernet (high speed, easy configuration) with real-time capability and suitability for small amounts of data. The use of Internet technologies on the I/O level is also possible with EtherCAT.

As with other field busses, device profiles are used for communication with EtherCAT. No new device profiles were defined for EtherCAT, however. Instead, interfaces for existing device profiles (already familiar to the user) are provided. This facilitates the migration from the existing field bus to EtherCAT, both for the user and for the device manufacturer.

The Ethernet with EtherCAT master option module supports CANopen over EtherCAT (CoE) and Ethernet over EtherCAT (EoE).

Additional information is found on the website of the EtherCAT Technology Group (ETG) at www.EtherCAT.org.

CANopen over EtherCAT (CoE)

CANopen devices and application profiles are available for a wide variety of device classes and applications, such as

- I/O modules
- drives
- encoders
- proportional valves and hydraulic regulators

and application profiles, such as

- plastic processing
- textile machines

EtherCAT can provide the same communication mechanisms as CANopen:

- object directory

- PDO (Process Data Objects)
- SDO (Service Data Objects)

Even the network management is comparable.

Ethernet over EtherCAT (EoE)

The EtherCAT technology is not only fully Ethernet-compatible, it's also characterized by its great openness "by design". The protocol is compatible with other Ethernet-based services and protocols on the same physical network, generally with only a minimal loss of performance. Any desired Ethernet devices can be connected within the EtherCAT line via a switch port. The Ethernet frames are tunneled via the EtherCAT protocol, as is usual with Internet technologies. The EtherCAT network is fully transparent to the Ethernet device, and the real-time properties are not taken into account. EtherCAT devices can also use other Ethernet protocols, and thus act like a standard Ethernet subscriber on the exterior. Here, the master functions as a layer 2 switch that passes on the frames to the corresponding subscribers using the address information.

This means that all the Internet technologies, e.g.

- integrated webservers
- e-mail
- FTP transfer

can also be used in the EtherCAT environment.

To exchange data with the EtherCAT master option module via EtherCAT, use ProMaster (and ProEtherCAT) for easy and quick data exchange configuration. Quick and easy programming of machine functioning then occurs with the motion control function modules and the network variables in ProProg wt III.

Example projects:

ProMaster project	Example_1_3_2.bmxml
IEC project in ProProg wt III	Example_BM4_ECT02_MA_2.mwt/.zwt

See ▹ProMaster, ProEtherCAT, ProProg wt III and motion control◃ from Page 51.

4.2 Fundamentals of the EtherCAT and EtherCAT networks

EtherCAT overcomes the principle limitations of other Ethernet solutions. An Ethernet telegram (containing the EtherCAT protocol) is no longer first received in each interface and then interpreted, followed by further copying of the process data. Instead, each EtherCAT slave takes the data specified for it from the telegram (e.g. set values) while the telegram runs through the EtherCAT slave. The input data (e.g. actual values) is also entered into the telegram while the telegram runs through the EtherCAT slave.

The Ethernet telegrams are delayed by only a few nanoseconds here.

Since an Ethernet telegram reaches many EtherCAT slaves both in the transmitting as well as the receiving direction, the user data rate rises to over 90%.

Protocol

The EtherCAT protocol optimized for process data is transported directly in the Ethernet telegram. EtherCAT uses only standard telegrams (standard frames) pursuant to IEEE802.3.

Network

EtherCAT supports the bus and line structure used for field busses.

Fast Ethernet physics (100BaseTX) allow a cable length up to 100 m between two devices on the EtherCAT bus. With the BM4-O-ECT-02 option module, up to 1023 devices can be connected to the EtherCAT bus.

Optional slaves

The extension „Optional slaves“ is available at EtherCAT master in version number

BM4-O-ECT-02-01-01-000-003

BMC-M-ECT-02-11-01-000-002

and higher.

The feature "**Optional slaves**" enables realizing different stages of extension of a modular machine concept using only one broad field bus configuration and one ProProg project (PLC program).

The maximum dismounting of the EtherCAT bus of the modular machine is to be configurated with ProMaster. That EtherCAT slaves, which may be absent in miscellaneous stages of extension, will be labeled with the attribute „optional“.

The EtherCAT master recognizes the availability or the absence of optional slaves at the start of the EtherCAT bus, manages the bus accordingly and keeps this informations ready for the PLC. Further application specific dependency on the optional components can be implemented in the PLC program (e.g. operation of the optional component A is allowed only if either component C or D is available).

The user must assign a unique identifier for each optional slave to identify the optional slave on the bus. This identifier must be configurated both in ProMaster (see ▹Configuring EtherCAT slave communication◃ from Page 60) and in the real device e.g. adjusted via decode switches or otherwise. For this see the Operating Instructions of the respective EtherCAT slave.

NOTE

The codomain of the identifier for optional slaves: 1 to 65535.

The value „0“ means „identifier not set“.

The identifier can (but doesn’t need) agree with EtherCAT node ID (see example). The identifier must be unique within a field bus configuration and can be assigned according to other viewpoints also (e.g. double usage of a decode switch for setting the TCP/IP address and for the identifier).

Example:

	mandatory / optional	Node ID	Identifier / decode switch	IP address	Maximum stages of extension	Version 1	Version 2	Version 3
Axis_1	m	1	0 „not set“	192.168.1.1	x	x	x	x
Axis_2	o	2	1	192.168.1.2	x	x	-	x
Axis_3	o	3	2	192.168.1.3	x	-	-	x
PLC_2	m	4	51	192.168.1.52	x	x	x	x
Axis_4	o	5	3	192.168.1.4	x	x	x	-
PLC_3	o	6	52	192.168.1.53	x	x	x	-
IO-Block	m	7	-	-	x	x	x	x

NOTE

An identifier, which is used to identify optional slaves from the EtherCAT master, can not be set on all EtherCAT slaves. For this see the Operating Instructions of the respective EtherCAT slave.

4.3 ProMaster, ProEtherCAT, ProProg wt III and motion control

This section explains how you can quickly and easily commission an EtherCAT network with the b maXX drive PLC and the EtherCAT master option module. This section is accompanied by a ProMaster example project.

An EtherCAT network consists of an EtherCAT master and one or more EtherCAT slaves.

The machine configuration is created from these devices with the Baumüller ProMaster engineering framework. EtherCAT communication between the individual devices is configured with the ProEtherCAT field bus configuration tool. By using motion control, the configuration settings are reduced to a minimum and the user can fully concentrate on the actual application (machine function) in the IEC 61131-3 programming system ProProg wt III.

The data and settings for EtherCAT arising from the configuration of EtherCAT communication are stored on the EtherCAT master module. This concerns the data for the EtherCAT master itself and the data for the EtherCAT slaves. The EtherCAT slaves do not receive their data from the EtherCAT master until the machine is switched on.

The data and settings for ProProg wt III arising during the configuration of EtherCAT communication are stored in the IEC project (ProProg wt III project) of the device with the EtherCAT master module. This IEC project is loaded on the b maXX drive PLC to which the EtherCAT master module is attached.

4.3.1 Requirements

The requirements are

- a PC with
  - the ProMaster engineering framework
  - the IEC 61131-3 programming system ProProg wt III with the firmware libraries
    - Bit_UTIL_30bd00
    - OmegaOS_30bd00
    - SYSTEM2_PLC01_30bd00
    - MC_SYS_30bd00

    and the libraries
    - BM_TYPES_30bd00
    - SYSTEM1_PLC01_30bd00
    - MOTION_TYPES_30bd02
    - MOTION_CONTROL_30bd03
    - MOTION_MULTI_AXIS_30bd03 (if multi-axis motion control is used)
- Physical commissioning of the components of the EtherCAT network. In our example, they are the EtherCAT master
  - b maXX 4400 with
    - b maXX drive PLC (BM4-O-PLC-0x-...)
    - EtherCAT master option module (BM4-O-ECT-02-...)

  and the EtherCAT slave
  - b maXX 4400 with

- EtherCAT slave option module (BM4-O-ECT-01-...)

For this purpose, see the respective operating instructions.

The extension „Optional slaves“ is available at EtherCAT master in version number

BM4-O-ECT-02-01-01-000-003

BMC-M-ECT-02-11-01-000-002

and higher. See ▹Optional slaves◃ in chapter "4.2 Fundamentals of the EtherCAT and EtherCAT networks" from page 49 onward.

4.3.2 Procedure to be carried out

To use the EtherCAT master option module for bmaXX drive PLC for motion control on an EtherCAT network, the following steps must be carried out.

1 Commissioning of the EtherCAT network

2 Creation of an IEC 61131-3 project for ProProg wt III with a motion-control template

3 Creation of a machine configuration in ProMaster

4 Configuration of EtherCAT communication with ProEtherCAT and subsequent downloading to the EtherCAT master
(with testing of the EtherCAT network with this configuration if appropriate)

5 Configuration of the PLC with linking of the IEC project and ProMaster, configuration of the cam data

6 Possible programming of the IEC 61131-3 project for ProProg wt III (application) and subsequent downloading to the b maXX drive PLC

7 Operation of the application on the EtherCAT network

Naturally, you can also begin with Step 2 and not perform physical commissioning (Step 1) until just prior to Step 6.

The ProMaster project **Example_1_3_2.bmxml** and the IEC 61131-3 project (ProProg wt III Projekt) **Example_BM4_ECT02_MA_2.mwt/.zwt** are created here.

4.3.3 Commissioning of the EtherCAT network

For details on commissioning the EtherCAT network, please refer to the respective operating instructions for the BM4-O-ECT-02 option module and the other EtherCAT network nodes.

After physically commissioning the network and switching in the power supply for the b maXX 4400 unit, the EtherCAT-Master option module is ready for operation after approximately 5 seconds.

Example		Meaning
H7 (green)	flickering	Connection to the Ethernet network is active
H5 (green)	flickering	Connection to the EtherCAT network is active

4.3.4 Creating an IEC project with ProProg wt III

The following explains how a motion control IEC project is created.

Multi-axis motion control is set in the ProMaster default settings for motion control. For this reason, it makes sense to first create an IEC project for multi-axis motion control.

Open ProProg wt III and create a new project by selecting an IEC template project via the "File\Open Project / Unzip Project…" menu.
In our example, we use the IEC template project "Tmpl_PLC01_MA_2_0104.zwt" (for BM4_O_PLC01).
This project is found in the directory <ProMaster installation directory>\projects\IEC templates, e.g. in the ProMaster standard installation directory C:\Baumueller\ProMasterNET \projects\IEC-Templates.
The IEC template project "Tmpl_PLC01_MA_2_0104.zwt" is unziped as "Example_BM4_ECT02_MA_2.mwt".

The template contains the libraries

- Bit_UTIL_30bd00
- BM_TYPES_30bd01
- MC_SYS_30bd00
- MOTION_TYPES_30bd02
- MOTION_CONTROL_30bd03
- MOTION_MULTI_AXIS_30bd03

Save the project. In our example, we use the project name "**Example_BM4_ECT02_MA_2.mwt**".

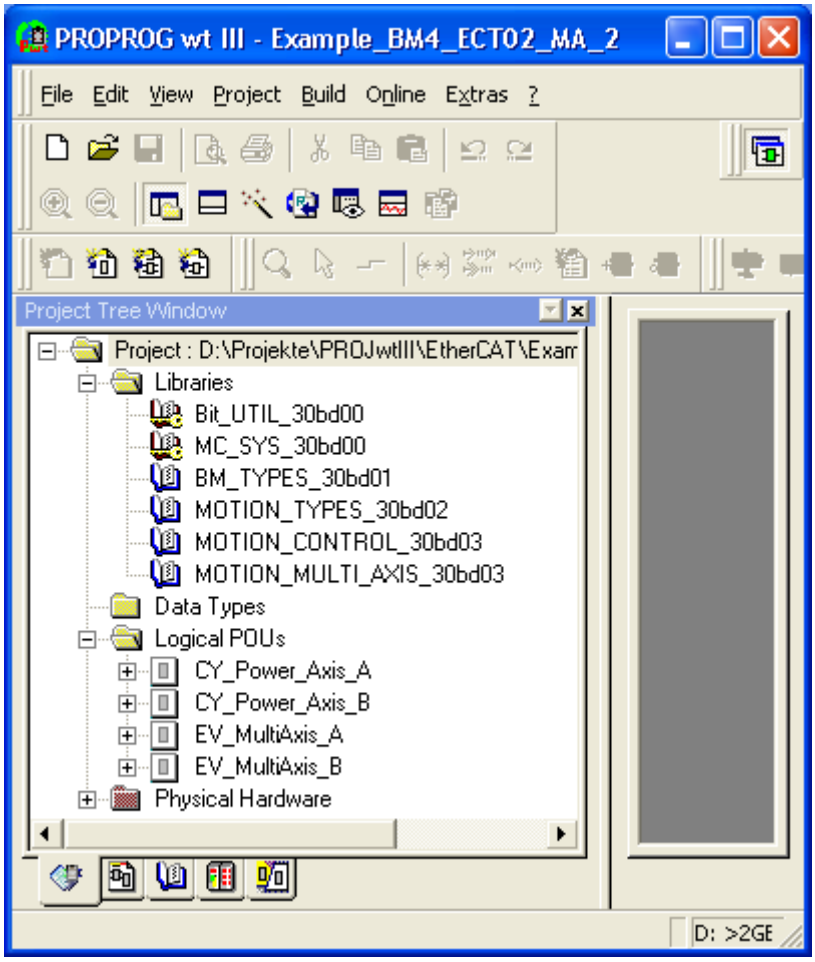


Figure 27: ProProg wt III - Creating the project "Example_BM4_ECT02_MA_2.mwt"

The communication settings must now be configured. The serial interface COM1 is set by default. If you would like to use Ethernet, see Section ▹3 Ethernet◃ from page 15 onward.

When using an IEC project in ProMaster, it is absolutely necessary that the "Create boot project when compiling" checkbox in the "Resource Settings" window is activated for ProProg wt III. This creates the "bootfile.pro" file, which is required for the download to the b maXX drive PLC.

NOTE

The "Create boot project when compiling" checkbox under "Resource Settings" must be activated in ProProg wt III.

The project "Example_BM4_ECT02_MA_2.mwt" is linked to the ProMaster project later on in Section ▹4.3.7 Configuration of the PLC◃ from page 78 onward.

Now close ProProg wt III via the menu “File\Exit”.

4.3.5 Creation of a machine configuration in ProMaster

This section explains how a project is created in ProMaster and how a machine configuration is created.

Open ProMaster.

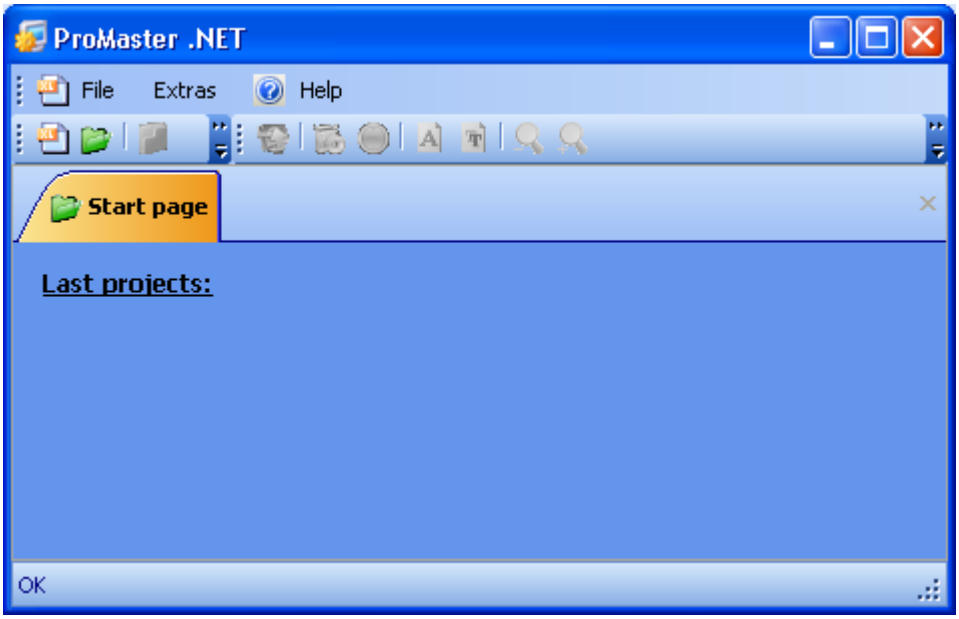


Figure 28: ProMaster with no projects

Create a new ProMaster project by opening the "Project Settings" window via the menu "File\New Project".

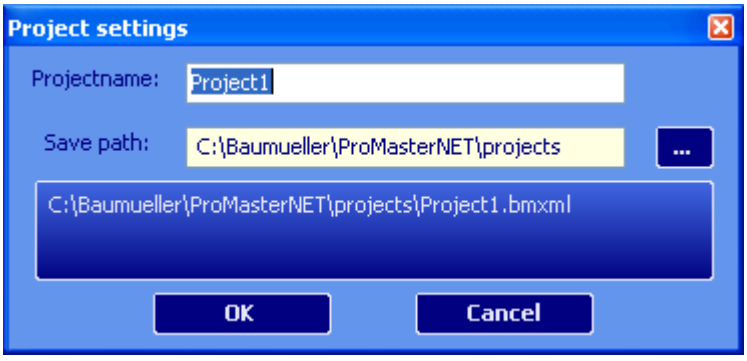


Figure 29: ProMaster - Project name

Enter the name (in the edit box) and the save location (via path selection) of the project. We use the name "Example_1_3_2.bmxml" for our example project.

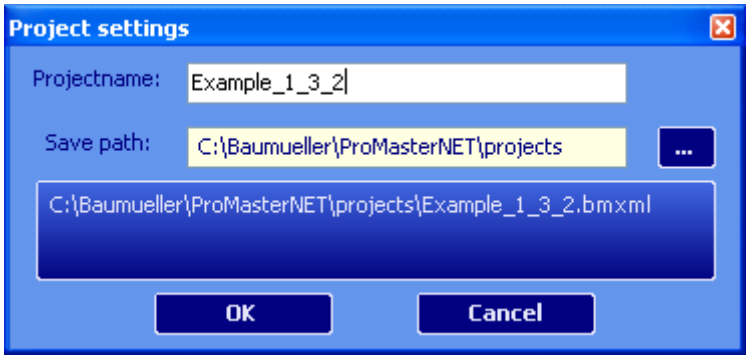


Figure 30: ProMaster - Issuing a project name

Accept the settings from the "Project Settings" window by clicking the "OK" button. ProMaster now switches to the network view.

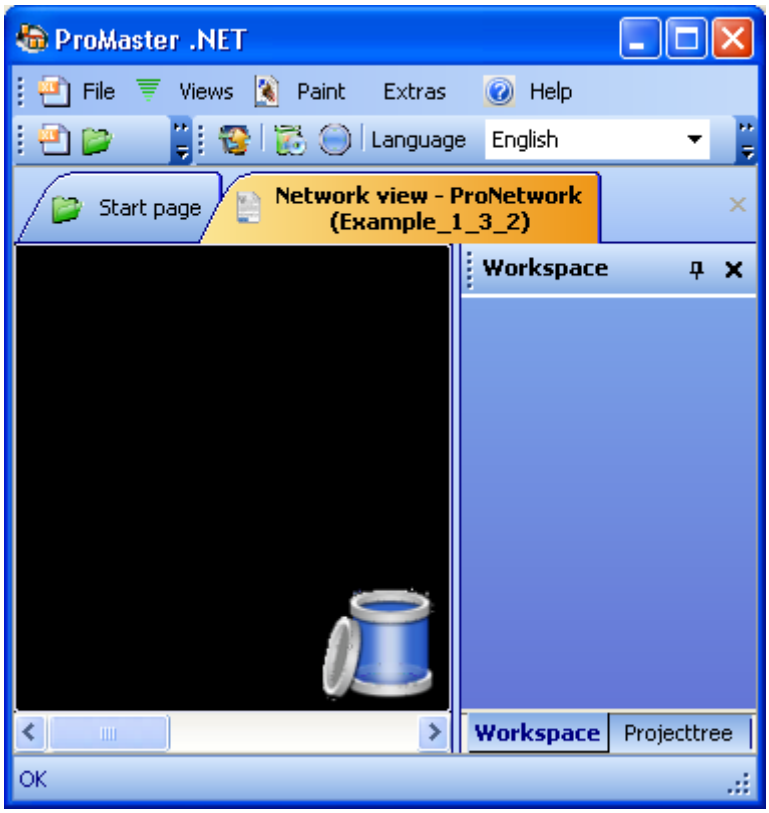


Figure 31: ProMaster - Issuing a project name

Now open the Baumüller catalog via the menu "Views\Catalog". The Baumüller catalog contains the devices, bus system and components provided by Baumüller by default.

In our example, we use

- the devices
  - "b maXX 4000 EtherCAT master (2-row)" (from the group "b maXX 4000 drive"); b maXX 4000 with two slots,

Ethernet with EtherCAT master option module in slot G,
b maXX drive PLC option module in slot H

- "b maXX 4000 EtherCAT slave (2-row)" (from the group "b maXX 4000 drive");
  b maXX 4000 with two slots,
  Ethernet with EtherCAT slave option module in slot G,

- the bus system
  - "EtherCAT bus" is created automatically

Drag the "b maXX 4000 EtherCAT master (2-row)" device from the Baumüller catalog and drop it in the ProMaster network view. The bus system is connected automatically. Now drag the "b maXX 4000 EtherCAT slave (2-row)" device and drop it on the EtherCAT bus (only possible when the bus changes color).

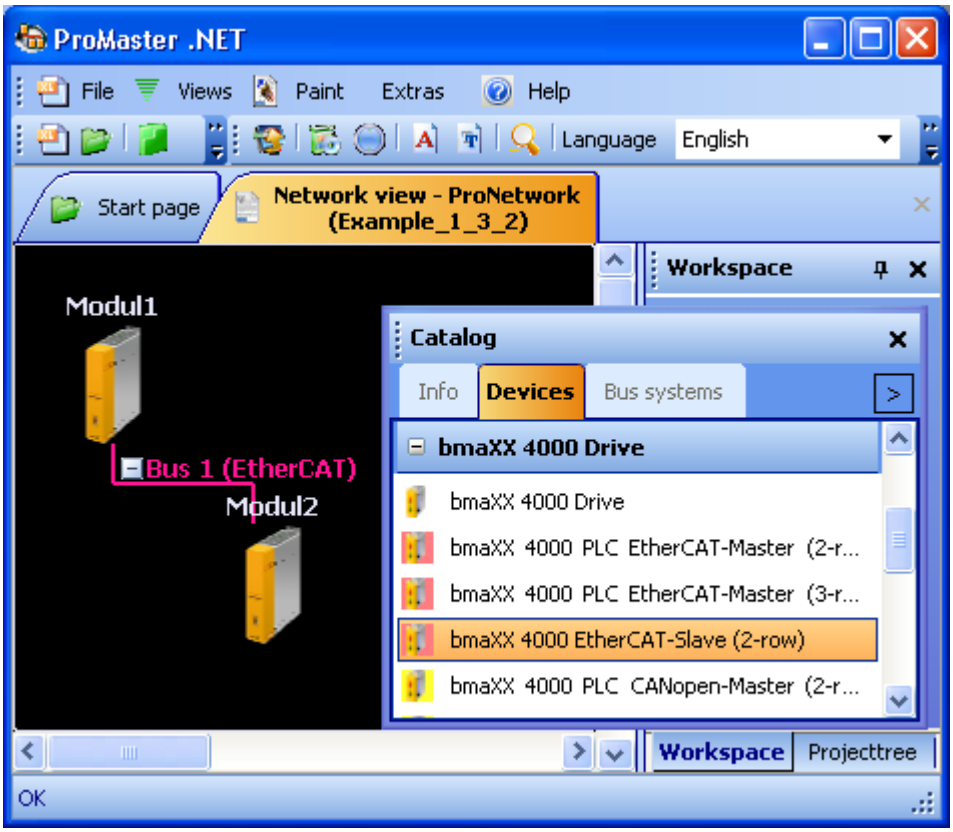


Figure 32: ProMaster - Machine configuration created

In the picture, you can now see the EtherCAT master (Modul1) and the EtherCAT slave (Modul2). Additional information on the modules can be obtained via the respective tool tip or the "Project Tree" window. The "Catalog" window is now closed to make things clearer.

The names (Modul1 and Modul2) are changed by clicking the respective module and then opening the properties window of the respective module via the context menu and “Properties”.

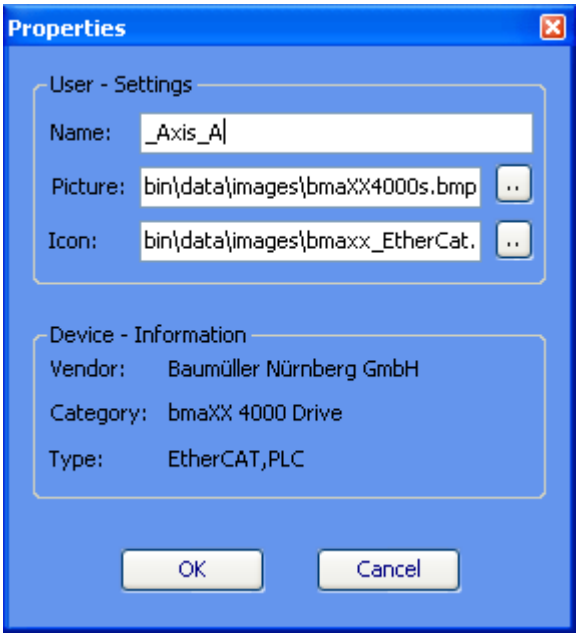


Figure 33: ProMaster - Module properties

In our example, we issue the name _Axis_A to the EtherCAT master and the name _Axis_B to the EtherCAT slave (_Axis_A and _Axis_B are the default axis variable names of the motion-control axis in our IEC project). We retain the respective images (*.bmp). The new names now appear in the ProMaster network view and in the project tree.

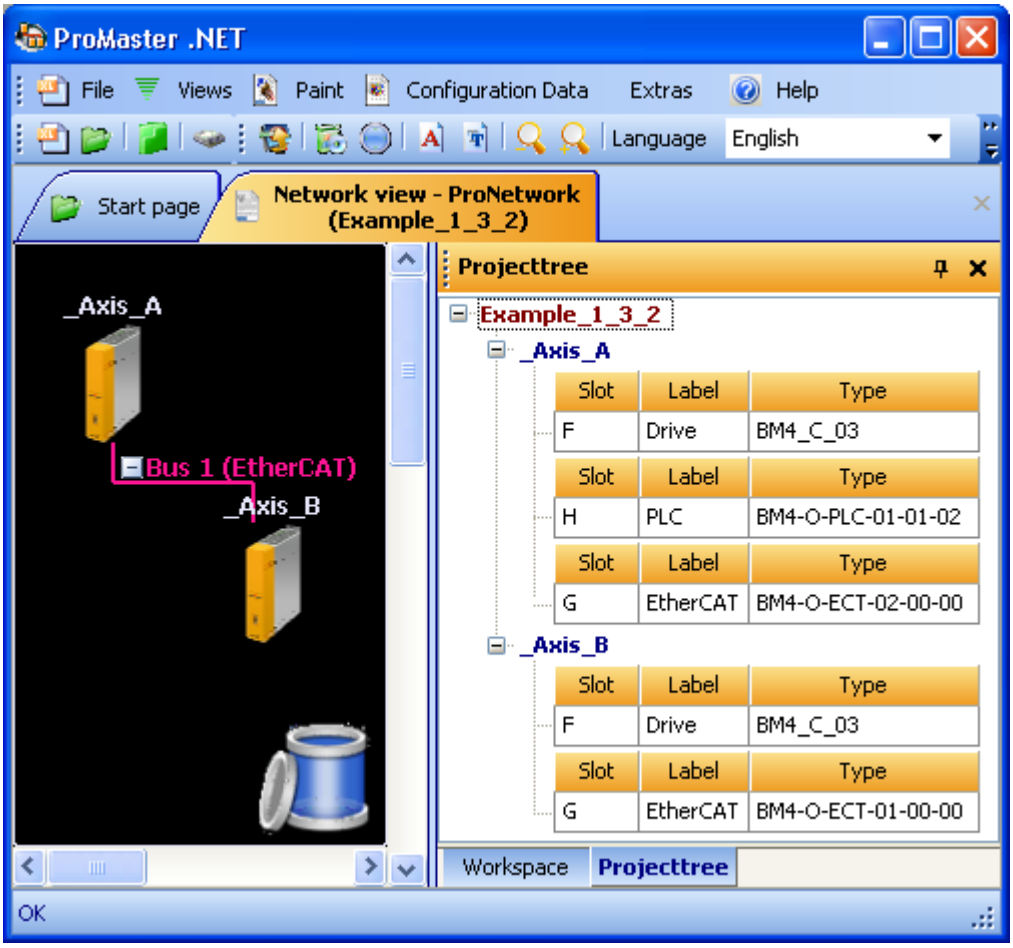


Figure 34: ProMaster - Machine configuration - Renamed module

The communication settings for the communication from the ProMaster to the b maXX drive PLC can be set by clicking the “_Axis_A” device and selecting "Communication settings" via the context menu. Then save the project via the menu “File\Save Project”.

Motion control is activated by default in ProMaster for the components with EtherCAT (EtherCAT master drive, EtherCAT slave drive). This reduces the settings to be configured to a minimum.

If you would like to use the default settings, you do not need to configure EtherCAT communication with ProEtherCAT.

In this case, continue with ▹4.3.7 Configuration of the PLC◃ from page 78 onward.

4.3.6 Configuring EtherCAT communication with ProEtherCAT

Motion control is activated by default in ProMaster for the components with EtherCAT (EtherCAT master drive, EtherCAT slave drive). This reduces the settings to be configured to a minimum.

If you would like to use the default settings, you do not need to configure EtherCAT communication with ProEtherCAT.

In this case, continue with ▹4.3.7 Configuration of the PLC◃ from page 78 onward.

Changing the EtherCAT communication settings is described in the following section.

The EtherCAT slaves are configured first, and the EtherCAT master is configured afterward.

4.3.6.1 Configuring EtherCAT slave communication

EtherCAT slave communication is configured individually for each EtherCAT slave.

Open ProEtherCAT for our EtherCAT slave by clicking the “_Axis_B” device and then selecting "Configuration Data (Components)\EtherCAT Slave (Slot G)\Configure Slave Bus (ProEtherCAT)" via the context menu.

When the machine configuration was created, the EtherCAT slave was set by default so that the EtherCAT settings are configured for motion control (dragging and dropping of the slave on a master for which motion control is activated).

The window for EtherCAT slave communication configuration is opened.

Ident tab

You can set the slot of the optional EtherCAT slave in the b maXX 4400 and the name of this module (device name) in ProMaster and the node ID for EtherCAT on the "Ident" tab for motion control.

Changing the device name and node ID affects the ProMaster project.

Changing the device name also affects the motion-control axis names in the IEC 61131-3 project in ProProg wt III.

NOTE

Changes to the device name also affect the motion-control axis names in the global variable worksheet in the IEC 61131-3 project in ProProg wt III.

This change is **not** automatically carried out in the local variable worksheets and in the code worksheets in the IEC 61131-3 project in ProProg wt III.

This explicitly remains a task for the user (for security reasons).

In addition, CANopen-specific information, such as device type, CANopen profile, manufacturer/vendor etc., is displayed.

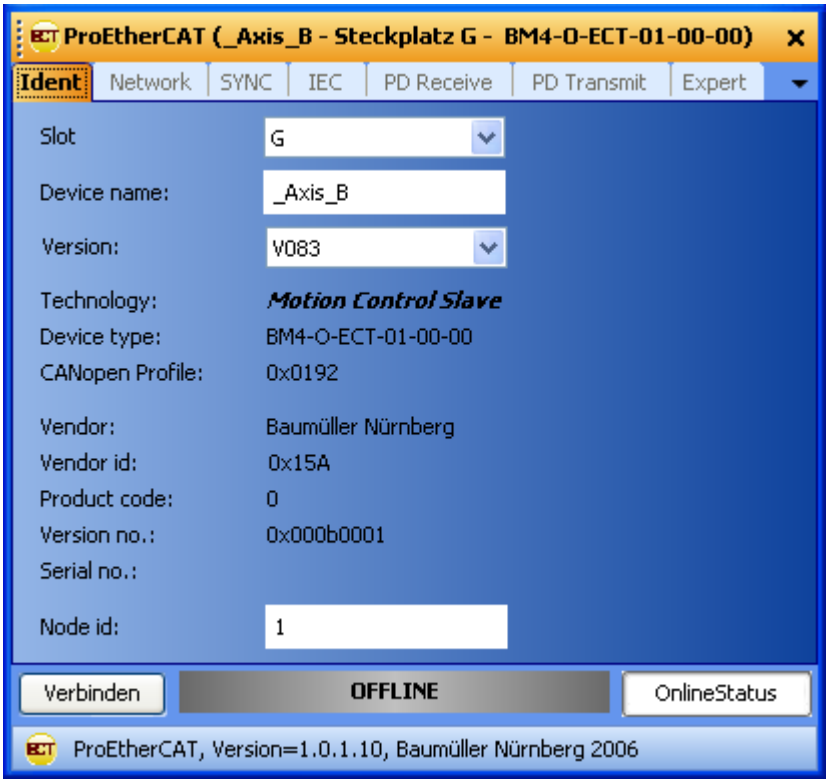


Figure 35: ProEtherCAT - EtherCAT slave communication configuration on Ident tab

Network tab

On the "Network" tab, you can set the slave status control interval and the mailbox scan interval.

The slave status control interval indicates the interval at which the EtherCAT master checks the NMT state of the respective slave.

The default value for the slave status control interval is 200 ms.

The mailbox scan interval indicates the time interval during which the EtherCAT master checks whether service data orders are present at the respective slave. The higher the value, the less frequent the service data orders are checked and the slower the processing of the service data order seems.

The default value for the mailbox scan interval is 5 ms.

The TCP/IP address of the EtherCAT slave for communication via Ethernet over EtherCAT (EoE), among other things, is set in the Tunneling (EoE) section.

The TCP/IP address must be set to the EtherCAT slave. For information on setting the IP address for EtherCAT slaves, please refer to the respective operating instructions of the EtherCAT slave.

If you activate the radio button „Yes“ at „Mandatory slave“, you determine that the slave must exist at the bus always.

Alternatively you can define a slave as „optional“ on this site.

For this activate the radio button „No“ at „Mandatory slave“ and write in the hereupon emerging input box „Ident“ (Identifier) the identifier setting on the slave (decode switch). See section ▹Optional slaves◃ in chapter "4.2 Fundamentals of the EtherCAT and EtherCAT networks" from page 49 onward.

NOTE

An identifier, which is used to identify optional slaves from the EtherCAT master, can not be set on all EtherCAT slaves. For this see the Operating Instructions of the respective EtherCAT slave.

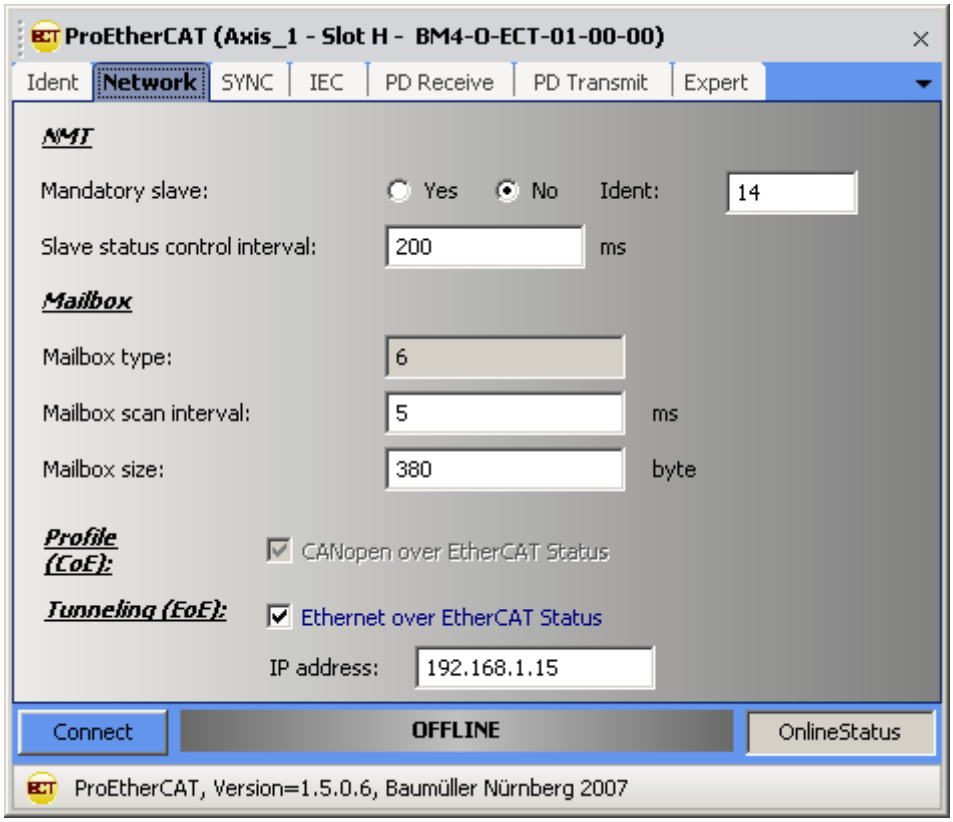


Figure 36: ProEtherCAT - EtherCAT slave communication configuration on Network tab

SYNC tab

On the „SYNC“ tab, you can set the „Cycle time of sync“ as well as the settings of „Distributed clocks“.

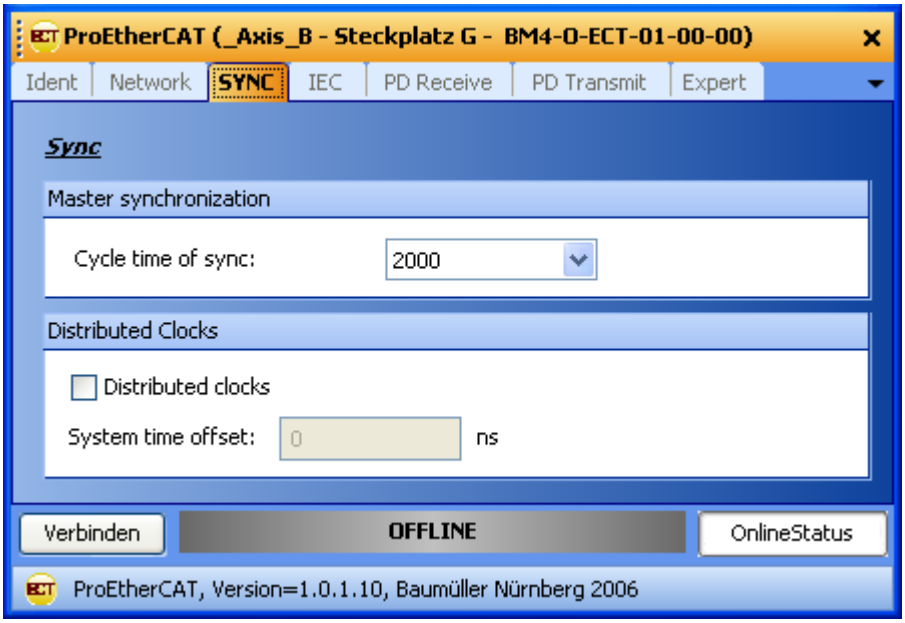


Figure 37: ProEtherCAT - EtherCAT slave communication configuration on SYNC tab

The „Cycle time of sync“ is the interval at which process data are exchanged between the EtherCAT master and the EtherCAT slaves.

Changes to the ProMaster project „Cycle time of sync“ affects all slaves on the same EtherCAT bus and the master. The changed „Cycle time of sync“ also affects the IEC project Motion Control event task.

NOTE

Any change to the „Cycle time of sync“ is applied automatically to the entire EtherCAT bus, i.e. the EtherCAT master and all connected EtherCAT slaves.

To activate the „Distributed clocks“ on the EtherCAT bus for this EtherCAT slave use the check box „Distributed clocks“.

In the EtherCAT slave b maXX System the „System time offset“ shifts the generation of the SYNC signal by distributed clocks.

IEC tab

When EtherCAT slave communication is configured, the EtherCAT slave objects are assigned to the network variables for the IEC project on the PLC (via EtherCAT master objects).

The variables displayed on the IEC tab are also called network variables. The process data is written and read in the IEC project via the network variables.

If motion control is used, some of the assignments are created by default.

The network variables and their data type in the IEC project on the PLC are displayed on the “IEC” tab.

Calculation of the (IEC) address of the network variables does not occur until after “Update list” in Section ▹Download tab◃ in Chapter "4.3.6.2 Configuration of EtherCAT master communication" from Page 76.

The configuration of other network variables occurs on the PD Receive and PD Transmit tabs.

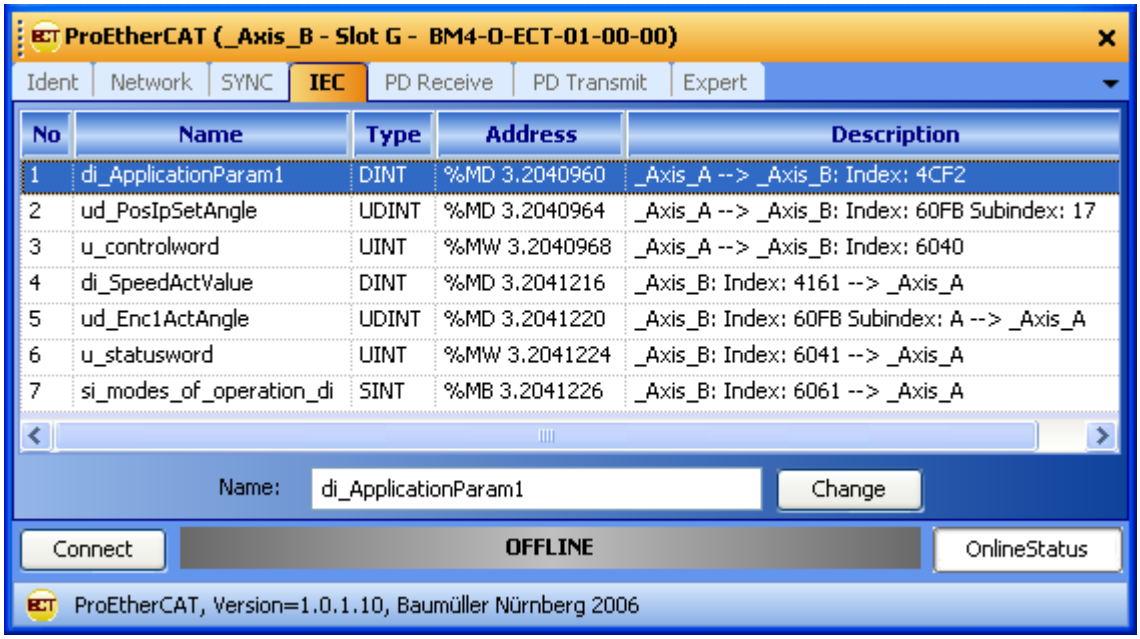


Figure 38: ProEtherCAT – EtherCAT slave communication configuration IEC tab after configuration of the EtherCAT slaves and “Update list” (EtherCAT master communication configuration, Download tab)

PD Receive tab

Communication from the master to the slave is configured on the “PD Receive” tab.

The default configuration is already set for motion control. You can expand on the configuration. Note that the motion-control configuration has a higher priority than the configuration by the user.

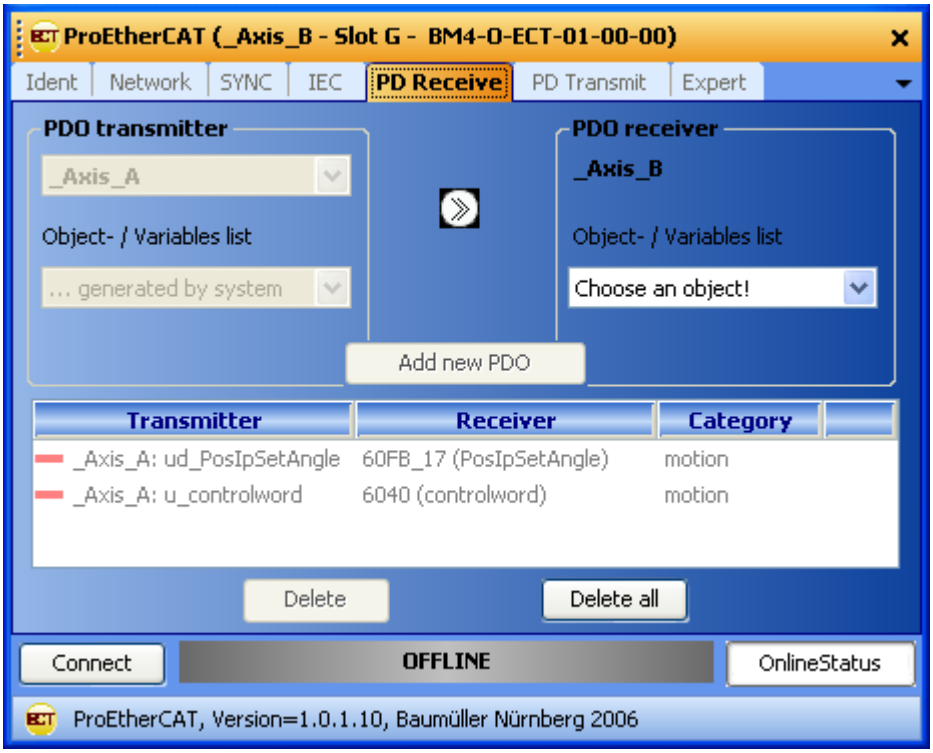


Figure 39: ProEtherCAT - Motion control EtherCAT slave communication configuration, “PD Receive” tab with default motion control configuration

With CANopen over EtherCAT (CoE), the PDO transmitter is the EtherCAT master.

Linking of an object in the EtherCAT slave with a network variable in the IEC project of the PLC with the EtherCAT master occurs here. This network variable is written from the IEC project.

Procedure:

- Select an object of the EtherCAT slave in the "Object/Variable list" combo box in the "PDO Receiver" section.

  Example: We select the EtherCAT slave object 0x4CF2 (application parameter 1) in the "Object/Variable list" combo box in the "PDO Receiver" section.

- The link that was just set is checked for conformity with the PDO entries and entered in the list with the "Add New PDO" button.

  The EtherCAT slave object 0x4CF2 (application parameter 1) is now linked to the network variables di_ApplicationParam1 of the EtherCAT master (or the IEC project of the _Axis_A device). The value written to di_ApplicationParam1 in the IEC project is written to the object 0x4CF2 of the EtherCAT slave.

  See ▹IEC tab◃ in Chapter "4.3.6.2 Configuration of EtherCAT master communication" from Page 75.

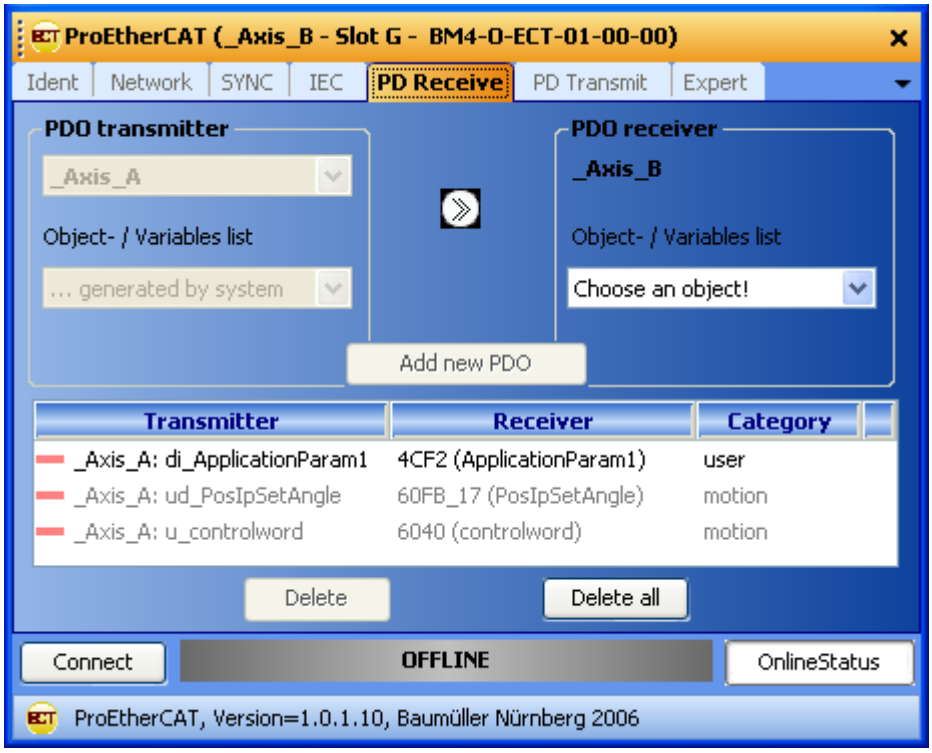


Figure 40: ProEtherCAT - Motion control EtherCAT slave communication configuration, “PD Receive” tab with default motion control configuration and additional object

NOTE

ProMaster automatically takes the 30-character limit for variable names in the IEC project in ProProg wt III into account.

The variable names from the additional configuration can be changed later on when configuring the PLC in Section ▹4.3.7.4 EtherCAT◃ from page 84 onward.

PD Transmit tab

Communication from the slave to the master is configured on the “PD Transmit” tab.

The default configuration is already set for motion control. You can carry out an additional configuration. Note that the motion-control configuration has a higher priority than the configuration by the user.

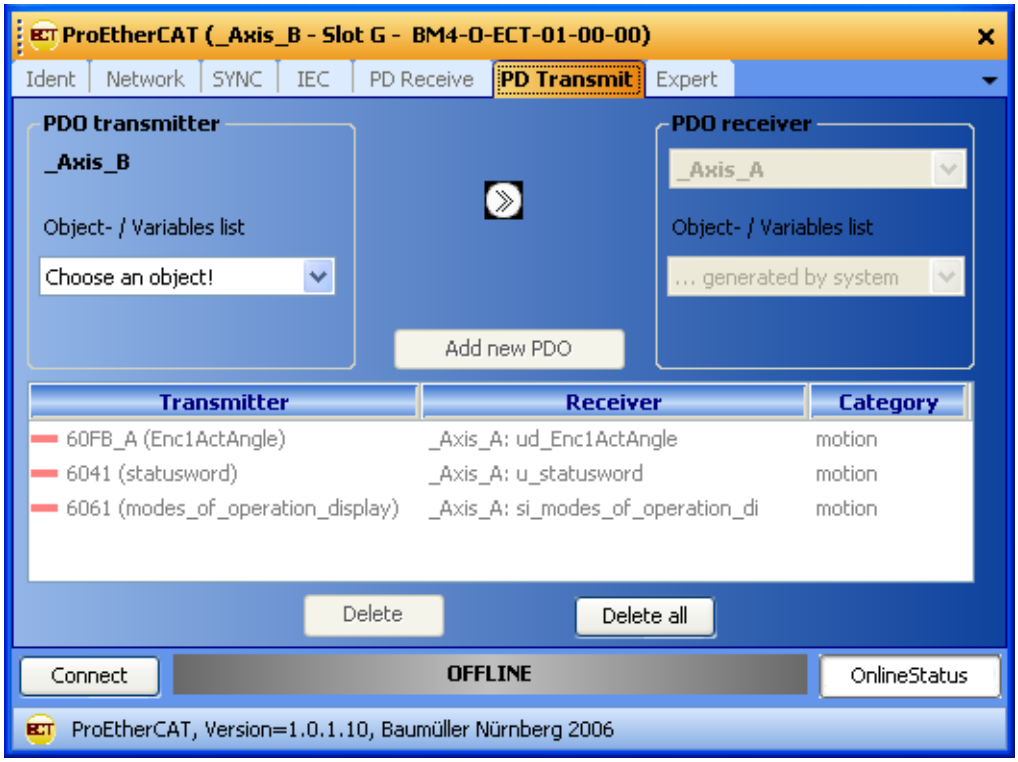


Figure 41: ProEtherCAT - Motion control EtherCAT slave communication configuration, “PD Transmit” tab with default motion control configuration

With CANopen over EtherCAT (CoE), the PDO receiver is the EtherCAT master.

Linking of an object in the EtherCAT slave with a network variable in the IEC project of the PLC with the EtherCAT master occurs here. This network variable is read in the IEC project.

Procedure:

- Select the object to be sent in the "Object/Variable list" combo box in the "PDO Sender" section.

  Example: We select the EtherCAT slave object 0x4161 (actual speed value) in the "Object/Variable list" combo box in the "PDO Sender" section.

- The link that was just set is checked for conformity with the PDO entries and entered in the list with the "Add New PDO" button.

  The EtherCAT slave object 0x4161 (actual speed value) is now linked to the network variables di_SpeedActValue of the EtherCAT master (or the IEC project of the _Axis_A device). The value of object 0x4161 of the EtherCAT slave is read from di_SpeedActValue in the IEC project.

  See ▹IEC tab◃ in Chapter "4.3.6.2 Configuration of EtherCAT master communication" from Page 75.

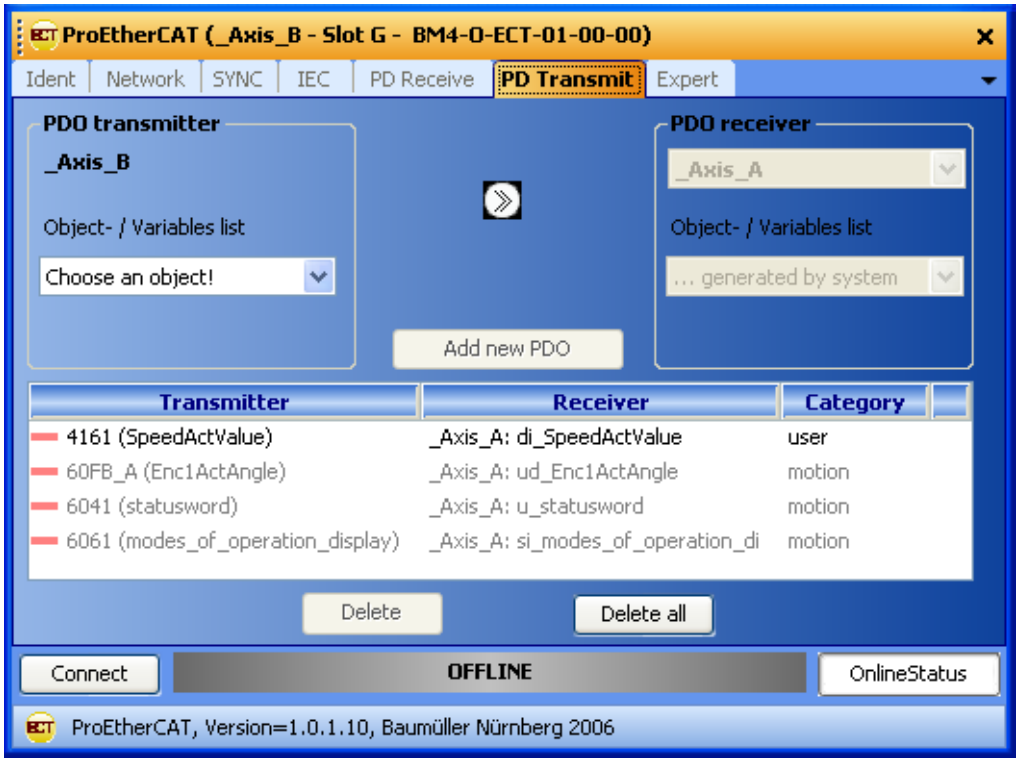


Figure 42: ProEtherCAT - Motion control EtherCAT slave communication configuration, “PD Transmit” tab with default motion control configuration and additional object

NOTE

ProMaster automatically takes the 30-character limit for variable names in the IEC project in ProProg wt III into account.

The variable names from the additional configuration can be changed later on when configuring the PLC in Section ▹4.3.7.4 EtherCAT◃ from page 84 onward.

Expert tab

If motion control is used, the "Expert" tab is used to display the objects available in the EtherCAT slave and their values.

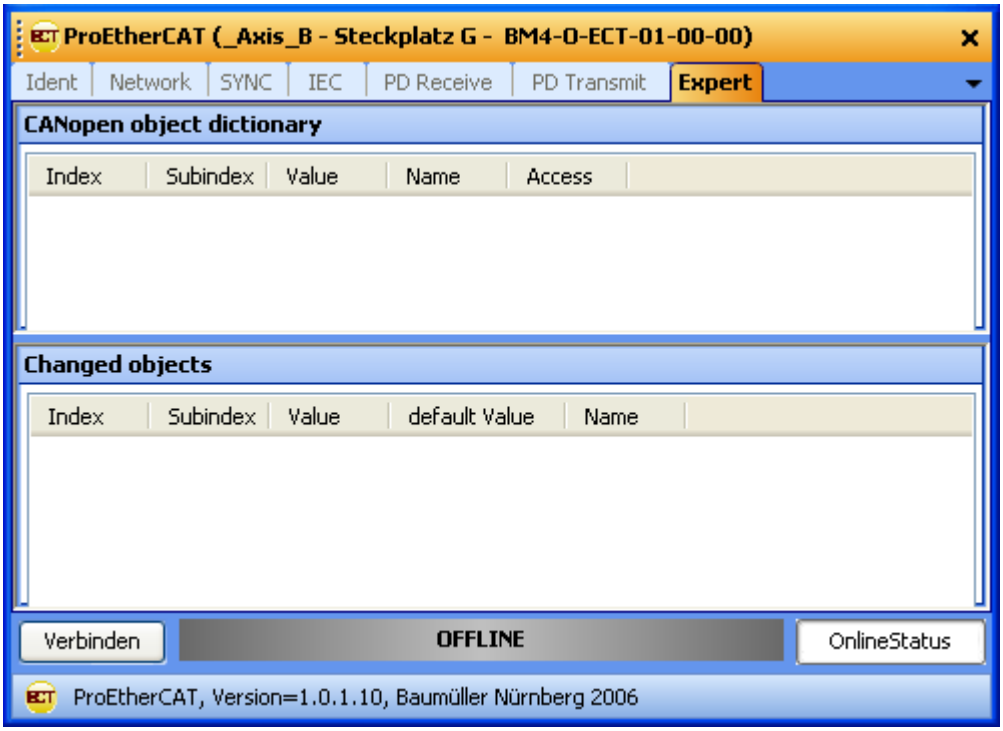


Figure 43: ProEtherCAT - EtherCAT slave communication configuration, "Expert" tab

4.3.6.2 Configuration of EtherCAT master communication

Open ProEtherCAT for our EtherCAT master by clicking the “_Axis_A” device and then selecting "Configuration Data (Components)\EtherCAT Master (Slot G)\Configure Master Bus (ProEtherCAT)" via the context menu.

When the machine configuration was created, the EtherCAT master was set by default so that the EtherCAT settings are configured for motion control.

The window for the EtherCAT master communication configuration is opened.

Ident tab

You can set the slot of the EtherCAT master module at the bmaXX drive PLC and the name of the PLC (device name) for motion control.

Changing the module number and the device name affects the ProMaster project.

In addition, the technology and CANopen-specific information, such as device type, CANopen profile, manufacturer/vendor etc., are displayed.

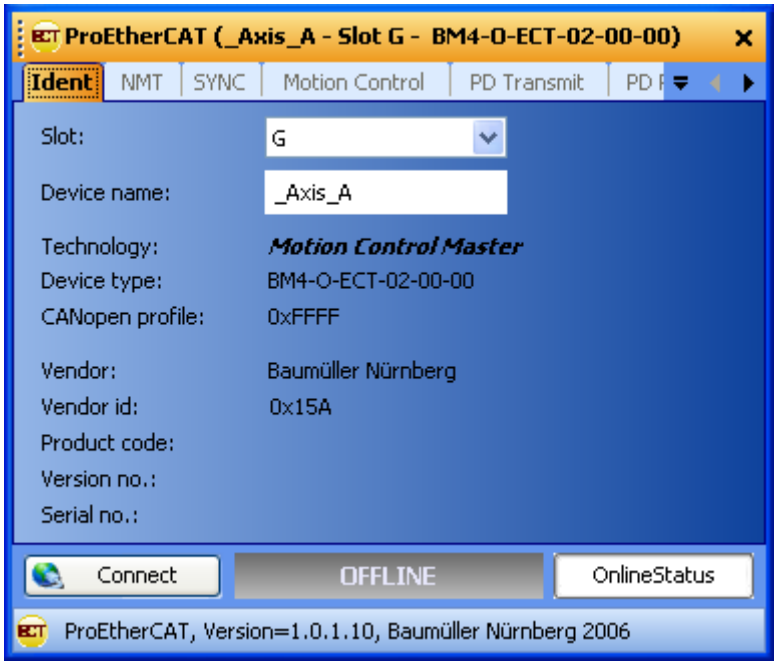


Figure 44: ProEtherCAT - EtherCAT master communication configuration, Ident tab

NMT tab

If motion control is used, the settings on the "NMT" tab remain unchanged.

You can specify the amount of time the EtherCAT master is to wait for a change of state by the slaves in the "Slave wait time" edit box.

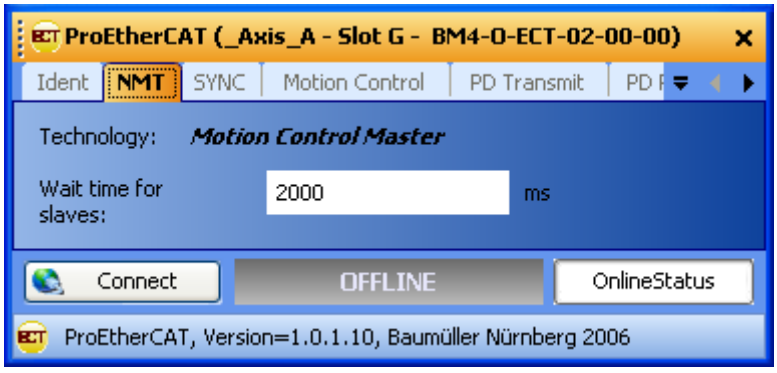


Figure 45: ProEtherCAT - EtherCAT master communication configuration, NMT tab

SYNC tab

If motion control is used, you can set the SYNC cycle time and the synchronization signal between the EtherCAT master option module and the b maXX drive PLC on the "SYNC" tab.

Furthermore you can set the „Distributed clocks“.

Changing the SYNC cycle time and the synchronization signal affects the ProMaster project.

You can select from "SYNC signal 1" and "SYNC signal 2" for "Synchronization with PLC (event)" for the EtherCAT master module. Other signals are not permissible. We use "SYNC signal 1" in our example. The signal must be generated on the b maXX drive PLC. If motion control is used, the "SNYC signal 1" is used and automatically generated by the b maXX drive PLC by default.

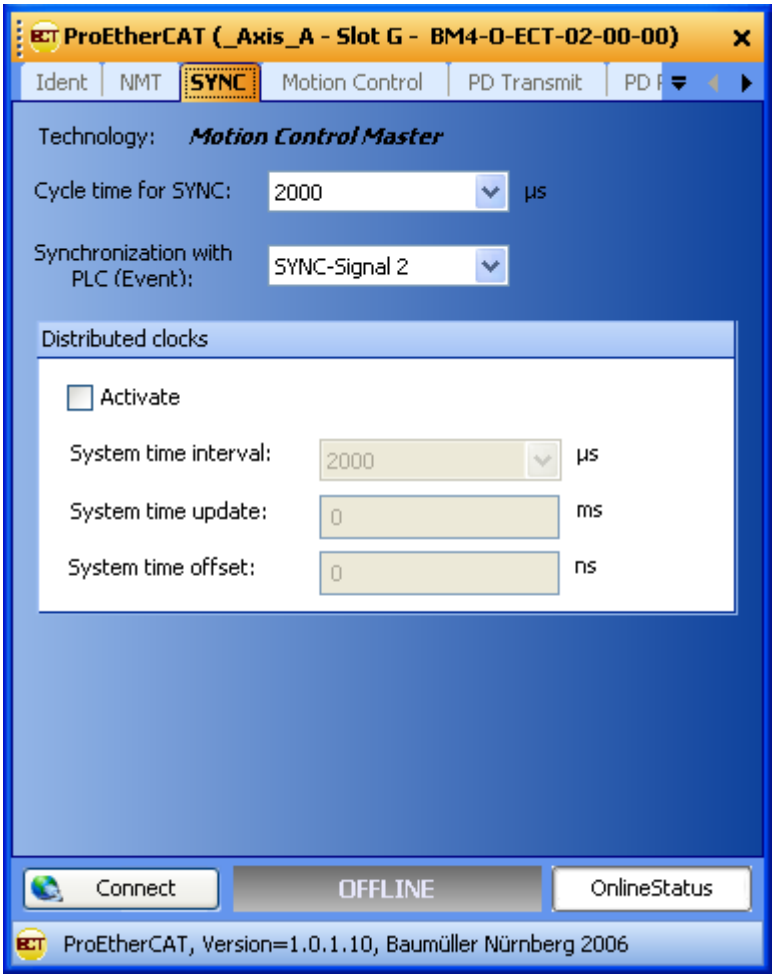


Figure 46: ProEtherCAT - EtherCAT master communication configuration, SYNC tab

To activate the „Distributed clocks“ on EtherCAT bus use the check box „Activate“.

The „System time interval“ is the interval at which process data are exchanged between the EtherCAT master and the EtherCAT slaves.

The „System time interval“ affects to the „Cyclic time of SYNC“.

Changes to the ProMaster project „System time interval“ or „Cycle time of SYNC“ affect all slaves on the same EtherCAT bus and the master. Furthermore the changed „Cycle time of sync“ affects the IEC project Motion Control event task.

NOTE

Any change to the „System time interval“ and „Cycle time of sync“ is applied automatically to the entire EtherCAT bus, i.e. the EtherCAT master and all connected EtherCAT slaves.

The „System time update“ is the interval at which the system times at the EtherCAT bus (master and slaves) are updated.

In the EtherCAT master b maXX system the „System time offset“ shifts the generation of the SYNC signal by distributed clocks.

Motion Control tab

The settings for motion control are made on the “Motion Control” tab.

For the machine configuration, you can specify whether motion control is to be used. Selection is made possible by choosing the master in the project tree.

It is also possible to specify whether or not individual drives are controlled via motion control and to change the name of individual drives. Selection is made possible by choosing the respective drive in the project tree. The changes affect the ProMaster project.

NOTE

If you use drives without motion control in the configuration, they must be configured in any case (refer to ▹Configuring EtherCAT slave communication◃ in Chapter "4.3 ProMaster, ProEtherCAT, ProProg wt III and motion control" from Page 60).

In addition, motion control-specific information, such as the motion control cycle time, and EtherCAT-specific informations, such as the node ID of the respective drive is displayed.

In this tab you can change among other things the axis name of the EtherCAT slaves, as well as the TCP/IP address for the Ethernet over EtherCAT (EoE) connection if necessary.

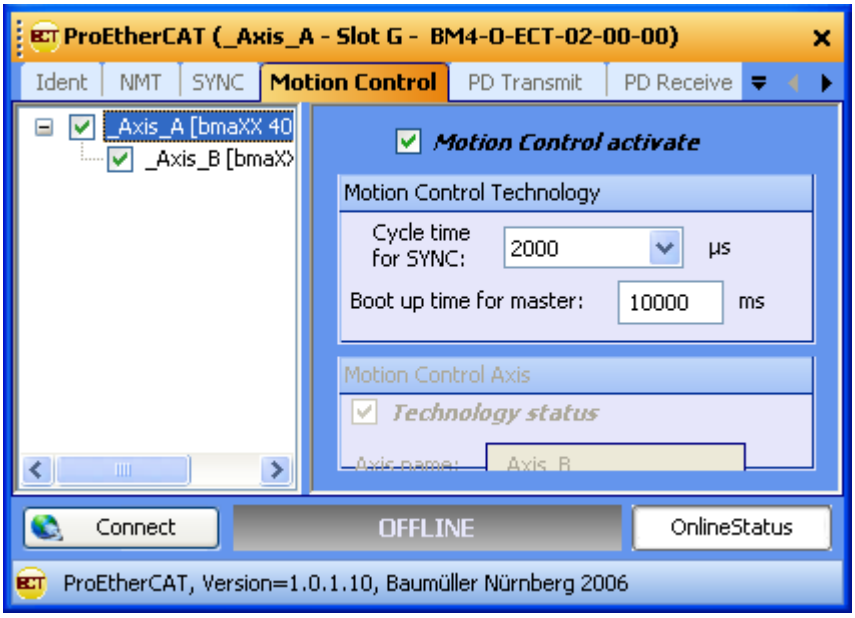


Figure 47: ProEtherCAT - EtherCAT master communication configuration, Motion Control tab

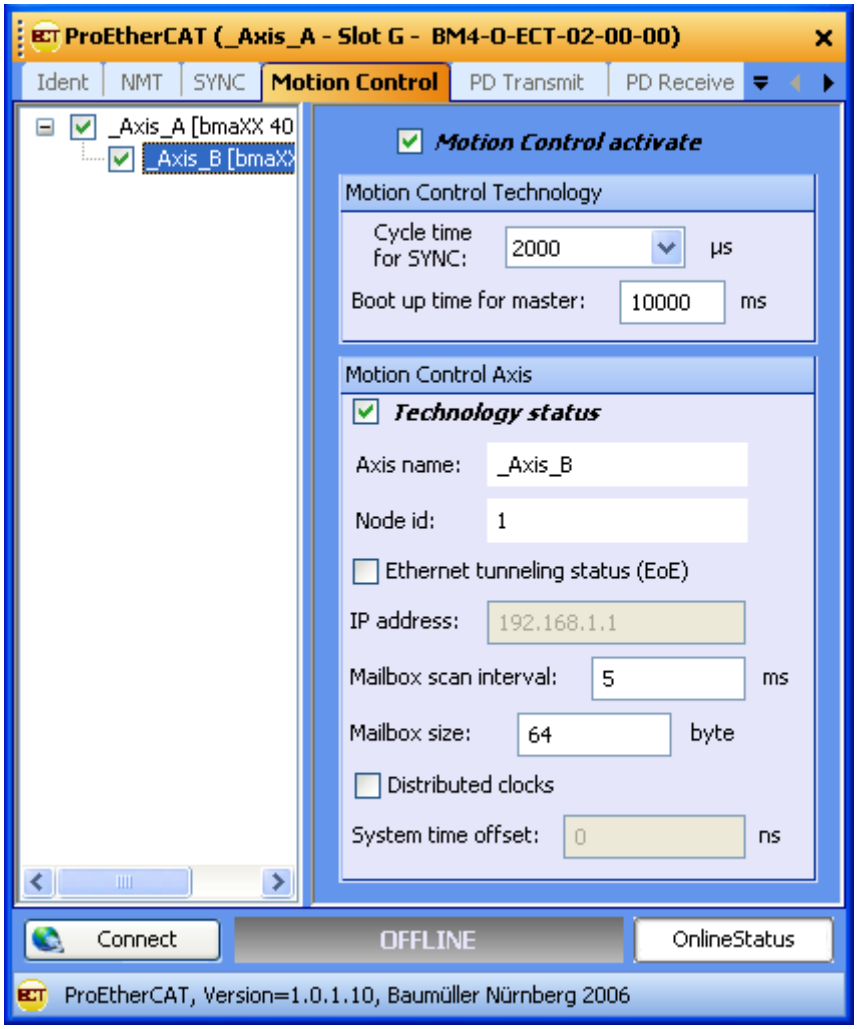


Figure 48: ProEtherCAT - EtherCAT master communication configuration on Motion Control tab

PD Transmit and PD Receive tabs

The data on these pages is used for display.

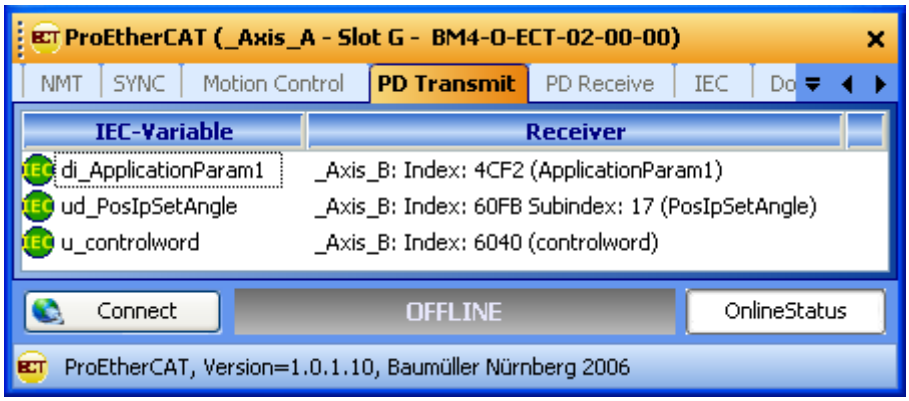


Figure 49: ProEtherCAT - EtherCAT master communication configuration, PD Transmit tab

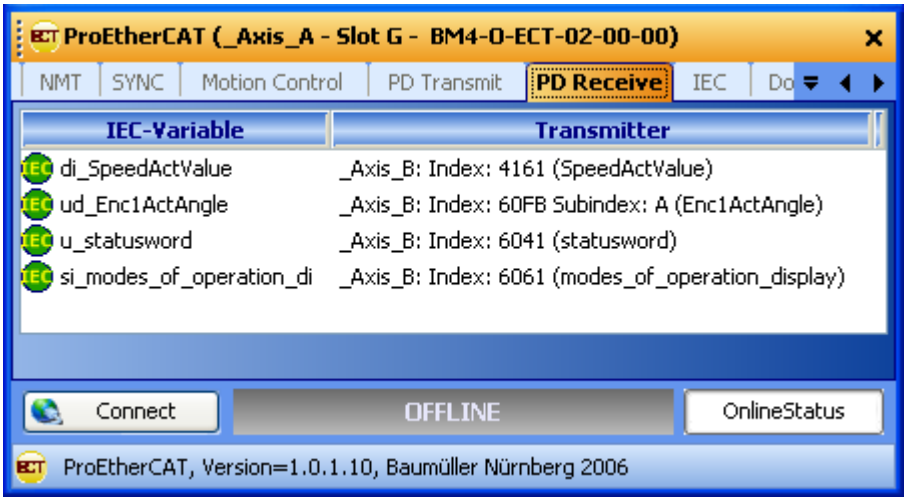


Figure 50: ProEtherCAT - EtherCAT master communication configuration, PD Receive tab

The IEC variables (network variables) for transmitting and receiving are displayed on the "PD Transmit" and "PD Receive" tabs, respectively.

Assignment of the network variables (via the EtherCAT master objects) to the EtherCAT slave objects occurs automatically with ProMaster and ProEtherCAT.

The assignment of additional EtherCAT slave objects to network variables occurs when configuring EtherCAT slave communication of the EtherCAT slaves. After this configuration of the EtherCAT slaves, the corresponding data is displayed on the “PD Transmit” and “PD Receive” tabs of the configuration of EtherCAT master communication.

For configuration of EtherCAT communication of the EtherCAT slaves, see Section ▹4.3.6.1 Configuring EtherCAT slave communication◃ from page 60 onward.

IEC tab

When EtherCAT slave communication is configured, the EtherCAT slave objects are assigned to the network variables for the IEC project on the PLC (via EtherCAT master objects).

The variables displayed on the IEC tab are also called network variables. The process data is written and read in the IEC project via the network variables.

If motion control is used, some of the assignments are created by default.

The network variables and their data type in the IEC project on the PLC are displayed on the “IEC” tab.

Calculation of the (IEC) address of the network variables does not occur until after “Update list” in Section ▹Download tab◃ in Chapter "4.3.6.2 Configuration of EtherCAT master communication" from Page 76.

For configuration of EtherCAT communication of the EtherCAT slaves, see ▹4.3.6.1 Configuring EtherCAT slave communication◃ from page 60 onward.

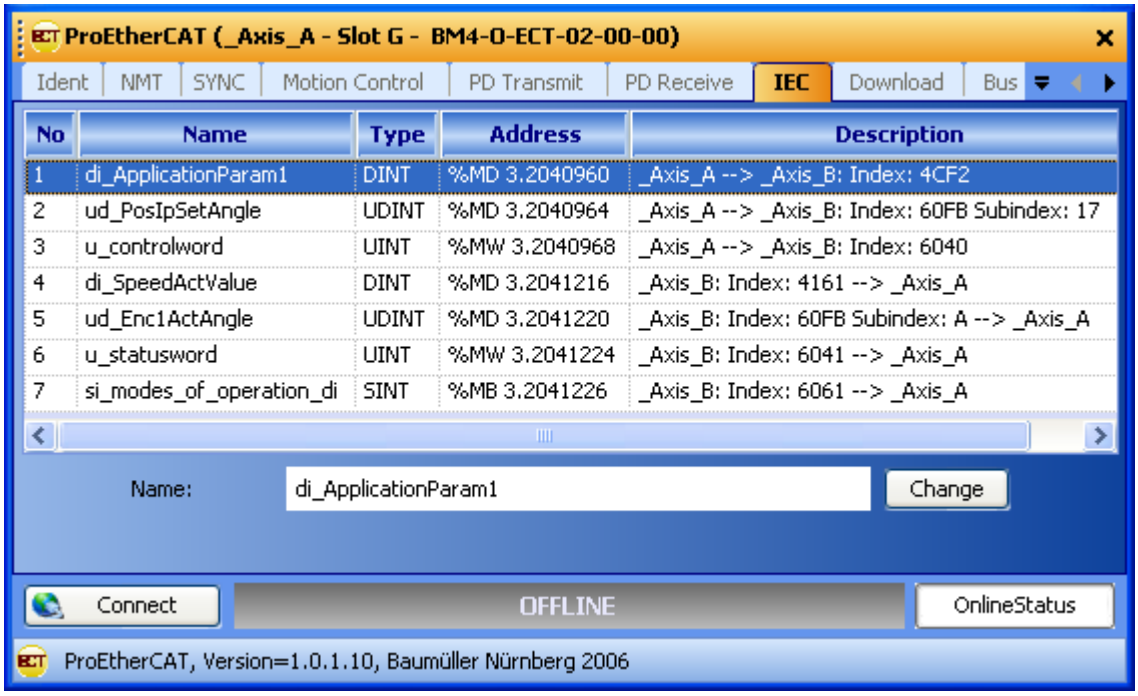


Figure 51: ProEtherCAT - EtherCAT master communication configuration IEC tab after configuration of the EtherCAT slaves and “Update list” (Download tab)

You can change the name of the network variables by selecting the network variable names from the “Name” column, editing the network variable name in the "Name" edit box and then clicking the "Change" button.

NOTE

Note the 30-character limit for variable names in the IEC project in ProProg wt III.

Download tab

The Download tab shows the objects that were downloaded to the EtherCAT master after configuration of EtherCAT communication of the EtherCAT master and the EtherCAT slave. As long as the configuration is not complete, no data is displayed on the Download tab, i.e. no data can be downloaded to the EtherCAT master.

Now press the “Update List” button (bottom right). This generates the configuration data (both master and slave) for the download to the EtherCAT master.

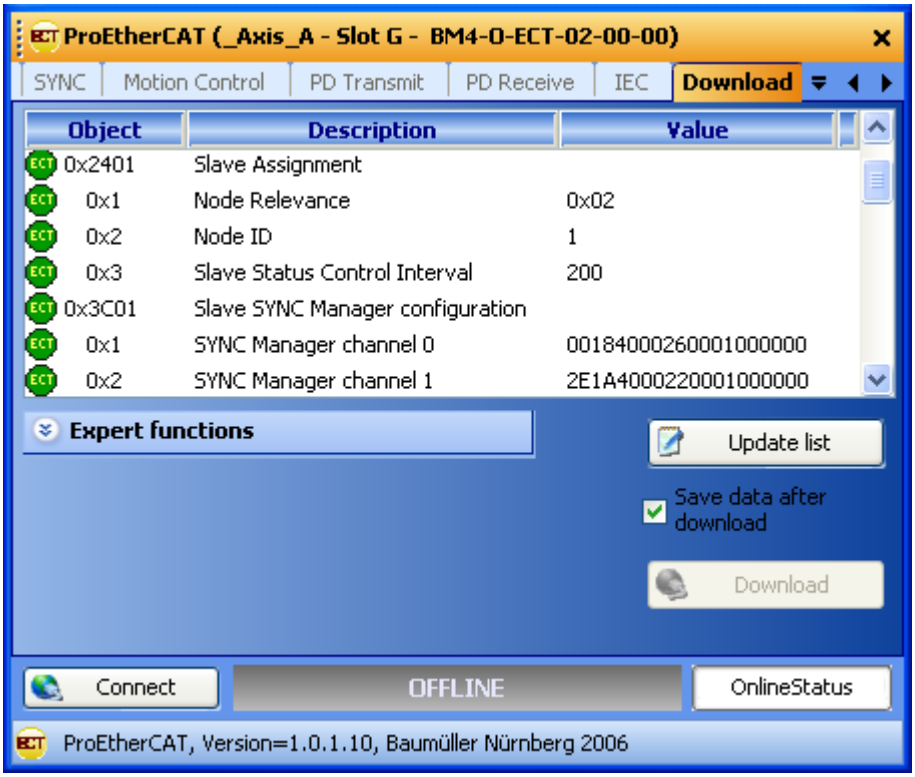


Figure 52: ProEtherCAT - EtherCAT master communication configuration on Download tab

Now press the “Download” button (after pressing „Connect“ button). The connection from the ProMaster to the EtherCAT master (BM4-O-ECT-02 option module) is established via the b maXX drive PLC (BM4-O-PLC-01 option module). The EtherCAT configuration data (both master and slave) is then sent to the EtherCAT master.

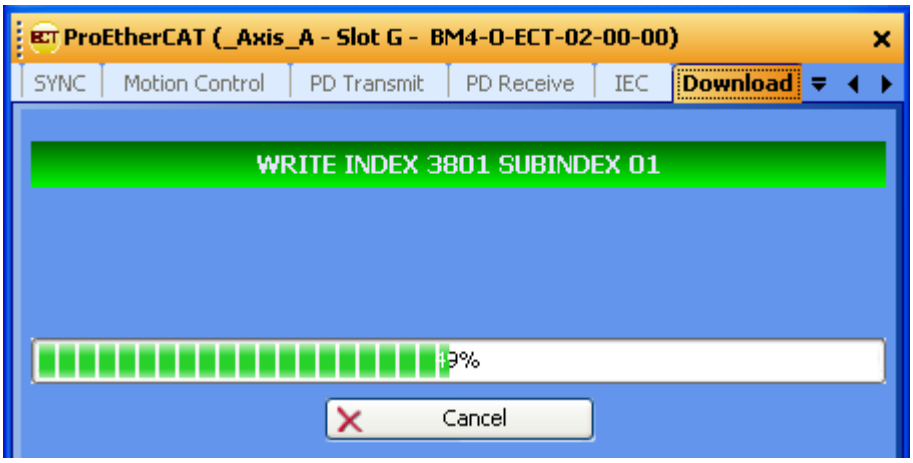


Figure 53: ProEtherCAT - EtherCAT communication configuration, Download tab

If errors are reported here, it may be necessary to switch the EtherCAT master to the EtherCAT status "Reset" via the "Connect" and "Reset" buttons on the "Bus Control" tab and then restart the download on the "Download" tab.

Once the download is complete, the download object list is displayed again.

The EtherCAT master now contains both the EtherCAT configuration data for the EtherCAT master and the EtherCAT configuration data for the EtherCAT slave, which is sent by the master to the slave when the EtherCAT network is started.

Bus Control tab

Experienced EtherCAT users can control the EtherCAT network "manually" on the “Bus Control” tab after downloading the EtherCAT configuration data. This can be helpful, e.g. when physically commissioning the EtherCAT network.

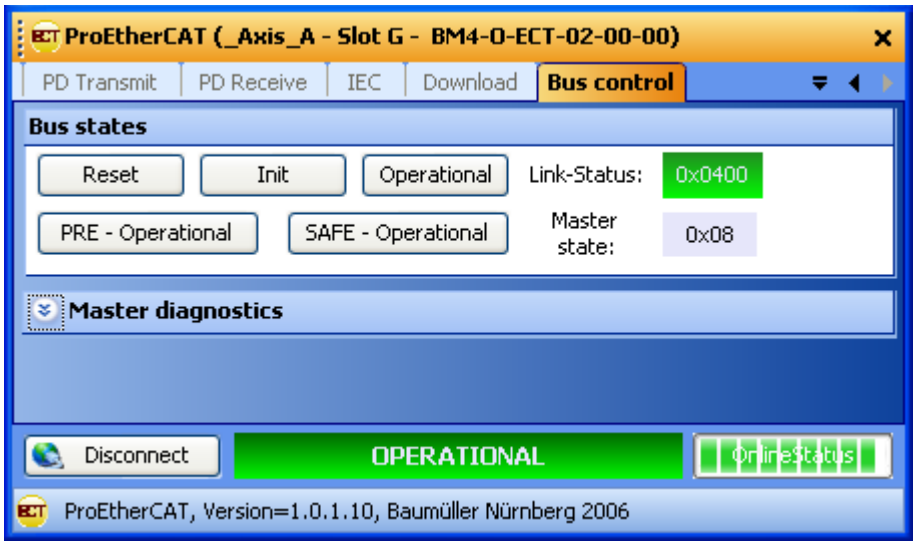


Figure 54: ProEtherCAT - EtherCAT master communication configuration, “Bus Control” tab

4.3.7 Configuration of the PLC

Once the configuration of EtherCAT network communication and the creation of the IEC project have occurred, the data for the b maXX drive PLC is to be configured.

For this purpose,

- the IEC project is included in the ProMaster project
- the cam data sets are selected

In addition,

- cam data can be generated with the cam editor ProCAM

Then click the "Update total IEC project" button in the "ProPLC" window. The data for the machine configuration, both for the EtherCAT network and the b maXX drive PLC, is then generated.

4.3.7.1 IEC

This section explains how an IEC project is linked with ProMaster.

Open the "ProPLC" window in the network view for our EtherCAT master in the ProMaster project. Click the “_Axis_A” device and then select "Configuration Data (Components)\PLC (Slot H)\PLC - Configuration (ProPLC)" via the context menu and then the "IEC" tab.

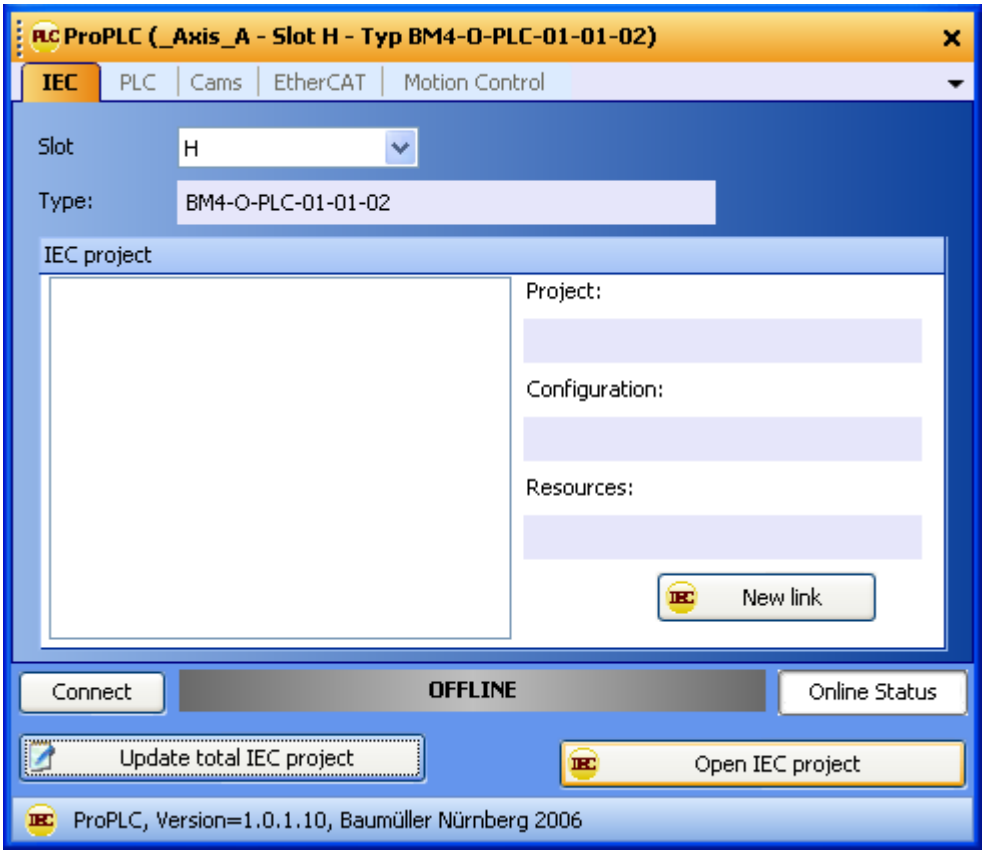


Figure 55: ProPLC - Connecting with example IEC project

NOTE

Only ProProg wt III projects can be opened with ProMaster. If you would like to use an existing PROPROG wt II project, you must first open it with ProProg wt III (thus converting it) and can then open and use it in ProMaster.

Note that your PROPROG wt II libraries will also be converted here!

Click the "New link" button and select our example IEC project "Example_BM4_ECT02_MA_2.mwt" created in Section ▹4.3.4 Creating an IEC project with ProProg wt III◃ from page 53 onward.

The IEC project just selected is displayed on the "IEC" tab.

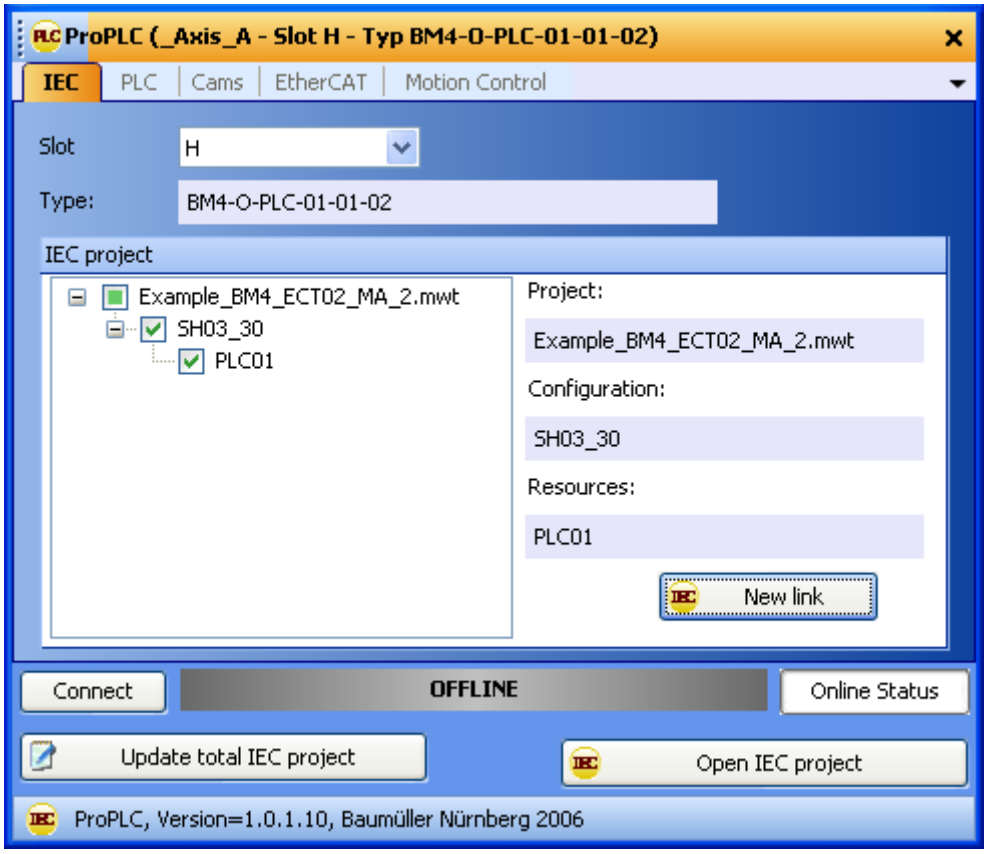


Figure 56: ProPLC - IEC tab

If you link an existing application (or template) from ProProg wt III with the ProMaster project, the names of the devices in the ProMaster may not match those in the IEC project. In this case, see Section ▹4.3.9 Programming the IEC project◃ from page 88 onward.

4.3.7.2 PLC

After downloading the data to the EtherCAT master and b maXX drive PLC (see ▹Downloading the data to the EtherCAT master and the b maXX drive PLC◃ from Page 87), the IEC project on the b maXX drive PLC can be activated and started on the PLC tab ("PLC Status" section).

In addition, data of the b maXX drive PLC, such as the operating system version, firmware version, IEC project and boot project, is also displayed in the "PLC Info" section. Information on the files in the flash memory of the b maXX drive PLC, e.g. on the cam records (see ▹4.3.7.3 Cams◃) is displayed in the "Flash Directory" section (after "Connect" and "Update")

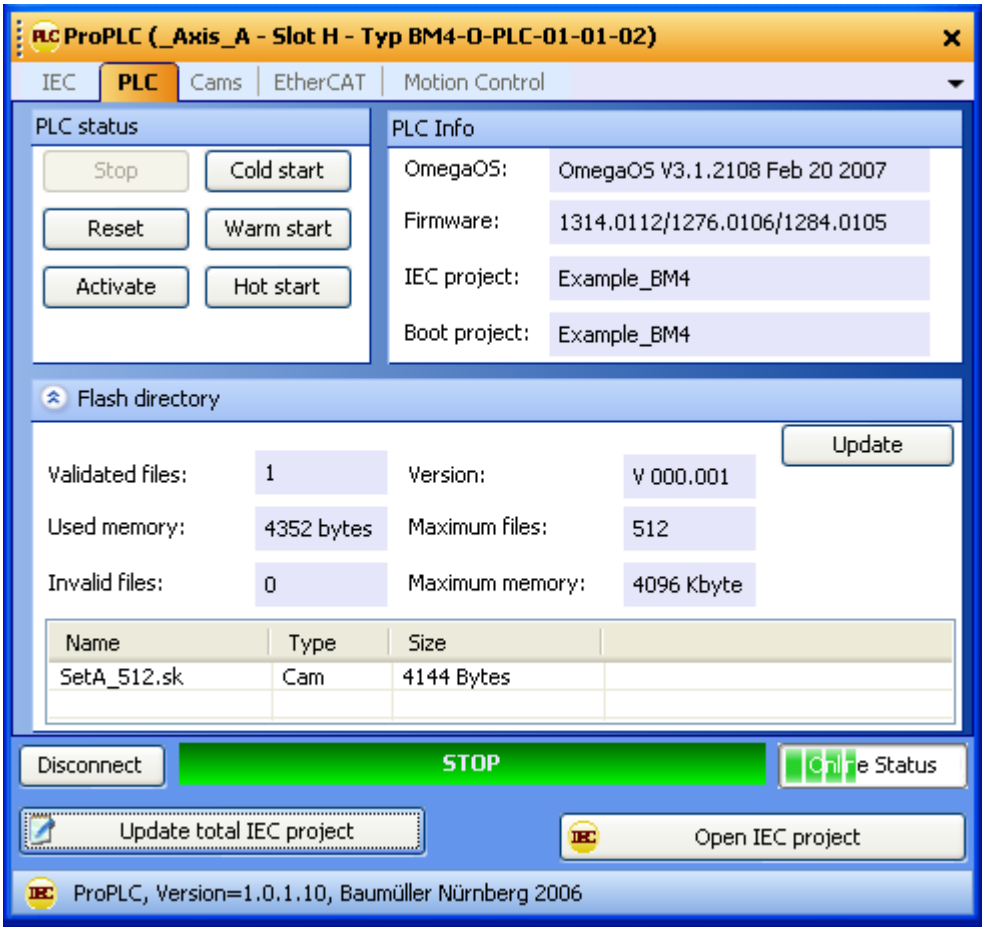


Figure 57: ProPLC - PLC tab after download of the cam (cam tab)

4.3.7.3 Cams

This section explains how a cam data set is generated and linked with ProMaster.

Open the "PLC Configuration" window in the network view for our EtherCAT master in the ProMaster project by clicking the “_Axis_A” device and then select "Configuration Data (Components)\PLC (Slot H)\PLC - Configuration (ProPLC)" via the context menu and then the "Cams" tab.

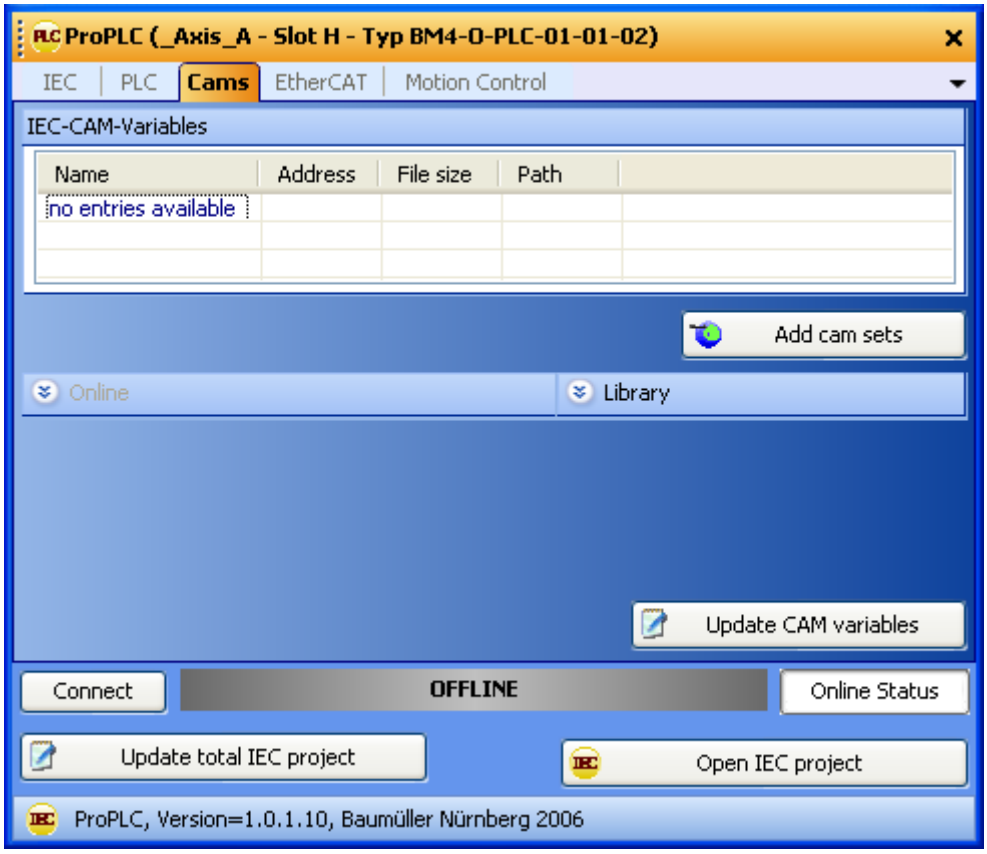


Figure 58: ProPLC - “Cams” tab with no cam data set

Press the "Add cam sets" button. The "ProMaster.NET - Cams" window is opened.

Press the "Add" button in the "Hard Disk" section and select the path on your hard disk that contains your cam data.

NOTE

Example cam data is found in the installation directory of ProCAM in the "examples" subfolder (e.g. C:\...\Baumueller\ProCam 2\examples).

If you would like to create a cam, click the "ProCAM" button, which opens the cam editor. You can edit your cams there.

Now create your cam data set (in the "*.sk Files" section) by dragging and dropping different cam data (in the "Hard Disk" section).

In our example, we select the cam data "ExampleGerade_MC.kbin" and "ExamplePendelP5_MC.kbin", which are combined in the “SetA_512” cam data set.

NOTE

You can display the cam data graphically (in the respective section) via the "View data" button. This simplifies cam selection.

Then click the "Save" button. This creates the cam data set and makes it available in ProMaster.

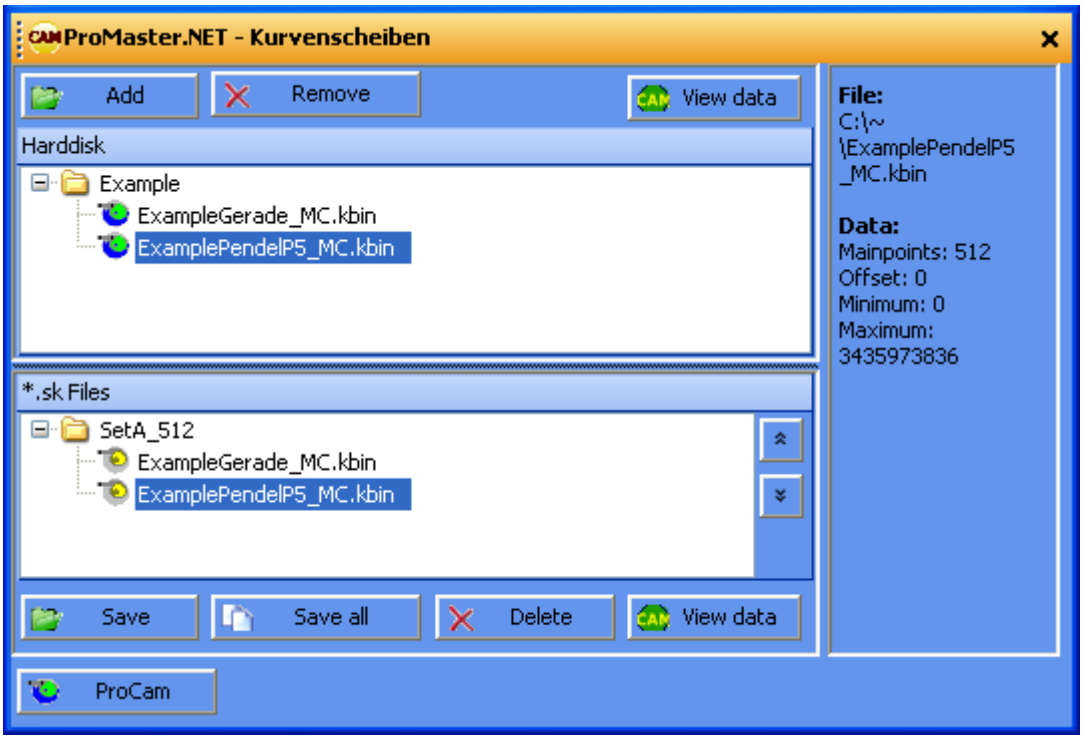


Figure 59: ProPLC (- Cams tab -) Generating cam data set

Now close the "ProMaster.NET - Cams" window by clicking "x" on the top right.

You can now see the name of the variables of the cam data set in the IEC project, its address in the IEC project and the file name of the cam data set on the hard disk via the "Cams" tab.

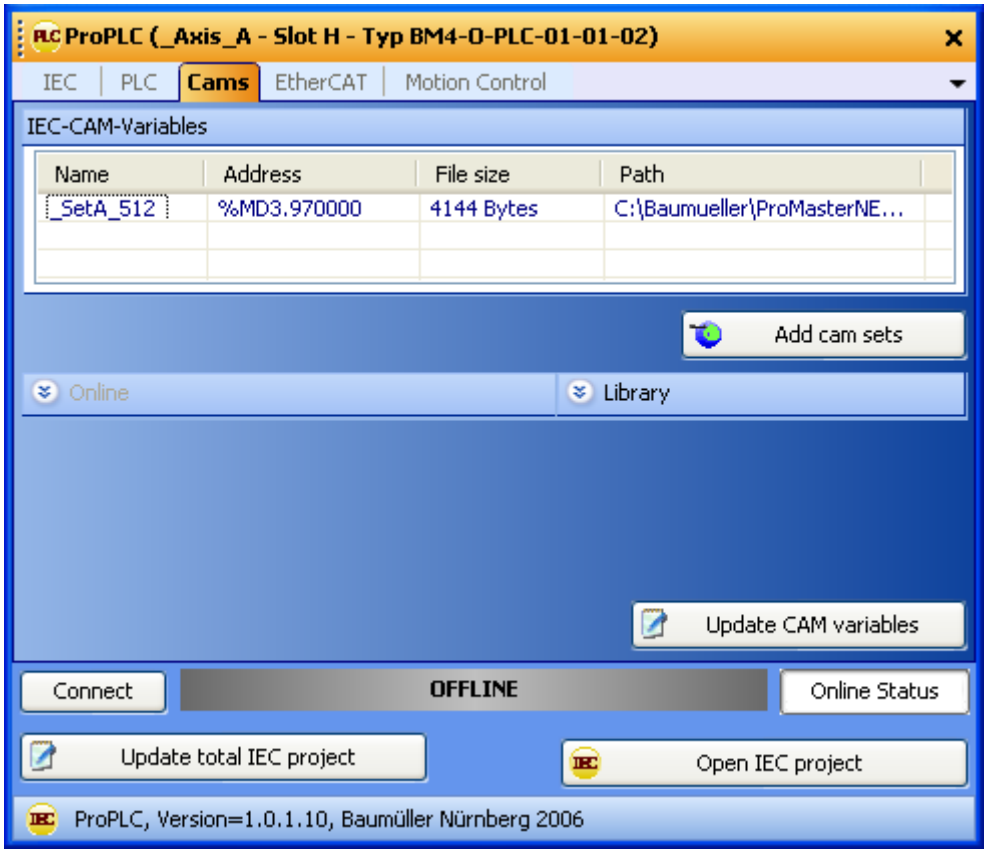


Figure 60: ProPLC - “Cams” tab with cam data set

If you link an existing application (or template) from ProProg wt III with the ProMaster project, the cam data set name in ProMaster may not match that in the IEC project. In this case, see Section ▹4.3.9 Programming the IEC project◃ from page 88 onward.

From the "Cams" tab, you can clear the flash memory on the b maXX drive PLC via the "Clear Flash" button ("⇓ Online" section). This may be necessary if the flash memory is "full" because of various downloads (both cam data sets and IEC projects).

In addition, you can download cam data sets to the b maXX drive PLC individually via the "Download Data Sets" button on the "Cams" tab.

Now load the cam data sets onto the b maXX drive PLC via the “Download Data Sets” button ("⇓ Online" section).

4.3.7.4 EtherCAT

The name, data type, address and comment of the network variables of the EtherCAT slave in the IEC project are displayed on the "EtherCAT" tab.

If motion control is used, communication occurs via the axis variable. ProMaster recreates the axis variable with the device name of the EtherCAT slave after the “Update total IEC project” button is pressed (see Section ▹4.3.8 Downloading the data to the EtherCAT master and the b maXX drive PLC◃ from page 87 onward). ProMaster also creates network variables in the IEC project. These are divided into standard network variables for motion control and, if additional objects were created during configuration of the EtherCAT slave, additional network variables for EtherCAT.

The "EtherCAT" tab displays these network variables, both the standard network variables for motion control and the additional network variables for EtherCAT.

The following must be heeded:

The standard network variables for motion control may be read, but not written, by the user.

These are the standard motion control network variables for the set values:

"u_controlword.."	for the control word of the axis
"ud_PosIpSetAngel.."	for the synchronization position set angle of the axis

These are the standard motion control network variables for the actual values:

"u_statusword.."	for the status word of the axis
"ud_Enc1ActAngle.."	for the synchronization position actual angle of the axis
"si_modes_of_operation_display.."	for the operating mode of the axis

NOTE

The standard network variables for motion control may be read by the user, but not written.

The number after the network variable name is an internal number used to differentiate between identically sounding, automatically generated network variable names.

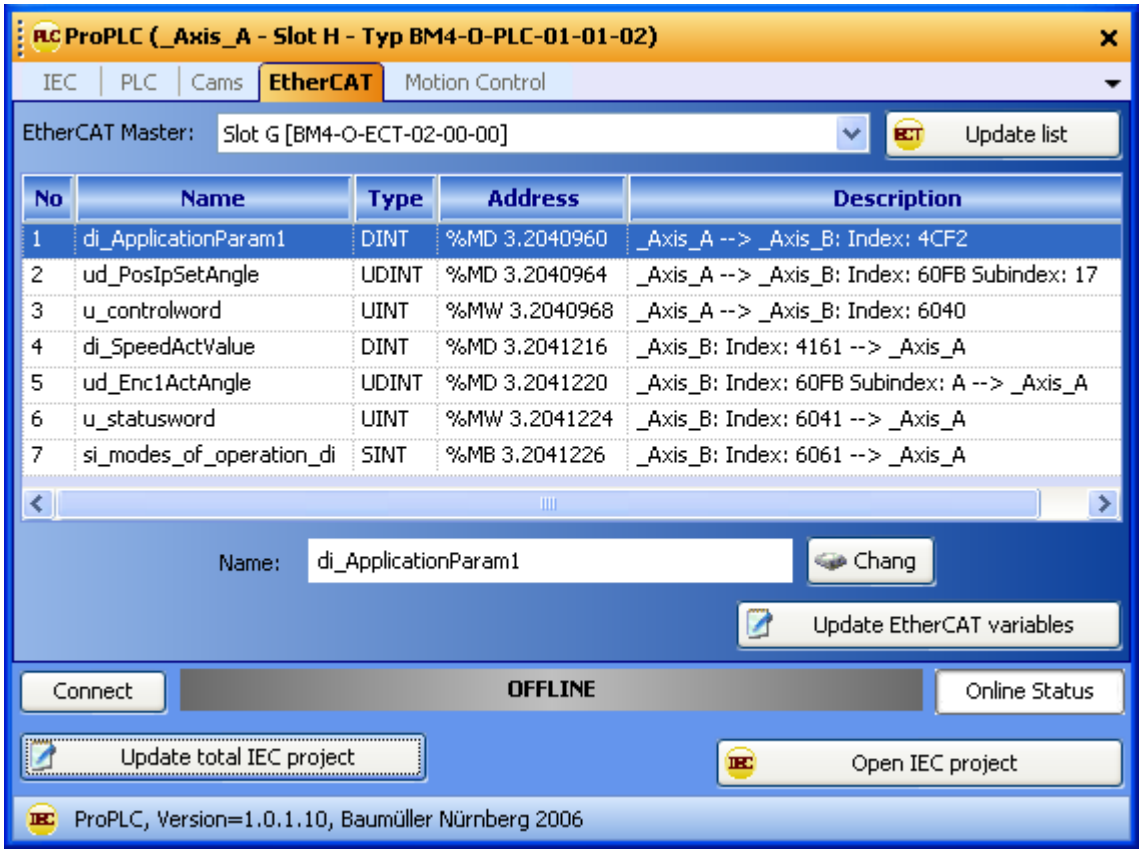


Figure 61: ProPLC - EtherCAT tab

If you created the additional network variables (and their link to EtherCAT slaves) that were suggested when configuring EtherCAT slave communication in Sections ▹PD Receive tab◃ from Page 64 and ▹PD Transmit tab◃ from Page 66, you will see the additional network variables.

The additional network variable for the set value is:

"di_ApplicationParam1.." for application parameter 1 of the axis

The additional network variable for the actual value is:

"di_SpeedActValue.." for the actual speed value of the axis

When changing the network variable names, note that each variable name can be issued only once in the IEC project. This is especially important to note when using several EtherCAT master option modules on the b maXX drive PLC.

NOTE

Each variable name may be issued only once in the IEC project.

NOTE

The additional network variables for set values must be written in the motion control event task.

The additional network variables for set values must be written at every call of the motion control event task.

4.3.7.5 Motion Control

All general settings for motion control will be made on the "Motion Control" tab in the future.

If you would like to use the default motion control settings, you do not need to make any settings here.

4.3.8 Downloading the data to the EtherCAT master and the b maXX drive PLC

At present, downloading of the data for the EtherCAT master occurs via ▹Download tab◃ in Chapter "4.3.6.2 Configuration of EtherCAT master communication" from Page 76.

Downloading of the cam data sets for the b maXX drive PLC currently occurs via the “Cams” tab of the PLC configuration. For this purpose, see Section ▹4.3.7.3 Cams◃ from page 81 onward.

Downloading the IEC project for the b maXX drive PLC currently occurs via ProProg wt III, as usual.

Now click the "Update total IEC project" button in the "PLC Configuration" window. The data for the b maXX drive PLC is now generated. This procedure can take some time, since ProProg wt III is opened and the configured IEC variables are written in the global variable worksheet in the IEC project, among other things.

Afterwards you can compile the IEC project by confirming the correspondent request with „Yes“.

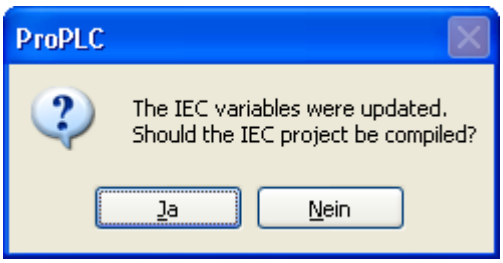


Figure 62: ProMaster - compile the updated IEC project

You can compile alternatively the IEC project in ProProg wt III via the ProProg wt III menu “Build\Rebuild Project”.

Now download the IEC project to the b maXX drive PLC (ProProg wt III menu “Online\Project control...” → “Send” → “Send” Boot Project).

Now perform a reset of the bmaXX drive PLC and then switch the PLC to the “RUN” state (alternatively, you can switch the entire EtherCAT network off and then on again). You can now control the local axis _Axis_A and the EtherCAT slave axis _Axis_B via the IEC project in ProProg wt III.

4.3.9 Programming the IEC project

4.3.9.1 General information

For information on how to program a motion control application in the IEC project in ProProg wt III, please refer to the motion control application manual and the Online Help System of ProProg wt III.

ProProg wt III is opened with our IEC project "Example_BM4_ECT02_MA_2.mwt" via the context menu on _Axis_A "Configuration Data (Components)\PLC (Slo H)\IEC Programming (ProProg wt III)". You can edit the IEC project as usual.

ProMaster has written

- the motion-control axis variables (in the MC_AxisVariables section)
- the cam data set (in the MC_CamDataSet section)
- the network variables (in the EtherCATVariables section) (standard network variables for motion control and additional network variables for EtherCAT, see ▹4.3.9.2 Exchanging data◃ from page 90 onward)
- the local axis variables (in the LocalAxisVariables section)

in the global variable worksheet "Global_Variables".

Global_Variables:SH03_30.PLC01

	Name	Type	Usage	Description	Address
⊞	**Global_Variables**				
⊟	**MyApplication**				
	_MyMaster	AXI...	VAR_GLO...	Not from ProMaster	
⊟	**MC_AxisVariables**				
	_Axis_A	AXI...	VAR_GLO...	1. Achse	%MD3.451500
	_Axis_B	AXI...	VAR_GLO...	2. Achse	%MD3.457300
⊞	**MC_MasterRef**				
⊞	**MC_SystemReserved**				
⊞	**MC_AxisRef**				
⊞	**MC_BaciInitRef**				
⊞	**MC_AxisPDORef**				
⊞	**MC_TriggerRef**				
⊞	**MC_IRPRef**				
⊟	**MC_CamDataSet**				
	s_File_SetA_512	File...	VAR_GLO...		%MD3.970000
	_SetA_512	MC_...	VAR_GLO...		%MD3.970000
⊟	**EtherCAT Variables**				
	di_ApplicationParam1	DINT	VAR_GLO...	_Axis_A --> _Axis_B: Index: 4CF2	%MD 3.2040960
	ud_PosIpSetAngle	UDINT	VAR_GLO...	_Axis_A --> _Axis_B: Index: 60FB Subindex: 17	%MD 3.2040964
	u_controlword	UINT	VAR_GLO...	_Axis_A --> _Axis_B: Index: 6040	%MW 3.2040968
	di_SpeedActValue	DINT	VAR_GLO...	_Axis_B: Index: 4161 --> _Axis_A	%MD 3.2041216
	ud_Enc1ActAngle	UDINT	VAR_GLO...	_Axis_B: Index: 60FB Subindex: A --> _Axis_A	%MD 3.2041220
	u_statusword	UINT	VAR_GLO...	_Axis_B: Index: 6041 --> _Axis_A	%MW 3.2041224
	si_modes_of_operation_di	SINT	VAR_GLO...	_Axis_B: Index: 6061 --> _Axis_A	%MB 3.2041226
⊞	**LocalAxisVariables**				

Figure 63: ProProg wt III - Global variable worksheet with the ProMaster data

The following, in particular, must be heeded:

The device name of the device on the EtherCAT bus in ProMaster is also the name of the axis in the IEC project in ProProg wt III. This means that if you have linked an existing application (or template) from ProProg wt III with the ProMaster project, the names of the devices in ProMaster may not match those in the IEC project. In this case, there are two possible solutions:

1 ProMaster changes the axis names to the device names in ProMaster in the "MC_AxisVariables" section of the "Global_Variables" variable worksheet in ProProg wt III, and the user changes the axis names in the POEs to the device names in ProMaster (via the ProProg wt III function "Global Replace").

2 You change the device names in ProMaster to the axis names in ProProg wt III.

In our example, we have given the local axis the device name _Axis_A and the EtherCAT slave the device name _Axis_B in ProMaster. These are also the axis names from our motion control template used for our IEC project.

This also applies for the cam data set name in the IEC project in ProProg wt III. This means that if you link an existing application (or template) from ProProg wt III with the ProMaster project, the name of the cam data set in ProMaster may not match that in the IEC project. In this case, the following applies:

ProMaster changes the cam data set names to the cam data set names in ProMaster in the "MC_CamDataSet" section of the "Global_Variables" variable worksheet in ProProg wt III, and the user changes the cam data set names in the POEs to the cam data set names in ProMaster (via the ProProg wt III function "Global Replace").

4.3.9.2 Exchanging data

A global variable is required in each case for exchanging data between

- the function blocks in the IEC project on the b maXX drive PLC (BM4-O-PLC-0x) and the local axis and
- the function blocks in the IEC project on the b maXX drive PLC (BM4-O-PLC-0x), the EtherCAT master module (BMC-M-ETH-02) and (via the EtherCAT field bus) the EtherCAT slave.

These variables are also called axis variables and have an axis name in the IEC project.

Our axis variable for the local axis has the axis name _Axis_A.

Our axis variable for the EtherCAT slave has the axis name _Axis_B.

We have given our local axis the device name _Axis_A and our EtherCAT slave the (same-sounding) name _Axis_B in ProMaster, which is why we do not need to make any adjustments here.

The same applies for the variable for the cam data, which has the cam data set name _SetA_512 in the IEC project and ProMaster.

The axis variable and the cam data variable are connected to the motion control function blocks in the IEC project. The machine functions are programmed with the motion control function blocks.

Information on using the motion control function blocks is found in the motion control application manual.

If motion control is used, communication occurs via the axis variable. ProMaster creates network variables in the IEC project. These are divided into standard network variables for motion control and, if additional objects were created during configuration of the EtherCAT slave, additional network variables for EtherCAT.

The following must be heeded:

The standard network variables for motion control may be read, but not written, by the user.

They are the standard motion control network variables for the set values:

"u_controlword.."	for the control word of the axis
"ud_PosIpSetAngel.."	for the synchronization position set angle of the axis

They are the standard motion control network variables for the actual values:

"u_statusword.."	for the status word of the axis
"ud_Enc1ActAngle.."	for the synchronization position actual angle of the axis
"si_modes_of_operation_display.."	for the operating mode of the axis

The additional network variables for EtherCAT are written (set values) and read (actual values) by the user.

If you created the additional network variables (and their link to EtherCAT slaves) that were suggested when configuring EtherCAT slave communication in Sections ▹PD Receive tab◃ from Page 64 and ▹PD Transmit tab◃ from Page 66, the additional network variable for the set values are:

"di_ApplicationParam1.." for application parameter 1 of the axis

The additional network variable for the actual value is:

"di_SpeedActValue.." for the actual speed value of the axis

NOTE

The standard network variables for motion control may be read, but not written, by the user.

The additional network variables for EtherCAT are written (set values) and read (actual values) by the user.

NOTE

If motion control is used, the network is started automatically via motion control.

NOTE

The additional network variables for set values must be written in the motion control event task.

The additional network variables for set values must be written at every call of the motion control event task.

APPENDIX A - ABBREVIATIONS

API	Applications Programming Interface
ARP	Address Resolution Protocol
BACI	Baumüller Component Interface
BUB	Ballast unit
BUC	Baumüller feed/return feed unit
BUG	Baumüller converter basic feed unit
BUM	Baumüller individual power unit
BUS	Baumüller power module
CAL	CAN Application Layer
CAN	Controller Area Network
CiA	CAN in Automation
COB	Communication Object
COB-ID	Communication Object Identifier
CSMA/CD	Carrier Sense Multiple Access / Collision Detection
CSMA/CA	Carrier Sense Multiple Access / Collision Avoidance
CPU	Central Processing Unit
DC	d.c. current
DCF	Device Configuration File
DHCP	Dynamic Host Configuration Protocol
DIN	Deutsches Institut für Normung e.V. (German Standards Institute)
DP-RAM	Dual-port RAM
DR	Draft Recommendation
DS	Draft Standard
DSP	Draft Standard Proposal
EDS	Electronic Data Sheet
EMC	Electromagnetic compatibility
EN	European standard
EPROM	Erasable Programmable Read Only Memory
ESD	Electrostatic Sensitive Device
FTP	File Transfer Protocol
HD	Hamming distance
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
I/O	Input/Output
ICMP	Internet Control Message Protocol
IP	Internet Protocol
IRP	Interrupt
ISO	International Organization for Standardization
LAN	Local Area Network
LED	Light Emitting Diode
LSS	Layer Setting Services
MAC	Media Access Control
OSI	Open Systems Interconnect
PDD	Process Data Directory
PDO	Process Data Object
PLC	Process Loop Controller (PLC)
RAM	Random Access Memory
SAP	Service Access Point
SDO	Service Data Object
SMS	Short Message System
SMTP	Simple Mail Transfer Protocol
PLC	Programmable Logic Controller
SRD	SDO Requesting Device
SRDO	Safety Relevant Data Object

TCP	Transport Control Protocol
Telnet	Terminal over Network
UDP	User Datagram Protocol
URL	Uniform Resource Locator
USS	USS protocol function module
USS®	Trademark of Siemens, universal serial interface
VDE	Verband deutscher Elektrotechniker (German Association of Electrical Engineers)
WWW	World Wide Web
16#	Prefix for hexadecimal numbers

Index

be in motion

Baumüller Nürnberg GmbH Ostendstraße 80-90 90482 Nuremberg Tel: +49(0)911-5432-0 Fax: +49(0)911-5432-130 **www.baumueller.de**