Bonfiglioli Vectron

AXIA VERT

Communication Module Manual
CMA-IE-01-EIP for Ethernet/IP™

Frequency inverter 230 V / 400 V
0,25 kW ... 15 kW

Bonfiglioli

CONTENTS

1 General Information about the Documentation

For better clarity, the documentation of the frequency inverter is structured according to the customer-specific requirements.

The present manual was created in the German language. The German manual is the original version. Other language versions are translated.

Quick Start Guide
The "Quick Start Guide" describes the basic steps required for mechanical and electrical installation of the frequency inverter. The guided commissioning supports you in the selection of necessary objects and the configuration of the software of the frequency inverter.

User manual
The user manual documents the complete functionality of the frequency inverter. The objects required for special purposes, for adjustment to the application and the numerous additional functions are described in detail.

Separate user manuals are supplied for optional components for the frequency inverter. These manuals complement the operating instructions and the "Quick Start Guide" for the frequency inverter.

Application manual
The application manual complements the documentation to ensure goal-directed installation and commissioning of the frequency inverter. Information on various topics in connection with the use of the frequency inverter is described in context with the specific application.

Installation instructions
The installation manual describes the installation and use of devices, complementing the "Quick Start Guide" and the user manual.

1.1 This document

The present user manual for the CMA-IE-01-EIP communication module complements the Operating Instructions and the "Quick Start Guide" for the frequency inverters of the AXIA device series.

The user manual contains important information on the installation and use of the communication module CMA-IE-01-EIP in its specified application range. Compliance with user documentation contributes to avoiding risks, minimizing repair cost and downtimes and increasing the reliability and service live of the frequency inverter.

For this reason, make sure you read the user manual carefully.

IMPORTANT:
Compliance with the documentation is required to ensure safe operation of the frequency inverter. BONFIGLIOLI Deutschland GmbH shall not be held liable for any damage caused by any non-compliance with the documentation.

In case any problems occur which are not covered by the documentation sufficiently, please contact the manufacturer.

For safe commissioning and operation of the AXIA series, the following documentation must be complied with:

- The Operating Instructions Document
- Safety manual "Functional Safety Manual"

1.2 Warranty and liability

BONFIGLIOLI Deutschland GmbH (hereinafter referred to as "manufacturer") notes that the contents of this Operating Instructions document do not form part of any previous or existing agreement, assurance or legal relationship between the manufacturer and the user of these Operating Instructions (hereinafter referred to as the "User"). Neither are they intended to supplement or replace such agreements, assurances or legal relationships. Any obligations of the manufacturer shall solely be based on the relevant purchase agreement which also includes the complete and solely valid warranty stipulations. These contractual warranty provisions are neither extended nor limited by the specifications contained in this documentation.

The manufacturer reserves the right to correct or amend the specifications, product information and omissions in these operating instructions without prior notice. The manufacturer assumes no responsibility to update these Operating Instructions. The manufacturer shall not be liable for any damage, injuries or costs which may be caused by the aforementioned reasons.

In addition, the manufacturer excludes any warranty and disclaims all liability, including without limitation direct, indirect, special, punitive, incidental, exemplary or consequential damages arising out of or in connection with one or more of the following causes:

- inappropriate use of the frequency inverter,
- non-compliance with the instructions, warnings and prohibitions contained in the documentation,
- unauthorized modifications of the frequency inverter,
- insufficient monitoring of parts of the machine/plant which are subject to wear,
- repair work at the machine/plant not carried out properly or in time,
- catastrophes by external impact and Force Majeure.

1.3 Obligation

These Operating Instructions must be read before commissioning and complied with. Anybody entrusted with tasks in connection with the

- transport,
- assembly,
- installation of the frequency inverter and
- operation of the frequency inverter

must have read and understood the Operating Instructions and, in particular, the safety instructions in order to prevent personal and material losses.

1.4 Copyright

In accordance with applicable law any copyrights relating to this document shall remain with

Bonfiglioli Deutschland GmbH
Europark Fichtenhain B6
47807 Krefeld
Germany

This document is intended for the operator of the frequency inverter. Any disclosure or copying of this document, exploitation and communication of its contents (as hardcopy or electronically) shall be forbidden, unless permitted expressly.

Any non-compliance will constitute an offense against the copyright law, the law against unfair competition and the German Civil Code and may result in claims for damages. All rights relating to patent, utility model or design registration reserved.

1.5 Storage

The documentation forms an integral part of the frequency inverter. It must be stored such that it is accessible to operating staff at all times. If the frequency inverter is sold on to other users, then the documentation must also be handed over.

2 General safety instructions and information on use

The chapter "General safety instructions and information on use" contains general safety instructions for the Operator and the Operating Staff. At the beginning of certain main chapters, some safety instructions are included which apply to all work described in the relevant chapter. Special work-specific safety instructions are provided before each safety-relevant work step.

2.1 Terminology

According to the documentation, different activities must be performed by certain persons with certain qualifications.

The groups of persons with the required qualification are defined as follows:

Operator

This is the entrepreneur/company who/which operates the frequency inverter and uses it as per the specifications or has it operated by qualified and instructed staff.

Operating staff

The term Operating Staff covers persons instructed by the Operator of the frequency inverter and assigned the task of operating the frequency inverter.

Skilled Personnel The term **Skilled Personnel** covers staff that are assigned special tasks by the Operator of the frequency inverter, e.g. installation, maintenance and service/repair and troubleshooting. Based on their qualification and/or know-how, **Skilled Personnel** must be capable of identifying defects and assessing functions.

Qualified electrician

The term Qualified Electrician covers qualified and trained staff that has special technical know-how and experience with electrical installations. In addition, Qualified Electricians must be familiar with the applicable standards and regulations, they must be able to assess the assigned tasks properly and identify and eliminate potential hazards.

Instructed person

The term Instructed Person covers staff that was instructed and trained about/in the assigned tasks and the potential hazards that might result from inappropriate behavior. In addition, instructed persons must have been instructed in the required protection provisions, protective measures, the applicable directives, accident prevention regulations as well as the operating conditions and verified their qualification.

Expert

The term Expert covers qualified and trained staff that has special technical know-how and experience relating to the frequency inverter. Experts must be familiar with the applicable government work safety directives, accident prevention regulations, guidelines and generally accepted rules of technology in order to assess the operationally safe condition of the frequency inverter.

2.2 Designated use

The frequency inverter is designed according to the state of the art and recognized safety regulations.

The frequency inverters are electrical drive components intended for installation in industrial plants or machines. Commissioning and start of operation is not allowed until it has been verified that the machine meets the requirements of the EC Machinery Directive 2006/42/EC and DIN EN 60204-1.

The frequency inverters meet the requirements of the low voltage directive 2014/35/EU and DIN EN 61800-5-1. CE-labelling is based on these standards. Responsibility for compliance with the EMC Directive 2014/30/EU lies with the operator. Frequency inverters are only available at specialized dealers and are exclusively intended for commercial use as per EN 61000-3-2.

No capacitive loads may be connected to the frequency inverter.

The technical data, connection specifications and information on ambient conditions are indicated on the rating plate and in the documentation and must be complied with in any case.

2.3 Misuse

Any use other than that described in "Designated use" shall not be permissible and shall be considered as misuse.

For, example, the machine/plant must not be operated

- by uninstructed staff,
- while it is not in perfect condition,
- without protection enclosure (e.g. covers),
- without safety equipment or with safety equipment deactivated,
- when general requirements, such as operating conditions and technical data, are not met.

The manufacturer shall not be held liable for any damage resulting from such misuse. The sole risk shall be borne by the operator.

Explosion protection

The frequency inverter is an IP 20 ingress protection rating device. For this reason, use of the device in explosive atmospheres is not permitted.

2.4 Residual risks

Residual risks are special hazards involved in handling of the frequency inverter which cannot be eliminated despite the safety-compliant design of the device. Residual risks are not obviously identifiable and can be a potential source of injury or a health hazard.

Typical residual hazards include:

- Electrical hazard
- Danger of contact with energized components due to a defect, opened covers or enclosures or improper working on electrical equipment.
- Danger of contact with energized components in frequency inverter if no external disconnection device was installed by the operator.

During operation, all covers must be installed correctly, and all electrical cabinet doors must be closed to minimize electrical hazards.

When LEDs and other indicating elements on the frequency inverter go out, this does not necessarily mean that the device is deenergized. Before carrying out any Work at the device where contact with energized parts might be possible, it must be checked in any case, i.e. irrespective of the status of any indicating elements that may be installed, if the device is deenergized.

2.5 Safety and warning signs on the frequency inverter

- Comply with all safety instructions and danger information provided on the frequency inverter.
- Safety information and warnings on the frequency inverter must not be removed.

2.6 Warning information and symbols used in the user manual

2.6.1 Hazard classes

The following hazard identifications and symbols are used to mark particularly important information:

DANGER

Identification of immediate threat holding a **high** risk of death or serious injury if not avoided.

WARNING

Identification of immediate threat holding a **medium** risk of death or serious injury if not avoided.

CAUTION

Identification of immediate threat holding a **low** risk of minor or moderate physical injury if not avoided.

NOTICE

Identification of a threat holding a risk of material damage if not avoided.

2.6.2 Hazard symbols

Symbol	Meaning	Symbol	Meaning
	General hazard		Suspended load
	Electrical voltage		Hot surfaces

2.6.3 Prohibition signs

Symbol	Meaning
	No switching; it is forbidden to switch the machine/plant, assembly on

2.6.4 Personal safety equipment

Symbol	Meaning
	Wear body protection

2.6.5 Recycling

Symbol	Meaning
	Recycling, to avoid waste, collect all materials for reuse

2.6.6 Grounding symbol

Symbol	Meaning
	Ground connection

2.6.7 ESD symbol

Symbol	Meaning
	ESD: Electrostatic Discharge (can damage components and assemblies)

2.6.8 Information signs

Symbol	Meaning
	Tips and information making using the frequency inverter easier.

2.6.9 Font style in documentation

Example	Font style	Use
0x1234	bold	Representation of object numbers
/01	bold, underlined	Representation of sub-index numbers
/d01	bold, underlined	Representation of data set numbers
Object	Italic, Font Times New Roman	Representation of object names
P.1234	bold	Representation of object numbers without name, e.g. in formulas
Q.1234	bold	Representation of source numbers

2.7 Directives and guidelines to be adhered to by the operator

The operator must follow the following directives and regulations:

- Ensure that the applicable workplace-related accident prevention regulations as well as other applicable national regulation are accessible to the staff.
- An authorized person must ensure, before using the frequency inverter, that the device is used in compliance with its designated use and that all safety requirements are met.
- Additionally, comply with the applicable laws, regulations and directives of the country in which the frequency inverter is used.
- Any additional guidelines and directives that may be required additionally shall be defined by the operator of the machine/plant considering the operating environment.

2.8 Operator's general plant documentation

- In addition to the user manual, the operator should issue separate internal operating instructions for the frequency inverter. The Operating Instructions of the frequency inverter must be included in the user manual of the whole plant.

2.9 Operator's/operating staff's responsibilities

2.9.1 Selection and qualification of staff

- Any work on the frequency inverter may only be carried out by qualified technical staff. The staff must not be under the influence of any drugs. Note the minimum age required by law. Define the staff's responsibility in connection with all work on the frequency inverter clearly.
- Work on the electrical components may only be performed by a qualified electrician according to the applicable rules of electrical engineering.
- The operating staff must be trained for the relevant work to be performed.

2.9.2 General work safety

- In addition to the user manual of the machine/plant, any applicable legal or other regulations relating to accident prevention and environmental protection must be complied with. The staff must be instructed accordingly.
  Such regulations and/or requirements may include, for example, handling of hazardous media and materials or provision/use of personal protective equipment.
- In addition to this user manual, issue any additional directives that may be required to meet specific operating requirements, including supervision and reporting requirements, e.g. directives relating to work organization, workflow and employed staff.
- Unless approved of expressly by the manufacturer, do not modify the frequency inverter in any way, including addition of attachments or retrofits.
- Only use the frequency inverter if the rated connection and setup values specified by the manufacturer are met.
- Provide appropriate tools as may be required for performing all work on the frequency inverter properly.

2.10 Organizational measures

2.10.1 General

- Train your staff in the handling and use of the frequency inverter and the machine/plant as well as the risks involved.
- Use of any individual parts or components of the frequency inverter in other parts of the operator's machine/plant is prohibited.
- Optional components for the frequency inverter must be used in accordance with their designated use and in compliance with the relevant documentation.

2.10.2 Use in combination with third-party products

- Please note that BONFIGLIOLI GmbH will not accept any responsibility for compatibility with third-party products (e.g. motors, cables or filters).
- In order to enable optimum system compatibility, BONFIGLIOLI GmbH office components facilitating commissioning and providing optimum synchronization of the machine/plant parts in operation.
- If you use the frequency inverter in combination with third-party products, you do this at your own risk.

2.10.3 Transport and storage

- The frequency inverters must be transported and stored in an appropriate way. During transport and storage the devices must remain in their original packaging.
- The units may only be stored in dry rooms which are protected against dust and moisture and are exposed to small temperature deviations only. The requirements of DIN EN 60721-3-1 for storage, DIN EN 60721-3-2 for transport and labeling on the packaging must be met.
- The duration of storage without connection to the permissible nominal voltage may not exceed one year.

2.10.4 Handling and installation

- Do not commission any damaged or destroyed components.
- Prevent any mechanical overloading of the frequency inverter. Do not bend any components and never change the isolation distances.
- Do not touch any electronic construction elements and contacts. The frequency inverter is equipped with components which are sensitive to electrostatic energy and can be damaged if handled improperly. Any use of damaged or destroyed components will endanger the machine/plant safety and shall be considered as a non-compliance with the applicable standards.
- Only install the frequency inverter in a suitable operating environment. The frequency inverter is exclusively designed for installation in industrial environments.
- If seals are removed from the case, this can result in the warranty becoming null and void.

2.10.5 Electrical connections

- The five safety rules must be complied with.
- Never touch live terminals. The DC link may have dangerous voltage levels even up to three minutes after shutdown.
- When performing any work on/with the frequency inverter, always comply with the applicable national and international regulations/laws on work on electrical equipment/plants of the country in which the frequency inverter is used.
- The cables connected to the frequency inverters may not be subjected to high-voltage insulation tests unless appropriate circuitry measures are taken before.
- Only connect the frequency inverter to suitable supply mains.

The five safety rules

When working on/in electrical plants, always follow the five safety rules:

1 Isolate

2 Secure to prevent restarting

3 Check isolation

4 Earth and short-circuit,

5 Cover or shield neighboring live parts.

2.10.6 Safe operation

- During operation of the frequency inverter, always comply with the applicable national and international regulations/laws on work on electrical equipment/plants.
- Before commissioning and the start of the operation, make sure to fix all covers and check the terminals. Check the additional monitoring and protective devices according to the applicable national and international safety directives.
- During operation, never open the machine/plant
- Do not connect/disconnect any components/equipment during operation.

- The machine/plant holds high voltage levels during operation, is equipped with rotating parts (fan) and has hot surfaces. Any unauthorized removal of covers, improper use, wrong installation or operation may result in serious injuries or material damage.
- Some components, e.g. the heat sink or brake resistor, may be hot even some time after the machine/plant was shut down. Don't touch any surfaces directly after shutdown. Wear safety gloves where necessary.
- The frequency inverter may hold dangerous voltage levels until the capacitor in the DC link is discharged. Wait for at least 3 minutes after shutdown before starting electrical or mechanical work on the frequency inverter. Even after this waiting time, make sure that the equipment is deenergized in accordance with the safety rules before starting the work.
- In order to avoid accidents or damage, only qualified staff and electricians may carry out the work such as installation, commissioning or setup.
- In the case of a defect of terminals and/or cables, immediately disconnect the frequency inverter from mains supply.
- Persons not familiar with the operation of frequency inverters must not have access to the frequency inverter. Do not bypass nor decommission any protective facilities.
- The frequency inverter may be connected to power supply every 60 s. This must be considered when operating a mains contactor in jog operation mode. For commissioning or after an emergency stop, a non-recurrent, direct restart is permissible.
- After a failure and restoration of the power supply, the motor may start unexpectedly if the Auto Start function is activated.
  If staff are endangered, a restart of the motor must be prevented by means of external circuitry.
- Before commissioning and the start of the operation, make sure to fix all covers and check the terminals. Check the additional monitoring and protective devices according to EN 60204 and applicable the safety directives (e.g. Working Machines Act or Accident Prevention Directives).

2.10.7 Maintenance and service/troubleshooting

- Visually inspect the frequency inverter when carrying out the required maintenance work and inspections at the machine/plant.
- Perform the maintenance work and inspections prescribed for the machine carefully, including the specifications on parts/equipment replacement.
- Work on the electrical components may only be performed by a qualified electrician according to the applicable rules of electrical engineering. Only use original spare parts.
- Unauthorized opening and improper interventions in the machine/plant can lead to personal injury or material damage. Repairs on the frequency inverters may only be carried out by the manufacturer or persons authorized by the manufacturer. Check protective equipment regularly.
- Before performing any maintenance work, the machine/plant must be disconnected from mains supply and secured against restarting. The five safety rules must be complied with.

2.10.8 Final decommissioning

Unless separate return or disposal agreements were made, recycle the disassembled frequency inverter components:

- Scrap metal materials
- Recycle plastic elements
- Sort and dispose of other component materials

Electric scrap, electronic components, lubricants and other utility materials must be treated as special waste and may only be disposed of by specialized companies.

Always comply with any applicable national disposal regulations as regards environmentally compatible disposal of the frequency inverter. For more details, contact the competent local authorities.

After the end of product service life, the user/operator must take the device out of operation.

For more information about the decommissioning of the device refer to the applicable operating instructions document.

Disposal requirements under European Union WEEE regulations

The product is marked with the WEEE symbol shown below.

This product cannot be disposed as general household waste. Users responsible for the final disposal must make sure that it is carried out in accordance with the European Directive 2012/19/EU, where required, as well as the relative national transposition rules. Fulfil disposal also in according with any other legislation in force in the country.

3 Introduction

This document describes the possibilities and properties of the EtherNet/IP™ communication module for the frequency inverters of the *AXIA* series of devices. It shows how to connect the hardware and describes the relevant available objects.

The module CMA-IE-01-EIP is a separate component and must be fitted by the user, 4. For the network connection 5.

This manual only describes the EtherNet/IP™ communication module. This manual is not to be understood as providing general/basic information on EtherNet/IP™ interfaces or frequency inverters.

Basic knowledge of the methods and function of EtherNet/IP™ interfaces and EtherNet/IP™ protocol are required in order to understand and implement the instructions contained in this document.

NOTICE

With the communication module, it is possible to access ALL frequency inverter objects via a controller. There are access restrictions via the user management within the software. Changing objects, the functions of which are not known to the user, can result in unintended movements and material and/or personal losses as well as inoperativeness of the frequency inverter.

Be cautious, if the function of a particular object is not fully clear to you.

The EtherNet/IP™ implementation is 100% compatible with the Common Industrial Protocol (CIP) specification.

Ethernet properties:

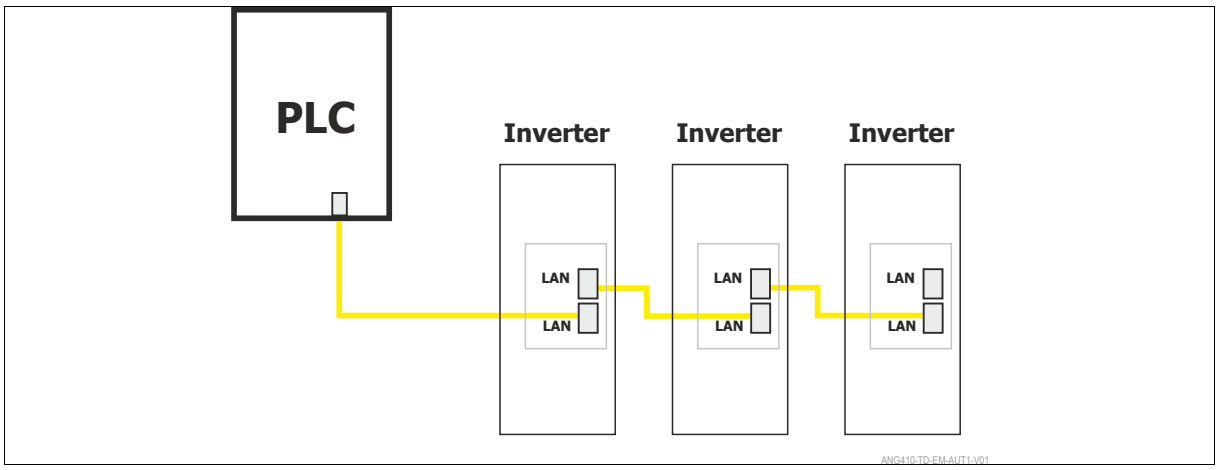
- 10/100 MB (10Base-T/100Base-T)
- Automatic identification (Auto negotiation)

Ports:

The module supports two logic TCP/IP ports for Ethernet communication.

3.1 Connection of Unit

The EtherNet/IP™ module is connected to PLC or switch using RJ45 connectors (LAN) with standard twisted-pair Ethernet cables.



EtherNet/IP™ topology

The EtherNet/IP™ module supports switch functionality for easy daisy-chaining of inverters.

The transmission speed of the communication module CMA-IE-01-EIP is 10/100 MB and the maximum cable length between two nodes is 100 m.

4 Installation/Disassembly of the communication module

The CMA-IE-01-EIP communication module is delivered in a separate case ready for assembly.

4.1 Installation

CAUTION

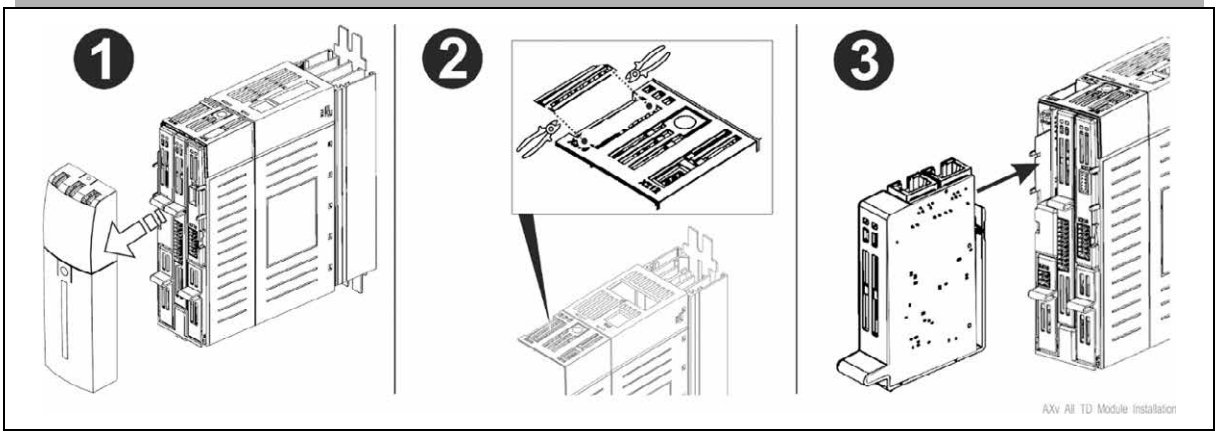
Destroying inverter and/or communication module

Connecting/disconnecting the module while the inverter is connected to live voltage can destroy the module and/or the inverter.

- Disconnect the frequency inverter from the power supply before installation of the communication module. Assembly under live voltage is not permissible.
- Do not touch the PCB visible on the back of the module, otherwise components may be damaged.

Work steps:

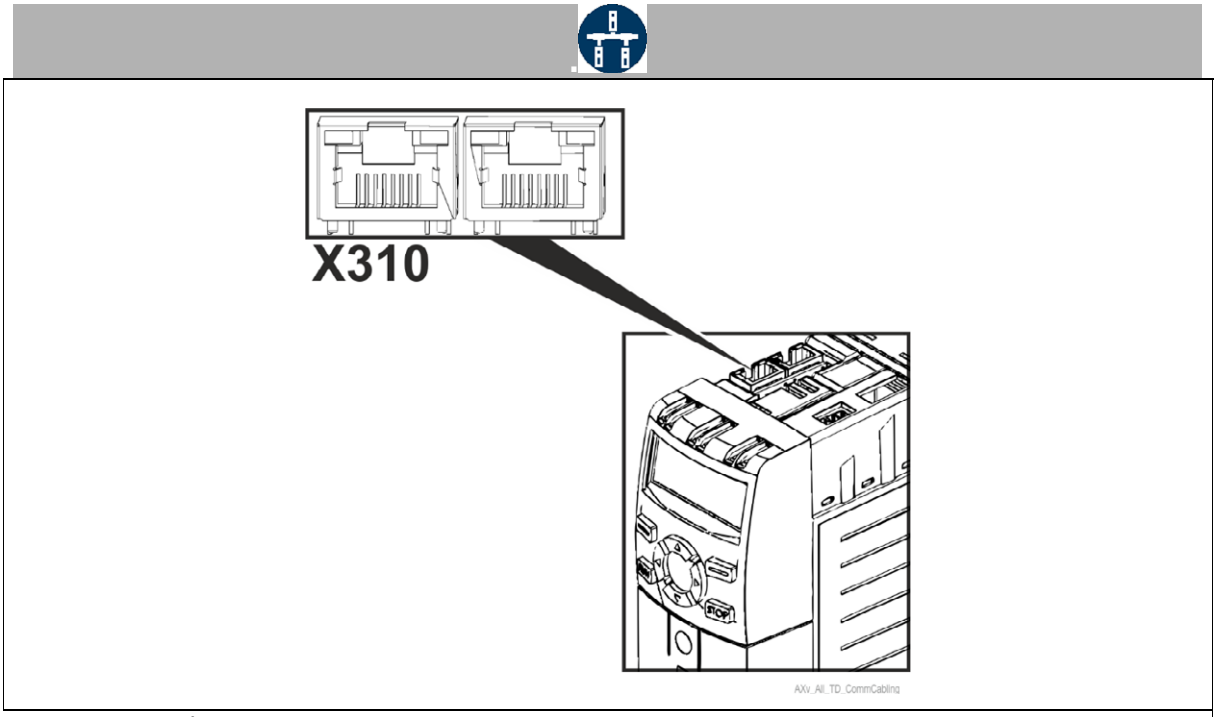
- Disconnect the frequency inverter from the mains voltage and protect it against being energized unintentionally.
- Disconnect the frequency inverter from the external 24 V if used and protect it against being energized unintentionally.



1. Remove covers of the frequency inverter. The upper left slot for the communication module is now accessible.
2. In the upper cover, break out the pre-punched cut-out for the interface X310, if necessary.
3. Insert the communication module into the slot until it engages audibly.

4.1.1 Connector assignment

The CMA-IE-01-EIP module is connected to the PLC or switch using RJ45 connectors (LAN) with standard twisted-pair Ethernet cables.



X310	Ethernet, RJ45

The interface is electrically isolated from the inverter.

The CMA-IE-01-EIP module is connected via the Ethernet connector socket **X310**.

Technical characteristics: EtherNet/IP™ interface X310

- 2 RJ45 connectors
- Ethernet standard: IEEE 802.3, 100Base-TX (fast Ethernet)
- Cable type: S/FTP
  - cable with braided shield
  - ISO/IEC 11801 or EN 50173, Straight Through or Cross Over
- The cable length is restricted by the BAUD rate, cables must not exceed a length of 100 m.

4.1.2 Status LEDs

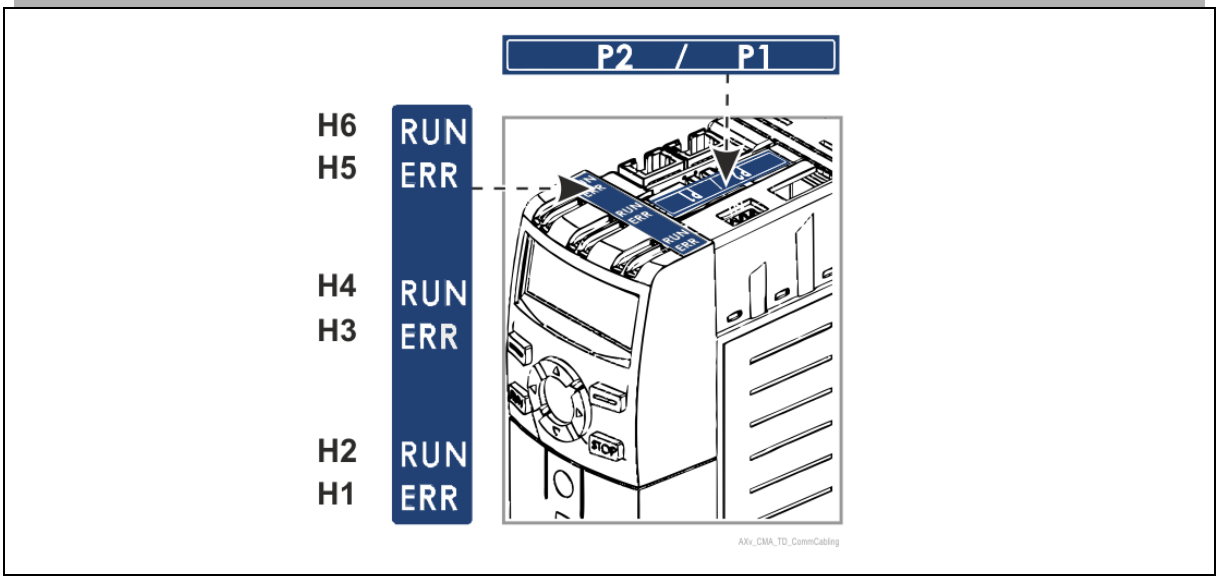
NOTICE

Residual risk

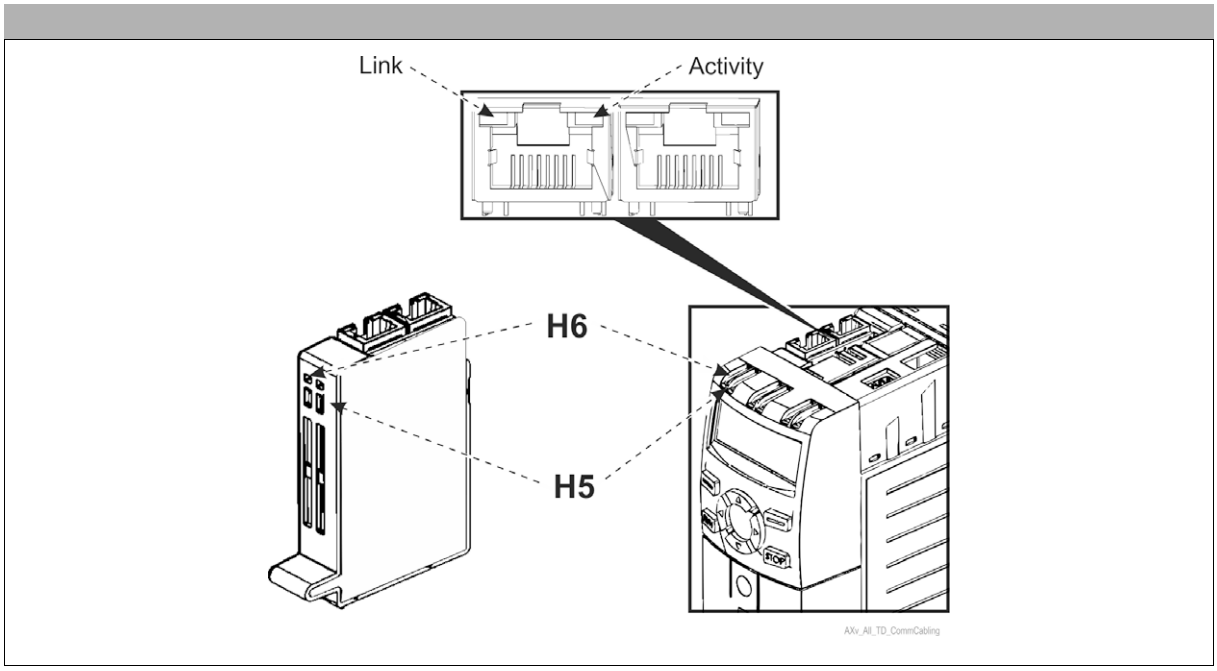
When LEDs and/or other indicating elements on the frequency inverter are not active, the inverter still may be energized.

- Before carrying out any work with the device, where contact with energized parts might be possible, always check if the device is deenergized, irrespective of the status of any indicating elements.

- Make sure the adhesive label has been placed correctly according to CEMA leaflet included in the scope of supply.



Status output via communication module LEDs:
The front LEDs (H5, H6) indicate the current status of the corresponding port of the communication module.



Link/Activity	RJ45 connector LED indicators
H5	Network status LED
H6	Module LED

Link/Activity: RJ45 connector LED indicators

The LEDs in the RJ45 connector indicate data activity (green) and the link (yellow) status of the corresponding port of the communication module.

Link/Activity LED	
LED State	**Description**
Off	No link, no activity
Green	Link (100 Mbit/s) established
Green, flickering	Activity (100 Mbit/s)
Yellow	Link (10 Mbit/s) established
Yellow, flickering	Activity (10 Mbit/s)

H5: Network status LED (RUN indicator)

This LED indicates the current status of the network connection.

Network Status LED	
LED State	**Description**
Off	No power or no IP address.
Green	Online, one or more connections established (CIP class 1 or 3).
Green, flashing	Online, no connections established.
Red	Duplicate IP address, FATAL error.
Red, flashing	One or more connections timed out (CIP class 1 or 3).

H6 Module LED (Status indicator)

The module LED indicates the current status of the module.

Module Status LED	
LED State	**Description**
Off	No power.
Green	Controlled by a Scanner in Run state and, if CIP Sync is enabled, time is synchronized to a Grandmaster clock.
Green, flashing	Not configured, Scanner in Idle state, or, if CIP Sync is enabled, time is not synchronized to a Grandmaster clock.
Red	Major fault (EXCEPTION-state, FATAL error.
Red, flashing	Recoverable fault(s). Module is configured, but stored parameters differ from currently used parameters.

4.2 Disassembly

- Disconnect the frequency inverter from mains voltage and external 24 V and protect it against being energized unintentionally.
- Remove covers of the frequency inverter, if necessary.
- Disconnect/unplug any connected lines.
- Unplug the communication module from its slot by unlocking the locking hooks on the right- and left-hand side of the module from the case of the frequency inverter using a small screwdriver.
- Proceed carefully to not damage the module.

5 Initial Settings

The IP-Address and other network objects can be set in the AxiaManager GUI (Menu> Configure Communication, 📖 AxiaManager GUI manual) or with the Axia TCP Configurator, 📄 5.3.1).

5.1 Setting the Fieldbus System to Ethernet/IP™

The communication module CMA-IE-01 can be used for various fieldbus systems with RJ45 connectors. The fieldbus system must be selected at the first start. Use the following object for this purpose:

Index	Sub-index	Description	Object code	Data type
0x3904	17	Change Fieldbus	Record	UInt32

These values are available:

Change Fieldbus **0x3904/17**	
0x00000001	EtherCAT®
0x00000002	PROFINET® I/O
0x00000003	Ethernet/IP™
0x00000004	Powerlink

- Set the fieldbus system to **0x0000003** `Ethernet/IP`.

5.2 TCP/IP Settings

In order to establish connection with a frequency inverter featuring the CMA-IE-01 you will have to configure IP settings for this module.

For all TCP/IP settings use the following object:

Index: 0x3903

Name: Ethernet Settings

By default, the objects of the communication module CMA-IE-01-EIP is set up as follows:

Index	Sub-index	Description	Object code	Data type
0x3903	1	IP Address (Ref)	Read/write	UInt 32
0x3903	2	Subnetmask (Ref)	Read/write	UInt 32
0x3903	3	Gateway (Ref)	Read/write	UInt 32
0x3903	6	SNTP Server (Ref)	Read/write	UInt 32
0x3903	7	DHCP Option (Ref)	Read/write	Boolean
0x3903	9	Hostname	Read/write	String (max. 20)
0x3903	10	Activate IP Settings	Read/write	UInt8
0x3903	18	TCP Config activated	Read/write	Boolean

The object settings must be adapted to the actual application.

5.2.1 Setting the TCP/IP Address and Subnet

For proper identification, each frequency inverter is assigned an IP address which must be unique in the system.

5.2.1.1 Network without DHCP Server

(DHCP: Dynamic Host Configuration Protocol.) The address is set via object *IP-Address* **0x3903/1**. In addition, the *Subnetmask* **0x3903/2** and the address of the *Gateway* **0x3903/3** must be entered properly for the local network.

Object		Settings		
No.	**Description**	**Min.**	**Max.**	**Default setting**
0x3903/1	IP Address (Ref)	0	0xFFFFFFFF	0
0x3903/2	Subnetmask (Ref)	0	0xFFFFFFFF	0
0x3903/3	Gateway	0	0xFFFFFFFF	0
0x3903/10	Activate IP Settings	0	1	0

The settings made with these objects will only become active when setting the object *Activate IP Settings* (**0x3903/10**) to `1`.

5.2.1.2 Network with DHCP Server

When a DHCP server is used, manual network configuration is not required. Set *DHCP Option* **0x3903/7** to `1 - Enabled` if you wish to use the DHCP function.

Object		Settings		
No.	**Description**	**Min.**	**Max.**	**Default setting**
0x3903/7	DHCP Option (Ref)	0	1	0

DHCP (Ref) **0x3903/7**	Function
0 - Disabled	Module must be configured manually, no DHCP server is used (**factory setting**).
1 - Enabled	The settings are made by a DHCP server.

When *Activate IP Settings* (**0x3903/10**) is enabled, the IP settings in the RAM are set to 0.

5.2.2 Setting the SNTP Server

The SNTP server is set via sub index 6.

Object		Settings		
No.	**Description**	**Min.**	**Max.**	**Default setting**
0x3903/6	SNTP Server (Ref)	0	0xFFFFFFFF	0

The SNTP server only becomes active when setting the object *Activate IP Settings* (**0x3903/10**) to `1`.

5.2.3 Setting the Hostname

The hostname is written via sub index 9. Changes become active after a restart.

Object		Settings		
No.	**Description**	**Min.**	**Max.**	**Default setting**
0x3903/9	Hostname	1	23	axia-device

5.2.4 Activate Changing the IP Settings via UDP Broadcast

(UDP: User Datagram Protocol) The object **0x3903/18** can be used to enable or disable the change of the IP settings via UDP Broadcast (Security).

Object		Settings		
No.	**Description**	**Min.**	**Max.**	**Default setting**
0x3903/18	TCP Config activated	0	1	0

5.3 Reading Actual TCP/IP Settings

For reading the actual TCP/IP settings use the following **0x3903** objects' sub-indexes:

5.3.1 Reading IP Settings

These objects are used to read the active IP settings directly from the TCP stack interface:

Index	Sub-index	Description	Object code	Data type
0x3903	11	IP Address (Actual)	Read	UInt 32
0x3903	12	Subnetmask (Actual)	Read	UInt 32
0x3903	13	Gateway (Actual)	Read	UInt 32
0x3903	14	DNS Server 1 (Actual)	Read	UInt 32
0x3903	15	DNS Server 2 (Actual)	Read	UInt 32
0x3903	16	SNTP (Actual)	Read	UInt 32

5.3.2 Certificate Busy

Object **0x3903/19** indicates that a certificate is currently being generated.

Index	Sub-index	Description	Object code	Data type
0x3903	19	Certificate Busy	Read	Boolean

5.3.3 Reading MAC addresses

Via the objects **0x3903/20**, **21**, and **22** the MAC addresses can be read out (from serial flash).

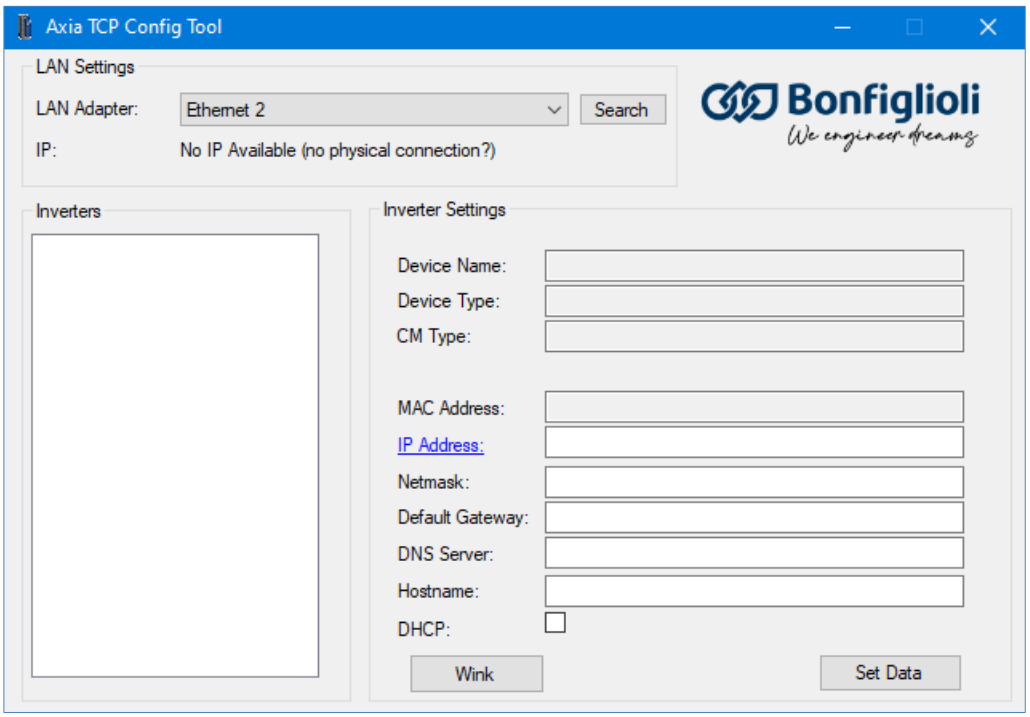
Index	Sub-index	Description	Object code	Data type
0x3903	20	MAC address	Read	Byte String (6 Bytes)
0x3903	21	MAC address Port 1	Read	Byte String (6 Bytes)
0x3903	22	MAC address Port 2	Read	Byte String (6 Bytes)

5.4 Axia TCP Configurator

The Axia TCP Configurator is a software tool for configuring IP settings for a TCP/IP connection. It can be started directly out of AxiaManager GUI.

Click on the TCP button in the AxiaManager GUI toolbar or go to "Menu Tools>Axia TCP Configurator".

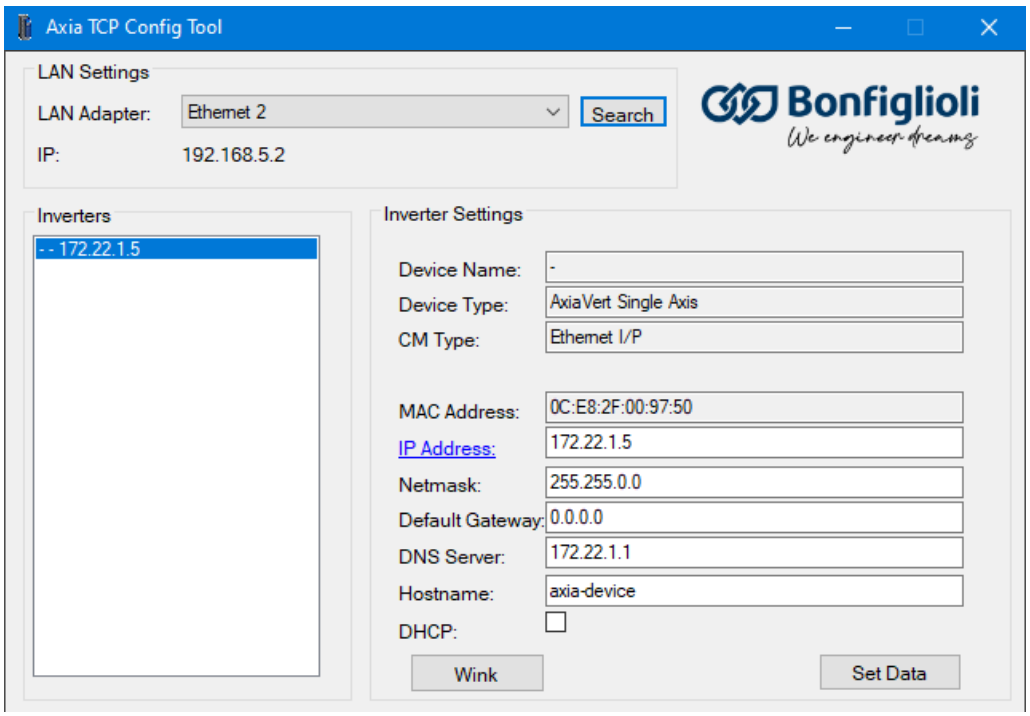
This opens the Axia TCP Config Tool where you can select the Ethernet adapter via which the frequency inverter is connected.



Select the Ethernet adapter and click on *Search* to search for available frequency inverters.

Upon completion of the search all frequency inverters found will be listed.

Click on the frequency inverter to be configured.



The right part of the window shows information on the device and input fields for various settings.

Enter the device settings required for your application. In order to apply the settings to the device, click on *Set Data*.

Before clicking on *Set Data*, make sure that you are connected with the right frequency inverter. To check this, click on the *Wink* button. The connected inverter's LEDs (green and red) will flash for 10 seconds.

5.5 OS Synchronization

The operating system (OS) of the frequency inverter can be synchronized with a PLC or other device. Synchronization of the operating system will improve the operating characteristics of the machine. Synchronization is used to eliminate CPU **phase** shifting between master and slave devices to make sure that calculations are carried out at the same time. The synchronization time must be a natural number (multiple of 1 ms for AXIAvert, multiple of 125 µs for AXIAmove/logic).

Synchronization via CANopen:
When using CANopen® without Systembus, the synchronization can be switched on and off. Synchronization can be done with CANopen® SYNC telegrams.

Synchronization via Systembus:
When using CANopen® simultaneously with Systembus, the synchronization can be set to either CANopen, Systembus or it can be switched off. Synchronization can be done with Systembus SYNC telegrams.

When synchronizing the OS via CANopen®, the master must support the synchronization mechanisms of CANopen®.

Sync Source 0x3906/14	
Operation mode	**Function**
0 - Off	The OS is not synchronized with other devices.
1 - Automatic (default)	The synchronization source is selected automatically by the frequency inverter.
2 - CM Module	The OS is synchronized via Ethernet Module.
3 - CANopen®	The OS is synchronized via CANopen®.
4 - Systembus	The OS is synchronized via Systembus.
5 - IO Module	The OS is synchronized via IO module.

Automatic mode: Selection is done based on the decision table:

CANopen® active	Systembus active	Synchronization
Yes	Yes	Synchronisation via CANopen®
Yes	No	
No	Yes	Synchronization via Systembus
No	No	No Synchronization activated

In the object **0x3906/15** *Active Sync Source* you can read the active sync source.

The CANopen "active status for synchronization" is recognized by the object setting **0x3911/1** *CANopen Node ID* >0 and a running synchronous PDO.

5.5.1 Synchronization via CANopen

Status "Synchronization via CANopen active" is identified via object setting **0x3911/1** *CANopen Node ID* and a running synchronous PDO.

For more information on the Communication Interface Configuration via CANopen, ☞📖 AXIA Operating instructions VEC2en1.

5.5.2 Synchronization via Systembus

Status "Synchronization via systembus active" is identified via object **0x3910/1** *Systembus Node ID* > `1`. In addition, object **0x3910/20** *Sysbus Nmt/Sync Master* must be set to `TRUE`. The master node then sends the sync telegram at intervals set in **0x3910/5**. If not set to `TRUE`, the node remains set to `FALSE` by default, and thus a slave node.

For more information on the Communication Interface Configuration via Systembus, ☞📖 AXIA Operating instructions VEC2en1.

6 Operational behavior on communication failure

The operating behavior in the case of failure of the bus system can be parameterized. The required behavior can be set via the object **0x6007** *Abort conn. option code* .

Index	Sub-index	Designation	Data type	Access	Map	Def.-Value
0x6007	0	Abort conn. option code	Integer16	rw	No	1

Object *Abort conn. option code* defines the operating characteristics of the frequency inverter in the case of an error in the bus connection due to BusOff, RxPDO length error or NMT state change (leaving of NMT state "Operational").

Depending on the setting of **0x2200** *Control Mode*, the response of the setting of object **0x6007** will change as shown in the following table:

Object 0x6007		Setting in object 0x2200	
Operation mode		**Function with "Control via Statemachine"**	**Function in other control modes**
0	No action	Operating point is maintained.	Operating point is maintainied.
1	Fault signal	"Fault" status will be activated immediately. **Default**.	The controller (state machine) switches to "Fault" state immediately.
2	Disable voltage command	Control command "Disable voltage" and switch to "switch on disabled" status.	
3	Quick Stop command	Control command "Quit Stop" and switch to "switch on disabled" status.	
-1	Slow Down Ramp, Fault	Control command "Disable operating" and switch to "Error" status once the drive has been shut down.	
-2	Quick Down Ramp, Fault	Control command "Quick stop" and switch to "Error" status once the drive has been shut down.	

When object **0x6007** was written and a parameter saving instruction (object **0x1010**) was generated after that, the value of **0x6007** will be saved in the non-volatile memory.

7 EtherNet/IP™ communication

The EtherNet/IP™ implementation in Bonfiglioli inverters is compatible with the ODVA/CIP EtherNet/IP™ specification.

Common Industrial Protocol (CIP™) makes a distinction between input and output assemblies. "Input" and "Output" in this context are viewed **from the perspective of the controlling element** (e.g., a PLC/PAC). An input assembly in a device collects data from the input application (e.g., field wiring terminal, proximity sensor, etc.) and produces it on the network, where it is **consumed** by the controlling device and/or operator interface. An output assembly in a device consumes data that the controlling element sends to the network and writes that data to the output application (e.g., field wiring terminals, motor speed control, etc.).

The following chapters describe the implemented classes/instances/attributes of EtherNet/IP™. For I/O data exchange, according to the CIP-defined AC drives device profile, output assemblies 20/21/22/24 and input assemblies 70/71/72/74 are available.

In addition, the vendorspecific output assembly 100 and input assembly 101 are available. These assemblies support objects defined in the Bonfiglioli CANopen implementation, including "dynamic mapping" as known from CANopen Rx/TxPDO mapping.

All of the CANopen functions are available, **with the exception of** motion controls "interpolated position mode" and "cyclic synchronous position mode".

7.1 Access to Inverter Parameters

EtherNet/IP™ supports two types of connection:

- Explicit Messaging – confirmed services for get/set values
- I/O – confirmed/unconfirmed services for R/W process data

7.1.1 Explicit Messaging

All values of classes/instances/attributes are accessible by explicit message (get/read or set/write, from the PLC's point of view).

A complete explicit message consists of the following elements:

- **Node Address**: Identification value assigned to each node on a CIP Network
- **Class ID**: Identification value assigned to each object class accessible from the network
- **Instance ID**: Identification value assigned to an object instance that identifies it among all instances of the same class.
- **Attribute ID**: Identification value assigned to a class or instance attribute.
- **Service Code**: Identification value which denotes an action request that can be directed at a particular object instance or object attribute.

Inverter parameters are addressed by object number (**0xnnnn**) and sub index or data set number.

Bonfiglioli inverter objects can be accessed via CIP Class 0x80, see chapter 7.5.9.

Refer to the object list of the respective configuration. ☞ 🕮 AXIA Operating instructions VEC2en1 to see the objects and the sub indexes that can be set. The object list states whether an object is data set change-over capable (data set = `1` to `4`) or only exists once (data set = `0`).

The object list also provides information about the display format of an object and its type (int/uint/long/string).

7.1.2 I/O – Confirmed/Unconfirmed Services

For I/O, different types of communication are available:

- Polled
- Change of state
- Cyclic

Polled
With the I/O setting "Polled", the PLC sends OUT data to the inverter. After receiving OUT data the inverter sends IN data to the PLC.

Cyclic
With the "Cyclic" setting, the PLC sends OUT data cyclically to the inverter. The inverter sends IN data cyclically to the PLC.

I/O-Connections:

- PLC - output data - Polled
- Inverter - input data - Polled, Change-of-State, Cyclic

7.2 How to create EDS Files

EtherNet/IP™ uses type-specific EDS files for each type of inverter (size, voltage, current). These EDS files are required for configuration utilities, e.g., Rockwell RSLogix 5000.

EDS files can be created in two ways:

1. Read the device using an EthernetIP_EDS Generator tool. This is possible if you are connected to an inverter by PC and USB (via Keypad) or Ethernet.
   In this case, the EDS file is created online.
2. Create the file manually using the EZ-EDS tool freely available at www.odva.org.
   In this case, the EDS file is created offline.

Irrespective of the method used, a window opens where a default filename for the EDS file is offered and a folder where to store this file. The EDS filename and storage location can be changed by the user.

If there are changes in the EDS file content due to new inverter software, version updates for EDS file templates are provided by the inverter configuration utility and the integrated software update utility (online internet access required).

The EDS file contains the following data:

VendCode:	= 1202
VendName:	= Bonfiglioli GmbH
ProdCode:	= 1 DevDat_sFCUDevDat.sGeneral.ui32DeviceID The DeviceID is explicit.
ProdTypeStr:	= AxiaVert Single Axis
Catalog:	The catalog number is entered manually in the EDS Generator.
ProdName:	= „AxiaVert Single Axis“ Comm_sOutAI.acManufDeviceName The product name is a string containing information of software, hardware, type of commissioning; see next table

ProdCode	ProdName	Line	Safety	Axis
1	AxiaVert Single Axis	„Vert“	“”	“Single”
	AxiaVert Double Axis			“Double”
	AxiaVert Safety Single Axis		“Safety“	“Single”
	AxiaVert Safety Double Axis			“Double”
	AxiaMove …	“Move”	…	…
	AxiaLogic …	“Logic”	…	…
	AxiaAgile …	“Agile”	…	…

The name of the EDS file is structured as follows:

Axia`ProdName`_EIP.eds

or: Axia`ProdCode`_EIP.eds

7.3 Supported configurations

AXIA inverters support different types of control and reference values:

- Standard (without Positioning functions)
- Positioning via contacts
- Positioning via Motion Control Interface (MCI) via field bus

To use the full functionality of the Motion Control Interface object **0x2200** *Control Mode* is set to mode `3 –State machine`.

The inverter's behavior with respect to *Control Word / Status Word* (**0x6040**/**0x6041**) and *Modes of operation / modes of operation display* (**0x6060**/**0x6061**) varies in the two different types of configurations.

7.3.1 Standard

Necessary settings:

- *Control Mode* **0x2200** = `1 - IOs`

The control (Start, Stop, Frequency change over, etc.) is typically carried out via:

- Digital contacts
- Remote contacts via Field bus

Reference values result from the selected configuration. Typical configurations:

Reference speed / Reference frequency:

- Analog input
- Fixed values from parameters
- Target velocity

Percentage reference value for technology controller or Torque control:

- Analog input
- Fixed values from parameters

7.3.2 Positioning via contacts (or remote contacts)

Necessary settings:

- *Control Mode* **0x2200**= `1 - IOs`

The control (Start, Stop, Target position change over, etc.) is carried out typically via:

- Digital contacts
- Remote contacts via Field bus

Reference values result from the selected configuration. Typical configurations:

- Reference speed / Reference frequency
- Reference target position

7.3.3 MCI (Motion Control Interface – Positioning via Field bus)

Necessary settings:

- *Control Mode* **0x2200** = `3 –Statemachine`

The control (Start, Stop, mode change over, etc.) is carried out via the Control word.

Reference values result from the selected Modes of Operation.

Typical Modes of Operation are:

- Reference speed via target velocity
- Target position

The usage of the Motion Control Interface is described in ☞📖 AXIA Operating instructions VEC2en1.

7.4 Initialization time

When the frequency inverter is turned on, the automation module must be initialized in addition to the frequency inverter. The initialization can take up to 20 seconds.

Wait until the initialization phase is complete before starting the communication. Module status LED A (☞ 4.1.2).

7.5 EtherNet/IP™ overview

7.5.1 Class 0x01, Identity Object

Instance	Attribute	Name	Service	Data Type	Value
0	1	Revision	get	UINT	2
1	1	VendorID	get	UINT	**1202** ***Bonfiglioli Deutschland GmbH***
	2	DeviceType	get	UINT	2 - AC/DC-Drive
	3	ProductCode	get	UINT	value depends on inverter type
	4	Revision	get	Struct USINT/USINT	0x01 0x03
	5	Status	get	WORD	current device state
	6	Serial Number	get	UDINT	
	7	Product Name	get	SHORT_ STRING	Axia Vert Single Axis

7.5.2 Class 0xF5, TCP/IP Object

The TCP/IP Interface Object provides the mechanism to configure a device's TCP/IP network interface.

The following Instances and Attributes are supported:

Instance	Attribute	Name	Service	Data Type	Value
0	1	Revision	get	UINT	4
1	1	Status	get	DWORD	see below
	2	Configuration Capability	get	DWORD	0x000000B4 DHCP, Configuration settable
	3	Configuration Control	get/set	DWORD	Default = 0 (fixed IP)
	4	- Physical Link Object - Path size - Path	get	STRUCT of: UINT Padded EPATH	 No. of 16 bit words in Path The path is restricted to one logical class segment and one logical instance segment. The max. size is 12 bytes
	5	Interface Configuration - IP Address - Network Mask - Gateway Address - Name Server (Primary name server) - Name Server 2 (Secondary name server) - Domain Name (Default domain name)	get/set	STRUCT of: UDINT UDINT UDINT UDINT UDINT STRING	see below
	6	Hostname	get/set	STRING	ASCII, max. 64 chars., padded to an even no. of chars. (pad not included in length). A length of 0 indicates no Host Name is configured.
1	8	TTL Value (Time-to-Live value for IP multicast packets)	get/set[1)]	USINT	see below
	9	Mcast Config - Alloc Control (Multicast address allocation control word) - Reserved - Num Mcast (No. of IP multicast addreses to allocate for EtherNet/IP™) - Mcast Start Addr(Starting multic astaddress from which to begin allocation)	get/set[1)]	STRUCTof: USINTUSINT UINT UDINT	

1) If either TTL Value or Mcast Config is implemented as settable, both must be implemented as settable.

Status

Bit(s)	Name	Value
0-3	Interface Configuration Status	0 – The Interface Configuration attribute has not been configured. 1 – The Interface Configuration attribute contains configuration obtained from BOOTP, DHCP or nonvolatile storage. 2 – The IP address member of the Interface Configuration attribute contains the configuration, obtained from hardware settings (e.g.: pushwheel, thumbwheel, etc.). 3-15 – Reserved for future use.
4	Mcast Pending	Indicates a pending configuration change in the TTL Value and/or Mcast Config attributes. This bit shall be set when either the TTL value or Mcast Config attribute is set, and shall be cleared the next time the device starts.
5	Interface Configuration Pending	Indicates a pending configuration change in the Interface Configuration attribute. This bit shall be 1 (TRUE) when Interface Configuration attribute are set and the device requires a reset in order for the configuration change to take effect (as indicated in the Configuration Capability attribute). The intent of the Interface Config Pending bit is to allow client software to detect that a device's IP configuration has changed, but will not take effect until the device is reset.
6	AcdStatus	not supported
7	AcdFault	not supported
8-31	Reserved	0

Configuration Capability

Bit(s)	Name	Value
0	BOOTP Client	1 (TRUE) = device capable of obtaining its network configuration via BOOTP.
1	DNS Client	1 (TRUE) = device capable of resolving host names by querying a DNS server.
2	DHCP Client	1 (TRUE) = device capable of obtaining its network configuration via DHCP.
3	DHCP-DNS Update	Shall be 0, behavior to be defined in a future specification edition.
4	Configuration Settable	1 (TRUE) = Interface Configuration attribute is settable.
5	Hardware Configurable	1 (TRUE) = the IP Address member of the Interface Configuration attribute can be obtained from hardware settings (e.g., pushwheel, thumbwheel, etc.). If this bit is FALSE the Status Instance Attribute (1), Interface Configuration Status field value shall never be 2 (The Interface Configuration attribute contains valid configuration, obtained from hardware settings).
6	Interface Configuration Change Requires Reset	1 (TRUE) = device requires a restart in order for a change to the Interface Configuration attribute to take effect. If this bit is FALSE a change in the Interface Configuration attribute will take effect immediately.
7	AcdCapable	1 (TRUE) = device is ACD capable
8-31	Reserved	0

Configuration Control

Bit(s)	Name	Value
0-3	Configuration Method	0 – The device shall use statically-assigned IP configuration values. 1 – The device shall obtain its configuration values via BOOTP. 2 – The device shall obtain its configuration values via DHCP. 3-15 – Reserved for future use.
4	DNS Enable	If 1 (TRUE), the device shall resolve host names by querying a DNS server.
5-31	Reserved	0

Interface Configuration

Field Name	Data Type	Value
IP Address	UDINT	A value of 0 indicates no IP address has been configured. Otherwise, the IP address shall be set to a valid Class A, B, or C address and shall not be set to the loopback address (127.0.0.1).
Network Mask	UDINT	A value of 0 indicates no Network Mask address has been configured.
Gateway Address	UDINT	A value of 0 indicates no Gateway address has been configured. Otherwise, the Gateway IP address shall be set to a valid Class A, B, or C address and shall not be set to the loopback address (127.0.0.1).
Name Server	UDINT	A value of 0 indicates no Name Server address has been configured. Otherwise, the Name Server address shall be set to a valid Class A, B, or C address.
Name Server 2	UDINT	A value of 0 indicates no Secondary Name Server address has been configured. Otherwise, the Secondary Name Server address shall be set to a valid Class A, B, or C address.
Domain Name	STRING	ASCII, max. 48 chars., padded to an even no. of chars. (pad not included in length). A length of 0 indicates no Domain Name is configured.

TTL Value

The *TTL Value* is the value a device shall use for the IP header Time-to-Live field when sending EtherNet/IP™ packets via IP multicast.

Data Type: USINT Min./Default value is 1, Max. value is 255

Unicast packets shall use the TTL as configured for the TCP/IP stack, and not the TTL Value configured in this attribute.

Mcast Config

The *Mcast Config* attribute contains the configuration of the device's IP multicast addresses to be used for EtherNet/IP™ multicast packets. There are three elements to the *Mcast Config* structure:

- Alloc Control,
- Num Mcast, and
- Mcast Start Addr.

Alloc Control

Value	Definition
0	Multicast addresses shall be generated using the default allocation specified in the CIP EtherNet/IP™ specification. When this value is specified on a set-attribute or set-attributes-all, the values of *Num Mcast* and *Mcast Start Addr* in the set-attribute request shall be 0.
1	Multicast addresses shall be allocated according to the values specified in *Num Mcast* and *Mcast Start Addr*.
2	Reserved

Num Mcast

Num Mcast is the number of IP multicast addresses allocated. The maximum number of multicast addresses is device specific, but shall not exceed the number of EtherNet/IP™ multicast connections supported by the device.

Mcast Start Addr

Mcast Start Addr is the starting multicast address from which *Num Mcast* addresses are allocated.

7.5.3 Class 0xF6, Ethernet Link Object

The *Ethernet Link* Object maintains link-specific counters and status information for an IEEE 802.3 communications interface.

The following instances and attributes are supported:

Instance	Attribute	Name	Service	Data Type	Value
0	1	Revision	get	UINT	4
1	1	Interface Speed	get	UDINT	Speed in Mbps
	2	Interface Flags (Interface status flags)	get	DWORD	Bitmap of Interface Flags, see below
	3	Physical Address (MAC layer address)	get	USINT[6]	see below

Interface Flags

Bit(s)	Name	Value
0	Link Status	0 – inactive link 1 – active link
1	Half/Full Duplex	0 – half duplex 1 – full duplex Note that if the Link Status flag is 0, then the value of this flag is indeterminate.
2-4	Negotiation Status	Indicates the status of link auto-negotiation 0 – Auto-negotiation in progress 1 – Auto-negotiation and speed detection failed. Using default values for speed and duplex. 2 – Auto negotiation failed but detected speed. Duplex was defaulted. 3 – Successfully negotiated speed and duplex. 4 – Auto-negotiation not attempted. Forced speed and duplex.
5	Manual Setting Requires Reset	0 – The interface can activate changes to link parameters (auto-negotiate, duplex mode, interface speed) automatically. 1 – The device requires a Reset service be issued to its Identity Object in order for the changes to take effect.
6	Local Hardware Fault	0 – The interface detects no local hardware fault. 1 – A local hardware fault is detected.
7-31	Reserved	0

Physical Address

Contains the interface's MAC layer address and is an array of octets.

The recommended display format is "XX-XX-XX-XX-XX-XX", starting with the first octet.

7.5.4 Class 0x04, Assembly Object

The *Assembly Object* provides explicit message access to assembly data. Assembly data can be written/read alternating with I/O data access.

The instance number defines the assembly. Assemblies 20, 21, 22, 24, 70, 71, 72 and 74 have fixed length (number of bytes). Assemblies 100/101 (vendor specific) support dynamic mapping and therefore have variable length (depending on the objects mapped).

Note that the assemblies 20-24 and 70-74 are only available for axis one.
If two axes are needed Assembly 100/101 must be used.

Instance	Attribute	Name	Service	Data Type	Value
0	1	Revision	get	UINT	2
20	3	data	set	ARRAY of BYTE	
	4	size	get	UINT	4
21	3	data	set	ARRAY of BYTE	
	4	size	get	UINT	4
22	3	data	set	ARRAY of BYTE	
	4	size	get	UINT	6
24	3	data	set	ARRAY of BYTE	
	4	size	get	UINT	6
100	3	data	set	ARRAY of BYTE	
	4	size	get	UINT	Depends on mapping, see 7.5.9
70	3	data	get	ARRAY of BYTE	
	4	size	get	UINT	4
71	3	data	get	ARRAY of BYTE	
	4	size	get	UINT	4
72	3	data	get	ARRAY of BYTE	
	4	size	get	UINT	6
74	3	data	get	ARRAY of BYTE	
	4	size	get	UINT	6
101	3	data	get	ARRAY of BYTE	
	4	size	get	UINT	Depends on mapping, see 7.5.9

7.5.5 Class 0x0F, Parameter Object

The *Parameter Object* refers to EtherNet/IP™ parameters that can be read/written. These parameters are used for different settings of the inverter.

Instance	Attribute	Name	Service	Data Type	Value
0	2	Max Instance	get	UINT	40
	8	Parameter Class Descriptor	get	WORD	0x0B Bit 0 = 1: supports parameter instances Bit 2 = 1: must do nonvolatile storage save command
	9	Configuration assembly Instances	get	UINT	0 – configuration assembly is not supported

Available Services for Class 0x0F / Instance 0 / Attribute do not care:

- *Reset*: Resets all parameters to the factory default
- *Restore*:Restores all parameter values from non-volatile memory
- *Save*: Saves all parameter values to non-volatile memory

7.5.5.1 Input Assembly Selection, Parameter 1

Instance	Attribute	Name	Service	Data Type	Value
1, 2	1	Param 1 InputAssemblySelection	get/set	UINT	70 – Assembly 70 71 – Assembly 71 72 – Assembly 72 74 – Assembly 74 101 – Assembly 101

Object **0x3952/1** *InputAssemblySelection* defines the used input assembly. The object content of the different assemblies ☞ 7.6.

Input assemblies 70/71/72/74 have fixed objects and length.

Input assembly 101 is a vendor specific assembly with variable mapping and up to 12 objects/ 24 bytes provided by CANopen definitions. For a detailed description ☞ 7.6.2.

This parameter is only settable if no I/O connection is established.

7.5.5.2 Input Assembly Length, Parameter 2

Instance	Attribute	Name	Service	Data Type	Value
2	1	Param 2 InputAssemblyLength	get	UINT	Max. length: 32*number of axes

Object **0x3952/2** "*InputAssemblyLength*" contains the length of the currently set input assembly.

7.5.5.3 Input Assembly Mapping for Vendor Specific Input Assembly 101, Parameters 3 to 14

Instance	Attribute	Name	Service	Data Type	Value
3	1	Param 3 InputObject1	get	UINT	See list
4	1	Param 4 InputObject2	get	UINT	
5	1	Param 5 InputObject3	get	UINT	
6	1	Param 6 InputObject4	get	UINT	
7	1	Param 7 InputObject5	get	UINT	
8	1	Param 8 InputObject6	get	UINT	
9	1	Param 9 InputObject7	get	UINT	
10	1	Param 10 InputObject8	get	UINT	
11	1	Param 11 InputObject9	get	UINT	
12	1	Param 12 InputObject10	get	UINT	
13	1	Param 13 InputObject11	get	UINT	
14	1	Param 14 InputObject12	get	UINT	

Objects **0x3952/3** *InputObject1* to **0x3952/14** *InputObject12*, define the content of vendor specific Input Assembly 101. A choice list is available for these objects.

These objects are only settable if no I/O-connection is established or the connection is not to Assembly 101.

Choice list for objects **0x3952/3** to **0x3952/14**:

No.	No.	No.	Choice Name
0x00000000	no object mapped	0x0000005F	0x6064 (4B) position actual value"
0x00000029	0x3001 (2B) Digital In actual value	0x00000064	0x606C (4B) velocity actual value
0x0000002A	0x3002 (2B) Digital Out actual value	0x0000006A	0x6077 (2B) Torque actual value
0x0000002C	0x3004 (2B) Boolean Mux	0x000000B	0x6078 (2B) Current actual value
0x0000002F	0x3007 (2B) Percentage actual value	0x0000006C	0x6079 (4B) DC link voltage
0x00000030	0x3008 (2B) Percentage actual valu2	0x0000007E	0x60F4 (4B) following error value
0x00000031	0x3011 (2B) Act. value word1 P.1415	0x00000083	0x60B9 (2B) touch probe status
0x00000032	0x3012 (2B) Act. value word2 P.1416	0x00000084	0x60BA (4B) touch probe 1 position
0x00000033	0x3021 (4B) Act. value long1 P.1417	0x00000085	0x60BB (4B) touch probe 1 position
0x00000034	0x3022 (4B) Act. value long2 P.1418	0x00000086	0x60BC (4B) touch probe 2 position
0x0000004C	0x5FF0 (1B) active motion block ttr	0x00000087	0x60BD (4B) touch probe 2 position
0x0000004D	0x5FF1 (1B) motion block to resume	0x00000088	0x60D5 (2B) touch probe 1 pos edge
0x00000051	0x6041 (2B) statusword",	0x00000089	0x60D6 (2B) touch probe 1 neg edge
0x00000053	0x6043 (2B) velocity demand	0x0000008A	0x60D7 (2B) touch probe 2 pos edge
0x00000054	0x6044 (2B) control effort	0x0000008B	0x60D8 (2B) touch probe 2 neg edge
0x0000005E	0x6061 (1B) modes of operation disp		

This selection above lists the CANopen objects which can be mapped to Input Assembly 101. The values in brackets, e.g. (2B) or (4B), define the number of bytes of the corresponding object. The maximum number of bytes for input assembly is 24.

For a detailed description 7.6.2.

7.5.5.4 Output Assembly Selection, Parameter 15

Instance	Attribute	Name	Service	Data Type	Value
15	1	Param 15 OutputAssemblySelection	get/set	UINT	20 – Assembly 20 21 – Assembly 21 22 – Assembly 22 24 – Assembly 24 100 – Assembly 100

Object **0x3952/15** "*OutputAssemblySelection*" defines the used output assembly. The object content of the different assemblies is described here: 7.6.

Output assemblies 20/21/22/24 have fixed objects and length.

Output assembly 100 is a vendor specific assembly with variable mapping and up to 12 objects/ 24 bytes. For a detailed description, 7.6.2.

These parameters are only settable if no I/O connection is established.

7.5.5.5 Output Assembly Length, Parameter 16

Instance	Attribute	Name	Service	Data Type	Value
16	1	Param 16 OutputAssemblyLength	get	UINT	Fixed length: 32*number of axes

Object **0x3952/16** "*OutputAssemblyLength*" contains the length of the currently set output assembly.

7.5.5.6 Output Assembly Mapping for Vendor specific Output Assembly 100, Parameters 17 to 28

Instance	Attribute	Name	Service	Data Type	Value
17	1	Param 17 OutputObject1	get	UINT	See list
18	1	Param 18 OutputObject2	get	UINT	
19	1	Param 19 InputObject3	get	UINT	
20	1	Param 20 OutputObject4	get	UINT	
21	1	Param 21 InputObject5	get	UINT	
22	1	Param 22 OutputObject6	get	UINT	
23	1	Param 23 OutputObject7	get	UINT	
24	1	Param 24 OutputObject8	get	UINT	
25	1	Param 25 OutputObject9	get	UINT	
26	1	Param 26 OutputObject10	get	UINT	
27	1	Param 27 OutputObject11	get	UINT	
28	1	Param 28 OutputObject12	get	UINT	

Object **3952/17** *OutputObject1* to object **3952/28** *OutputObject12* define the content of vendor specific Output Assembly 100. A choice list is available for these objects.

These parameters are only settable if no I/O connection is established, or the connection is not to Assembly 100.

Choice list for object **3952/17** to **3952/28**:

No.	Choice Name	No.	Choice Name
0x00000000	no object mapped	0x00000052	0x6042 (2B) target velocity
0x0000002B	0x3003 (1B) Digital Out set values	0x0000005D	0x6060 (1B) modes of operation
0x0000002D	0x3005 (2B) Boolean Demux	0x00000069	0x6071 (2B) Target torque
0x0000002E	0x3006 (2B) Percentage set value	0x0000006D	0x607A (4B) target position
0x00000035	0x3111 (2B) Ref. value word1 (S.762)	0x0000006F	0x6081 (4B) profile velocity
0x00000036	0x3112 (2B) Ref. value word2 (S.763)	0x00000070	0x6083 (4B) profile acceleration
0x00000037	0x3121 (4B) Ref. value long1 (S.764)	0x00000071	0x6084 (4B) profile deceleration
0x00000038	0x3122 (4B) Ref. value long2 (S.765)	0x00000072	0x6085 (4B) quick stop deceleration
0x00000039	0x5F10_1 (2B) Numerator	0x00000080	0x60FF (4B) target velocity
0x0000003A	0x5F10_2 (2B) Denominator	0x00000082	0x60B8 (4B) touch probe function
0x00000050	0x6040 (2B) controlword		

This selection defines CANopen objects to be mapped to Output Assembly 100. The values in brackets, e.g. (2B) or (4B), define the number of bytes of the corresponding object. The maximum number of bytes for output assembly is 24.

For a detailed description, ☞ 7.6.2.

7.5.5.7 Reserved Parameters 29–32

Objects **0x3952/29** to **0x3952/32** have no function and are reserved for future use.

7.5.5.8 CIP Abort Behavior, Parameter 33

Instance	Attribute	Name	Service	Data Type	Value
33	1	Param 33 CIP Abort Behavior	get	USINT	See list

Object **0x3952/33** "*CIP Abort Behavior*" defines inverter behavior in the event of bus errors.

This has the same effect as writing inverter object **0x1029** *Error Behavior*.

Choice list for Parameter **33**:

No.	Choice Name	No.	Choice Name
0x00000000	No Action	0x00000003	Quick Stop Command
0x00000001	Fault Signal	0x00000004	Slow Down Ramp, Fault
0x00000002	Disable Voltage Command	0x00000005	Quick Down Ramp, Fault

Settings `2`, `3`, `4` and `5` are only executed as described by the inverter, if object *Control mode* **0x2200** is set to "`3 –State Machine`". With all other settings of **0x2200,** the settings `2`, `3`, `4` and `5` trigger a fault reaction as in choice `1`.

7.5.5.9 CIP COS (Change of State) Mask, Parameters 34 to 39

Instance	Attribute	Name	Service	Data Type	Value
34	1	Param 34 COS Mask 0	get/set	DWORD	0x00000000 to 0xFFFFFFFF (Default = 0xFFFFFFFF)
35	1	Param 35 COS Mask 1	get/set	DWORD	
36	1	Param 36 COS Mask 2	get/set	DWORD	
37	1	Param 37 COS Mask 3	get/set	DWORD	
38	1	Param 38 COS Mask 4	get/set	DWORD	
39	1	Param 39 COS Mask 5	get/set	DWORD	

Object **0x3952/34** to 0x3952/39 have no function and are reserved for compatibility reasons with ACU410/ANG inverter series.

7.5.5.10 Reserved Parameter 40

Instance	Attribute	Name	Service	Data Type	Value
40	1	Param 40	-	-	reserved for future use

Object **0x3952/40** has no function and is reserved for future use.

7.5.6 Class 0x28, Motor Data Object

The values of the Motor Data Object attributes are read from internal inverter objects.
They are **always** read from data set 1, even if data sets **1** to **4** have different values.
This object supports Instances 1 and 2, one for each axis.

Instance	Attribute	Name	Service	Data Type	Value
1, 2	3	MotorType	get	USINT	0 - Non-standard Motor 1 - PM DC Motor 2 - FC DC Motor 3 - PM Synchronous Motor 3 - PM Synchronous Motor 4 - FC Synchronous Motor 5 - Switched Reluctance Motor 6 - Wound Rotor Induction Motor 7 - Squirrel Cage Induction Motor 8 - Stepper Motor 9 - Sinusoidal PM BL Motor 10 - Trapezoidal PM BL Motor
	6	RatedCurrent	get	UINT	xxx [100 mA]
	7	RatedVoltage	get	UINT	xxx [V]
	8	RatedPower	get	UDINT	xxx [W]
	9	RatedFrequency	get	UINT	xxx [Hz]
	12	PoleCount	get	UINT	xxx
	15	BaseSpeed	get	UINT	xxx [RPM]

The Motor Data objects are accessed by the following Axia objects:

Attribute	Assigned motor data	Axia Object	Object Name
3	MotorType	**0x2001/1…4**	*Motor Type*
	0 - Non-standard Motor		3 – Grid
	1 - PM DC Motor		
	2 - FC DC Motor		
	3 - PM Synchronous Motor		2 – PMSM
	4 - FC Synchronous Motor		
	5 - Switched Reluctance Motor		1 – SynRM
	6 - Wound Rotor Induction Mo		
	7 - Squirrel Cage Induction Motor		0 – ASM
	8 - Stepper Motor		
	9 - Sinusoidal PM BL Motor		
	10 - Trapezoidal PM BL Motor		
6	RatedCurrent [100mA]	**0x2003/1…4**	*Rated Current [A]*
7	RatedVoltage [V]	**0x2002/1…4**	*Rated Voltage [V]*
8	RatedPower [W]	**0x2008/1…4**	*Rated Mech. Power [W]*
9	RatedFreq [Hz] (elec)	**0x2007/1…4**	*Rated Frequency [Hz]*
12	PoleCount(not pair)	**0x2005/1…4**	*No. of Pole Pairs*
15	BaseSpeed [rpm]	**0x2004/1…4**	*Rated Speed [rpm]*

The CIP class 0x28 can also be accessed in AxiaManager via object **0x21F3** *CIP 28h*. The CIP instances correspond to the axes (index offset 0x0800), the CIP attributes correspond to the sub indexes.

Object	Sub-index	Object Name
0x21F3	0	*CIP 28h Motor Data*
	3	*CIP 28h Motor Type*
	6	*CIP 28h Rated Current*
	7	*CIP 28h Rated Voltage*
	8	*CIP 28h Rated Power*
	9	*CIP 28h Rated Frequency*
	12	*CIP 28h Pole Count*
	15	*CIP 28h Base Speed*

Example:

0x21F3/6 = Instance 1 (axis 1), *Rated Current*

0x29F3/15 = Instance 2 (axis 2), *Base Speed*

7.5.7 Class 0x29, Control Supervisor Object

All attributes of the *Control Supervisor Object* can be read / written by explicit message. They are assembled in various pre-defined assemblies for I/O data exchange. 7.6 for correct handling of attributes and necessary basic settings of inverter objects.

This object supports Instances 1 and 2, one for each axis.

Instance	Attribute	Name	Service	Data Type	Value
1, 2	3	Run 1	get/set	BOOL	1 – Run Forward (=Clockwise?)
	4	Run 2	get/set	BOOL	1 – Run Reverse (=Anticlockwise?)
	5	NetCtrl	get/set	BOOL	0 – Local control 1 – Network control
	6	State	get	USINT	1 – Startup 2 – Not_Ready 3 – Ready 4 – Enabled 5 – Stopping 6 – Fault_Stop 7 – Faulted
	7	Running1	get	BOOL	1 – Enabled and Run1 (= Clockwise)
	8	Running2	get	BOOL	1 – Enabled and Run2 (= Anticlockwise)
	9	Ready	get	BOOL	1 – Ready or Enabled or Stopping
	10	Faulted	get	BOOL	1 – Fault Occurred
	11	Warning	get	BOOL	1 – Warning
	12	FaultRst	get/set	BOOL	0 → 1 Fault Reset
	15	CtrlFromNet	get	BOOL	0 – Control is local 1 – Control is from network

The attributes are accessed by the following Axia objects:

Attribute	Name	Axia Object	Object name	Remarks
3*	RunFwd	**0x6040/0**	Control Word	
4*	RunRev	**0x6040/0**	Control Word	
5*	NetCtrl 1 0	**0x2200/ds** (ds = data sets)	Control Mode Mode of Control: State Machine Restores previous value	The writing of NetCtrl changes **0x2200/ds** only in RAM. The value of **0x25E0** must be set to `0x00604000: SRC Control Word`.
6	State	**0x6040/0**	Status Word	
7	Running1	**0x6041/0**	Status Word	
8	Running2	**0x6041/0**	Status Word	

Attribute	Name	Axia Object	Object name	Remarks
9	Ready	**0x6041/0**	Status Word	
10	Faulted	**0x6041/0**	Status Word	
11	Warning	**0x6041/0**	Status Word	
12*	FaultRst	**0x6040/0**	Control Word	
15	CtrlFromNet		Echo from 5 NetCtrl	

*) When writing this EIP object, the Axia object is changed in RAM. The written value can not always be read back in AxiaManager.

The CIP class 0x29 can also be accessed via object **0x21F4**. The CIP instances correspond to the axes (index offset 0x0800), the CIP attributes correspond to the sub indexes.

Object	Sub-index	Object Name
0x21F4	0	CIP 29h Control Supervisor
	3	CIP 29h RunFwd
	4	CIP 29h RunRev
	5	CIP 29h NetCtrl
	6	CIP 29h State
	7	CIP 29h Running1
	8	CIP 29h Running2
	9	CIP 29h Ready
	10	CIP 29h Faulted
	11	CIP 29h Warning
	12	CIP 29h FaultRst
	15	CIP 29h CtrlFromNet

Example:

0x21F4/3 = Instance 1 (axis 1), *Run Forward*

0x29F4/10 = Instance 2 (axis 2), *Faulted*

7.5.8 Class 0x2A, AC/DC-Drive Object

All attributes of the *AC/DC-Drive Object* can be read / written by explicit message. They are assembled in various pre-defined assemblies for I/O data exchange. ☞ 7.6 for correct handling of attributes and necessary basic settings of inverter objects.

This object supports Instances 1 and 2, one for each axis.

A distinction is made between two operating modes, **Frequency Inverter** and **Torque or Current controlled**.

The operating modes are set via the DriveMode attribute (6). Writing a DriveMode causes the writing of the following objects in the RAM:

- **0x6060/0** *ModeOfOperation*,
- **0x2080/0** *MotorControl*

and, if applicable

- **0x2321/ds** *PID Operation Mode*,
- **0x2313/ds** *PID Setpoint Source*,
- **0x2314/ds** *PID Actual Value Source*.

Depending on the **0x6060/0** *Mode of Operation*, the setpoints are transferred in different attributes and forwarded to different objects (inverters). The same applies to actual values (see next table below).

For remote operation of the inverter, the attributes NetRef (4) or NetProc (5) and NetCtrl (5) from Class Control Supervisor (0x29) must be true (1).

Objects in unit "VL Units" correspond to [RPM] as long as objects VL Set-Point Factor Numerator+ Denumerator (606B:1+2) are set to 1.

The attributes are accessed by the following Axia objects:

Attribute	Name	Axia Object	Remarks
6*	DriveMode: Write		CIP 29 Ctrl FromNet must be "1"
	0 Vendor specific mode	**0x6060/0** **0x2080/0**	FB Axia Speed Control Unchanged
	1 Open loop speed control	**0x6060/0** **0x2080/0**	02 Velocity Mode 00 V/f Scalar Control)
	2 Closed loop speed control	**0x6060/0** **0x2080/0**	02 Velocity Mode 01 Field Oriented Control
	3 Torque control	**0x6060/0** **0x2080/0**	04 Torque Profile Mode Unchanged
	4 Process control	**0x6060/0** **0x2080/0** **0x2321/ds** **0x2313/ds** **0x2314/ds**	02 Velocity Mode 01 Field Oriented Control *PID Operation mode* = Percentage *PID Setpoint source = s0021F50E* *PID Actual value source = Percentage Analog in (if 0x2314 = 0)*
	5 Position control	**0x6060/0** **0x2080/0**	01 Profile Position Mode Unchanged
	Read: Value is generated in 6061:0		(all 6060 Modes:)
	0 Vendor specific mode		00 No mode stopped
	1 Open loop speed control "x" = last value written	**0x2080/0=0** **0x2080/0=1**	02 Velocity mode
	0 Vendor specific mode		03 Profile velocity mode
	3 Torque control		04 Torque profile mode
	0 Vendor specific mode		05 Reserved
	0 Vendor specific mode		06 Homing mode
	0 Vendor specific mode		07 Interpolated position mode
	0 Vendor specific mode		08 Cyclic synchronous position
	0 Vendor specific mode		09 Cyclic synchronous velocity
	0 Vendor specific mode		0A Cyclic synchronous torque
	0 Vendor specific mode		F6 Axia automatic tuning
	0 Vendor specific mode		FB Axia speed control
	0 Vendor specific mode		FC Axia torque control
	0 Vendor specific mode		FD Axia electronic gear
	0 Vendor specific mode		FE Axia move away from limit switch
	0 Vendor specific mode		FF Axia table travel record)
7	SpeedActual [RPM]	*s00224003 * 60	MMSFDep_asOutAD[axis].f32FreqMech * 60 (rpm)
8	SpeedRef [RPM]		(depending on 6061:0 and 2080:0)
	FB Axia Speed	s0021F508	EIP Speed Ref (SpeedRef / 60)
	02 Velocity Uf	s00604200	Target Velocity (VL Units = rpm)
	02 Velocity FOC	s00604200 (0x6042/0 is reading „0")	Target Velocity (VL Units = rpm)
	xx	--	Not used

*) When writing this EIP object, the Axia object is changed in RAM. The written value can not always be read back in AxiaManager.

Attribute	Name	Axia Object	Remarks
11	TorqueActual [Nm] (depending on 6061:0) 04 Torque profile mode FC Axia torque control xx	 0x6077/0 *0x2580/3 --	 Torque [1ms] (Nm) unfiltered Default s00404411 = ActVal_asOutAD [ui8Axis].sUnfilterd.f32Torque_*NM*
12	TorqueRef [Nm] (depending on 6061/0) 04 Torque profile mode FC Axia torque control xx	 s00607110 s0021F50C --	 TorqueNom = (2009/ds) Torque Ref / TorqueNom * 1000 Torque Ref Not used
13	ProcessActual [%] (PID Controller = Technology controller)	 *0x2314/ds	 PID_asInParamAD[0].aeChoise-ListActualValue[0] via Default Percentage Analog In
14	ProcessRef [%]: -DrvMode 4 -DrvMode x	 S0021F50E	 Eip Process Ref [%] -1000 < x < 1000 Not used

*) When writing this EIP object, the Axia object is changed in RAM. The written value can not always be read back in AxiaManager.

If the error "CIP Status = 16 (0x10) Ext = None => Device state conflict" occurs during writing the SpeedRef, TorqueRef or ProcessRef, one of the following values is not set:

- Class Control Supervisor (0x29) NetCtrl (5)
- Class Control Supervisor (0x29) NetRef (4)
- Class Control Supervisor (0x29) NetProc (5)
- Class AC/DC Drive (0x2A) DriveMode (6).

The CIP class 0x2A can also be accessed via object **0x21F5**. The CIP instances correspond to the axes (index offset 0x0800), the CIP attributes correspond to the sub indexes.

Object	Sub-index	Object Name
0x21F5	0	*CIP 2Ah AC_DC Drive*
	3	*CIP 2Ah AtReference*
	4	*CIP 2Ah NetReference*
	5	*CIP 2Ah NetProc*
	6	*CIP 2Ah DriveMode*
	7	*CIP 2Ah SpeedAct*
	8	*CIP 2Ah SpeedRef*
	11	*CIP 2Ah TorqueAct*
	12	*CIP 2Ah TorqueRef*
	13	*CIP 2Ah ProcessAct*
	14	*CIP 2Ah ProcessRef*
	22	*CIP 2Ah SpeedScale*
	23	*CIP 2Ah CurrentScale*
	24	*CIP 2Ah TorqueScale*
	25	*CIP 2Ah ProcessScale*
	26	*CIP 2Ah PowerScale*
	27	*CIP 2Ah VoltageScale*
	28	*CIP 2Ah TimeScale*
	29	*CIP 2Ah RefFromNet*
	30	*CIP 2Ah ProcFromNet*

Example:

0x21F5/4 = Instance 1 (axis 1), *NetReference*

0x29F5/8 = Instance 2 (axis 2), *SpeedRef*

7.5.9 Class 0x80, Access to Inverter Objects

Vendor specific inverter objects are accessed via CIP Class 0x80. The instances for R/W access by explicit message comprise a 32-bit value.

Instance				
Byte	**0**	**1**	**2**	**3**
	Not used	Sub index	Object number	

The following attributes are supported:

Attribute	Name	Service	Data Type	Description
1	Name	get	Array of CHAR	ADI name (can be multilingual)
2	DataType	get	Array of UINT8	Each UINT8 defines the data type of the corresponding element of the instance value for structures and variables. For arrays, one UINT8 defines the data type for all subelements of the corresponding array element.
3	Number of elements	get	UINT8	Number of elements in attribute #5. It is strongly recommended not to use ADIs with Number of elements set to zero since this is not accepted by some networks.
4	Descriptor	get	Array of UINT8	Each UINT8 is a bit field specifying the access rights etc. for the corresponding element of the instance value for structures and variables. For arays, one UINT8 defines the descriptor for all subelements of the corresponding array element. b3 and o4 are mandatory if remapping of Process Data is supported. Bit: Access: b3 and b4 are mandatory if remapping of Process Data is supported. 0: 1: Get Access 1: 1: Set Access 2: –: (reserved, set to zero) 3: 1: Can be mapped as Write Process Data 4: 1: Can be mapped as Read ProcessData 5 1: The ADI instance is a non-volatile storage parameter in the application 6 1: Data notification is enabled
5	Value(s)	Determined by attribute #4	Determined by attribute #2	ADI value(s) Indexed elements can be of different types and sizes as specified in attribute #2. This attribute consists of all elements packed together with bit alignment. No implicit padding should be used.
6	Max. value	get	Determined by attribute #2	The maximum permitted ADI value. Implementation or this attribute is optional. If not implemented, the module will use the maximum value of the specified data type for this attribute.
7	Min. value	get	Determined by attribute #2	The minimum permitted ADI value. Implementation of the attribute is optional. If not implemented, the module will use the minimum value of the specified data type for this attribute.
8	Default value	get	Determined by attribute #2	The default ADI value. Implementation of the attribute is optional. A zero value (float: +Min. valu) will be used if not implemented.

Attribute	Name	Service	Data Type	Description
9	Number of subelements	get	Array of UINT16	Each UINT16 defines the number of subelements of the corresponding element of the instance value for structres and variables. Implementation of this attribute is optional, and it must not be implemented for arrays. Data types with bit alignment cannot be used with more than 1 subelement. If this attribute is not implemented, one (1) subelement for each element is assumed.
10	Element name	get	Struct of Strings (Array of CHAR, separated by NULL byte)	This attribute is used to enable reading the name of each element in an ADI of class Structure. Each string is separated by a NULL byte. There is no NULL byte at the end of the last string. The attribute reflects the element names used on the network. The number of elements in the structure has to be equal to the value of attribute #3 (Number of elements). Commands possible for this attribute are Get_Attribute (response includes strings and separating NULL bytes} and Get_Indexed_Attribute (the string is returned without any NULL byte). The entire response must fit into the message data field. The largest response accepted is 255 or 1524 bytes, depending on used channel.

Free mapping is possible via assembly 100/101 of the Assembly class 0x04, see chapter 7.5.4. The mapping can be viewed via objects **0x3953** and **0x3954**, see chapter 7.6.2.

7.5.9.1 Compatibility with legacy inverters

To keep compatibility with ACU, ACT and ANG inverters, there are also the objects **0x3952/03-14** and **0x3952/17-28**.

For these objects, choice lists are available, ☞ 7.5.5.3 and 7.5.5.6.

Object **0x3952** only reflects the EIP access to the parameter class 0x0F, Instance = sub index, Attribute = 5 (Value), but can also be read via the AxiaManager tool.

Mappings via object **0x3952** are only available for axis 1.

Mappings with objects for axis 2 can only be carried out using assembly 100/101 or objects **0x3953** and **0x3954**.

7.6 I/O Assemblies

I/O-Assemblies are pre-defined object assemblies for I/O data with defined content. The assemblies to be used are selected by:

- Object **0x3952/1** *InputAssemblySelection* for input values to the PLC
  - choice list for input assemblies: 70 (default)/71/72/74/101
- Object **0x3952/15** *OutputAssemblySelection* for output values from the PLC
  - choice list for output assemblies 20 (default)/21/22/24/100

The selection can be made by explicit message, or with the AxiaManager control software.

These objects are only settable if no I/O Connection is established. When connecting an inverter these settings are overwritten.

AxiaManager

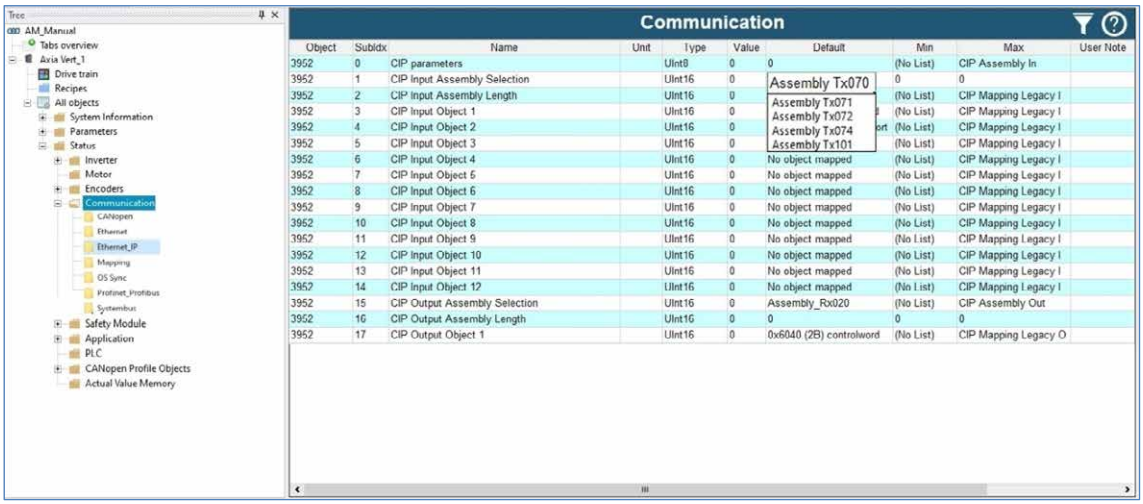
The assembly selection is carried out by writing the following inverter objects:

- *Input Assembly Selection* **0x3952/1**
- *Output Assembly Selection* **0x3952/15**

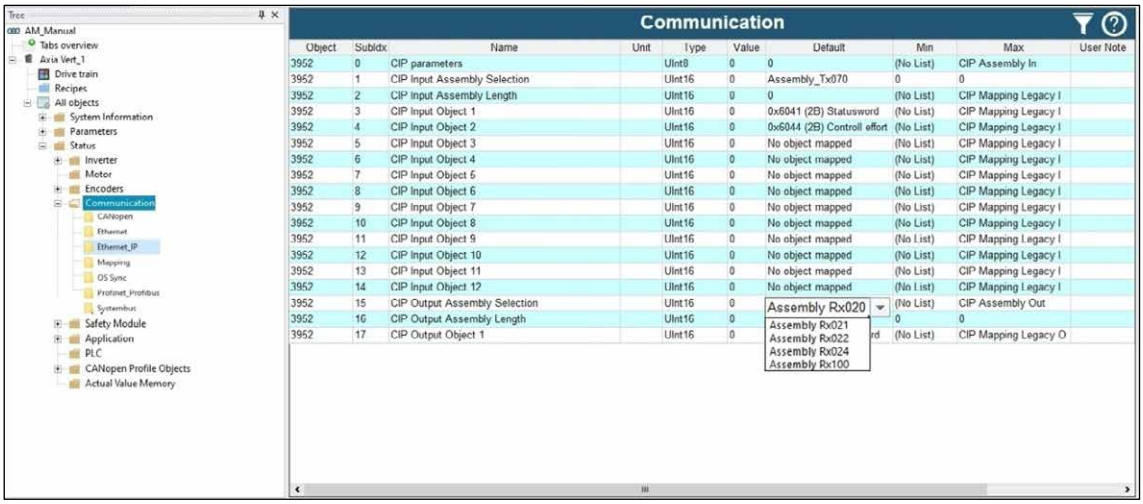
The length of the currently selected assembly can be read back from the inverter with:

- *Input Assembly Length* **0x3952/2**
- *Output Assembly Length* **0x3952/16**

Input Assembly

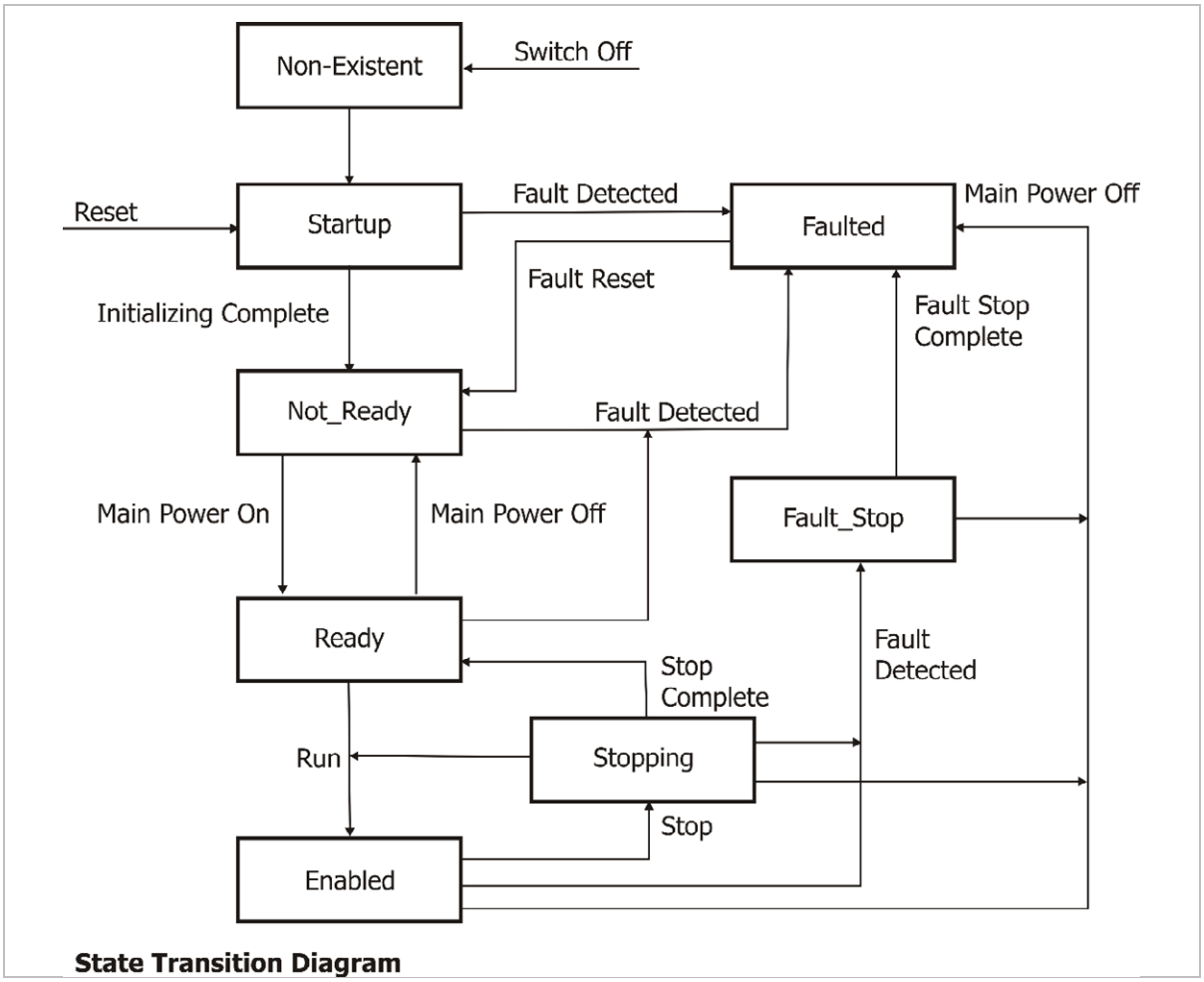


Output Assembly



7.6.1 CIP Assemblies 20/21/22/24 + 70/71/72/74

Note that the assemblies 20-24 and 70-74 are only available for one axis.
If two axes are needed Assembly 100 must be used.



State Transition Diagram

The actual state of the drive is displayed in the *Control Supervisor Object* (Class 0x29, Instance 1 or 2, Attribute 6 "state").

- 1 - Startup
- 2 - Not_Ready
- 3 - Ready
- 4 - Enabled
- 5 - Stopping
- 6 - Fault_Stop
- 7 - Faulted

7.6.1.1 Output Assembly 20

Object **0x3955/20** *Assembly Rx020* defines the content of the output assembly selection chosen with object **0x3952/15**.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	-	-	-	-	-	Fault Reset	-	Run Forward
1	-	-	-	-	-	-	-	-
2	Speed Reference (low byte)							
3	Speed Reference (high byte)							

- unused, no function

Run Forward = `1` drive runs clockwise

Fault Reset = `0` → 1, acknowledge fault

***Speed Reference* = nnnn [RPM]**

The handling of Speed Reference value depends on object **0x6061/0** *Mode Of Operation*:

If **0x6061/0** is set to `0x00000002 Velocity mode`, the value reference speed (rpm) is written to either EIP_ECL2ADriveMode "Closed Loop Speed" or "Open Loop Speed".

The value can be read via object **0x6042/0** *vl target velocity*.

If in **0x6061/0** object value `0xFFFFFFF8 Axia Speed Control` is selected, the speed value is written to EIP ECL2A DriveMode "VendorSpecific".

The value can be read via object **0x21F5/8** *CIP 2Ah SpeedRef*.

7.6.1.2 Output Assembly 21

Object **0x3955/21** *Assembly Rx021* defines the content of the output assembly selection chosen with object **0x3952/15**.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	-	Net Ref	Net Ctrl	-	-	Fault Reset	Run Reverse	Run Forward
1	-	-	-	-	-	-	-	-
2	Speed Reference (low byte)							
3	Speed Reference (high byte)							

- unused, no function

Run Forward = `1` drive runs clockwise

Run Reverse = `1` drive runs anticlockwise

Setting both objects, *Run Forward* and *Run Reverse*, to `1` shall have no effect.

Fault Reset = 0 → 1, acknowledge fault

***Speed Reference* = nnnn [RPM]**

For the handling of Speed Reference value, ☞ 7.6.1.2

7.6.1.3 Output Assembly 22

Object **0x3955/22** *Assembly Rx022* defines the content of the output assembly selection chosen with object **0x3952/15**.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	-	-	-	-	-	Fault Reset	-	Run Forward
1	-	-	-	-	-	-	-	-
2	Speed Reference (low byte)							
3	Speed Reference (high byte)							
4	Torque Reference (low byte)							
5	Torque Reference (high byte)							

- unused, no function

Run Forward = `1` drive runs clockwise

Fault Reset = `0` → `1`, acknowledge fault

Speed Reference **= nnnn [RPM]**

For the handling of Speed Reference value, ☞ 7.6.1.2

Torque Reference **= nnnn [Nm]**

The handling of Torque Reference value depends on object **0x6061/0** *Mode Of Operation*:

If **0x6061/0** is set to `0xFFFFFFFB Axia Speed Control`, the speed reference only is saved internally and will not be used any further. The value can be read and written via object **21F5/8**.

The source for **0x21F5/8** speed reference is `0x0021F508` for the first axis and `0x0029F508` for the second axis, if available.

Note that the second axis is only available for assemblies 100 and 101.

Axia Speed Control can only work if one of these objects are selected as source:

- **0x2511** *Speed Reference 1*
- **0x2512** *Speed Reference 2*
- **0x2513** *Speed Reference 3*
- **0x2514** *Speed Reference 4*

7.6.1.4 Output Assembly 24

Object **0x3955/24** *Assembly Rx024* defines the content of the output assembly selection chosen with object **0x3952/15**.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	-	-	-	-	-	Fault Reset	-	Run Forward
1	-	-	-	-	-	-	-	-
2	Speed Reference (low byte)							
3	Speed Reference (high byte)							
4	Process Reference (low byte)							
5	Process Reference (high byte)							

- unused, no function

Run Forward = 1 drive runs clockwise

Fault Reset = 0 → 1, acknowledge fault

***Speed Reference* = nnnn [RPM]**

For the handling of Speed Reference value, 7.6.1.2

***Process Reference* = nnnn [%]**

The handling of Process Reference value depends on object **0x6061/0** *Mode Of Operation*:

If **0x6061/0** is set to 0xFFFFFFFB Axia Speed Control, the speed reference only is saved internally and will not be used any further. The value can be read and written via object **21F5/8**.

The source for **0x21F5/8** speed reference is 0x0021F508 for the first axis and 0x0029F508 for the second axis, if available.

Note that the second axis is only available for assemblies 100 and 101.

Axia Speed Control can only work if one of these objects are selected as source:

- **0x2511** *Speed Reference 1*
- **0x2512** *Speed Reference 2*
- **0x2513** *Speed Reference 3*
- **0x2514** *Speed Reference 4*

7.6.1.5 Input Assembly 70

Object **0x3955/70** *Assembly Tx070* defines the content of the input assembly selection chosen with object **0x3952/1**.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	-	-	-	-	-	Running1	-	Faulted
1	-	-	-	-	-	-	-	-
2	Speed Actual (low byte)							
3	Speed Actual (high byte)							

- unused, no function

Faulted = 1 fault detected

Running1 = 1 drive running

***Speed Actual* = nnnn [RPM]**

The data type of Speed Actual is INT. *Speed Actual* has negative values when the drive is running anticlockwise.

7.6.1.6 Input Assembly 71

Object **0x3955/71** *Assembly Tx071* defines the content of the input assembly selection chosen with object **0x3952/1**.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	At Reference	Ref From Net	Ctrl From Net	Ready	Running2 (Rev)	Running1 (Fwd)	Warning	Faulted
1	Drive State							
2	Speed Actual (low byte)							
3	Speed Actual (high byte)							

Faulted = 1 fault detected

Running1 (Fwd) = 1 drive running forward

Running2 (Rev) = 1 drive running reverse

Ready = 1 drive state Ready

Ctrl From Net = 1 inverter controlled by EtherNet/IP™

Ctrl From Net = 0 inverter controlled by local control

Ref From Net = 1 speed reference from EtherNet/IP™

Ref From Net = 0 speed reference from local control

Drive State

- 1 – Startup
- 2 – Not_Ready
- 3 – Ready
- 4 – Enabled
- 5 – Stopping
- 6 – Fault_Stop
- 7 - Faulted

***Speed Actual* = nnnn [RPM]**

The data type of *Speed Actual* is INT. *Speed Actual* has negative values when the drive is running anticlockwise.

7.6.1.7 Input Assembly 72

Object **0x3955/72** *Assembly Tx072* defines the content of the input assembly selection chosen with object **0x3952/1**.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	-	-	-	-	-	Running1	-	Faulted
1	-	-	-	-	-	-	-	-
2	Speed Actual (low byte)							
3	Speed Actual (high byte)							
4	Torque Actual (low byte)							
5	Torque Actual (high byte)							

- unused, no function

Faulted = 1 fault detected

Running1 = 1 drive running

***Speed Actual* = nnnn [RPM]**

The data type of Speed Actual is INT. *Speed Actual* has negative values when the drive is running anticlockwise.

***Torque Actual* = nnnn [Nm]**

7.6.1.8 Input Assembly 74

Object **0x3955/74** *Assembly Tx074* defines the content of the input assembly selection chosen with object **0x3952/1**.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	-	-	-	-	-	Running1	-	Faulted
1	-	-	-	-	-	-	-	-
2	Speed Actual (low byte)							
3	Speed Actual (high byte)							
4	Process Actual (low byte)							
5	Process Actual (high byte)							

- unused, no function

Faulted = 1 fault detected

Running1 = 1 drive running

***Speed Actual* = nnnn [RPM]**

The data type of Speed Actual is INT. *Speed Actual* has negative values when the drive is running anticlockwise.

***Process Actual* = nnnn [%]**

7.6.2 Vendor Specific Assemblies 100/101

The vendor specific assemblies 100 and 101 allow the creation of assemblies with variable object mapping. The method is similar to PDO mapping with CANopen. A list of CANopen objects is available for mapping.

Index	Sub-index	Name	Designation	Access
0x21F3	3–15	CIP Input Objects 1–2	CIP Class 0x28 Motor Data	Axis-dependent
0x21F4	3–15		CIP Class 0x29	Axis-dependent
0x21F5	3–30		CIP Class 0x2A	Axis-dependent
0x3952	1–40	CIP Parameter	CIP Params	AGL, ACx, ANG (Legacy, see 7.5.9.1) Mapping ReadOnly
0x3953	0–32	CIP Mapping 100 Out	Mapping for CIP Assembly 100 Out (Choicelist: see table below)	Read/Write, CIP does not support multiple axes
0x3954	0–32	CIP Mapping 101 In	Mapping for CIP Assembly 101 In (Choicelist: see table below)	Read/Write, CIP does not support multiple axes
0x3955	1 51	Assembly Rx Control Bits Assembly Tx Status Bits	CIP Assembly Control, Status	Set value/actual value
0x3955	20–24 70–74 100 101	Assemblies Rx20–24 Assemblies Tx70–74 Assembly Rx100 Assembly Tx101	CIP Assemblies	Set value/actual value
0x4080	0–OD_NR_OF_PDORX_MAPPINGS_PER_AXIS	Rx PDO Mapping	RxPDO mapping as string	Axis-dependent, not CIP specific, only reflects 0x3953 if assembly 100 is selected
0x4081	0–OD_NR_OF_PDOTX_MAPPINGS_PER_AXIS	Tx PDO Mapping	TxPDO mapping as string	Axis-dependent, not CIP specific, only reflects 0x3954 if assembly 101 is selected

The length is independent from the number of mapped objects.

These assemblies are only settable if an IO-Connection is **not** established. When connecting an inverter these settings are overwritten.

Output Assembly 100:

- Parameter Object Class 0x0F, Instance 15, Attribute 1 = **100**
- Parameter Object Class 0x0F, Instance 16, Attribute 1 = *Output Assembly length*

Input Assembly 101:

- Parameter Object Class 0x0F, Instance 1, Attribute 1 = **101**
- Parameter Object Class 0x0F, Instance 2, Attribute 1 = *Input Assembly length*

The assemblies can also be selected with the inverter configuration utility AxiaManager. The relevant objects are located under Communication > EtherNet/IP™:

- **0x3952/1** *Input assembly selection* `0x00000065 Assembly Tx101`
- **0x3952/2** *Input assembly length*
- **0x3955/15** *Output assembly selection* `0x00000064 Assembly Rx100`
- **0x3952/16** *Output assembly length*

7.6.2.1 Object Mapping - Vendor Specific Output Assembly 100

With AxiaManager

When using the inverter utility AxiaManager, object **0x3955/100** offers a selection list where one of the listed objects can be chosen for mapping. Choosing `0x00000000 No object mapped` marks the end of the mapped objects.

The mapping **MUST** begin with Index 1. Unused Indices / Mapping Positions must be set to `No object mapped`.

All mapping settings unequal to `No object mapped` following the first Index after `No object mapped` are ignored.

The mapping parameters are only settable if an IO-Connection is **not** established, or the Connection is **not** set to Assembly 100/101.

Mapping example:

Index 1 - `0x6040 (2B) controlword`

Index 2 - `0x6042 (2B) target velocity`

Index 3 - no object mapped ➔ end of mapping, `control word` and `target velocity` are included in output assembly 100 and processed

Index 4 - `0x3003 (1B) Digital Out set values` **IGNORED**

Index 5 **IGNORED**

…

Index 12 **IGNORED**

If objects are ignored, there will be **no error message** generated by AxiaManager. The ignored objects are automatically overwritten with `No object mapped`, however this is only visible after the next screen update.

0xnnnn:	CANopen object index
(xB):	number of bytes (object length)
Text:	CANopen object name

7.6.2.2 Object Mapping - Vendor Specific Input Assembly 101

With AxiaManager

When using the inverter utility AxiaManager, object **0x3955/101** Indices `1` to `24` offer a choice box where one of the listed objects can be chosen for mapping. Choosing `No object mapped` marks the end of the mapped objects.

The mapping **MUST** begin with Index 1. Unused Indices / Mapping Positions must be set to "`No object mapped`".

All mapping settings unequal to "`No object mapped`" following the first Index after "`No object mapped`" are ignored.

The mapping parameters are only settable if an IO-Connection is **not** established or the Connection is **not** set to Assembly 100/101.

Mapping example:

Index 1 - `0x6041 (2B) Status word`

Index 2 - `0x6044 (2B) Control effort`

Index 3 - no object mapped ➔ end of mapping, `status word` and `control effort` are included in input assembly 101 and processed

Index 4 - `0x3001 (2B) Digital In` actual values **IGNORED**

Index 5 **IGNORED**

…

Index 12 **IGNORED**

If objects are ignored, there will be **no error message** generated by AxiaManager. The ignored objects are automatically overwritten with "`No object mapped`", however this is only visible after the next screen update.

0xnnnn:	CANopen object index
(xB):	number of bytes (object length)
Text:	CANopen object name

7.6.3 Handling of I/O Assemblies, Setup Device/PLC

All samples and screenshots in this chapter are taken from the Rockwell PLC.

CompactLogix *L23E* QBFC1B
with **RSLogix 5000** (V20.01).

For the support/use of EDS files, RSLogix 5000 version V20 or higher is required.

STEP1 – Initial Inverter Setup

The inverter must be equipped with the module CMA-IE-01-EIP. Connect the inverter to the network and assign the IP Address (☞ 5.2).

The required I/O Assemblies must first be selected with AxiaManager (☞ 7.6).

If vendor specific Assemblies 100/101 are selected, then check/select the mapped objects with AxiaManager (☞ 7.6.2).

From here on, the meaning of parameters is from EtherNet/IP™'s point of view. These EtherNet/IP™ parameters are accessible **via the network only**.

STEP2 – Register the Inverter's EDS File

- Connect the configuration PC to the PLC and start RSLogix 5000.
- Create a new project or select an existing project.
- Register the EDS file(s) for the used inverter(s), 7.2.
- To register the new device, select "EDS Hardware Installation Tool" in the Tools menu.

  The EDS Wizard is then started:
- Follow the instructions in the EDS Wizard.
- After successful registration, continue with the creation of the new Module:

STEP3 – Create the New Module

In the Controller Organizer window:

- right click on the "Ethernet" branch and select "New Module" in the menu list:

  The "Select ModuleType" dialog box is displayed, with a list of all registered modules in the database:
- Enter the Catalog Number of the device in the "Enter Search Text for Module Type…" box.

  The corresponding Catalog numbers are automatically displayed while you type.
- Whenever the required number is displayed, select it and then click the "Create" button.

  The "`New Module`" dialog box is displayed:
- Enter the Module Name, the IP Address and then click on the "Change" button.

  This opens the "Module Definition" dialog box:
  The Module Revision is automatically displayed (taken from the EDS File) and can be changed.
- Select the used Assemblies from the drop-down list box, and then set the Assembly Sizes and Tag Suffix, if necessary.
- Finally, check the Tag Suffix and change it, if necessary.
- Click the "OK" button to accept the settings or click "Cancel" to quit.

If changes were made a message box is displayed for confirmation.

The "Module Definition" dialog box is then closed and the "New Module" dialog box reactivated:

- Select the "Connection" tab for further settings:
- Click "Ok" to accept any changes or "Cancel" to quit.

7.7 Sending Explicit Messages from the PLC Program

This chapter describes how to send explicit messages from the RSLogix5000 PLC MainProgram.

The following steps are required:

STEP1

- Register the Inverter EDS File and create the Inverter Module (7.2)

STEP2

- Define the required messages (7.7.1).

STEP3

- Add the appropriate code to the PLC MainProgram (7.7.2)

7.7.1 How to create a MESSAGE Tag

A Message is defined by creating a Controller Tag of Data Type *MESSAGE*.

This chapter describes the basic steps required for creating a Message Tag. For further details, please refer to the RSLogix 5000 Manual or Online Help.

The example below describes how to create the following Message:

Name: MsgCtrlWrite

Service Code: 0x10, Set Attribute Single

Class: 4, Assembly Object

Instance: 21, Output Assembly 21 (7.6.1.2)

Attribute: 3, Data value

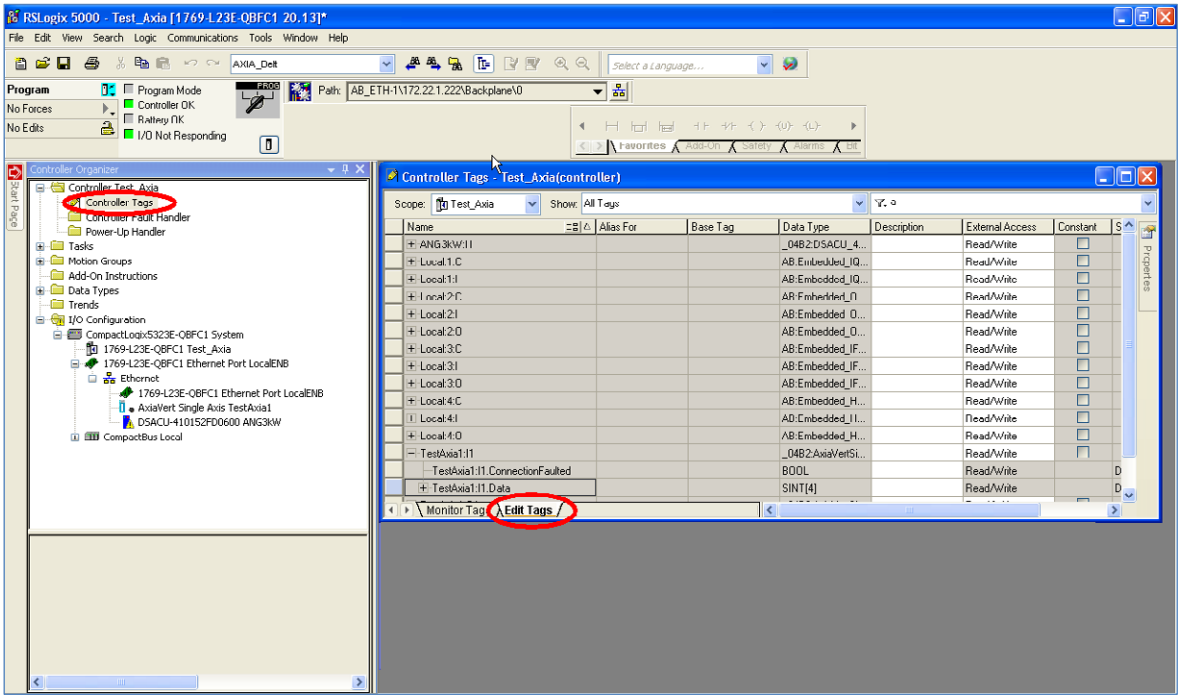
The following four data bytes are predefined for Assembly 21.

Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
0	-	Net Ref	Net Ctrl	-	-	Fault Reset	Run Reverse	Run Forward
1	-	-	-	-	-	-	-	-
2	Speed Reference (low byte)							
3	Speed Reference (high byte)							

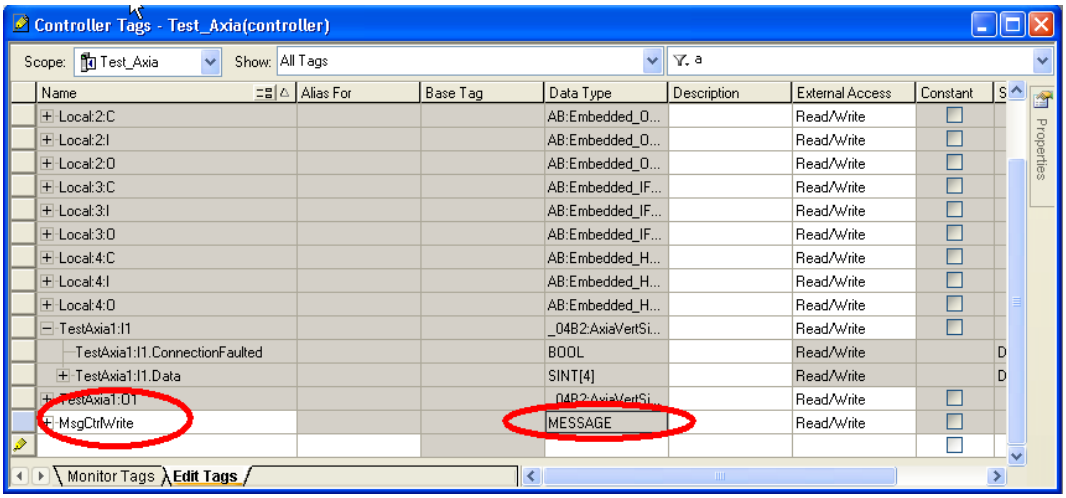
- unused, no function

This Message can be used to set the *NetCtrl* and *NetRef* objects when using Output Assembly 20, 22 or 24 (0) as these Assemblies do not contain the *NetCtrl* and *NetRef* objects.

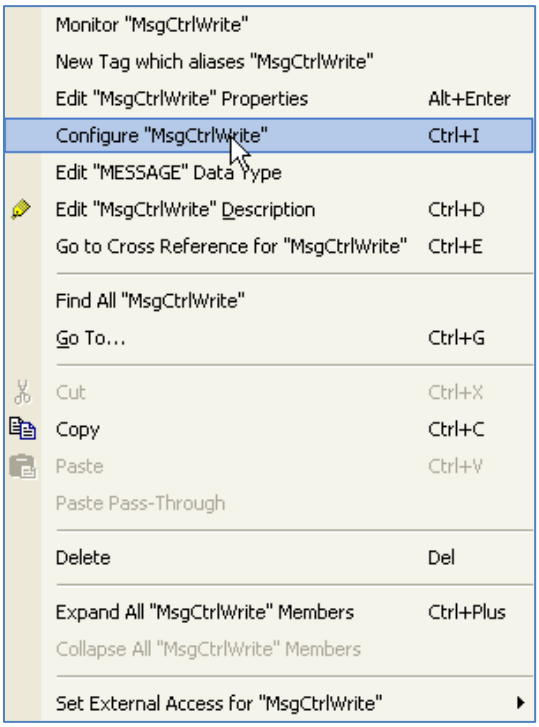
Open the *Controller Tags* window, and click on the *Edit Tags* tab to enter the new tag:



Enter the Tag name "`MsgCtrlWrite`" in the data field provided and enter or select the Data Type *MESSAGE*.



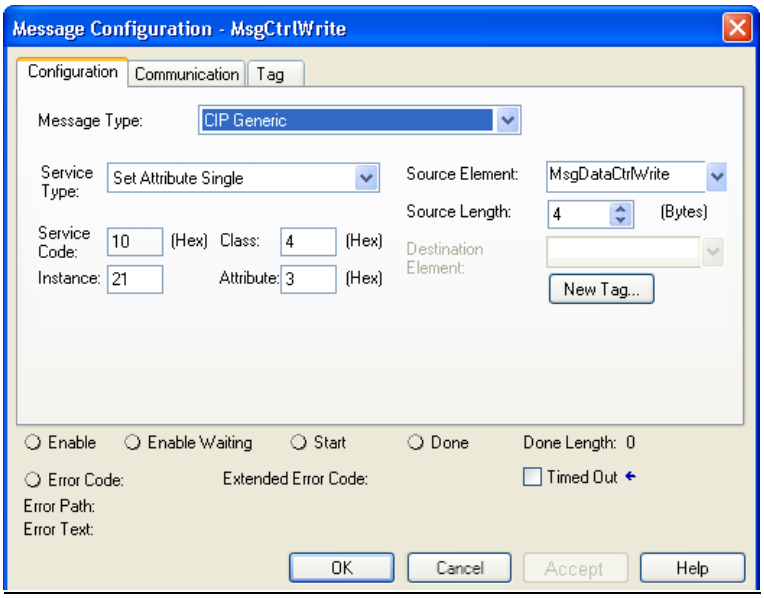
Right click on the Tag Selection button to the left of the Tag Name and select *Configure "MsgCtrlWrite"* from the menu to display the Message Configuration dialog box:



Select Service Type *Set Attribute Single* from the pull-down menu.

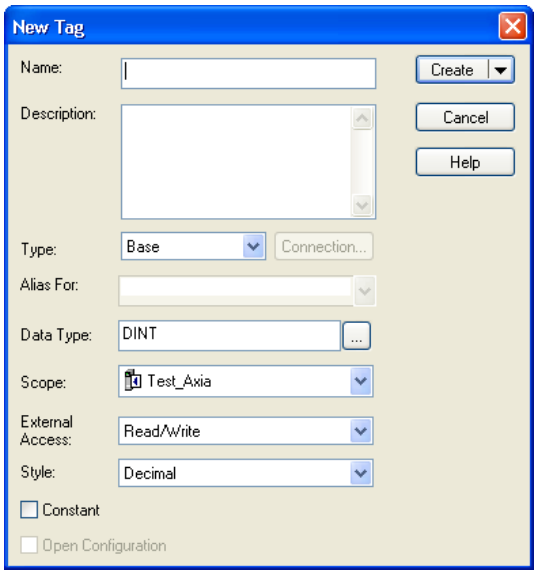
The Service Code (`10`) is displayed automatically.

Set the Class to `4`, Instance to `21`, Attribute to `3` and the Source Length to `4`.



If not already defined, a new tag must be defined for the Source Element:

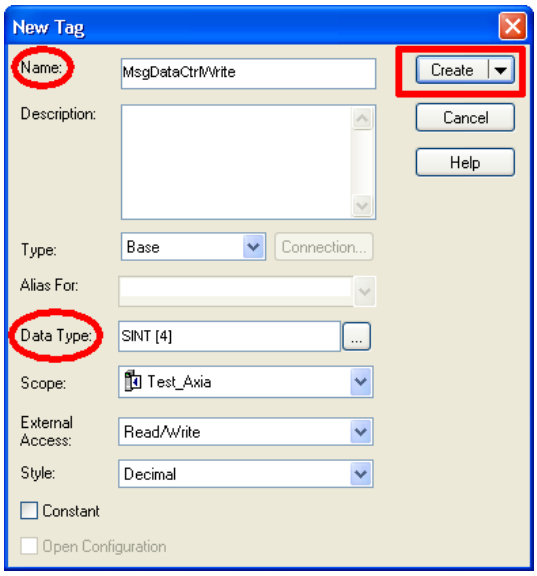
Click on the *New Tag* button to display the "New Tag" dialog box:



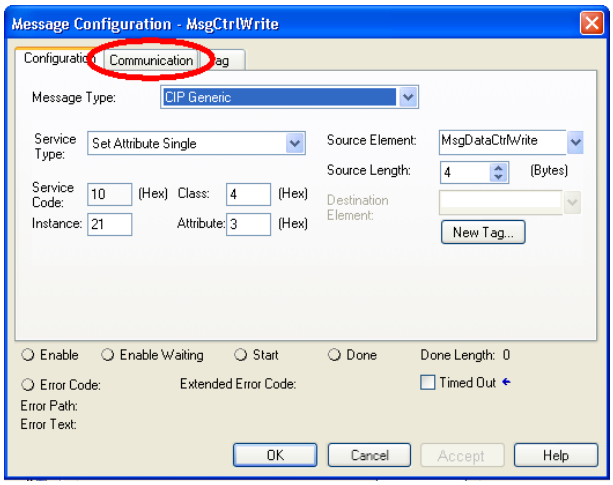
Enter the following data:

- Name: `MsgDataCtrlWrite`
- Data Type: `SINT[4]`

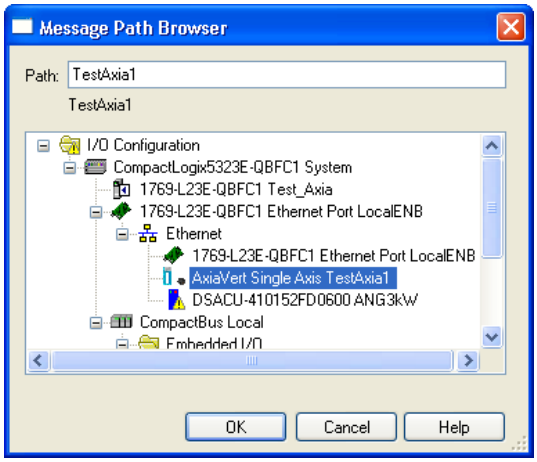
Then click on the *Create* button (or one of the options) to create the tag.
Close the "New Tag" dialog box, if not done automatically, and select the newly created tag for the Source Element.



Select the *Communication* tab and click on the *Browse* button to select the Message Path:



Select the required device and Click *OK* to accept the selection and *OK* again in the "Message Configuration" dialog box.



The message is now created.

Other messages can be created in the same manner.

Once all messages are created, code can be added to the PLC Program to send the messages. This is described in the next chapter.

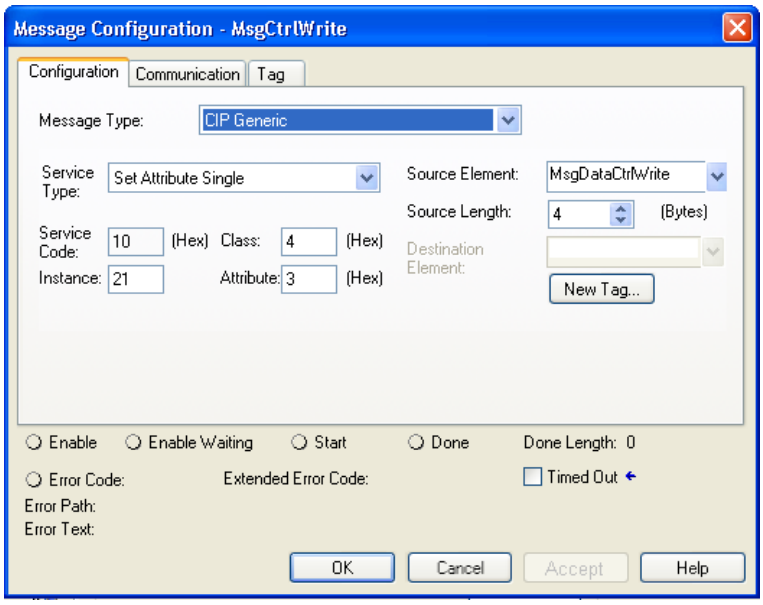
7.7.2 PLC Program Code Examples

Example 1: Setting *NetCtrl* and *NetRef* via Output Assembly *21*.

This example uses the following tags, created as described in the previous chapter:

MsgCtrlWrite

Tag of Data Type `MESSAGE`, defined as follows:



MsgDataCtrlWrite

Tag of Data Type `SINT[4]` – 4 Byte Buffer for the Message Data as defined by Output Assembly *21*.

The following code sets the required data and sends the message:

```
// Set Assembly Data
MsgDataCtrlWrite[0]    := 16#60;  // NetRef = 1, NetCtrl = 1
MsgDataCtrlWrite[1]    := 0;      // Byte 1 not used
MsgDataCtrlWrite[2]    := 0;      // Speed Reference (low byte)
MsgDataCtrlWrite[3]    := 0;      // Speed Reference (low byte)
// Set Data Length
MsgCtrlWrite.REQ_LEN := 4;      // length of Assembly 21
// Send Message
MSG(MsgCtrlWrite);
```

The following flags of the MsgCtrlRead message tag should then be monitored to evaluate the message state:

Flag	Name	Description
.EN	Enable	Message processing enabled
.DN	Done	Message sent successfully
.ER	Error	Message error The error code and extended error code are in the *.ERR* and *.EXERR* data fields of the *MsgCtrlRead* message structure.

For a full description of the *MSG* instruction, *MESSAGE* structure and other related topics, please refer to the RSLogix 5000 Manuals or Online Help.

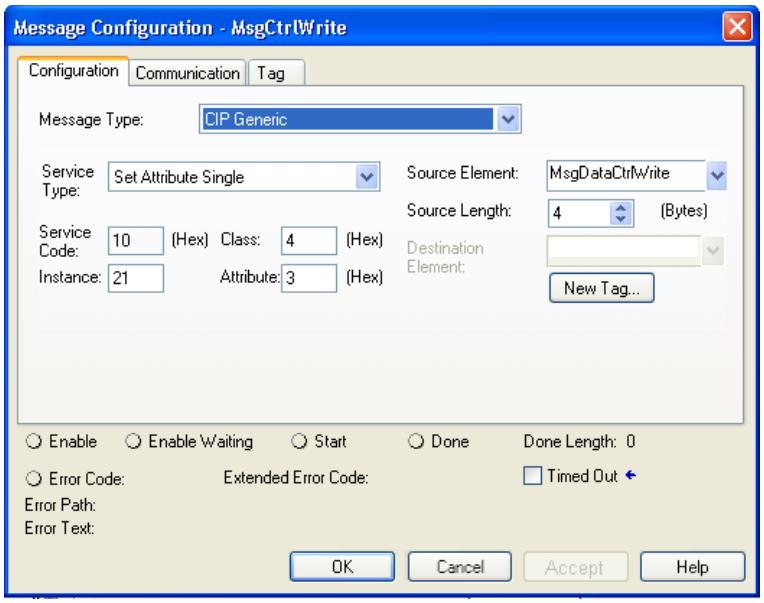
Example 2: Reading back *NetCtrl* and *NetRef* via Input Assembly *71*.

This can be used for verification of the data sent as described in Example 1.

The following tags are required:

MsgCtrlRead

Tag of Data Type `MESSAGE`, defined as follows:



MsgDataCtrlRead

Tag of Data Type `SINT[4]` – 4 byte Buffer for the Message Data as defined by Input Assembly *71*.

The following code sends the read request message:

```
// Read NetCtrl and NetRef via Assembly 71
// ------------------------------------------------
// Send Message
MSG(MsgCtrlRead);
```

The following flags of the *MsgCtrlRead* message tag can then be monitored to evaluate the message state:

Flag	Name	Description
.EN	Enable	Message processing enabled
.DN	Done	Message sent successfully. The data read is in tag *MsgDataCtrlRead*
.ER	Error	Message error The error code and extended error code are in the *.ERR* and *.EXERR* data fields of the *MsgCtrlRead* message structure.

For a full description of the *MSG* instruction, *MESSAGE* structure and other related topics, please refer to the RSLogix 5000 Manuals or Online Help.

7.8 Diagnosis of EtherNet/IP™ state

The EtherNet/IP™ State is displayed via the inverter actual value object *CANopen Nmt State* **0x3911/9**.

7.9 Resetting errors

Depending on the settings and operating state of the device, errors can be reset in various ways:

When using control via object *Control Mode* **0x2200** = `3 - Statemachine`:

- Set bit 7 in **0x6040** *Control word* = `0x0080`.

When using control via Keypad, object *Control mode* **0x2200** = `2 - Keypad`:

Press the STOP button of the keypad.

– Resetting by pressing the STOP button is only possible if object *Control Mode* **0x2200** permits control via the keypad.

When using control via IOs, object *Control mode* **0x2200** = `1 - IOs`:

Reset the error by activating the corresponding digital input.

– Resetting via digital signal can only be carried out when object *Control Mode* **0x2200** permits this or when an input with the additional (hardware) is selected in the case of physical inputs.

Some errors will occur again after an error reset. In such cases, it may be necessary to take certain measures (e.g. moving from a limit switch in the non-disabled direction).

8 Object structure

The available objects are marked with Index and Sub-index and must be addressed via this ID.

The objects are listed in the following tables. The following definitions apply:

Access type			
Read only	The PLC can only read data from the frequency inverter.		
Read/Write	The PLC is granted access (reading and writing) to the frequency inverter data.		
Data type			
Unsigned32	32 Bit value: (UDINT)	$0…2^{32}$-1 0…0xFFFF FFFF	(0…4294967295)
Unsigned16	16 Bit value: (UINT)	$0…2^{16}$-1 0…0x FFFF	(0…65535)
Unsigned8	8 Bit value: (USINT)	$0…2^{8}$-1 0…0xFF	(0…255)
Integer32	Signed 32 Bit value: (DINT)	$-2^{31}…2^{31}$-1 0x8000 0000…0x7FFF FFFF	(-2147483648 … 2147483647)
Integer16	Signed 16 Bit value: (INT)	$2^{15}…2^{15}$-1 0x8000…0x7FFF	(-32768…32767)
Integer8	Signed 8 Bit value: (SINT)	$2^{7}…2^{7}$-1 0x80…0x7F	(-128…127)
PDO Mapping			
No	This object cannot be used for exchange of PDO. Only SDO can be used.		
Tx	This object can be transmitted from the frequency inverter in a TxPDO.		
Rx	This object can be transmitted to the frequency inverter in a RxPDO.		

"Highest sub-index supported" shows the highest sub-index supported by the object.

8.1 Objects table

Object Grouping

Every object is addressed via a 16 Bit index, which is displayed as a 4-digit hexadecimal number.

The object indexes are sorted in groups as follows:

- CANopen Communication Objects: **0x1000** – **0x1FFF**
- Bonfiglioli-specific objects: **0x2001** – **0x5FFF** with

Axis-dependent object ranges:

- **0x2001** – **0x27FF**
- **0x4000** – **0x47FF**

The Bonfiglioli-specific objects can be subdivided in axis-dependent objects versus axis-independent objects. There is an offset of 80x00 per axis in the axis-dependent range.

For example:

0x2001 *Motor Type* on Axis 1 and **0x2801** *Motor Type* on Axis 2. The Bonfiglioli-specific objects in the range

- **0x3000** – **0x37FF**
- **0x5800** - **0x5FFF**

are not axis-dependent.

- CANopen Drive Profile objects: **0x6000** – **0x7FFF**

Object-no		Group
from	**to**	
0x2001	0x27FF	Configuration: Axis 1, Settings for Axis 1 Example: 0x2001 for motor type Axis 1
0x2801	0x28FF	Configuration: Axis 2, Settings for Axis 2 Example: 0x2801 for motor type Axis 2
0x3800	0x3FFF	Configuration: Axis independent settings Example: 0x3801 for serial-no. of Axia device
0x4000	0x47FF	Actual Values: Readings for Axis 1 Example: 0x4001 for active data set Axis1
0x4800	0x4FFF	Actual Values: Readings for Axis 2 Example: 0x4801 for active data set Axis2
0x5800	0x5FFF	Actual Values: axis independent readings Example: 0x5801 for DC-link Voltage
0x6000	0x67FF	CiA 402 objects Axis 1
0x6800	0x6FFF	CiA 402 objects Axis 2

9 Available Objects

9.1 0x3952 CIP Parameters

Index	Sub-index	Designation	Data type	Access	Default
0x3952	0	CIP parameters	UInt8	ro	0
	1	CIP Input Assembly Selection	UInt16	ro	Assembly_Tx070
	2	CIP Input Assembly Length	UInt16	ro	0
	3	CIP Input Object 1	UInt16	ro	0x6041 (2B) Statusword
	4	CIP Input Object 2	UInt16	ro	0x6044 (2B) Controll effort
	5	CIP Input Object 3	UInt16	ro	No object mapped
	6	CIP Input Object 4	UInt16	ro	No object mapped
	7	CIP Input Object 5	UInt16	ro	No object mapped
	8	CIP Input Object 6	UInt16	ro	No object mapped
	9	CIP Input Object 7	UInt16	ro	No object mapped
	10	CIP Input Object 8	UInt16	ro	No object mapped
	11	CIP Input Object 9	UInt16	ro	No object mapped
	12	CIP Input Object 10	UInt16	ro	No object mapped
	13	CIP Input Object 11	UInt16	ro	No object mapped
	14	CIP Input Object 12	UInt16	ro	No object mapped
	15	CIP Output Assembly Selection	UInt16	ro	Assembly_Rx020
	16	CIP Output Assembly Length	UInt16	ro	0
	17	CIP Output Object 1	UInt16	ro	0x6040 (2B) controlword
	18	CIP Output Object 2	UInt16	ro	0x6042 (2B) target velocity
	19	CIP Output Object 3	UInt16	ro	No object mapped
	20	CIP Output Object 4	UInt16	ro	No object mapped
	21	CIP Output Object 5	UInt16	ro	No object mapped
	22	CIP Output Object 6	UInt16	ro	No object mapped
	23	CIP Output Object 7	UInt16	ro	No object mapped
	24	CIP Output Object 8	UInt16	ro	No object mapped
	25	CIP Output Object 9	UInt16	ro	No object mapped
	26	CIP Output Object 10	UInt16	ro	No object mapped
	27	CIP Output Object 11	UInt16	ro	No object mapped
	28	CIP Output Object 12	UInt16	ro	No object mapped
	29-32	Reserve	UInt16	ro	0
	33	CIP Abort Behaviour	Int8	ro	Fault Signal
	34	CIP COS Mask 0	UInt32	ro	0xFFFFFFFF
	35	CIP COS Mask 1	UInt32	ro	0xFFFFFFFF
	36	CIP COS Mask 2	UInt32	ro	0xFFFFFFFF
	37	CIP COS Mask 3	UInt32	ro	0xFFFFFFFF
	38	CIP COS Mask 4	UInt32	ro	0xFFFFFFFF
	39	CIP COS Mask 5	UInt32	ro	0xFFFFFFFF
	40	Reserve	UInt16	ro	0
	41	CIP Mapping Commit	Int8	ro	Off

7.6

9.2 0x3953/255 ClP Mapping 100 Out

Index	Sub-index	Designation	Data type	Access	Default
0x3953	255	ClP Mapping 100 Out	UInt32	rw	*(Multiple values)*

7.6.2.1

9.3 0x3954/255 ClP Mapping 101 In

Index	Sub-index	Designation	Data type	Access	Default
0x3954	255	ClP Mapping 101 In	UInt32	rw	*(Multiple values)*

7.6.2.2

9.4 0x3955 CIP Assemblies

Index	Sub-index	Designation	Data type	Access	Default
0x3955	0	CIP Assemblies	UInt8	ro	0
	1	Assembly Rx Control Bits	UInt8	rw	0
	20	Assembly Rx020	UInt32	rw	0
	21	Assembly Rx021	UInt32	rw	0
	22	Assembly Rx022	String	rw	000000
	24	Assembly Rx024	String	rw	000000
	51	Assembly Tx Status Bits	UInt8	ro	0
	70	Assembly Tx070	UInt32	ro	0
	71	Assembly Tx071	UInt32	ro	0
	72	Assembly Tx072	String	ro	
	74	Assembly Tx074	String	ro	
	100	Assembly Rx100	String	rw	0
	101	Assembly Tx101	String	ro	

7.6

9.5 0x4080/0 RxPDO Mapping

Index	Sub-index	Designation	Data type	Access	Default
0x4080	0	RxPDO Mapping	UInt8	ro	0

7.5.9

9.6 0x4081/0 TxPDO Mapping

Index	Sub-index	Designation	Data type	Access	Default
0x4081	0	TxPDO Mapping	UInt8	ro	0

7.5.9

Index

AUSTRALIA

Bonfiglioli Transmission (Aust.) Pty Ltd
2, Cox Place Glendenning NSW 2761
Locked Bag 1000 Plumpton NSW 2761
Tel. +61 2 8811 8000

BRAZIL

Bonfiglioli Redutores do Brasil Ltda Travessa Cláudio Armando 171 - Bloco 3 CEP 09861-730 - Bairro Assunção
São Bernardo do Campo - São Paulo
Tel. +55 11 4344 2322

CHINA

Bonfiglioli Drives (Shanghai) Co. Ltd. #68, Hui-Lian Road, QingPu District, 201707 Shanghai
Tel. +86 21 6700 2000

#568, Longpan Road, Jiading District, 201707 Shanghai

Bonfiglioli Trading (Shanghai) Co. Ltd. Room 423, 4th Floor, #38, Yinglun Road, China (Shanghai) Pilot Free Trade Zone, Shanghai

FRANCE

Bonfiglioli Transmission s.a.
14 Rue Eugène Pottier
Zone Industrielle de Moimont II
95670 Marly la Ville
Tel. +33 1 34474510

GERMANY

Bonfiglioli Deutschland GmbH
Sperberweg 12 - 41468 Neuss
Tel. +49 0 2131 2988 0

Bonfiglioli Deutschland GmbH Krefeld
Europark Fichtenhain B6 - 47807 Krefeld
Tel. +49 0 2151 8396 0

O&K Antriebstechnik GmbH
Ruhrallee 8-12 - 45525 Hattingen
Tel. +49 0 2324 2050 1

PRODUCTION
ASSEMBLY
SALES
SERVICE

INDIA

Bonfiglioli Transmission Pvt. Ltd.
Mobility & Wind Industries
AC 7 - AC 11 Sidco Industrial Estate
Thirumudivakkam Chennai - 600 044
Tel. +91 844 844 8649

Discrete Manufacturing & Process Industries - Motion & Robotics
Survey No. 528/1,
Perambakkam High Road Mannur Village,
Sriperumbudur Taluk Chennai - 602105
Tel. +91 844 844 8649

Discrete Manufacturing & Process Industries
Plot No.A-9/5, Phase IV MIDC Chakan,
Village Nighoje Pune - 410 501
Tel. +91 844 844 8649

ITALY

Bonfiglioli Riduttori S.p.A. **Discrete Manufacturing & Process Industries**
Via Cav. Clementino Bonfiglioli, 1
40012 Calderara di Reno
Tel. +39 051 6473111

Discrete Manufacturing & Process Industries
Via Sandro Pertini, lotto 7b
20080 Carpiano
Tel. +39 02985081

Discrete Manufacturing & Process Industries
Via Saliceto, 15 - 40010 Bentivoglio

Mobility & Wind Industries
Via Enrico Mattei, 12 Z.I. Villa Selva
47122 Forlì
Tel. +39 0543 789111

Motion & Robotics
Via Unione, 49 - 38068 Rovereto
Tel. +39 0464 443435/36

NEW ZEALAND

Bonfiglioli Transmission (Aust.) Pty Ltd
88 Hastie Avenue, Mangere Bridge,
2022 Auckland
PO Box 11795, Ellerslie
Tel. +64 09 634 6441

SINGAPORE

Bonfiglioli South East Asia Pte Ltd
8 Boon Lay Way, #04-09,
8@ Tadehub 21, Singapore 609964
Tel. +65 6268 9869

SLOVAKIA

Bonfiglioli Slovakia s.r.o.
Robotnícka 2129
Považská Bystrica, 01701 Slovakia
Tel. +421 42 430 75 64

SOUTH AFRICA

Bonfiglioli South Africa Pty Ltd.
55 Galaxy Avenue, Linbro Business Park,
Sandton, Johannesburg
2090 South Africa
Tel. +27 11 608 2030

SPAIN

Tecnotrans Bonfiglioli S.A
Pol. Ind. Zona Franca, Sector C,
Calle F, n°6 - 08040 Barcelona
Tel. +34 93 447 84 00

TURKEY

Bonfiglioli Turkey Jsc
Atatürk Organize Sanayi Bölgesi,
10007 Sk. No. 30
Atatürk Organize Sanayi Bölgesi,
35620 Çiğli - Izmir
Tel. +90 0 232 328 22 77

UNITED KINGDOM

Bonfiglioli UK Ltd.
Unit 1 Calver Quay, Calver Road, Winwick
Warrington, Cheshire - WA2 8UD
Tel. +44 1925 852667

USA

Bonfiglioli USA Inc.
3541 Hargrave Drive
Hebron, Kentucky 41048
Tel. +1 859 334 3333

VIETNAM

Bonfiglioli Vietnam Ltd.
Lot C-9D-CN My Phuoc Industrial Park 3
Ben Cat - Binh Duong Province
Tel. +84 650 3577411

Abbiamo un'inflessibile dedizione per l'eccellenza, l'innovazione e la sostenibilità. Il nostro Team crea, distribuisce e supporta soluzioni di Trasmissioni e Controllo di Potenza per mantenere il mondo in movimento

We have a relentless commitment to excellence, innovation & sustainability. Our team creates, distributes and services world-class power transmission & drive solutions to keep the world in motion.

Wir verpflichten uns kompromisslos zu Qualität, Innovation und Nachhaltigkeit. Unser Team entwickelt, vertreibt und wartet erstklassige Energieübertragungs- und Antriebslösungen, um die Welt in Bewegung zu halten

Notre engagement envers l'excellence, l'innovation et le développement durable guide notre quotidien. Notre Équipe crée, distribue et entretient des solutions de transmission de puissance et de contrôle du mouvement contribuant ainsi à maintenir le monde en mouvement.

Tenemos un firme compromiso con la excelencia, la innovación y la sostenibilidad. Nuestro equipo crea, distribuye y da soporte en soluciones de transmisión y control de potencia para que el mundo siga en movimiento.

COD. VEC 2en35 R0

HEADQUARTERS
Bonfiglioli S.p.A
Registered Office: Via Bazzane, 33
40012 Calderara di Reno BO
Head office: Via Isonzo, 65/67/69
40033 Casalecchio di Reno BO
ITALY

tel: +39 051647 3111
www.bonfiglioli.com