Bonfiglioli

Vectron

Active Next Generation

Operating Instruction
Expansion Module EM-AUT-21
ANG210 / ANG 410 / ANG510 / ANG610

Bonfiglioli

Table of contents

1 General Information on Documentation

1.1 Instruction manuals

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

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 parameters and the configuration of the frequency inverter by the software.

Operating instructions

The Operating Instructions describe all functions of the frequency inverter. The parameters required for adapting the frequency inverter to specific applications and several additional functions are described in detail.

Application manual

The application manual supplements the documentation for purposeful installation and commissioning of the frequency inverter. Information on various subjects related to the use of the frequency inverter is given in a way specific to the application.

If you need a copy of the documentation or additional information, contact your local representative of BONFIGLIOLI.

The present documentation was prepared with great care and was subjected to extensive and repeated reviews. For reasons of clarity, it was not possible to include all details of all types of the product in the documentation. Neither was it possible to consider all conceivable installation, operation or maintenance situations. If you require further information or if you encounter specific problems which are not dealt with in the documentation, contact your local BONFIGLIOLI agent.

The present document was created in English. Other language versions are translations.

1.2 This document

The information in this document pertains specifically to the *Automation Interface EM-AUT-21* of *ACTIVE NEXT GENERATION (ANG)* series. You will find further information in the basic operating instructions document "**VEC1105**".

This document describes the Automation Interface module EM-AUT ACTIVE NEXT GENERATION (ANG) series.

This document contains important information on the installation and use of the product in its specified application range. Compliance with this document 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 document carefully.

IMPORTANT:

Compliance with the documentation is required to ensure safe operation of the frequency inverter. BONFIGLIOLI VECTRON 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.

This document exclusively describes the EM-AUT-21 expansion modules. It is not to be understood as fundamental information for the operation of the frequency inverters of the ANG series.

This document is not to be understood as providing general/basic information on Ethernet/IP. It requires basic knowledge of the methods and effects of Ethernet/IP be known by the user.

Software or Hardware designations and label names of the companies mentioned here are usually protected by law concerning trademark, patent and brand name. Licensing right and copyright are held by the respective companies. Contents of this document may not be copied or duplicated outside the legal regulations without written permission.

This document applies to expansion modules of the following device series:

- EM-AUT-21 (Ethernet-IP)
- EM2-AUT-21 (Ethernet-IP)

In this manual the designation "EM-AUT-21" also applies to EM2-AUT-21 modules.

1.3 Warranty and liability

BONFIGLIOLI VECTRON GmbH (hereinafter referred to as "manufacturer") notes that the contents of this document do not form part of any previous or existing agreement, assurance or legal relationship between the manufacturer and the user of the document (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 the document. 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 solar 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.

1.4 Obligation

This document must be read before commissioning. 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.

1.5 Copyright

Any copyrights relating to this document shall remain with

BONFIGLIOLI VECTRON 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.

1.6 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. In case the frequency inverter is sold to other users, this Operating Instructions document 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 according to 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 tasked with 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 who have 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 who were 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 who have special technical know-how and experience relating to 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 2006/95/EEC and DIN EN 61800-5-1. CE-labeling is based on these standards. Responsibility for compliance with the EMC Directive 2004/108/EC 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.

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

2.3.1 Explosion protection

The frequency inverter is an IP 20 protection class 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. Remaining hazards are not obvious and can be a source of possible injury or health damage.

Typical residual risks 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.

Electrostatic charging

Touching electronic components bears the risk of electrostatic discharges.

Thermal hazards

Risk of accidents by hot machine/plant surfaces, e.g. heat sink, transformer, fuse or sine filter.

Charged capacitors in DC link

The DC link may have dangerous voltage levels even up to three minutes (or up to eight minutes with bigger devices) after shutdown.

Danger of equipment falling down/over, e.g. during transport

The center of gravity is not the in middle of the electric cabinet modules.

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 Operating Instructions

2.6.1 Hazard classes

The following hazard identifications and symbols are used in the Operating Instructions 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
⚠	Danger of crushing		

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
●	Wear ear protectors

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 Sensitive Devices, i.e. components and assemblies sensitive to electrostatic energy

2.6.8 Information signs

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

2.6.9 Font style in documentation

Example	Font style	Use
1234	bold	Representation of parameter numbers
Parameter	inclined, font: Times New Roman	Representation of parameter names
1234	font: Courier New	Representation of parameter values and other values
P.1234	bold	Representation of parameter numbers without name, e.g. in formulas
Q.1234	bold	Representation of source numbers
•	Bullet point	Indicates an action to be taken
–	Bullet dash	Indicates a list of items

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 Operating Instructions, the operator should issue separate internal user manuals for the frequency inverter. The Operating Instructions of the frequency inverter must be included in the Operating Instructions 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 Skilled Personnel. The staff must not be under the influence of any drugs. Note the minimum age required by law. Define the staff's responsibility pertaining to 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 Operating Instructions 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 these Operating Instructions, 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.

2.9.3 Ear protectors

- The frequency inverter produces noise. For this reason it should be installed in areas where people normally don't stay.
- Noise emission in operation is < 85 dB(A) in the case of sizes 1 through 7.
- Noise emission in operation is approx. 86 dB(A) in the case of size 8. Ear protectors must be used when staying near the frequency inverter

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 Vectron MDS 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 Vectron MDS GmbH offers 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 so at your own risk.

2.10.3 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 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.

2.10.4 Electrical connections

- The five safety rules must be complied with.
- Never touch live terminals. In sizes 1 through 7, the DC-link may have dangerous voltage levels up to 3 minutes after shutdown. In size 8, the DC-link may have dangerous voltage levels up to 10 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 frequency inverter may be operated in TN, TT and IT grid types. Precautions must be taken for operation in IT grids. Operation in a corner-grounded TN grid shall not be permissible.

2.10.4.1 The five safety rules

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

1 Disconnect
2 Secure to prevent restarting
3 check for absence of voltage,
4 carry out earthing and short-circuiting
5 cover or shield neighboring live parts

2.10.5 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, all covers must be installed correctly, and all electrical cabinet doors must be closed. During operation, never open the machine/plant.
- No connection work shall be carried out while power supply is on.
- 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 braking resistor, may be hot even some time after the machine/plant was shut down. Do not 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. After shutdown, wait for at least 3 minutes (sizes 1 through 7) and at least 10 minutes (size 8) before starting any electrical or mechanical work on the frequency inverter. Even after this waiting time, make sure that the equipment is de-energized 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 the frequency inverter and children must not have access to the device.
- Do not bypass nor decommission any protective devices.
- 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 AutoStart 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).

2.10.6 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. Any repair work may only be carried out by the manufacturer or persons approved/licensed by the manufacturer. Any repair work must be carried out by qualified electricians. 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.

2.10.7 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.

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

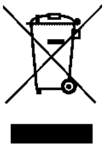
After the end of product service life, the user/operator must take the device out of operation.

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 Technical data

When using the EM-AUT expansion modules, the technical data **of the frequency inverter** must be also considered.

Control terminal X410A		
	Main function	**Other function**
X410A.1	Resolver Ref +	-
X410A.2	Resolver Ref -	-
X410A.3	Resolver Sin -	Encoder B +
X410A.4	Resolver Sin +	Encoder B -
X410A.5	Resolver Cos -	Encoder A +
X410A.6	Resolver Cos+	Encoder A -
X410A.7	Ground GND	

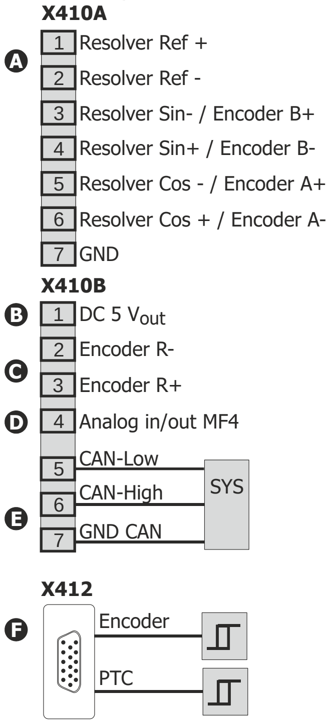
Control terminal X410B		
	Main function	**Other function**
X410B.1	-	Output 5V$_{DC}$[2]
X410B.2	-	Encoder R -
X410B.3	-	Encoder R +
X410B.4	Analog in/out MF4 [1][3]	
X410B.5	System bus, CAN low	
X410B.6	System bus, CAN high	
X410B.7	Ground GND$_{CAN}$	

[1] The control electronics parameters can be configured as required.

[2] The maximum power available is reduced by the other control outputs of the frequency inverter and expansion module.

[3] Also see chapter 13 Analog input/Analog output MF4 for more information

Block diagram



Ⓐ Ⓒ **Resolver / Speed sensor input EM-RES**
The encoder interface is suitable for typical market resolvers or TTL encoders.

Ⓑ **Voltage output for encoder supply**
DC 5 V, P$_{max}$ = 1 W, observe the maximum power supply

Ⓓ **EM-AUT: Analog input /Analog output MF4**
You can use the terminal optionally as analog input, analog output, PTC, KTY, PT1000 or digital input.

Ⓔ **Communication interface system bus**
Galvanic decoupled CAN-connection according to ISO-DIS 11898 (CAN High Speed), bus termination can be activated via internal switch. The Protocol CANopen or Systembus is set up via Parameter *CAN Interface (CAN-Systembus / CANopen)* **276**.

Ⓕ **Inputs for Absolute encoder and PTC/KTY/PT1000 (15 pole female connector HD-Sub-D)**
The additional encoder interface on the EM-AUT module is designed for connection of of standard commercial TTL encoders (optionally with reference track, without commutation tracks), SinCos encoders (with reference track, without commutation tracks), EnDat 2.1 (SinCoS track required), Hiperface and SSI encoders (optionally with TTL [RS-422] or SinCos track).
The encoder supply voltage at contacts X412.6 (VEnc) and X412.15 (0VL) can be adjusted through parameter Supply voltage 1187 in between DC 5.0 … 12 V. See chapter 7.4.4 “Supply voltage”. Maximum load: 1 W.

Technical data of control terminals X410

Resolver input (X410A.1) ... (X410A.6):

- Reference voltage $U_{REF\ eff}$ = 2.5 V, I_{max} = 40 mA (Ref)
- Input voltage $U_{min\ eff}$ = 2 V, voltage-proof up to 6 V_{rms} (Sin and Cos)
- Ratio U_{IN}/U_{REF} = 0.5
- Excitation frequency = 8 kHz
- Input impedance: >95 Ω at 8 kHz, Maximum pole pairs = 7,
- 30000 rpm at nº of pole pares = 1.

Speed sensor input (Alternative function) (X410A.3) ... (X410B.2):

- A/B tracks (TTL encoder)
- R track (Reference track, only evaluable with TTL and SinCos encoders)
- A/B tracks: constant part V =DC 2.5 V ±0.5 V, RS485 Standard
- R track: constant part V =DC 2.5 V ±0.5 V, RS485 Standard
- Frequency signal, f_{max} = 300 kHz, voltage-proof up to 6 V,
- TTL (push-pull) according to specification RS-422A / RS-485: U_{max} = 5 V

Voltage output DC 5 V for encoder supply (X410B.1):

P_{max} = 1 W. Depending on the load on the digital outputs of the frequency inverter and expansion module, this value may be lower.

Analog input / output MF4 (X410B.4):

- Analog signal: Input voltage: DC -10 V to 10 V (R_i = 69.5 kΩ), DC 0 V to 10 V (R_i = 69.5 kΩ), DC 0 ... 20mA (R_i = 249 Ω), PTC, KTY, PT1000.
- Resolution 13 Bit
- Output voltage: DC 0 V to 10 V
- Resolution 12 Bit
- Digital input: Low signal: DC 0 ... 4 V, High signal: DC 6 ... 30 V.

Conductor cross-section:

The control terminals are suitable for the following cable sizes:

– with ferrule:	0.25 … 1.0 mm²
– without ferrule:	0.14 … 1.5 mm²

Encoder and PTC input X412 (HD-Sub-D)

Encoder input:	**Motor temperature evaluation**
Internal resistance <120 Ω	**PTC-Input** Trigger resistance = 3.99 kΩ Hysteresis = 2.3 kΩ
A/B track: sine-shaped differential signal 0.6…1.2 Vpp	**KTY-Input** **PT1000 Input**
R-track: Differential signal 0.2…1.7 Vpp	
Clock and data Signal: V =DC 2.5 V ±0.5 V (RS485 Standard)	**PTC or bimetal temperature sensor (NC)**
Power supply encoder: V_{ENC} track: Supply DC 5 … 12 V (max. 1 W) $V_{ENC,Sense}$ track: encoder sensor cable	

CAUTION

Device damage

Exceeding the permitted power value for the DC 24V-Supply may damage the device.

The maximum total power load of the ANG DC 24V-Supply must not exceed 4 W. The Encoder supply voltages are supplied by the ANG DC 24V-Supply and must therefore be taken into account as well as the digital outputs of the ANG Controller.

- The manufacturer recommends connecting an external DC 24 V power supply to the inputs of the control terminal and to supply the input voltages of the digital inputs of the ANG-Controller directly from this external voltage.
- Note the encoder manufacturer's input power specifications.

NOTICE

Device damage

The inputs for motor temperature evaluation are **not** insulated by default. Inappropriately insulated components may result in device damage.

- Only motor temperature evaluations featuring safe insulation from the motor winding as per EN61800-5-1 may be connected.

BONFIGLIOLI servo motors of types BMD, BCR and BTD are provided with safe insulation from the motor winding.

Motor temperature input:

- PTC Trigger resistance = 3.99 kΩ, Hysteresis = 2.3 kΩ
- KTY input
- PT1000 input
- Bimetal temperature sensor (NC)

- Use PTC resistors with safe isolation from motor winding according to EN 61800-5-1.

4 Functional description of the expansion module

WARNING

Unintended movements! Device damage!

With the EM-AUT-21 module, it is possible to access **ALL** frequency inverter parameters from a controller. There is **no access control** levels as with the KP500 manual control unit or the VPlus PC software.

Changing parameters, the functions of which are not known to the user, can result in unintended movements of equipment and material and/or personal damage as well as permanent frequency inverter damage.

- Proceed with due caution and contact the manufacturer when in doubt.

The expansion modules EM-AUT-21 can only be used with frequency inverter of the ANG series.

The EM-AUT-21 expansion module is mounted on the frequency inverter **as an integrated component**.

- Do not remove this component.

This document exclusively describes the EM-AUT-21 expansion modules. It is not to be understood as fundamental information for the operation of the frequency inverters of the ANG series.

This document is not to be understood as providing general/basic information on Ethernet/IP. It requires basic knowledge of the methods and effects of Ethernet/IP be known by the user.

Software or Hardware designations and label names of the companies mentioned here are usually protected by law concerning trademark, patent and brand name. Licensing right and copyright are held by the respective companies. Contents of this document may not be copied or duplicated outside the legal regulations without written permission.

In some chapters, setting and display options via the PC software VPlus are described as an alternative to the KP500 control unit. If you wish to use the VPlus PC software, you will need an optional serial interface adapter KP232 or direct Ethernet connection with the system.

With older firmware versions, the EM2-AUT modules may show up as “EM-AUT” in VPlus in Parameter 1. Check the labels on your component/device to verify the status of your module.

EM2-AUT modules feature R-track evaluation for absolute encoder AND for SinCos-encoder. Check the labels on your component/device to verify the status of your module.

The EM-AUT-21 expansion modules are hardware components to complete the functionality of the frequency inverter. They allow the data exchange within a network and between directly connected components such as control and regulation elements.

The Ethernet-IP implementation is fully compatible with the CIP specification.

If data is written cyclically, comply with the instructions in Chapter 8.6 "Access to Inverter Parameters (Classes 0x64 … 0x73)".

4.1 Device Identification

The figure below shows a schematic representation of the EM-AUT module mounted on the frequency inverter.

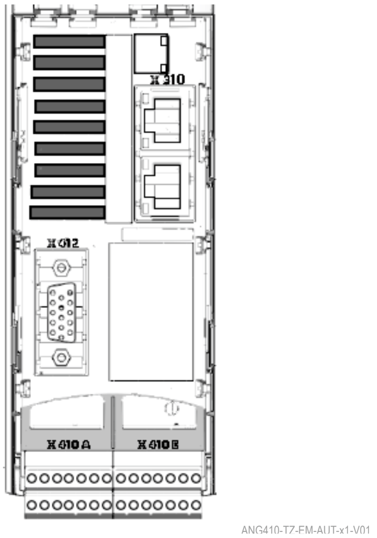


Figure 4-1: EM-AUT expansion module

4.2 Encoder range of applications

The EM-AUT-21 module supports, in the case of EnDat 2.1 encoders, a baud rate of 100 kBit/s. Other baud rates will not be supported.

Supported interfaces

- CAN interface (galvanic decoupled) for CANopen or System bus (CAN interface ISO-DIS 11898, CAN High Speed, max. 1 MBaud). See chapter 9 "CANopen".
- Ethernet I/P® communications.
  See chapter 8 "Ethernet/IP communication".
- Analog output DC 0…+10 V
  Usable either as analog input (DC -10…+10 V, DC 0…+10 V) or analog output (DC 0…+10 V) or as PTC, KTY, PT1000 input. See chapter 13.
- Additional Encoder interface X412 including PTC evaluation via HD-Sub-D female connector.

Supported encoder types:

- TTL (optionally with Reference track, without commutation tracks)
- SinCos (with Reference tracks, without commutation tracks)
- EnDat 2.1 (encoder type with SinCos track required)
- Hiperface
- SSI encoder (optionally with TTL [RS-422]- or SinCos track)

See chapter 7.4 "Encoder input 3 (X412) – Parameter descriptions".

The voltage level for encoder power supply can be set via a parameter (*Supply voltage* **1187**, see Chapter 7.4.4 "Supply voltage"). The voltage value can be controlled via a measuring cable (often referred to as "sense" line).

Depending on the motor and encoder type used there are restrictions as to usability in applications. The following sections describe the range of applications.

4.2.1 Asynchronous motor

- HTL (evaluation via basic device terminals X210)
- TTL (evaluation via X410) See chapter 7.2.
- SinCos,

- Hiperface,
- EnDat 2.1 with SinCos track,
- SSI with incremental track (TTL [RS-422] or SinCos)

can be used on asynchronous motors as:

- Motor encoders for speed feedback (e.g. Configuration 210)
- Motor encoders for speed feedback and parallel position feedback in non-slip systems (e.g. Configuration 240)
- Application encoder for position feedback with parallel speed feedback either via motor model (sensorless e.g. Configuration 440) or via HTL encoder (via terminals X210 on ANG basic device e.g. Configuration 240) or via TTL encoder (via terminals X410 on EM-AUT module i.e. Configuration 240).

SSI encoders without incremental track

can be used on asynchronous motors as:

- Application encoder for position feedback with speed feedback either via motor model (sensorless e.g. Configuration 440) or via HTL encoder (via terminals X210 on ANG basic device e.g. Configuration 240).

EnDat 2.1 without SinCos track cannot be used.

4.2.2 Synchronous motor

- Resolver
- Hiperface,
- EnDat 2.1 with SinCos track,
- SSI with incremental track (TTL [RS-422] or SinCos)

can be used on synchronous motors as:

- Motor encoders for speed feedback (e.g. Configuration 510).
- Motor encoders for speed feedback and parallel position feedback in non-slip systems (e.g. Configuration 540).
- Application encoder for position feedback with parallel speed feedback via motor model (sensorless e.g. Configuration 640).
- HTL (evaluation via basic device terminals X210)
- TTL (evaluation via X410)
- TTL (evaluation via X412)
- SinCos without commutation track
- SSI encoders without incremental track

can be used on synchronous motors as:

- Application encoder for position feedback with parallel speed feedback via motor model (sensorless e.g. Configuration 640).

EnDat 2.1 without SinCos track cannot be used.

5 Connecting the expansion module

5.1 General information

The electrical installation of the EM-AUT expansion module must be carried out by qualified personnel according to the general and regional safety directives as well as installation directives. For a safe operation of the frequency inverter it is necessary that the documentation and the device specifications be complied with during installation and commissioning. In the case of special applications, you may also have to comply with further guidelines and instructions.

The frequency inverters are designed according to the requirements and limit values of product norm EN 61800-3 with an interference immunity factor (EMI) for operation in industrial applications. The electromagnetic interference is to be avoided by expert installation and observation of the specific product information.

For further information, refer to the chapter "Electrical Installation" of the **frequency inverter operating instructions**.

WARNING

Live system! Electric shock hazard!

Direct contact with live components possible.

- All connection terminals, where dangerous voltage levels may be present (e.g. motor connection terminals, mains terminals, fuse connection terminals, etc.) must be protected against direct contact in the final installation.

5.2 Mechanical Installation

The EM-AUT-01 or EM-AUT-04 expansion module is mounted on the frequency inverter as an integrated component. Do not remove this component.

5.3 Electrical Installation

WARNING

Severe injury by electric current

If the following instructions are not complied with, there is direct danger with the possible consequences of death or severe injury by electric current. Further, failure to comply can lead to destruction of the frequency inverter and/or of the expansion module.

- Make sure that the frequency inverter is not energized during installation.
- Make sure that the voltage supply is securely disconnected and prevent unintended reconnection.

WARNING

Live system! Electric shock hazard!

When the frequency inverter is disconnected from power supply, the mains terminals, DC-link voltage terminals and motor terminals may still be live for some time.

- Wait until the DC link capacitors have discharged before starting work on the unit. The time to wait is at least 3 minutes in the case of sizes 1 through 7 and at least 10 minutes in the case of size 8.

NOTICE

Module damage

The MF4OA output used as analog output (terminal X410B.4) cannot permanently withstand short circuits and external voltages. External voltages can damage the module.

- Avoid short circuits and external voltages.

5.3.1 Control terminals

CAUTION

Device damage

Connecting or disconnecting control lines on a live system may damage the device.

- Switch off power supply before connecting or disconnecting the control inputs and outputs.

CAUTION

Device damage

Exceeding the permitted power value for the DC 24V-Supply may damage the device.

The maximum total power load of the ANG DC 24V-Supply must not exceed 4 W. The Encoder supply voltages are supplied by the ANG DC 24V-Supply and must therefore be taken into account as well as the digital outputs of the ANG Controller.

- The manufacturer recommends connecting an external DC 24 V power supply to the inputs of the control terminal and to supply the input voltages of the digital inputs of the ANG-Controller directly from this external voltage.
- Note the encoder manufacturer's input power specifications.

NOTICE

Communication faults

Without GND$_{CAN}$ connection telegram faults or telegram interruptions can occur.

- For the connections of two or more devices CAN Low, CAN High and GNDCAN must be connected.

NOTICE

Unexpected operating behavior

The power output on terminal **X410B.1** may be loaded with a maximum power of 1 W. Depending on the total load on the DC 24 V supply voltage, the available power output may decrease. Higher power loads can cause unexpected operating behavior.

- Avoid higher power loads.

NOTICE

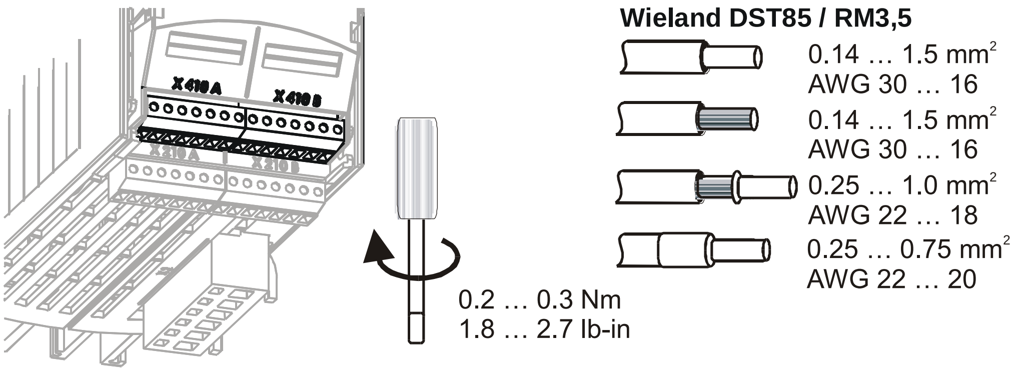
Electromagnetic Interference

Inappropriate shielding of lines may result in electromagnetic interferences.

- In order to minimize electromagnetic interference and to obtain a good signal quality, connect the shield of the cable to a ground plane (PE) at both ends.

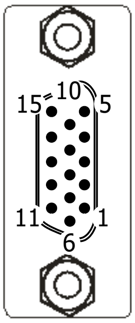
Please see chapter 3 “Technical data” for technical details.

Expansion module EM-AUT



Encoder and PTC input X412 (female connector HD-Sub-D)

Contact	Function				
	SinCos	TTL	Hiperface	EnDat 2.1	SSI
Housing	PE		PE	PE	PE
1				Clock-	Clock-
2				Clock+	Clock+
3	Cos-	B-	Cos-	Cos-	(optionally B- / Cos-)
4	Cos+	B+	Cos+	Cos+	(optionally B+ / Cos+)
5	TM_{PTC-}		TM_{PTC-}	TM_{PTC-}	TM_{PTC-}
6	V_{Enc}		V_{Enc}	V_{Enc}	V_{Enc}
7	R-	-			
8	-	R-	Data-	Data-	Data-
9	Sin-	A-	Sin-	Sin-	(optionally A- / Sin-)
10	TM_{PTC+}		TM_{PTC+}	TM_{PTC+}	TM_{PTC+}
11	$V_{Enc,Sense}$			$V_{Enc,Sense}$	$V_{Enc,Sense}$
12	R+	-			
13	-	R+	Data+	Data+	Data+
14	Sin+	A+	Sin+	Sin+	(optionally A+ / Sin+)



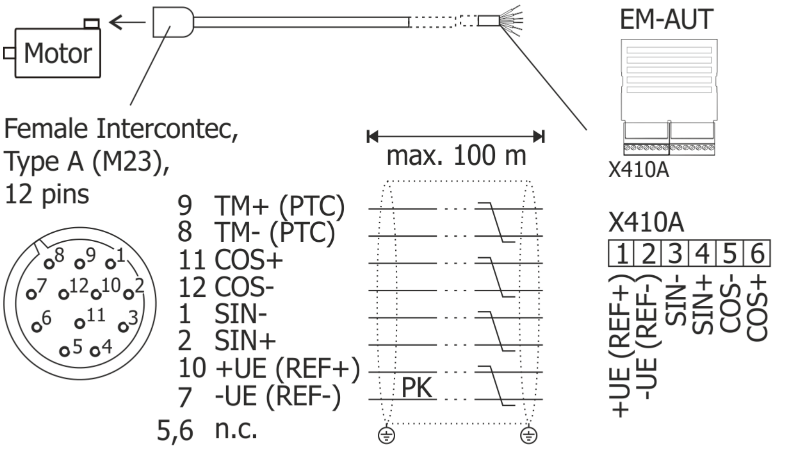
Function and signal

Function	Signal
Housing	Shield connected with PE
A+/A- Cos+/Cos- B+/B- Sin+/Sin-	0.6 V … 1.2 Vpp incremental signal In the case of SSI encoders, the A+/A- and B+/B- tracks can be used, as an option, for TTL [RS-422] or SinCos signals.
R+/R-	Reference track (RS485 Standard)
Clock+/Clock-	Clock signal (RS485 Standard)
Data+/Data-	Data signal (RS485 Standard)
TM_{PTC} + TM_{PTC} –	Motor Temperature evaluation
V_{Enc} GND	Encoder supply (DC 5 …12 V), max. load capacity 1 W
V_{EncS}	Measuring line for monitoring of V_{Enc} [2)]

[2)] Voltage control via the measuring line can be activated, as an option, through parameter *Power Supply* **1186**. See chapter 7.4.3 "Power supply".

5.3.2 Cable assembly Resolver

Contact assignment BONFIGLIOLI VECTRON assembled cable for connection of Resolvers



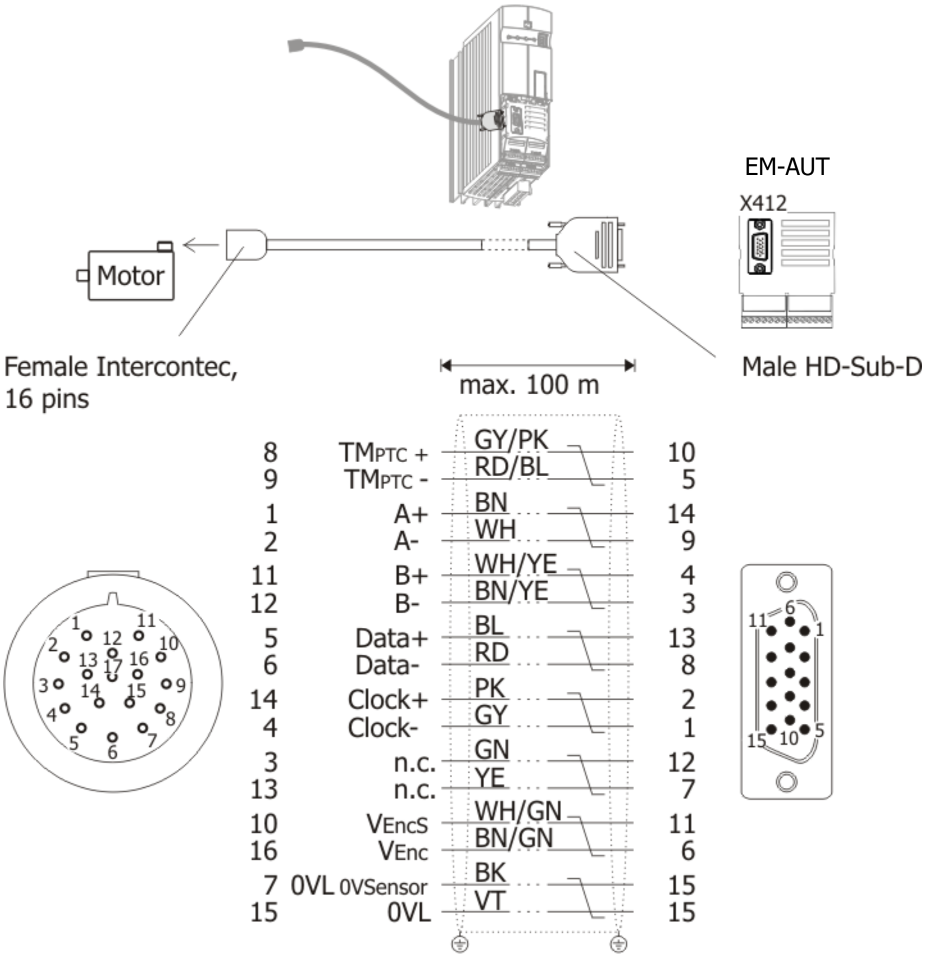
ANG_All_TD_ResolverMotorCabling_Resolver_BCR_BTD

BONFIGLIOLI VECTRON MDS assembled cable	
Cable	Resolver Flexible Wire
Cable size	4x0.25 mm² + 2x0.5 mm²
Length	3 m, 5 m or 10 m

The cable has open leads with wire ferrules at the inverter side. The wires for the encoder evaluation have to be connected to the expansion module terminal X410A. The wires for the motor temperature evaluation (PTC) have to be connected to the basic device terminal X210B.

5.3.3 Cable assembly EnDat 2.1

Contact assignment BONFIGLIOLI VECTRON assembled cable for connection of EnDat 2.1 encoders



8	$\mathrm{TM}_{\mathrm{PTC}}$ +	GY/PK	10
9	$\mathrm{TM}_{\mathrm{PTC}}$ -	RD/BL	5
1	A+	BN	14
2	A-	WH	9
11	B+	WH/YE	4
12	B-	BN/YE	3
5	Data+	BL	13
6	Data-	RD	8
14	Clock+	PK	2
4	Clock-	GY	1
3	n.c.	GN	12
13	n.c.	YE	7
10	$\mathrm{V}_{\mathrm{EncS}}$	WH/GN	11
16	$\mathrm{V}_{\mathrm{Enc}}$	BN/GN	6
7	0VL 0VSensor	BK	15
15	0VL	VT	15

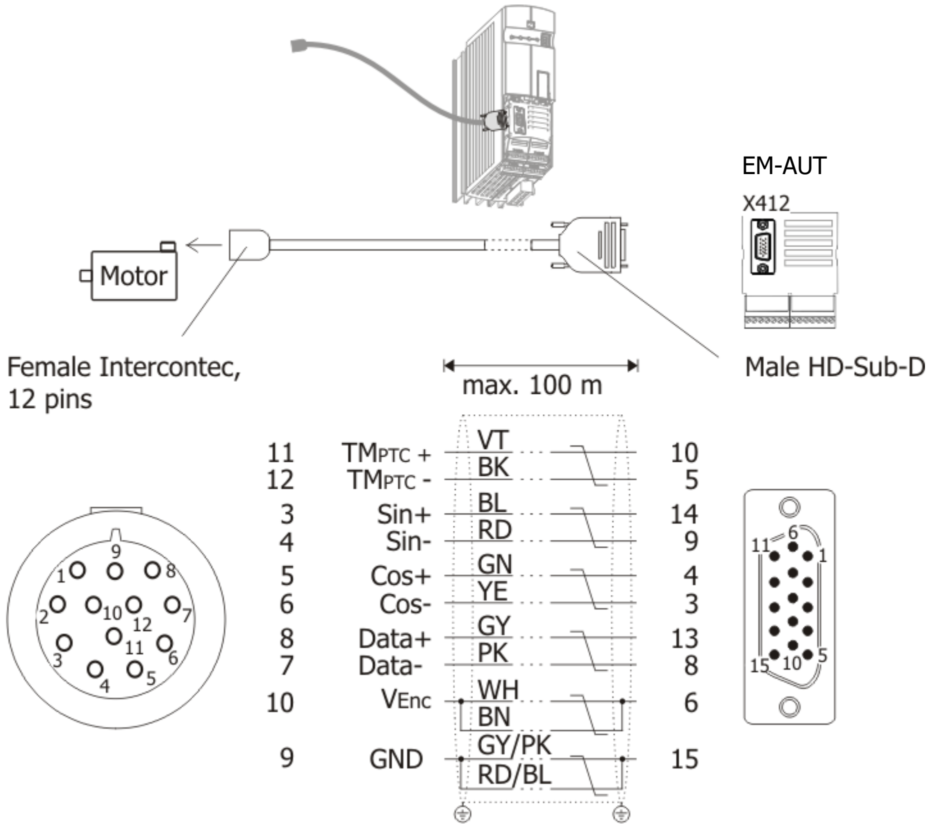
BONFIGLIOLI VECTRON MDS assembled cable	
Encoder cable	8 double-circuit twisted lines
Cable size	0.14 mm^2
Length	3 m, 5 m or 10 m

- Use PTC/KTY/PT1000 evaluations with safe insulation from motor winding according to EN 61800-5-1.
- Use shielded and twisted cables.
- Install encoder cable separate from motor cable.
- Connect the shield of the encoder line properly on both sides.

BONFIGLIOLI VECTRON MDS recommends using the pre-assembled cables for synchronous motor types BMD, BCR and BTD.

5.3.4 Cable assembly Hiperface

Contact assignment BONFIGLIOLI VECTRON assembled cable for connection of Hiperface encoders



BONFIGLIOLI VECTRON MDS assembled cable	
Encoder cable	8 double-circuit twisted lines
Cable size	0.14 mm^2
Length	3 m, 5 m or 10 m

- Use PTC/KTY/PT1000 evaluations with safe insulation from motor winding according to EN 61800-5-1.
- Use shielded and twisted cables.
- Install encoder cable separate from motor cable.
- Connect the shield of the encoder line properly on both sides.

BONFIGLIOLI VECTRON MDS recommends using the pre-assembled cables for synchronous motor types BMD, BCR and BTD.

5.3.5 EM-AUT: Speed Sensor Power supply

Encoder power can be supplied in several ways. Depending on the consumers connected, there are different encoder power supply requirements. The encoders connected at the X410 terminal are supplied via X410B.1 and X410A.7. The encoders connected at the X412 terminal are supplied via the X412.6 and X412.15 terminals.

NOTICE

Incorrect data!

If the encoder is not supplied with power before the inverter starts or after the inverter shuts down, it may produce faulty readings.

- Ensure that the encoder is supplied with sufficient power before the inverter starts evaluating and after the inverter shuts down to ensure the faulty readings are prevented.

Generally, there are three different application types:

- Low power demand (< 0.5 W) and power supply ≤ 12 V:
  ➔ Internal power supply.
- Medium power demand (0.5... 2 W) and power supply ≤ 12 V:
  ➔ Power supply to be looped via X410.
- High power demand (> 2 W) or power supply > 12 V:
  ➔ Connect encoder directly to external power supply.

Encoders with high power demand (> 2 W) or voltage higher than DC 12 V must be connected to an external power supply directly.

5.3.5.1 Internal power supply

Encoders with low power consumption (< 1 W) can be supplied, in most cases, by the internal power supply unit.

For voltage supply via X412 activate the *Power supply* **1186** with either setting "`1 - internal`" or "`5- internal, sense`".

See chapter 7.4.3 "Power supply" and .

The voltage value for the supply voltage via X412 can be set up via parameter *Supply voltage* **1187**. See chapter 7.4.4 "Supply voltage".

The voltage supply via X410B.1/X410A.7 is fixed at DC 5 V.

If power supply of the encoders is done via the internal power supply, a total power of 4 W is available for all consumers connected to digital, analog interfaces and encoder interfaces. This includes all interfaces.

For each individual encoder a maximum of 1 W is available.

5.3.5.2 Direct connection of external power supply to the encoder

CAUTION

Faults and system shutdown

Setting *Power supply* **1186** to modes with a "sense" line while leaving the power supply terminals unconnected will result in faults and subsequently in system shutdown.

- Do not set *Power supply* **1186** to modes with a "sense" line in this configuration.

Encoders with high power demand (> 1 W) or voltage higher than DC 12 V (X412) or DC 5 V (X410) must be connected to an external power supply directly.

When using encoder input 3 (X412):

- Set parameter *Power supply* **1186** to "`1-internal`". See chapter 7.4.3.

This setting must be used for proper function of the evaluation. However, the power supply terminals do not have to be connected but should remain open.

The voltage level set in *Supply voltage* **1187** is irrelevant when the terminal is open. See chapter 7.4.4 "Supply voltage".

6 Commissioning the expansion module

For basic information on commissioning refer to the operating instructions document "**VEC1105**".

The Ethernet/IP module is already mounted on the frequency inverter, ex works. No further installation by user is necessary.

For first commissioning, you should be familiar with the followings steps and the described functions:

Selection of device control *Local/Remote* **412:** Chapter 12

Commissioning of device functions via PLC:

- Network settings: Chapter 6.1
- Setting the process data: Chapter 8
- Fault Reaction: Chapter 9.6
- Resetting errors: Chapter 6.3.3
- Parameter access: Chapter 9.9

Setting reference values:

- Reference speed in speed-controlled configuration x10, x11, x15, x16, x30, x60: Chapter 12.3
- Reference value in position configuration x40: Chapters 11 and 12.4
  - Velocity mode vl: Chapter 12.4.1
  - Profile velocity mode pv: Chapter 12.4.2
  - Profile position mode: Chapter 12.4.3
  - Homing mode: Chapter 12.4.4
  - Table travel record mode: Chapter 12.4.5
  - Move Away from Limit Switch: Chapter 12.4.6
  - Mode of Operation change: Chapter 9.9.5.12

Diagnosis: Chapter 8.9

6.1 Protocol Selection CANopen or Systembus

With Parameter *CAN Interface* **276** the CAN interface can be set to "CANopen" or "Systembus".

CAN Interface	
1 -	CAN-Systembus
2 -	CANopen

It is not possible to use the CANopen® communication and Systembus communication simultaneously.

WARNING

Communication interrupted

With the Selection "CANopen" the functionality "EthernetIP" is completely deactivated, an running EthernetIP communication is interrupted.

- Simultaneous operation of EthernetIP® and CANopen® is possible.
- Simultaneous operation of EthernetIP® and Systembus is possible.
- Simultaneous operation of Systembus **and** CANopen® is **not** possible.

- Exercise caution when changing the communication protocol.

6.2 Network configuration for Ethernet/IP

The IP-Address and other network parameters can be set in the *Communication – Ethernet* branch of the VPlus Parameter window or with the VPlus TCP/IP Configuration Utility:

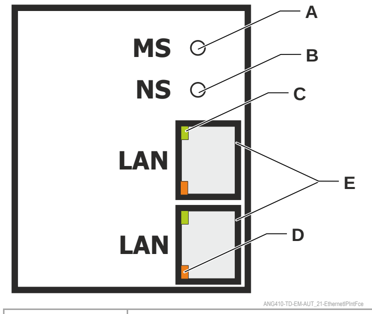
Parameter		Setting	
No.	**Description**	**Min.**	**Max.**
1432	IP-Address	0.0.0.0	255.255.255.255
1433	Netmask	0.0.0.0	" " " "

1434	Gateway	0.0.0.0	" " " "
1435	DNS Server	0.0.0.0	" " " "
1436	DHCP Option	0 - Disabled	1 - Enabled

In order to configure the network connection using the TCP/IP Utility or the VPlus software, ensure that the inverter is connected correctly to the network and is supplied with control voltage.

6.2.1 Connector assignment

The Ethernet/IP module is connected to the PLC or switch using RJ45 connectors (LAN).



A	Module Status LED
B	Network Status LED
C	Activity indication
D	Link indication
E	RJ45 Connectors

Figure 6-1: Ethernet connector

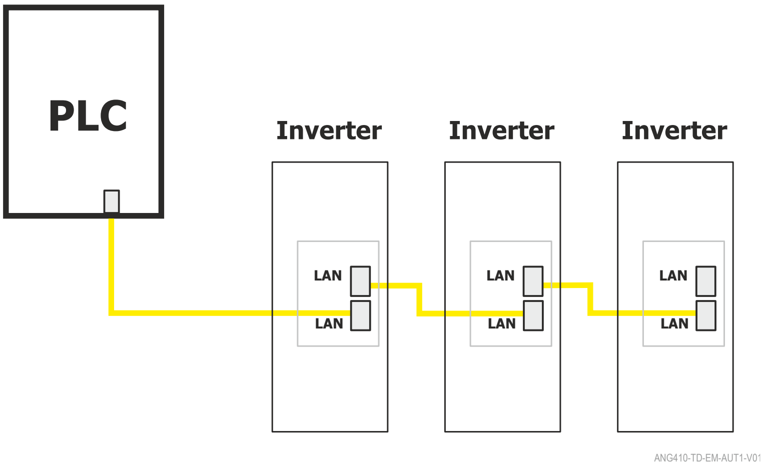


Figure 6-2: Ethernet/IP topology

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

LED status indicators

The module status LED (A) indicates the current status of the module.

LED Status	Module status
Off	Module is off.
On	Module is on and running.

The Network Status LED (B) indicates the current status of the connection.

LED Status	Module status
On	Module has no Ethernet connection.
Flashing	Module has Ethernet connection, no cyclic exchange of data is taking place.
Off	Exchange of Cyclic data is taking place.

The transmission speed of the Ethernet/IP communication module Ethernet/IP is 10/100 Mbit/s and the maximum cable length between two nodes is 100 m.

Module Status Indicator

The indication of module status is a single (red/green) LED indicator (C and D) that represents the state of the module.

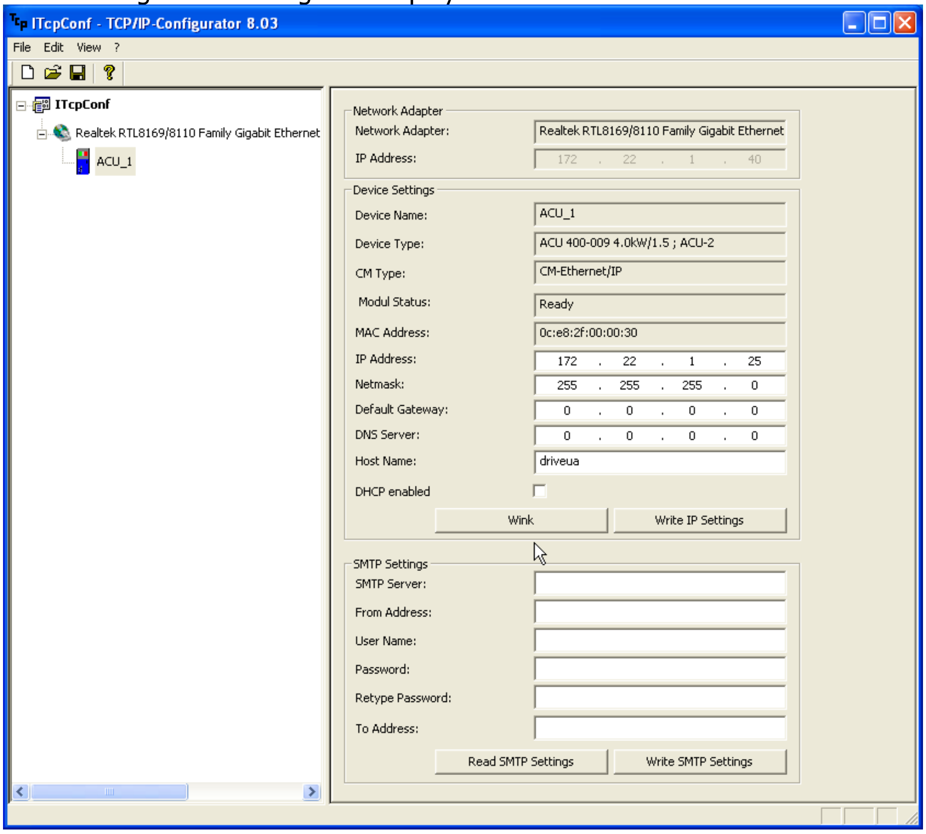
The following states are defined:

Indicator State	Meaning	Description
Steady Off	No power	No power is supplied to the device.
Steady Green	Device operational	The device is operating correctly.
Flashing Green	Standby	The device has not been configured.
Flashing Red	Minor fault	The device has detected a recoverable minor fault. **NOTE:** An incorrect or inconsistent configuration would be considered a minor fault.
Steady Red	Major fault	The device has detected a non-recoverable major fault.
Flashing Green / Red	Self-test	The device is performing its power-up testing.

6.2.2 TCP/IP Configuration Utility

- Ensure your control device (PC) is connected to the Ethernet network.
- Start the TCP/IP Configuration Utility.

The configuration dialogue is displayed as follows:



Network Adapter

Network Adapter: Realtek RTL8169/8110 Family Gigabit Ethernet

IP Address: 172 . 22 . 1 . 40

Device Settings

Device Name: ACU_1

Device Type: ACU 400-009 4.0kW/1.5 ; ACU-2

CM Type: CM-Ethernet/IP

Modul Status: Ready

MAC Address: 0c:e8:2f:00:00:30

IP Address: 172 . 22 . 1 . 25

Netmask: 255 . 255 . 255 . 0

Default Gateway: 0 . 0 . 0 . 0

DNS Server: 0 . 0 . 0 . 0

Host Name: driveua

DHCP enabled ☐

SMTP Settings

SMTP Server:

From Address:

User Name:

Password:

Retype Password:

To Address:

Figure 6-3: TCP/IP Configuration

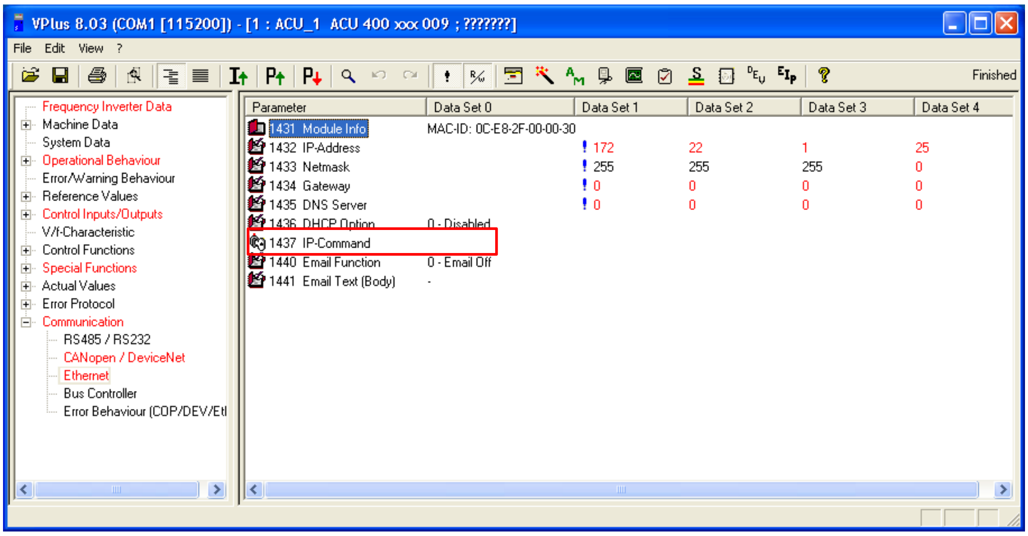
- Select the Ethernet interface as required.
- Select the inverter as required.
- Enter the network settings according to your required configuration.
- Apply the settings with the *Write IP Settings* command button.

The Wink Command can be sent from the TCP/IP Configuration Utility by clicking the "*Wink*" button. This will cause the Inverter's LEDs (green and red) to blink for 10 seconds.

6.2.3 VPlus

- Ensure your control device (PC) is connected to the Ethernet network.
- Start the VPlus software Utility.
- Select the Ethernet interface as required.
- Select the inverter as required.

The parameter setting dialogue is displayed as follows:


VPlus 8.03 (COM1 [115200]) - [1 : ACU_1 ACU 400 xxx 009 ; ???????]
File
Edit
View
?
Finished

- Frequency Inverter Data
- Machine Data
- System Data
- Operational Behaviour
- Error/Warning Behaviour
- Reference Values
- Control Inputs/Outputs
- V/f-Characteristic
- Control Functions
- Special Functions
- Actual Values
- Error Protocol
- Communication
  - RS485 / RS232
  - CANopen / DeviceNet
  - Ethernet
  - Bus Controller
  - Error Behaviour (COP/DEV/Et

Parameter	Data Set 0	Data Set 1	Data Set 2	Data Set 3	Data Set 4
1431 Module Info	MAC-ID: 0C-E8-2F-00-00-30				
1432 IP-Address		! 172	22	1	25
1433 Netmask		! 255	255	255	0
1434 Gateway		! 0	0	0	0
1435 DNS Server		! 0	0	0	0
1436 DHCP Option	0 - Disabled				
1437 IP-Command					
1440 Email Function	0 - Email Off				
1441 Email Text (Body)	-				

Figure 6-4: TCP/IP Configuration parameter setting VPlus

- Search for the parameters listed in the table above or select the branch *Communication –> Ethernet* in the parameter window of the VPlus dialogue.
- Enter the network settings according to your required configuration.

NOTICE

Risk of component damage!

If a maximum number of write cycles for the EEPROM is exceeded, it will be destroyed.

The values are entered automatically in the EEPROM of the controller. However, only a limited number of write cycles is permissible for the EEPROM (approx. 1 million cycles).

- If cyclic writing of data is required, use the RAM.

- Apply the chosen settings via parameter *IP-Command* **1437** = 0 (Apply). This command sends the settings to the Ethernet/IP Module and writes them to the inverter's EEPROM.

If settings have been changed but **not applied**, you can reload the last settings saved via parameter *IP-Command* **1437** = 1 (Reload).

6.2.4 Module Info

EM-Software version **016** shows the Software version of the EM-AUT module.

Parameter *Module Info* **1431** shows basic Ethernet based module data:

MAC-ID:	unambiguous MAC-ID
Sno:	Serial number
HW:	Hardware version
SW:	Ethernet Software version

6.3 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.

Note, that only small deviations of the CPU clock frequencies between devices (i.e. different CPU Quartz clock frequencies) of ±1 ‰ can be compensated.

Synchronization via CANopen®:

If CANopen is used, synchronization to a CANopen PLC can be turned on or off. Synchronization is done via CANopen SYNC telegrams.

Synchronization via systembus:

If CANopen is used simultaneously with systembus, synchronization can be done either on CANopen, systembus or turned off. Synchronization is effected through systembus SYNC telegrams or systembus RxPDO telegrams.

If the operating system is synchronized via CANopen, the CANopen master must support the CANopen synchronization mechanisms.

Synchronization via Ethernet/IP is not possible.

OS_SyncSource 1452		
Operation mode		**Function**
0 -	Auto	The synchronization source is selected automatically by the frequency inverter. **Factory setting.**
1 -	CANopen	The OS is synchronized via CANopen®.
2 -	Systembus	The OS is synchronized via Systembus.
3 -	Ind. Ethernet Module	The OS is synchronized via Ethernet Module
4 -	Synchronised Ind. Ethernet Module	The OS is synchronized via synchronized Ethernet Module e.g. EthernetIP
99 -	Off	The OS is not synchronized with other devices.

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

Fieldbus active	Systembus active	Synchronization
Yes	Yes	Synchronisation via Fieldbus
Yes	No	
No	Yes	Synchronization via Systembus
No	No	No Synchronization activated.

1453 *OS SyncSource Act* shows the active Synchronization source.

6.3.1 Synchronization via CANopen

Status "Synchronization via CANopen active" is identified via parameter setting **387** *CAN Node Number* > `1` and a running synchronous PDO.

The parameter **1451** *OS Synctime* can be used to shift the point of the synchronization inside of 1 ms. When you experience noises from a motor, shifting the CANopen *OS Synctime* can result in a better behavior.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
1451	OS Synctime	700 us	900 us	800 us

6.3.1.1 Synchronization via Systembus

Status "Synchronization via systembus active" is identified via parameter setting **900** *Node ID* > `1`. In addition, parameter **1180** *Synchronization* must be set to `SYNC` or `RxPDO`.

The source of the operating system (OS) synchronization is set via **1180** *Operation mode*. This defines the Sync event (RxPDO or SYNC telegram), which will be used for synchronization of PDOs:

930 *TxPDO1 Function*	**936** *RxPDO1 Function*
932 *TxPDO2 Function*	**937** *RxPDO2 Function*
934 *TxPDO3 Function*	**938** *RxPDO3 Function*

Synchronization Operation mode 1180

Operation mode	Function
0 - Off	Synchronization via systembus is deactivated. **Factory setting.**
1 - RxPDO1	Synchronization via systembus is activated via RxPDO1.
2 - RxPDO2	Synchronization via systembus is activated via RxPDO2.
3 - RxPDO3	Synchronization via systembus is activated via RxPDO3.
10 - SYNC	Synchronization via systembus is activated via SYNC.

6.3.1.2 Settings for electronic gear in configuration x40

If the function "electronic gear" of the positioning in ANG (configuration x40) is used in a slave, synchronization via SYNC or RxPDO1 must be set via systembus. Please check the following settings:

Use of RxPDO

A Master Identifier must correspond to the Slave Identifier.

Master	Slave
925 *TxPDO1 Identifier* **926** *TxPDO2 Identifier* **927** *TxPDO3 Identifier*	**924** *RxPDO1 Identifier*
930 *TxPDO1 Function* **932** *TxPDO2 Function* **934** *TxPDO3 Function*	**936** *RxPDO1 Function* = `1 – controlled by SYNC` (recommended)
	1180 *Operation mode* = 1- RxPDO

Use of SYNC

The Master Sync Identifier must correspond to the Slave Sync Identifier (e.g. 0 → Predefined Set 0x80 = `128`).

Master	Slave
	936 *RxPDO1 Function* = `1 – controlled by SYNC` (recommended)
918 *Sync Identifier* **919** *Sync Time*	**918** *Sync Identifier* **1180** *Operation mode*= 10-SYNC

Operation mode **1180** ensures synchronization of the operating systems of different devices and must be set up in configuration x40 in one of the two ways described.

RxPDO1 Function **936** should be set to "1 – controlled by SYNC" in order to synchronize the master position with the OS in the slave. Although this setting is optional, BONFIGLIOLI VECTRON recommends setting this parameter accordingly.

6.3.2 SDO Error code table

If an error occurs in reading or writing, the server SDO of the frequency inverter replies with the SDO abort message.

Error codes

Abort code high	Abort code low	Description as per CANopen®	Product-specific allocation
0x0601	0x0000	Unsupported access to an object	Parameter cannot be written or read
0x0602	0x0000	Object does not exist	Parameter does not exist
0x0604	0x0047	General internal incompatibility in the device	Data sets differ
0x0606	0x0000	Access failed due to a hardware error	EEPROM Error (R/W/checksum)
0x0607	0x0010	Data type does not match	Parameter has a different data type

Error codes			
Abort code high	**Abort code low**	**Description as per CANopen®**	**Product-specific allocation**
0x0607	0x0012	Data type does not match or length of Service telegram too big	Parameter has a different data type or telegram length not correct.
0x0607	0x0013	Data type does not match or length of Service telegram too small	Parameter has a different data type or telegram length not correct.
0x0609	0x0011	Subindex does not exist	Data set does not exist
0x0609	0x0030	Value range of parameter exceeded	Parameter value too large or too small
0x0609	0x0031	Value of parameter written too high.	Parameter value too large
0x0609	0x0032	Value of parameter written too low.	Parameter value too small
0x0800	0x0020	Data cannot be transmitted or saved	Invalid value for operation
0x0800	0x0021	Data cannot be transferred because of local control	Parameter cannot be written in operation
0x0800	0x0022	No data transfer because of present device state	NMT state machine is not in correct state

6.3.3 Resetting errors

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

– When using control via parameter *Local/Remote* **412** = Statemachine:

- Set bit 7 in 0x6040 Control word = 0x0080.
- Press the STOP button of the control panel.

– Resetting by pressing the STOP button is only possible, if Parameter *Local/Remote* **412** permits control via the control panel.

- Reset via parameter *Error Acknowledgement* **103** which is assigned a logic signal or a digital input.

– Resetting by digital signal is only possible, if Parameter *Local/Remote* **412** permits this or if an input with the addition (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).

6.4 Scope sources

For the Vplus Scope function, the following sources are available for diagnosis:

Operation mode		**Function**
731 -	B: Sync. OS <→ Sysbus Ok	1 = Synchronization OS to systembus OK, 0 = Synchronization OS to systembus not OK
852-	SysBus SYNC time [us]	Represents the synchronization time cycles. Should show the set SYNC time or TxPDO of the transmitting master.
853	SysBus SYNC position 1ms Task [us]	Represents the synchronization time within 1 ms. Should be constant with minor deviations.
854-	B: Sync. OS <→ CANopen Ok	1 = Synchronization OS to CANopen OK, 0 = Synchronization OS to CANopen not OK
848-	SYNC time [us]	Represents the synchronization time cycles. Should show the SYNC time of object 0x1006.
849-	CANopen SYNC position 1ms Task [us]	Represents the synchronization time within 1 ms. Should be constant with minor deviations.

For more information on VPlus, see chapter 17.

7 Commissioning the encoder interface

This chapter describes how the different encoder types are commissioned.

General Information

Following feedback systems are supported:

Feedback System	Speed Sensor 1 (X210)	Speed Sensor 2 (X410)	Speed Sensor 3 (X412)
HTL	Possible	Not possible	Not possible
Resolver	Not possible	Possible	Not possible
TTL (Tracks A, B, R)	Not possible	Possible	Possible
TTL (Tracks A, B)	Not possible	Possible	Possible
SinCos (Tracks Sin, Cos)	Not possible	Not possible	Possible
Hiperface	Not possible	Not possible	Possible
EnDat 2.1 (+SinCos tracks)	Not possible	Not possible	Possible
SSI (with or without SinCos or TTL tracks)	Not possible	Not possible	Possible
Output TTL Encoder emulation (A, B, R)	Not possible	Possible	Not possible

Each Speed Sensor Interface can be configured independently of the other Speed Sensor Interfaces. Each Speed Sensor Interface can be configured with one Feedback system from the following table:

Speed Sensor 1 (X210)	Speed Sensor 2 (X410)	Speed Sensor 3 (X412)
HTL or no Usage	Resolver or TTL or Output TTL Encoder emulation or no Usage	SinCos or or Hiperface or EnDat 2.1 or SSI or no Usage

Speed Sensor 1 is part of the Basic device and therefore described in the ANG Operating instructions. See the Operating Instructions document **VEC1105**.

The Speed Sensor 3 interface supports, with systems Hiperface, EnDat 2.1 and SSI, both Singleturn and Multiturn encoders. Multiturn encoders must be configured as such in order to avoid unwanted effects.

The internal resolution of encoder information is 32 bits, 16 bits for the position in one turn and 16 bits for the number of turns. Encoders with other properties will be converted to this format internally.

In the case of motor encoders with a Multiturn portion of more than 16 bits, clear identification of the position in the frequency inverter is not guaranteed.

In the case of motor encoders with a multiturn portion of less than 16 bits, the free bits are filled up to 16 bits and managed in a fail-safe manner, whereby the MSB represents the sign.

Example: An encoder has a multiturn portion of 13 bits. 3 bits are managed additionally in the inverter, thus 8 (=2^3) overflows of the multiturn portion are recognized.

This information may be lost in some situations, if the DC link is discharged very quickly due to external conditions.

In the case of usage in positioning applications (configuration x40), the incremental encoder or resolver can be used for the reference system directly in user units [u]. Using gear factors, a gear transmission between the encoder and the travel distance can be considered.

The input data of the encoder is evaluated via the reference systems. The evaluated parameters (e.g. motor frequency, drive speed in rev/s, position in rev.) are available for diagnosis via actual value parameters, see chapter 15.

- Check the power demand of the encoder to be connected. The internal power supply unit can only supply a maximum total of 4 W for all consumers connected. See chapter 5.2.5 "EM-AUT: Speed Sensor Power supply".
- Install encoder cables separately from motor cables to minimize interference.
- Upon first commissioning and during operation, make sure that the encoder and other electrical components can acclimatize in order to prevent condensation and resulting malfunction. The manufacturer recommends storing the device near the operation site for 24 hours before commissioning.

Information on use

After mains on, encoders may require an initialization, depending on the encoder type. This may take up to 5 seconds. This delay can be eliminated by powering the basic device and the encoder using an external DC 24 V supply.

When the encoder or motor (including motor encoder) are replaced, typically, re-calibration will be required, in order to obtain the absolute position. This applies to the encoder-internal value (depending on the encoder type used, this value cannot be changed), position angle *Offset* **1188** and, in positioning applications (configuration x40), referencing *Home-Offset* **1131**. After encoder replacement, always check the position angle *Offset* **1188** and carry out a referencing operation in the case of positioning applications (configuration x40).

When an absolute value encoder is used, referencing is not required after encoder or motor replacement to ensure **correct function of the ANG device**. Adjustments of *Home-Offset* **1131** are applied directly.

After encoder or motor replacement, **correct function of the system** is achieved by performing a referencing operation or offset adjustment.

The signals provided by the encoder are used in the expansion module for various plausibility checks. This makes the system more fail-safe and less prone to unwanted interference.

During operation, the encoders and communication with the encoder are monitored. Critical conditions are reported via device errors. Most error evaluations will only be performed when the power output stage is activated.

WARNING

Danger of system failure!

Changing the value while the system is in operation or standstill can result in significant failures of the system (independent from the motor system).

Some absolute value encoder types allow to "nullify" or re-calibrate the position transmitted by the encoder.

- Do not use this function in synchronous motors, as this will change the commutation angle for the synchronous motors via *Offset* **1188** and correct speed control is not guaranteed.

CAUTION

Inconsistent encoder values

Via parameter *Change Sense of Rotation* **1199**, you can change the direction of rotation of the motor system. In the case of absolute value encoders, a modification of **P.1199** will result in an unwanted modification of actual value.

- While modifying the value of **P.1199**, slave drives in an electronic gear must be switched off.

7.1 Speed sensor input 2 Resolver / TTL Encoder

The six speed sensor inputs can be set via the parameter *Operation mode* **493** and selection of the corresponding operation mode for the evaluation of a two-channel speed sensor (TTL incremental speed sensor) with reference impulse or via a resolver.

Operation mode		Function
0 -	Off	Speed measurement not active
4 -	Quadruple evaluation[1]	Two-channel speed sensor with recognition of direction of rotation via track signals A and B; four signal edges are evaluated per division mark.
104 -	Quadruple evaluation inverted	Like operation mode 4; the actual speed value is inverted (alternatively to exchanging the track signals).
1004 -	Quadruple evaluation with reference impulse	Two-channel speed sensor with recognition of direction of rotation via track signals A and B; four signal edges are evaluated per division mark. The reference impulse is used for speed sensor monitoring.
1104 -	Quadruple evaluation inverted with reference impulse	Like operation mode 1004; the actual speed value is inverted (alternatively to the exchange of the track signals).
10000 -	Resolver	Two signal Resolver via signals SIN and COS; four signal sinusoidal are evaluated.
10100 -	Resolver Inverted	Like operation mode 10000; the actual speed value is inverted.

[1] The speed sensor inputs are suitable according to specification RS-422A / RS-485 for a 5 V push-pull signal.

7.1.1 TTL incremental encoder – Division marks speed sensor 2

The number of increments of the connected speed sensor can be parameterized via the parameter *EC2 Division marks* **494.** Select the number of division marks of the speed sensor according to the speed range of the application.

The maximum number of division marks S_{max} is defined by the frequency limit of f_{max} = 300 kHz in the speed sensor inputs track A and track B.

$$S_{max} = 300000\,\mathrm{Hz} \cdot \frac{60\,\mathrm{s/min}}{n_{max}}$$

n_{max} = Max. speed of the motor in RPM

To ensure a good true running of the drive mechanism, a sensor signal must be evaluated at least every 2 ms (signal frequency f = 500 Hz). The minimum number of division marks S_{min} of the incremental speed sensor for a required minimum speed n_{min} can be calculated from this requirement. The evaluation of four signal edges per mark is firmly defined in the function of speed sensor 2.

$$S_{min} = 500\,\mathrm{Hz} \cdot \frac{60\,\mathrm{s/min}}{A \cdot n_{min}}$$

n_{min} = Min. speed of the motor in RPM
A = 4 (quadruple evaluation)

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Default**
494	Division marks speed sensor 2	1	8192	1024

7.1.2 Resolver evaluation

The resolver input is used for evaluating the position information of a resolver.

If the nº. of resolver pole pairs > 1, the measured electric angle runs through the range of 0º .. 360º several times during one mechanical revolution.

For the detection of the position angle of the rotor at synchronous motor, the ratio of the nº. of motor pole pairs must be an integer.

The nº. of pole pairs of the resolver can be adjusted via parameter *RES Nº. of Pole Pairs* **381.**

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Default**
381	RES Nº. of Pole Pairs	1	7	1

Offset of the Resolver

In order to enable the start of a synchronous machine, the absolute position of the rotor must be known. This information is required in order to actuate the stator windings in the right order depending on the position of the rotor. The position of the rotary field in the synchronous machine must be controlled in order to obtain a continuous movement of the rotor. During first commissioning, the position of the rotor winding of the resolver is adjusted to the rotor displacement angle of the synchronous motor by adjusting the offset. For operating a synchronous machine with resolver, the offset must be adjusted in order to obtain perfectly true running and a maximum torque.

The correct *RES Offset* **382** is adjusted when the *flux-forming voltage* **235** reaches the value 0 (approximately) while the motor is turning.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Default**
382	RES Offset	-360.0°	360.0°	0

BONFIGLIOLI servo motors of types BMD, BCR and BTD are set up with an offset of 0.0° ex works. For these motors an offset adjustment is not necessary.

Automatic setup

WARNING

Material damage! Personal Injury!

In certain circumstances, the motor speed may reach high values. This can result in personal injury and damage to the machine.

- Decouple the motor from the load to avoid possible hazards.
- Lock or barrier the motor shaft and make sure that no loose parts can be moved in an uncontrolled manner due to a sudden acceleration of the motor shaft.

- Ensure that the motor commissioning is finished before the automatic offset setup is executed.
- Set up additionally the *RES N°. of Pole Pairs* **381** of the resolver.

If a wrong direction is caused by an incorrect wiring, this is typically detected during the Automatic Offset setup.

- Before starting the setup, always ensure that the wiring is correct.

The offset can be determined and adjusted as follows:

- Set up parameter *SETUP Selection* **796** to "550 – Para-Ident. Offset, DS0" if the auto setup should be done for all data sets.

  For single data sets use the following settings.

  After the start via *SETUP Selection* **796** the Auto-Tuning of the Resolver is started. The status of the Auto-Tuning is displayed by *SETUP Status* **797**.
- For the Auto-Tuning a controller release via STO must be done. Switch the STO inputs when *SETUP Status* **797** shows "STO".

If a correct Resolver Offset was detected, this value is set up automatically in *Offset* **382**. Additionally the device executes a Reset. The Resolver is now tuned to the motor.

If U, V and W are connected correctly, the "clockwise" sense of rotation is defined as seen on the front of the motor shaft in accordance with DIN EN 60034-8.

Possible errors and corrections:

If during the auto-tuning error "F1420" shows, this is an indication of an incorrect direction of rotation of the resolver in comparison to the motor.

Execute one of the following actions to adjust the direction of rotation of the encoder to the motor:

- Replace two motor phases, e.g. U and V. Note the motor direction of rotation.
- Invert the direction of rotation of the resolver via parameter *Operation mode* **493**.
- Invert the direction of rotation of the resolver by exchanging Sin+ and Sin-.

Manual setup

The offset can be determined and adjusted as follows:

- During first commissioning "SEtUP" will be displayed in the control unit. Press ESC to stop this operation. The guided commissioning ("SETUP") is performed after adjusting the offset.
- Open the parameter menu "PARA" and enter the machine data indicated on the type plate or the data sheet of the motor.
- Adjust parameter *RES N°. of Pole Pairs* **381** to the number of pole pairs of the resolver.

Before adjusting the offset, take the following **safety precautions**:

- Disable the frequency inverter via STO (controller release).
- If possible, uncouple the motor from the load so that the motor shaft turns freely. If installed, release the mechanical brake.

If uncoupling is not possible, make sure that the motor is loaded as little as possible.

WARNING

Material damage and personal injury!

In certain circumstances, the motor speed may reach high values. If the motor is not uncoupled from the load, personal injury and material damage may result.

- To avoid such damage, make the following settings in any case.

- Set the max. permissible output frequency of the frequency inverter to a low frequency value via parameter *Switch-Off Limit* **417**. Select the frequency value such that uncontrolled acceleration of the motor ("overspeeding") is detected at an early stage. This limitation is necessary in order to avoid injury and material damage.
- Set parameter *Current Limit* **728** of the speed controller to a low current value (e.g. 10% of the rated motor current). In this way it is made sure that there are no excessive currents of the offset is set incorrectly.

WARNING

Unintended Starting!

If the Drive is supplied with voltage, it can suddenly start. This may lead to personal and material damage.

- Prior to starting the manual adjustments, disconnect the drive from voltage supply.
- Comply with the five safety rules.
- If possible, wear protective clothing.

- Turn motor shaft manually. Check the sense of rotation of the resolver via the actual value of parameter *Frequency Speed Sensor 2* **219**. In the case of a clock-wise rotation of the motor shaft, positive values are displayed for the actual frequency value. If the displayed sense of rotation does not correspond to the actual sense of rotation, change the connections SIN+ and SIN- at socket X410A of the frequency inverter.

The *Offset* **382** must be between 0° and 360°, divided by the number of motor pole pairs. If the number of resolver pole pairs is higher than 1, the possible range is between 0° and the max. offset.

$$\text{Max. Offset} = \frac{360\,^{\circ}}{\text{number of motor pole pairs / number of resolver pole pairs}}$$

If the adjusted value is changed by the max. offset, this does not affect the *flux-forming voltage* **235**.

- Adjust a low reference speed value (approx. 10% lower than the *Switch-off Limit Frequency* **417**), and enable the frequency inverter via digital input S1IND (controller release) and S2IND (start clock-wise operation) in order to accelerate the motor.
- If an overcurrent is detected or a fault message is issued due to an overload, the guided commissioning (setup) will start first. Confirm the machine and resolver data. After completion of the guided commissioning, adjust the parameter *Limit Current* **728** to a low value again because this value was overwritten during the guided commissioning.

Depending on the behavior of the motor after start, carry out the following steps:

Motor does not turn, or the motor shaft only turns to a new position and stops again:

- Check if the parameters *No. of Pole Pairs* **373** for the motor and *No. of Pole Pairs* **381** for the resolver are set correctly.

If these values are adjusted correctly, take the following measures complying with the safety instructions.

> **WARNING**
>
> The mains, direct voltage and motor sockets can be live with dangerous voltage after disconnection of the frequency inverter. Work only on the device after a waiting period of some minutes until the DC link capacitors have discharged.

- Before electrical installation work, de-energize the frequency inverter and take appropriate precautions to make sure it is not re-energized unintentionally. Make sure that the frequency inverter is de-energized.
- Exchange two motor phases (e.g. U and V) at the frequency inverter sockets because the senses of rotation of the motor and the resolver do not correspond to each other.
- Switch on the power supply again.
- As described above, adjust a low speed reference value and start the motor.

If the motor does not start despite the phase exchange:

- Increase the parameter value for *Offset* **382** by 90°, divided by the no. of motor pole pairs.

If the motor still does not turn, exchange the two motor phases (e.g. U and V) again.

The motor turns and accelerates until it reaches the *Frequency Switch-Off Limit* 417:

- Check the resolver lines and check the resolver connection contacts.
- In the case of fault message "Overfrequency" F1100: increase the parameter value for *Offset* **382** by 180°, divided by the no. of motor pole pairs.

If the motor turns at the adjusted speed and in the right direction, carry out the fine adjustment of the offset:

- Adjust the parameter value for *Offset* **382** in small steps (e.g. 2.5°) until the *flux-forming voltage* **235** is approximately 0.
- In case the flux-forming voltage deviates from 0 significantly, adjust the offset in bigger steps.
- In the case of a positive flux-forming voltage: increase the offset.
- In the case of a negative flux-forming voltage: reduce the offset.
- Adjust parameters *Frequency Switch-Off Limit* **417** and *Current Limit* **728** to the required values.
- Repeat the **fine adjustment** of the offset at 50% of the rated frequency.

This completes the offset adjustment.

- Start the guided commissioning. This is required for optimum current control.

7.1.3 Filter time constant speed sensor 2/Resolver

Via parameter *Encoder 2/Resolver: Filter time constant* **1194**, you can filter the speed of the encoder 2. This filter can be applied in cases where the encoder fluctuates (in example for mechanical reasons). Change the value in small steps and check each result and do not to change the value in too big steps.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
1194	Encoder 2/Resolver: Filter time constant	0 us	32000 us	0 us

7.1.4 Resolver Inverted Evaluation mode

If a synchronous motor which is not from BONFIGLIOLI should be connected to the resolver input it can be necessary to change the sign of the sinus track. This can be set via parameter *Evaluation Mode* **492**.

Evaluation Mode **492**		Function
0 -	Bonfiglioli	Factory setting. For Bonfiglioli synchronous motors.
1 -	inverted	The sign of the sinus track is changed.

7.1.5 Gear factor speed sensor 2

If the speed sensor is coupled to the motor via one or more gears, the transmission ratio between the motor and the encoder must be configured via *EC2 Gear Factor Numerator* **513** and *EC2 Gear Factor Denominator* **514**.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
513	EC2 Gear Factor Numerator	-300.00	300.00	1.00
514	EC2 Gear Factor Denominator	0.01	300.00	1.00

$$\frac{\text{Revolution s of the Motor shaft}}{\text{Revolution s of the EC2 encoder shaft}} = \frac{EC\;2\;Gear\;Factor\;Numerator\,\mathbf{513}}{EC\;2\;Gear\;Factor\;Denominator\,\mathbf{514}}$$

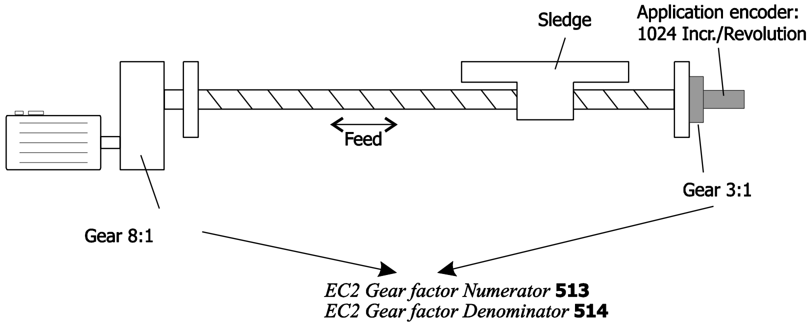
NOTICE

Gear factors *EC2 Gear Factor Numerator* **513** and *EC2 Gear Factor Denominator* **514** must always be referred to the motor.

Gear factors *EC2 Gear Factor Numerator* **513** and *EC2 Gear Factor Denominator* **514** must be set to 1/1 when used as motor encoder.

Example

On a linear axis, the motor is flange-connected via a gear (transmission ratio 8:1) and the application connector is flange-connected via a second gear (transmission ratio 3:1).


Sledge
Application encoder:
1024 Incr./Revolution
Feed
Gear 3:1
Gear 8:1
EC2 Gear factor Numerator **513**
EC2 Gear factor Denominator **514**

1 motor revolution = 1/8 turn on output side
= 1/8x3 encoder turn

$$\frac{EC\;2\;Gear\;Factor\;Numerator\,\mathbf{513}}{EC\;2\;Gear\;Factor\;Denominator\,\mathbf{514}} = \frac{\text{Revolution s of the Motor shaft}}{\text{Revolution s of the EC2 encoder shaft}} = \frac{8}{3}$$

7.2 Speed sensor 2 output (X410): TTL Encoder emulation

This chapter describes the commissioning of speed sensor 3 at X410 interface.

The emulation output is transmitted via the X410.A and X410.B terminals. `Speed sensor 1` or `speed sensor 3` can be selected as input signal for the encoder emulation.

The Encoder Emulation settings are located in the `Machine data --> TTL encoder emulation` branch of the VPlus software.

Using the parameter *Operation mode* **656** you can choose to perform the TTL encoder emulation via the `speed sensor 1` or `speed sensor 3`.

Using the parameter *Division marks* **657** you can select the emulation's output number of the division marks. The number of division marks always corresponds to one mechanical revolution of the motor shaft. The value range lies between 30 to 8192 marks.

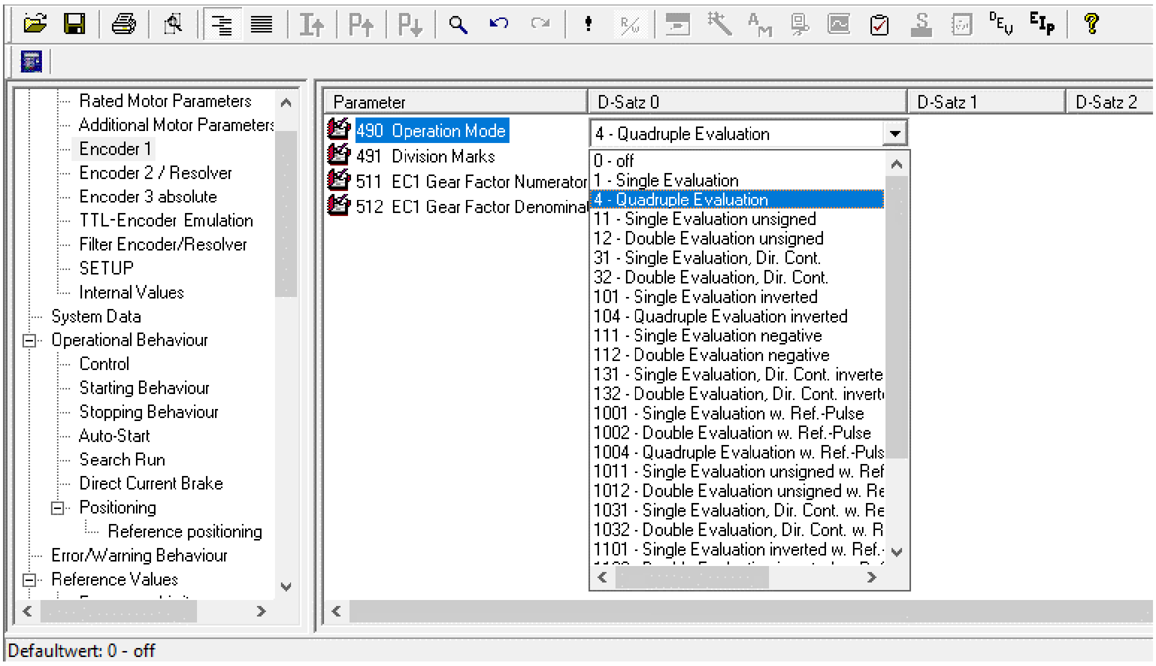
Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Default**
656	Operation mode	0 – Off		0 – Off
		1 – TTL-Encoder Emulation DG 1		
		2 – TTL-Encoder Emulation DG 3		
657	Division marks	30	8192	1024

Selecting Encoder

Fort he next step, select the encoder to utilize as input source of the emulation. Encoder DG1 and DG3 are available for selection here. Make sure to select only one of the encoders. If the encoder DG1 is selected while DG3 is active, the encoder DG3 shall be deactivated and vice versa.

Encoder DG1

DG1 is selected, when for Parameter *Operation Mode* **490** any selection other than zero is made.


Rated Motor Parameters
Additional Motor Parameters
Encoder 1
Encoder 2 / Resolver
Encoder 3 absolute
TTL-Encoder Emulation
Filter Encoder/Resolver
SETUP
Internal Values
System Data
Operational Behaviour
Control
Starting Behaviour
Stopping Behaviour
Auto-Start
Search Run
Direct Current Brake
Positioning
Reference positioning
Error/Warning Behaviour
Reference Values

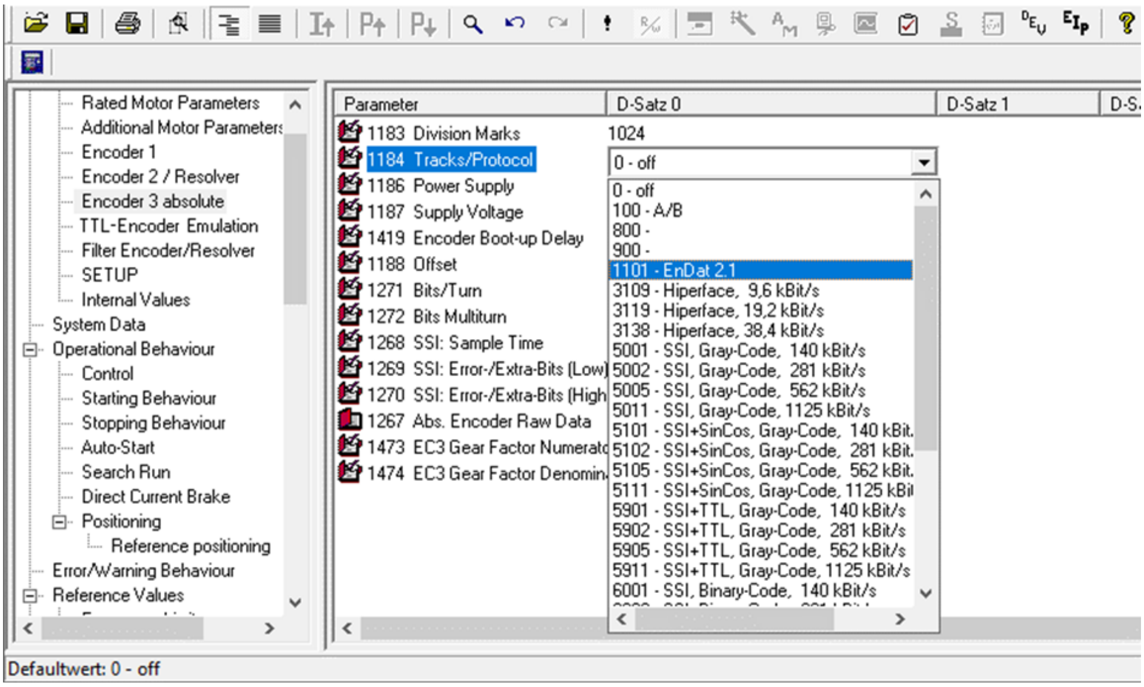
Parameter	D-Satz 0	D-Satz 1	D-Satz 2
490 Operation Mode	4 - Quadruple Evaluation		
491 Division Marks			
511 EC1 Gear Factor Numerator			
512 EC1 Gear Factor Denomina			

0 - off
1 - Single Evaluation
4 - Quadruple Evaluation
11 - Single Evaluation unsigned
12 - Double Evaluation unsigned
31 - Single Evaluation, Dir. Cont.
32 - Double Evaluation, Dir. Cont.
101 - Single Evaluation inverted
104 - Quadruple Evaluation inverted
111 - Single Evaluation negative
112 - Double Evaluation negative
131 - Single Evaluation, Dir. Cont. inverte
132 - Double Evaluation, Dir. Cont. invert
1001 - Single Evaluation w. Ref.-Pulse
1002 - Double Evaluation w. Ref.-Pulse
1004 - Quadruple Evaluation w. Ref.-Puls
1011 - Single Evaluation unsigned w. Ref
1012 - Double Evaluation unsigned w. Re
1031 - Single Evaluation, Dir. Cont. w. Re
1032 - Double Evaluation, Dir. Cont. w. R
1101 - Single Evaluation inverted w. Ref.-

Defaultwert: 0 - off

Encoder DG3

DG3 is selected, when for Parameter *Operation Mode* **1184** any selection other than zero is made.


Rated Motor Parameters
Additional Motor Parameters
Encoder 1
Encoder 2 / Resolver
Encoder 3 absolute
TTL-Encoder Emulation
Filter Encoder/Resolver
SETUP
Internal Values
System Data
Operational Behaviour
Control
Starting Behaviour
Stopping Behaviour
Auto-Start
Search Run
Direct Current Brake
Positioning
Reference positioning
Error/Warning Behaviour
Reference Values
0 - off
100 - A/B
800 -
900 -
1101 - EnDat 2.1
3109 - Hiperface, 9,6 kBit/s
3119 - Hiperface, 19,2 kBit/s
3138 - Hiperface, 38,4 kBit/s
5001 - SSI, Gray-Code, 140 kBit/s
5002 - SSI, Gray-Code, 281 kBit/s
5005 - SSI, Gray-Code, 562 kBit/s
5011 - SSI, Gray-Code, 1125 kBit/s
5101 - SSI+SinCos, Gray-Code, 140 kBit.
5102 - SSI+SinCos, Gray-Code, 281 kBit.
5105 - SSI+SinCos, Gray-Code, 562 kBit.
5111 - SSI+SinCos, Gray-Code, 1125 kBit
5901 - SSI+TTL, Gray-Code, 140 kBit/s
5902 - SSI+TTL, Gray-Code, 281 kBit/s
5905 - SSI+TTL, Gray-Code, 562 kBit/s
5911 - SSI+TTL, Gray-Code, 1125 kBit/s
6001 - SSI, Binary-Code, 140 kBit/s

Parameter	D-Satz 0	D-Satz 1	D-S.
1183 Division Marks	1024		
1184 Tracks/Protocol	0 - off		
1186 Power Supply			
1187 Supply Voltage			
1419 Encoder Boot-up Delay			
1188 Offset			
1271 Bits/Turn			
1272 Bits Multiturn			
1268 SSI: Sample Time			
1269 SSI: Error-/Extra-Bits (Low)			
1270 SSI: Error-/Extra-Bits (High			
1267 Abs. Encoder Raw Data			
1473 EC3 Gear Factor Numerato			
1474 EC3 Gear Factor Denomin.			

Defaultwert: 0 - off

7.3 Speed sensor 3 (X412): Description Commissioning

This chapter describes the commissioning of speed sensor 3.

7.3.1 SinCos encoders

This chapter describes how SinCos encoders are commissioned.

EM-AUT-21 does not support commutation tracks on SinCos encoders. The speed control of synchronous motors with SinCos encoders is therefore not possible.

Step 1: Turn the frequency inverter on for parameter configuration (mains voltage or DC 24 V).

Step 2: Configure the frequency inverter according to the following parameters.

- Adjust the *Division marks* **1183** according to the encoder data sheet (see Chapter 7.4.1), in the case of SinCos encoders, the value is typically 1024 pulses/turn.
- Set *Tracks/Protocol* **1184** to value 100 (please see chapter 7.4.2).
- Adjust the *Supply voltage* **1187** according to the encoder data sheet (see Chapter 7.4.4), in the case of SinCos encoders, the value is typically 5.0V.
- Adjust *Power supply* **1186** according to the connections (see chapter 7.4.3). Bonfiglioli Vectron recommends evaluating the sense line (settings: "5-intern, Sense" or "6-Via X410A, Sense"), if available and connected.

NOTICE

Always set the *Supply voltage* **1187** first and then set *Power supply* **1186**.

- If the encoder is used as a motor encoder for a synchronous servomotor, set *Offset* **1188** according to chapter 7.4.6. This step is not required in the case of asynchronous motors or if the encoder is used as an application encoder.

Step 3: Turn the frequency inverter off.

Step 4: Connect the SinCos encoder to the EM-AUT module. See chapter 5.2.

Step 5: Turn the frequency inverter on.

Step 6: Check the encoder for proper function.

SinCos encoders are no absolute value encoders. In configurations "Positioning" x40 you will have to carry out a referencing operation with SinCos encoders after mains on.

7.3.2 Hiperface encoders

This chapter describes how Hiperface encoders are commissioned.

Step 1: Turn the frequency inverter on for parameter configuration (mains voltage or DC 24 V).

Step 2: Configure the frequency inverter according to the following parameters.

- Adjust the *Division marks* **1183** according to the encoder data sheet (see Chapter 7.4.1), in the case of Hiperface encoders, the value is typically 1024 amplitudes/turn (in example SRS50/SRM50).
- Set *Tracks/Protocol* **1184** according to the encoder data sheet to value 3109, 3119, 3138 or 700 (please see chapter 7.4.2).
- Typical values:
- Sick SEK37/SEL37 & SEK52/SEL52: 9.6 kBaud → value 3109
- Sick SKS36/SKM36: 9.6 kBaud → = value 3109
- Sick SRS50/SRM50: 9.6 kBaud → = value 3109
- Adjust the *Supply voltage* **1187** according to the encoder data sheet (see Chapter 7.4.4), in the case of Hiperface encoders, the value is typically 8.0 V.
- Adjust *Power supply* **1186** according to the connections to "1-internal" or "2-Via X410A" (see chapter 7.4.3).
- In the case of Hiperface encoders, the sense line (settings "5-intern, Sense") is typically not used, as it is not defined in the Hiperface standard Specification. Thus, using the sense line is not required in the case of Hiperface encoders.

NOTICE

Always set the *Supply voltage* **1187** first and then set *Power supply* **1186**.

- Set the number of *Bits/Turn* **1271** according to the encoder data sheet (see chapter 7.4.7).
- Typical values:
- Sick SEK37/SEL37 & SEK52/SEL52: 9 bits/t
- Sick SKS36/SKM36: 12 bits/t
- Sick SRS50/SRM50: 15 bits/t
- Set the *Bits Multiturn* **1272** according to the encoder data sheet (see chapter 7.4.8),
- Typical values:
- Sick SEL37, SEL52, SKM36, SRM50: 12 bits/t

In the case of singleturn encoders (e.g. Sick SEK37, SKS36, SRS50), you will have to set *Bits Multiturn* **1272** = 0.

- If the encoder is used as a motor encoder for a synchronous servomotor, set *Offset* **1188** according to chapter 7.4.6. This step is not required in the case of asynchronous motors or if the encoder is used as an application encoder.

Step 3: Turn the frequency inverter off.

Step 4: Connect the Hiperface encoder to the EM-AUT module. Bonfiglioli Vectron recommends the use of pre-assembled cables (see chapter 5.2.3).

Step 5: Turn the frequency inverter on.

Step 6: Check the encoder for proper function.

Step 7: In configurations "Positioning" x40: Carry out referencing operation once.

If the data track cannot be evaluated, error "F1719 Dig. encoder: Protocol error" will be triggered. In this case, check *Tracks/Protocol* **1184** setting.

When the frequency inverter is turned on, the absolute position is read via the data tracks. Via the incremental tracks, the position is counted up internally and compared to the updated absolute position at regular intervals. This guarantees a very high positioning and speed accuracy at all supported transmission rates.

7.3.3 EnDat 2.1 encoders

This chapter describes how EnDat 2.1 encoders are commissioned.

The EM-AUT-21 module supports, in the case of EnDat 2.1 encoders, a baud rate of 100 kBit/s. Other baud rates will not be supported.

Step 1: Turn on the frequency inverter for parameter configuration (mains voltage or DC 24 V).

Step 2: Configure the frequency inverter according to the following parameters.

- Adjust the *Division marks* **1183** according to the encoder data sheet (see Chapter 7.4.1), in the case of EnDat 2.1 encoders, the value is typically 512 amplitudes/turn, (e.g. Heidenhain ECN 1113, EQN 1125).
- Set *Tracks/Protocol* **1184** to value 1101 (please see chapter 7.4.2).
- Adjust the *Supply voltage* **1187** according to the encoder data sheet (see Chapter 7.4.4), in the case of EnDat 2.1 encoders, the value is typically 5.0 V.
- Adjust *Power supply* **1186** according to the connections (see chapter 7.4.3). Bonfiglioli Vectron recommends evaluating the sense line (settings: "5-intern, Sense").

NOTICE

Always set the *Supply voltage* **1187** first and then set *Power supply* **1186**.

- If the encoder is used as a motor encoder for a synchronous servomotor, set *Offset* **1188** according to chapter 7.4.6. This step is not required in the case of asynchronous motors or if the encoder is used as an application encoder.

Parameters *Bits/Turn* **1271** and *Bits Multiturn* **1272** have no function in the case of EnDat 2.1 encoders. The required data is exchanged directly between the encoder and inverter.

Step 3: Turn the frequency inverter off.

Step 4: Connect the EnDat 2.1 encoder to the EM-AUT module. Bonfiglioli Vectron MDS recommends the use of pre-assembled cables (see chapter 5.2.2).

Step 5: Turn the frequency inverter on.

Step 6: Check the encoder for proper function.

Step 7: In configurations "Positioning" x40: Carry out referencing operation once.

If the data track cannot be evaluated, error "F1719 Dig. encoder: Protocol error" will be triggered. In this case, check *Tracks/Protocol* **1184** setting.

When the frequency inverter is turned on, the absolute position is read via the data tracks. Via the incremental tracks, the position is counted up internally and compared to the updated absolute position at regular intervals. This guarantees a very high positioning and speed accuracy at all supported transmission rates.

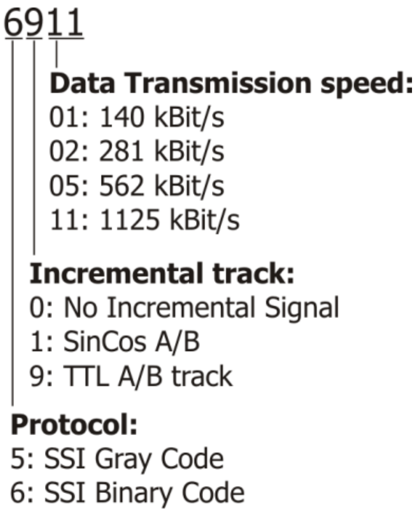
7.3.4 SSI encoders

This chapter describes how SSI encoders are commissioned. You can connect SSI encoders with binary evaluation and SSI encoders with Gray code evaluation.

> For a correct function of the speed control, an SSI encoder with incremental tracks (TTL [RS-422] level or SinCos tracks) must be used.
>
> If the SSI encoder is used for positioning (and not for speed feedback), you can also use a SSI encoder without incremental tracks.
>
> HTL tracks cannot be used as incremental tracks.

- **Step 1:** Turn the frequency inverter on for parameter configuration (mains voltage or DC 24 V).
- **Step 2:** Configure the frequency inverter according to the following parameters.
- Set *Tracks/Protocol* **1184** according to the encoder data sheet (please see chapter 7.4.2).

SSI operation modes key:



6911

Data Transmission speed:
01: 140 kBit/s
02: 281 kBit/s
05: 562 kBit/s
11: 1125 kBit/s

Incremental track:
0: No Incremental Signal
1: SinCos A/B
9: TTL A/B track

Protocol:
5: SSI Gray Code
6: SSI Binary Code

> If a SSI encoder without incremental track (*Tracks/Protocol* **1184** = 50xx or 60xx) is used for positioning, the speed of the data track must be as high as possible for optimum control quality.
>
> The usable transmission rate depends on the length of the encoder cable.

- Adjust the *Division marks* **1183** according to the encoder data sheet (see Chapter 7.4.1), in the case of SSI encoders, the value is typically 512 amplitudes/turn. If an encoder without incremental tracks is used (setting via Tracks/Protocol 1184), this information is not needed and the setting of this parameter will be ignored.
- Adjust the *Supply voltage* **1187** according to the encoder data sheet (see Chapter 7.4.4), in the case of SSI encoders with TTL [RS-422] or SinCos track, the value is typically 5.0V.
- Adjust *Power supply* **1186** according to the connections (see chapter 7.4.3). Bonfiglioli Vectron recommends evaluating the sense line (settings: "5-intern, Sense"), if available and connected.
- Set the number of *Bits/Turn* **1271** according to the encoder data sheet (see chapter 7.4.7).
- Set the Bits Multiturn 1272 according to the encoder data sheet (see chapter 7.4.8).
- Set *SSI: Error-/Extra-Bits (Low)* **1269** and *SSI: Error-/Extra-Bits (High)* **1270** , if additional information from the encoder is supported (see chapter 7.4.9).
- Adjust *SSI: Sample time* **1268** according to the encoder data (see chapter 7.4.10).
- If the encoder is used as a motor encoder for a synchronous servomotor, set Offset 1188 according to chapter 7.4.6. This step is not required in the case of asynchronous motors or if the encoder is used as an application encoder.

> In the case of single-turn encoders, you will have to set *Bits Multiturn* **1272** = 0.

- **Step 3:** Turn the frequency inverter off.
- **Step 4:** Connect the SSI encoder to the EM-AUT-31. See chapter 5.2.

- **Step 5:** Turn the frequency inverter on.
- **Step 6:** Check the encoder for proper function.
- **Step 7:** In configurations "Positioning" x40: Carry out referencing operation once.

If the data track cannot be evaluated, error "F1719 Dig. encoder: Protocol error" will be triggered. In this case, check *Tracks/Protocol* **1184** setting.

When the frequency inverter is turned on, the absolute position is read via the data tracks. Via the incremental tracks, the position is counted up internally and compared to the updated absolute position at regular intervals. This guarantees a very high positioning and speed accuracy at all supported transmission rates.

Encoders without incremental track can only be used as application encoders (for example for positioning applications).

7.3.5 Commissioning of linear encoders

In addition to the settings described in the previous chapters, the conversion from the rotary to the translatory system must be considered when it comes to commissioning a linear encoder. This conversion is influenced greatly by the diameter of the turning wheel.

The following applies:

Circumference = π * diameter

Linear encoders are normally not suitable for speed control, as the sampling time is too long to enable good speed control. For this reason, the following descriptions are based on the use as a position encoder in configuration x40.

For the calculations described in this chapter, an Excel worksheet was prepared by Bonfiglioli. Please contact your local sales agent. This Excel worksheet will help you to carry out the calculations required for commissioning linear encoders with ANG frequency inverters.

Linear encoders typically have a fixed resolution (e.g. 1 mm). In some linear encoders, the resolution can be configured. First check the resolution of the linear encoder using the data sheet or the parameter configuration.

The resolution of the linear encoder must be assigned in the frequency at the resolution of the selected user units. This is done using the four parameters *Bits/Turn* **1271**, *Bits Multiturn* **1272**, *EC2 Gear Factor Numerator* **513** and *EC2 Gear Factor Denominator* **514**.

The positioning reference system is always referred to the output side, in user units, through parameters *Feed constant* **1115**, *Gear Box: Driving Shaft Revolutions* **1116** and *Gear Box: Motor Shaft Revolutions* **1117**. Thus, these parameters must also be considered when configuring the linear encoder.

Parameters *Bits/Turn* **1271** and *Bits Multiturn* **1272** are virtual quantities in the case of a linear encoder and are determined by the mechanical properties of the system. Different properties of the mechanical system (e.g. gear transmission or turning wheel diameter) will lead to different parameter settings.

Shifting of a bit in parameters *Bits/Turn* **1271** and *Bits Multiturn* **1272** has the same effect as doubling or halving in parameters *EC2 Gear Factor Numerator* **513**/ *EC2 Gear Factor Denominator* **514**.

Reduction of *Bits/Turn* **1271** or increase of *Bits Multiturn* **1272** by 1 Bit

--> has the same effect as doubling of **513** / **514**

Increase of *Bits/Turn* **1271** or reduction of *Bits Multiturn* **1272** by 1 Bit

--> has the same effect as halving of **513** / **514**.

Required data:

The following data is needed for commissioning of the linear encoder:

- Gear transmission [] or input speed / output speed [rpm/rpm]
- Encoder resolution [bits]
- Running wheel diameter [m]
- Required accuracy [m] or resolution [increments/m]

- **Step 1:** Identify gear values reference system:

The input speed (motor speed) will determine the setting for parameter *Gear Box: Motor Shaft Revolutions* **1117**, the output speed will determine the setting for parameter *Gear Box: Driving Shaft Revolutions* **1116**.

The value should be entered as exactly as possible. Shifting of decimal places or multiplication with appropriate factors can increase accuracy.

Example:

- Input speed: 1401 rpm
- Output speed: 77.3 rpm   i = 18.12
- Encoder resolution: 24 Bit
- Diameter: 160 mm = 0.16 m
- Required accuracy: 0.01 mm = 0.00001 m

➔ Gear Box: Motor Shaft Revolutions **1117** = 14010

➔ Gear Box: Driving Shaft Revolutions **1116** = 773

- **Step 2:** Identify feed constant reference system:

The feed constant is calculated by multiplying the diameter and π by the resolution. The resolution is the reciprocal of the accuracy.

$$\text{Accuracy [m]} = \frac{1}{\text{Resolution}\left[\frac{\text{u}}{\text{m}}\right]}$$

$$\textit{Feed constant } \mathbf{1115}\ [\text{u}] = \frac{\pi \cdot \text{Diameter [m]}}{\text{Accuracy}[\frac{\text{m}}{\text{u}}]}$$

$$= \pi \cdot \text{Diameter [m]} \cdot \text{Resolution}\left[\frac{\text{u}}{m}\right]$$

Example:

- Diameter: 0.16 m = 160 mm
- Required resolution: 0.00001 m = 0.01 mm

➔ *Feed constant* **1115** = 50265 u

- **Step 3:** Calculate auxiliary quantity reference system

In the following step, the ratio of the *Feed constant* **1115** to *Gear Box: Driving Shaft Revolutions* **1116** and *Gear Box: Motor Shaft Revolutions* **1117** is used in the calculations frequently. For better clarity, auxiliary quantity "R" (=reference system) is calculated now:

$$\text{R} = \frac{\textit{Feed constant } \mathbf{1115}\ \frac{[\text{u}]}{\text{U}} \cdot GearBox : DrivingShaftRevolutions\,\mathbf{1116}}{GearBox : MotorShaftRevolutions\,\mathbf{1117}}$$

Example:

- *Feed constant* **1115 =** 50265 rev
- *Gear Box: Driving Shaft Revolutions* **1116 =** 773
- *Gear Box: Motor Shaft Revolutions* **1117 =** 14010

➔ R = 2773.365 rev = 50265 x 773 / 14010 rev

- **Step 4**: Determine the encoder resolution:

First determine the number of user units per encoder increment. If, for example, the encoder features a resolution of 1 mm and 0.01 is to be used as the "user unit", β = 100.

β = Number of user units per encoder increment

- **Step 5:** Calculate *Bits/Turn* **1271** :

The reference system and the number of user units per encoder increment β determine parameter *Bits/Turn* **1271**.

$$Bits\,/\,Revolution = \mathrm{Log}_2 \frac{Feed\ Constant\,\mathbf{1115}\ \frac{[\mathrm{u}]}{\mathrm{U}} \cdot GearBox : DrivingShaftRevolutions\,\mathbf{1116}}{\beta \cdot GearBox : MotorShaftRevolutions\,\mathbf{1117}\ \cdot}$$

or

$$Bits\,/\,Revolution = \mathrm{Log}_2 \frac{R}{\beta}$$

$$= \frac{1}{\mathrm{Ln}\ 2} \cdot \mathrm{Ln}\ \frac{R}{\beta}$$

Round the value up to the next natural number.

With the values above, *Bits/Turn* **1271**=5.

Conversion of logarithm base 2 and other bases:

$$\mathrm{Log}_2 a = \frac{\mathrm{Log}_{10} a}{\mathrm{Log}_{10} 2} = \frac{\mathrm{Ln}\ a}{\mathrm{Ln}\ 2}$$

- **Step 6:** Calculate *Bits Multiturn* **1272** :

Bits Multiturn **1272** is calculated from the subtraction of the total number of position bits of the encoder with the *Bits/Turn* **1271** calculated above.

$$Multiturn = EncoderBits - Bits/Turn$$

With the values above, *Bits Multiturn* **1272**=19.

Additionally, for encoders with additional analogue tracks (Hiperface, EnDat 2.1) it is imperative that the parameter *Division marks* **1183** is set correctly. Thereto, both digital resolution (A_d) of the encoder (e.g. 1mm/Bit) AND the analogue resolution (A_a) (e.g. 4 mm/amplitude) must be known.

Then, the setting value for the parameter **1183** is calculated as follows:

$$P.1183 = \frac{2^{Bits/Turn(P.1271)} \cdot A_d}{A_a} = \frac{2^5 \cdot 1[mm]}{4[mm]} = 8$$

- **Step 7**: Calculation of speed sensor 2 gear factors

For calculation of speed sensor 2 gear factors, the *preliminary numerator* is calculated first as follows:

- *Preliminary Numerator* **= 2 ^** *Bits/Turn* **1271**

Then, the preliminary denominator is calculated as follows:

$$PreliminaryDenominator = \frac{Feed\ Constant\,\mathbf{1115}\ \frac{[\mathrm{u}]}{\mathrm{U}} \cdot GearBox : DrivingShaftRevolutions\,\mathbf{1116}}{\beta \cdot GearBox : MotorShaftRevolutions\,\mathbf{1117}\ \cdot}$$

or

$$PreliminaryDenominator = \frac{R}{\beta}$$

With the example values, the following results are obtained:

Preliminary Numerator = 32.

Preliminary Denominator = 27.7336.

The values calculated in this way can be used directly for parameters *EC2 Gear Factor Numerator* **513** and *EC2 Gear Factor Denominator* **514**. To increase accuracy, the following intermediate "Optimization" step is recommended. This intermediate step is not necessary if accuracy is already sufficient.

- *EC2 Gear Factor Numerator* **513** = 32.00.
- *EC2 Gear Factor Denominator* **514** = 27.73

- **Step 8:** Optional: Optimization of gear factors

The steps carried out above will result (provided that calculation was made correctly) in a denominator which is smaller than the numerator. This advantage is used for optimization.

The following is set:

- *EC2 Gear Factor Numerator* **513** = 300.00.

Value 300.00 is always used to achieve maximum accuracy.

$$ConclusiveDenominator = 300.00 \cdot \frac{PreliminaryDenominator}{PreliminaryNumerator}$$

With the example values, the following results are obtained:

- EC2 Gear Factor Numerator **513** = 300.00
- EC2 Gear Factor Denominator **514** = 260.00

Parameter *EC2 Gear Factor Numerator* **513** is limited to value range -300.00...300.00. To maximize the value range of the factors, the maximum value 300.00 is used for optimization.

- **Step 9**: Optional: Check of accuracy:

This section describes the calculations required for determining the accuracy. The check is not required for proper function, it is solely for determining the accuracy limits.

Due to rounding operations in the parameters described above, there will be an error across the total travel distance. This error is calculated in the following steps:

$$(1)\ \ Distance\_ref[u] = \frac{Distance\_ref[m]}{Accuracy\left[\frac{m}{u}\right]}$$

$$(2)\ \ Distance\_\text{act}[\text{intemal}] = \text{RoundDown}\left(\frac{EC2GearFactorNumerator\ \mathbf{513}}{EC2GearFactorDenominator\ \mathbf{514}} \cdot \frac{Distance\_ref[u]}{\text{ß}} \cdot \frac{2^{16}}{2^{Bits/Revolution\ \mathbf{1271}}}\right)$$

$$(3)\ \ Distance\_act[u] = \text{RoundDown}\left(Distance\_act[\text{intern}] \cdot \frac{R}{2^{16}}\right)$$

$$(4)\ \ Error[u] = Distance\_act[u] - Distance\_ref[u]$$

$$(5)\ \ Error[m] = Distance\_act[u] \cdot Accuracy\left[\frac{m}{u}\right] - Distance\_ref[m]$$

The error can be reduced by increasing the accuracy of the gear factors. By using the 2 decimal places of parameters *EC2 Gear Factor Numerator* **513** and *EC2 Gear Factor Denominator* **514** and the optimization described in the previous step ("8 Optimization of gear factors"), accuracy can be increased.

At a maximum travel distance of 10 m, the following is obtained:

Non-optimized gear factors	**Optimized gear factors**
Distance_nominal [rev] = 1 000 000 rev	Distance_nominal [rev] = 1 000 000 rev
Distance_actual [internal] = 23 633 609	Distance_actual [internal] = 23 630 769
Distance_actual [rev] = 1 000 131 rev	Distance_actual [rev] = 1 000 011 rev
Error [rev] = 131 rev	Error [rev] = 11 rev
Error [m] = 0.00131 m	Error [m] = 0.00011 m
Error [mm] = 1.3 mm	Error [mm] = 0.11 mm

Parameter *EC2 Gear Factor Numerator* **513** is limited in value range -300.00...300.00, *EC2 Gear Factor Denominator* **514** is limited in value range 0.01 to 300.00. In many situations, choosing a modifier is useful which sets the greater of the two parameters to a value slightly below 300.00.

7.3.5.1 Checking the settings

Upon completion of the setup, check the system for proper function.

WARNING

Incorrect movements

Wrong setup of the linear encoder can result in incorrect movements or direction of movement.

The following requirements must be met when it comes to testing the linear encoder:

- Before the start of the test, make sure the hardware limit switches work properly.
- Before the start of the test, make sure the emergency stop works properly.
- Use
  - slow speeds
  - slow ramps
- Deactivate for the test the position controller by setting **1118** = 0.

To reduce the speeds, you can use the so-called "Speed Override" mode.

Via actual value parameter *Abs. encoder raw data* **1267**, you can monitor the encoder value transmitted. Carry out a travel operation across a distance which can be measured easily (e.g. 10 cm). Check if the actual value parameter *Abs. encoder raw data* **1267** changes and the *Act. Position* **1108** changes across the distance in accordance with your settings.

Via the scope function of VPlus, you can check the commissioning of the linear encoder.

Adjust the following scope sources:

- 1003 Act. Position * 1000
- 1007 Ref. Position * 1000
- 1013 Contouring Error *10 or 1012 Contouring Error *1
- 442 Hz: Act. Speed

For more information on VPlus, see chapter 17.

As the time base, choose the observation period for some seconds.

When starting a motion block or a travel command via field bus, Ref. Position is set to Act. Position. The two curves of sources 1003 and 1007 must be identical as from the start time of the travel command. If the two curves are not identical, the parameter factors have not been set correctly.

If the ramp Act. Position is steeper than the ramp of Ref. Position, the ratio 513/514 must be reduced.

If the ramp Act. Position is less steep than the ramp of Ref. Position, the ratio 513/514 must be increased.

Via the source of the contouring error, the quality of the settings can be checked additionally. The contouring error must not increase continuously. Due to the mechanical characteristics, a small constant contouring error is typical to the system, continuous (significant) increasing of the contouring error (also in negative direction) indicates that linear encoder parameters have be set up incorrectly.

When the position controlled is deactivated, rounding errors may result in a minor continuous increase in the contouring error. In most cases however, this is small enough to be distinguishable.

As soon as the settings have been checked for correctness, repeat the tests using sources 1002/ 1006 (resolution 10 times higher than sources 1007/1011), then using 1001 / 1005 and then using 1000 and 1004. In this way, the settings are checked again at a higher accuracy. Note that, with a higher accuracy, overflows may be displayed in Scope more frequently. This does not affect the function.

Depending on the reference system chosen (Parameter *Feed constant* **1115**, *Gear Box: Driving Shaft Revolutions* **1116** and *Gear Box: Motor Shaft Revolutions* **1117**), some sources may not have the required significance in Scope. Then, switch to the next smaller couple as shown above. Always start with the highest setting.

Activate the position controller again. Position controller *Limitation* **1118** settings must always match the reference system and the mechanical system.

A contouring error will typically build up during acceleration or deceleration. During constant travel operations, the contouring error should become smaller again. Note that the *Maximum frequency* **419** is exceeded by the output of the position controller. Ensure that the total of *Maximum frequency* **419** and position controller *Limitation* **1118** can be reached by the mechanical equipment. A reduction of the maximum frequency may be a good idea in certain applications in order to limit the total to the mechanically possible maximum.

In most application, limitation of position controller *Limitation* **1118** to approx. 10 % of the maximum frequency makes sense.

With the position controller activated, check the function again.

7.3.5.2 Initialize counting direction

First check if the counting direction of the user units meets the requirements. You can change the counting direction by inverting the parameter *EC2 Gear Factor Numerator* **513** (e.g. by inverting parameter *EC2 Gear Factor Numerator* **513** from 200.00 to -200.00).

WARNING

Application Errors

By changing parameter *EC2 Gear Factor Numerator* **513**, the encoder values will be re-calculated in the internal user unit format. As a result, the value of *Act. Position* **1108** may change. Especially when software limit switches are used or in the case of feedback to a PLC, this can result in warnings or application errors.

- For this reasons, after changing the parameters of the reference system and the encoder, always check the *Act. Position* **1108**, considering the permissible travel distance (e.g. *Pos. SW Limit Switch* **1145**).

7.3.5.3 Initializing home position

For positioning application, a certain point of the system is typically defined as the home position. After checking the correct reference system of the positioning and linear encoder (see Chapter 7.3.4) and setting the counting direction, the home position can be initialized.

Move (e.g. in JOG mode) to the required system home position. At this position, stop the drive. Set parameter *Home Offset* **1131** = 0.

By default, *Home Offset* **1131** is set to zero. Upon first commissioning, you do not have to change the value, but this step is required in the case of commissioning following a change.

Now, read the value in parameter *Act. Position* **1108**. Invert this value. Enter the inverted value in *Home Offset* **1131**.

Example:

Act. Position **1108** = 7654 u → *Home Offset* **1131** = -7654

- Once you have set up the home position offset, check the system for correct function again (see chapter 7.3.4).
- If required for the application, set up the software limit switches now.

Referencing using an absolute value encoder is not necessary after completion of first commissioning. The referencing setting *Operation mode* **1220** with setting "10 – No referencing required" can be used after initialization.

7.4 Encoder input 3 (X412) – Parameter descriptions

The encoder input is used for evaluating the position information from the encoder.

Depending on the encoder system used, certain parameters need to be set up. The following table describes the use of the individual parameters for the encoder systems.

Parameters		Encoder system			
No.	**Description**	**SinCos/TTL**	**Hiperface**	**EnDat 2.1**	**SSI**
1183	Division marks	X	X	X	(X)
1184	Encoder signals/log	X	X	X	X
1186	Power supply	X	X	X	X
1187	Supply voltage	X	X	X	X
1188	Offset	1)			
1268	SSI: Sampling interval	---	---	---	X
1269	SSI: Error-/Extra-Bits (Low)	---	---	---	X
1270	SSI: Error-/Extra-Bits (High)	---	---	---	X
1271	Bits/Turn	---	X	---	X
1272	Bits Multiturn	---	X	---	X
1473	EC3 Gear Factor Numerator	X	X	X	X
1474	EC3 Gear Factor Denominator	X	X	X	X

X: Parameter must be configured according to the encoder data sheet.

---: Parameter has no function for this encoder type.

(X): In the case of SSI encoders the evaluation of the division marks depends on the setting of *Tracks/Protocol* **1184**.

1): Setting the offset is required in the case of synchronous motors.

In addition, the following actual value parameters are available:

Parameters		Encoder system			
No.	**Description**	**SinCos**	**Hiperface**	**EnDat 2.1**	**SSI**
1267	Abs. encoder raw data	---	X	X	X
1274	Warning Dig. Encoder	---	---	X	---

If positioning (configurations x40) is used, please refer to the instructions in chapter 7.6.

Gear factors *EC3 Gear Factor Numerator* **1473** and *EC3 Gear Factor Denominator* **1474** must be set to 1/1 when used as motor encoder.

7.4.1 Division marks

In parameter *Division marks* **1183**, you can set the type-specific count of division marks of the encoder. The count of division marks is typically described in amplitudes/revolution in the case of encoders with SinCos tracks. Enter the division marks or amplitudes/revolution in parameter *Division marks* **1183**.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Default**
1183	Division marks	0	8192	1024

In the case of SSI absolute value encoders, evaluation of *Division marks* **1183** is active only if *Tracks/Protocol* **1184** is described in an operation mode for evaluation of TTL [RS-422] or SinCos tracks (settings 51xx, 59xx, 61xx and 69xx).

7.4.2 Tracks/Protocol

Via parameter *Tracks/Protocol* **1184**, you can specify the type-specific number of analog Tracks/Protocol of the encoder and evaluation of a reference track.

The EM-AUT-31 module supports, in the case of EnDat 2.1 encoders, a baud rate of 100 kBit/s. Other baud rates will not be supported.

Key of Tracks/Protocol:

6910

Data transmission speed:
01: 100 kBit/s
02: 250 kBit/s
05: 500 kBit/s
10: 1000 kBit/s
} EnDat 2.1 SSI

09: 9,6 kBit/s
19: 19,2 kBit/s
38: 38,4 kBit/s
} Hiperface

Incremental track:
0: No Incremental Signal
1: SinCos A/B
8: TTL A/B
9: TTL A/B + R track

Protocol:
0: SinCos without Absolute value
1: EnDat 2.1
3: Hiperface
5: SSI Gray Code
6: SSI Binär Code

The identifiers of track A/B and Sin/Cos are typically ambivalent and can be set to A = Sin and B = Cos.

	Tracks/Protocol 1184	Function
	0 - off	Evaluation is turned off. **Default**.
SinCos	100 - A/B	Evaluation of analog Tracks/Protocol A and B.
	500 - A/B, R	Evaluation of analog Tracks/Protocol A and B as well as reference track R. Monitoring and comparison of Tracks/Protocol.
TTL	800 - A/B	Evaluation of analog Tracks/Protocol A and B
	900 - A/B, R	Evaluation of analog Tracks/Protocol A and B as well as reference track R
EnDat 2.1	1101 EnDat 2.1	Evaluation of analog Tracks/Protocol A/B and the data and clock track with the EnDat 2.1 protocol. Monitoring and comparison of Tracks/Protocol.
Hiperface	3109 Hiperface, 9.6 kBit/s	Evaluation of analog Tracks/Protocol A/B and the data tracks with the Hiperface protocol. Monitoring and comparison of Tracks/Protocol. The data track is transmitted at 9.6 kBaud.
	3119 Hiperface, 19.2 kBit/s	Like 3109. The data track is transmitted at 19.2 kBaud.
	3138 Hiperface, 38.4 kBit/s	Like 3109. The data track is transmitted at 38.4 kBaud.
SSI Gray code	5001 SSI, Gray code, 141 kBit/s	Evaluation of data and clock tracks with the SSI protocol (without TTL or SinCos track). The data track is transmitted at 140.625 kBaud in Gray code.
	5002 SSI, Gray code, 281 kBit/s	Like 5001. The data track is transmitted at 281.25 kBaud in Gray code.
	5005 SSI, Gray code, 563 kBit/s	Like 5001. The data track is transmitted at 562.5 kBaud in Gray code.
	5011 SSI, Gray code, 1125 kBit/s	Like 5001. The data track is transmitted at 1125 kBaud in Gray code.
	5101 SSI+SINCOS, Gray code, 141 kBit/s	Evaluation of Tracks/Protocol A/B as SINCOS track and the data and clock tracks with the SSI protocol. The data track is transmitted at 140.625 kBaud in Gray code.
	5102 SSI+SINCOS, Gray code, 281 kBit/s	Like 5101. The data track is transmitted at 281.25 kBaud in Gray code.
	5105 SSI+SINCOS, Gray code, 563 kBit/s	Like 5101. The data track is transmitted at 562.5 kBaud in Gray code.
	5111 SSI+SINCOS, Gray code, 1125 kBit/s	Like 5101. The data track is transmitted at 1125 kBaud in Gray code.
	5901 SSI+TTL, Gray code, 141 kBit/s	Evaluation of Tracks/Protocol A/B as TTL [RS-422] track and the data and clock tracks with the SSI protocol. The data track is transmitted at 140.625 kBaud in Gray code.
	5902 SSI+TTL, Gray code, 281 kBit/s	Like 5901. The data track is transmitted at 281.25 kBaud in Gray code.
	5905 SSI+TTL, Gray code, 563 kBit/s	Like 5901. The data track is transmitted at 562.5 kBaud in Gray code.
	5911 SSI+TTL, Gray code, 1125 kBit/s	Like 5901. The data track is transmitted at 1125 kBaud in Gray code.

	Tracks/Protocol 1184		Function
SSI Binary code	6001	SSI, binary code, 141 kBit/s	Evaluation of data and clock tracks with the SSI protocol (without TTL or SinCos track). The data track is transmitted at 140.625 kBaud in binary code.
	6002	SSI, binary code, 281 kBit/s	Like 6001. The data track is transmitted at 281.25 kBaud in binary code.
	6005	SSI, binary code, 563 kBit/s	Like 6001. The data track is transmitted at 562.25 kBaud in binary code.
	6011	SSI, binary code, 1125 kBit/s	Like 6001. The data track is transmitted at 1125 kBaud in binary code.
	6101	SSI+SINCOS, binary code, 141 kBit/s	Evaluation of Tracks/Protocol A/B as SINCOS track and the data and clock tracks with the SSI protocol. The data track is transmitted at 140.625 kBaud in binary code.
	6102	SSI+SINCOS, binary code, 281 kBit/s	Like 6101. The data track is transmitted at 281.25 kBaud in binary code.
	6105	SSI+SINCOS, binary code, 563 kBit/s	Like 6101. The data track is transmitted at 562.25 kBaud in binary code.
	6111	SSI+SINCOS, binary code, 1125 kBit/s	Like 6101. The data track is transmitted at 1125 kBaud in binary code.
	6901	SSI+TTL, binary code, 141 kBit/s	Evaluation of Tracks/Protocol A/B as TTL [RS-422] track and the data and clock tracks with the SSI protocol. The data track is transmitted at 140.625 kBaud in binary code.
	6902	SSI+TTL, binary code, 281 kBit/s	Like 6901. The data track is transmitted at 281.25 kBaud in binary code.
	6905	SSI+TTL, binary code, 563 kBit/s	Like 6901. The data track is transmitted at 562.25 kBaud in binary code.
	6911	SSI+TTL, binary code, 1125 kBit/s	Like 6901. The data track is transmitted at 1125 kBaud in binary code.

For synchronous servomotors, an encoder with absolute value will be required. Settings 100, 500, 800 and 900 are only intended for operation with asynchronous motors for this reason. In the case of synchronous servomotors, set the *Offset* **1188** according to chapter 7.4.6.

Changeover of parameter *Tracks/Protocol* **1184** can only be done with the output stage disabled. After the parameter change, the new encoder type will have to be initialized. This may take up to 5 seconds.

After mains on, an initialization may have to be performed depending on the encoder type. This may take up to 5 seconds.

SSI encoder: The usable transmission rate depends on the length of the encoder cable. In case there are any transmission errors, reduce the transmission rate.

7.4.3 Power supply

Via parameter *Power supply* **1186**, you can activate the encoder power supply source.

The operation modes with meas. line "sense" (*Power supply* **1186** = "5 – intern, Sense") enable monitoring of the supply voltage of the encoder. In these settings, deviations will be compensated when the supply voltage of the encoder deviates from the set voltage level. To that end, the voltage is measured at the end of the supply line (at encoder).

In operation mode 1, the voltage is controlled at the EM-AUT-31 module, power losses during energy transmission via the supply line will not be compensated.

The encoder can be powered as follows:

– via contacts X412.6 (V_{Enc}) and X412.15 (0VL) of the female HD-Sub-D connector.

See chapters 5.3.1 "Control terminals" and 5.3.5 "EM-AUT: Speed Sensor Power supply".

CAUTION

High voltage levels

If parameters are misconfigured, the encoder might be destroyed by high voltage levels.

- Always set the *Supply voltage* **1187** first, and then set *Power supply* **1186**.

Power supply 1186	
0 - off	No power supply selected for the encoder. This setting is also used if the encoder is connected directly to an external power supply. **Default**.
1 - Internal	power supply to encoder at terminals X410A.5 (5 … 12 VDC) and X410A.7 (GND) at contacts X412.6 (V_{Enc}: 5 … 12 VDC) and X412.15 (0VL). Voltage source is provided internally by the frequency inverter, max. 2 W.
5 -internal, Sense	power supply to encoder at terminals X410A.5 (5 … 12 VDC) and X410A.7 (GND) at contacts X412.6 (V_{Enc}: 5 … 12 VDC) and X412.15 (0VL). Voltage source is provided internally by the frequency inverter, max. 2 W. A measuring line "sense" of the encoder must be connected in order to monitor the supply voltage.

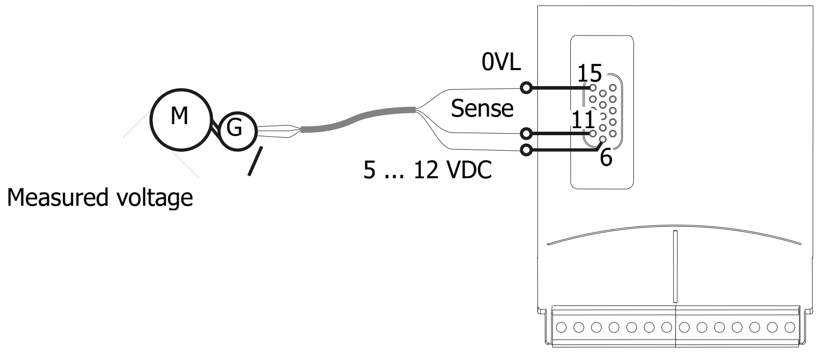
Even if the encoder features a measuring line "sense", you can chose operation mode 1 or 2.

In the case of Hiperface encoders, the sense line (settings "`5-intern, Sense`") is typically not used, as it is not defined in the Hiperface standard Specification. Thus, using the sense line is not required in the case of Hiperface encoders.

The maximum voltage of the power supply is DC 12 V. Via a sense line, the voltage can be monitored at the encoder, but the voltage output is limited to DC 12 V.

The voltage level can be set up via parameter *Supply voltage* **1187**. See chapter 7.4.4. See also chapter 5.3.5.2.

Measuring line "sense": constant voltage level at encoder



G: encoder

The encoder supply voltage is measured at the encoder and kept constant at the adjusted value of *Supply voltage* **1187** (DC 5 … 12 V).

7.4.4 Supply voltage encoder X412

Via parameter *Supply voltage* **1187**, you can select the voltage level for encoder power supply X412.

The encoder 3 can be powered as follows by the ANG:

- via contacts X412.6 (V_{Enc}) and X412.15 (0VL) of the female HD-Sub-D connector.

The parameter setting is effective on the terminals and the contact of the female HD-Sub-D connector.

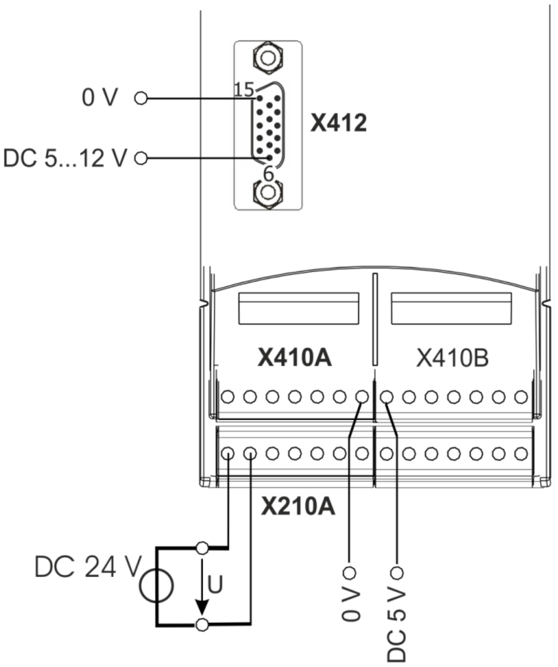
NOTICE

Note the encoder manufacturer's supply voltage specifications. Non-compliance may damage the encoder.

NOTICE

Always set the *Supply voltage* **1187** first and then set *Power supply* **1186**. Otherwise, the encoder might be destroyed by high voltage levels.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Default**
1187	Supply voltage	5.0 V	12.0 V	5.0 V



7.4.5 Filter time constant speed sensor 3

Via parameter *Abs. Encoder: Filter time constant* **1189**, you can filter high frequency of the encoder signals and limit the control band width.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Default**
1189	Abs. Encoder: Filter time constant	125 µs	8000 µs	125 µs

7.4.6 Offset Absolute encoder

In order to enable the start of a synchronous machine, the absolute position of the rotor must be known. This information is required in order to actuate the stator windings in the right order depending on the position of the rotor. The position of the rotary field in the synchronous machine must be controlled in order to obtain a continuous movement of the rotor. During first commissioning, the position of the rotor winding of the resolver is adjusted to the rotor displacement angle of the synchronous motor by adjusting the offset. For operating a synchronous machine with an encoder, the offset must be adjusted in order to obtain perfectly true running and a maximum torque.

The correct *Offset* **1188** is adjusted when the *flux-forming voltage* **235** reaches the value 0 (approximately) while the motor is turning and has approx. the same amount for both rotation senses.

Also note the information on fine-tuning at the ending of this chapter.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Fact. Sett.**
1188	Offset	-360.0°	360.0°	0

BONFIGLIOLI servo motors of types BMD, BCR and BTD are set up during the production with an offset of 0.0°. For these motors an offset adjustment is not necessary

Automatic setup

WARNING

Injury and damage to the machine

In certain circumstances, the motor speed may reach high values.

- Decouple the motor from the load to avoid possible hazards (injury and damage to the machine).
- Lock or close off the motor shaft and make sure that no loose parts can be moved in an uncontrolled manner through a sudden acceleration of the motor shaft.

NOTICE

If a wrong direction is caused by an incorrect wiring, this is typically detected during the Automatic Offset setup.

- Always ensure before starting the setup, that the wiring is correct.

The offset can be determined and adjusted as follows:

- Set up parameter *SETUP Selection* **796** to "550 – Para-Ident. Offset, DS0" if the auto setup should be done for all data sets.
- Use for single data sets the corresponding following settings.

  After the start via *SETUP Selection* **796** the Auto-Tuning of the Offset determination is started. The status of the Auto-Tuning is displayed by *SETUP Status* **797**.
- For the Auto-Tuning a controller release via STO must be done. Switch the STO inputs when *SETUP Status* **797** shows "STO".

If a correct Offset was detected, this value is set up automatically in *Offset* **1188**. Additionally the device executes a Reset. The Absolute encoder is now tuned to the motor.

If U, V and W are connected correctly, the sense of rotation "clockwise" with a view from the front is defined on the motor shaft in accordance with DIN EN 60034-8.

Possible errors and corrections:

If during the auto-tuning error "F1420" shows up, this is an indication of an incorrect direction of rotation of the absolute encoder in comparison to the motor.

Execute one of the following actions to adjust the direction of rotation of the encoder to the motor:

- Swap two motor phases, for example U and V. Note the direction of rotation of the motor.
- Invert the direction of rotation of the resolver by exchanging Sin+ and Sin-. Ensure that the analog part and digital part of the encoder use the same sense of direction.

Manual setup

The offset can be determined and adjusted as follows:

- During first commissioning "SEtUP" will be displayed in the control unit. Press ESC to stop this operation. The guided commissioning ("SEtUP") is performed after adjusting the offset.
- Open the parameter menu "PARA" and enter the machine data indicated on the type plate or the data sheet of the motor.

Before adjusting the offset, take the following **safety precautions**:

- Disable the frequency inverter via STO (controller release).
- If possible, uncouple the motor from the load so that the motor shaft turns freely. If installed, release the mechanical brake.
- If uncoupling is not possible, make sure that the motor is loaded as little as possible.

WARNING

Injury and damage to the machine

In certain circumstances, the motor speed may reach high values.

- Decouple the motor from the load to avoid possible hazards (injury and damage to the machine).
- Lock or close off the motor shaft and make sure that no loose parts can be moved in an uncontrolled manner through a sudden acceleration of the motor shaft.
- Adapt the setting.

- Set the maximum permissible output frequency of the frequency inverter to a low frequency value via parameter *Switch-Off Limit* **417**. Select the frequency value such that uncontrolled acceleration of the motor ("overspeeding") is detected at an early stage. This limitation is necessary in order to avoid personal and material damage.
- Set parameter *Current Limit* **728** of the speed controller to a low current value (e.g. 10% of the rated motor current). In this way it is made sure that there are no excessive currents of the offset is set incorrectly.

WARNING

Unintended Starting!

If the Drive is supplied with voltage, it can suddenly start. This may lead to personal and material damage.

- Prior to starting the manual adjustments, disconnect the drive from voltage supply.
- Comply with the five safety rules.
- If possible, wear protective clothing.

- Turn motor shaft manually. Check the sense of rotation of the resolver via the actual value of parameter *Frequency Speed Sensor 3* **279**. In the case of a clock-wise rotation of the motor shaft, positive values are displayed for the actual frequency value. If the displayed sense of rotation does not correspond to the actual sense of rotation, swap the connections of the tracks A and B of the frequency inverter.

The *Offset* **1188** must be between 0° and 360°, divided by the number of motor pole pairs.

$$\text{Max. Offset} = \frac{360^\circ}{\text{number of motor pole pairs}}$$

If the adjusted value is changed by the maximum offset, this does not affect the *flux-forming voltage* **235**.

- Adjust a low reference speed value (approx. 10% lower than the Switch-off Limit Frequency 417), and enable the frequency inverter via digital input S1IND (controller release) and S2IND (start clock-wise operation) in order to accelerate the motor.
- If an overcurrent is detected or a fault message is issued due to an overload, the guided commissioning (setup) will start first. Confirm the machine and encoder data. After completion of the guided commissioning, adjust the parameter Limit Current 728 to a low value again because this value was overwritten during the guided commissioning.

Depending on the behavior of the motor after start, carry out the following steps:

Motor does not turn, or the motor shaft only turns to a new position and stops again:

- Check if the parameters *No. of Pole Pairs* **373** for the motor is set correctly.

If these values are adjusted correctly, take the following measures complying with the safety instructions.

WARNING

Live system! Electric shock hazard!

When the frequency inverter is disconnected from power supply, the mains, DC-link voltage and motor terminals may still be live for some time.

- Wait until the DC link capacitors have discharged before starting work on the unit. The time to wait is at least 3 minutes in the case of sizes 1 through 7 and at least 10 minutes in the case of size 8.

- Before electrical installation work, de-energize the frequency inverter and take appropriate precautions to make sure it is not re-energized unintentionally. Make sure that the frequency inverter is de-energized.
- Exchange two motor phases (e.g. U and V) at the frequency inverter sockets because the senses of rotation of the motor and the encoder do not correspond to each other.
- Switch on the power supply again.
- As described above, adjust a low speed reference value and start the motor.
- If the motor does not start despite the phase exchange:
  Increase the parameter value for *Offset* **1188** by 90°, divided by the no. of motor pole pairs.
- If the motor still does not turn, exchange the two motor phases (e.g. U and V) again.

  The motor turns and accelerates until it reaches the *Frequency Switch-Off Limit* **417**:
- Check the encoder lines and check the encoder connection contacts.
- In the case of fault message "Overfrequency" F1100: increase the parameter value for Offset 1188 by 180°, divided by the no. of motor pole pairs.

If the motor turns at the adjusted speed and in the right direction, carry out the fine adjustment of the offset:

- Adjust the parameter value for *Offset* **1188** in small steps (e.g. 2.5°) until the *flux-forming voltage* **235** is approximately 0.
- In case the flux-forming voltage deviates from 0 significantly, adjust the offset in bigger steps.
- In the case of a positive flux-forming voltage, increase the offset.
- In the case of a negative flux-forming voltage, reduce the offset.
- Adjust parameters *Frequency Switch-Off Limit* **417** and *Current Limit* **728** to the required values.
- Repeat the **fine adjustment** of the offset at 50% of the rated frequency.

  This completes the offset adjustment.
- Start the guided commissioning. This is required for optimum current control.

WARNING

Some absolute encoder types offer the possibility to "zeros" or change the position transmitted from the encoder. Do not use this functionality since otherwise the commutation angle for *Offset* **1188** is changed and the correct speed control cannot be guaranteed.

7.4.7 Bits/Turn

If an absolute value encoder is used (EnDat 2.1, Hiperface, SSI), the number of Bits/Turn (referred to encoder) must be configured in the frequency inverter. In the case of Hiperface and SSI encoders, the value specified in the data sheet of the encoder used must be entered in parameter *Bits/Turn* **1271**.

In the case of EnDat 2.1, the value is read automatically from the EnDat encoder and used internally. Parameter *Bits/Turn* **1271** is not evaluated in the case of EnDat encoders.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Default**
1271	Bits/Turn	0 bits/t	32 bits/t	13 bits/t

The internal resolution of one motor revolution is 16 bit. The resolution of *Bits/Turn* **1271** is converted to the internal resolution if the encoder is used as a motor encoder.

In the case of application encoders, the reference between motor and application encoder is parameterized through the gear factors *EC2 Gear Factor Numerator* **513** and *EC2 Gear Factor Denominator* **514**.

In the case of linear encoders, this value must be set according to chapter 7.3.5.

7.4.8 Bits Multiturn

If a multiturn absolute value encoder is used (EnDat 2.1, Hiperface, SSI), the number of Bits/Turn (referred to encoder) for the multiturn resolution must be configured in the frequency inverter. In the case of Hiperface and SSI encoders, the value specified in the data sheet of the encoder used must be entered in parameter *Bits Multiturn* **1272**.

In the case of EnDat 2.1, the value is read automatically from the EnDat encoder and used internally. Parameter *Bits Multiturn* **1272** is not evaluated in the case of EnDat encoders.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Default**
1272	Bits Multiturn	0 Bit	32 Bit	13 Bit

The position of the motor is resolved at a total of 31 bits + sign bit. The lower 16 bits are used for the motor position angle the higher 16 bits are used for the number of motor revolutions and the sign.

If the absolute value encoder is used as a motor encoder, the following shall apply:

If the number of *Bits Multiturn* **1272** is smaller than 16 bits, the missing bits are filled internally in the frequency inverter. These additional bits are used for overflow saving of the revolutions so that 2^{16} revolutions (including one sign bit) can be managed safe against zero voltage.

If the number of *Bits Multiturn* **1272** is greater than 16 bits, the accuracy of the encoder exceeds the accuracy of the inherent resolution of the frequency inverter.

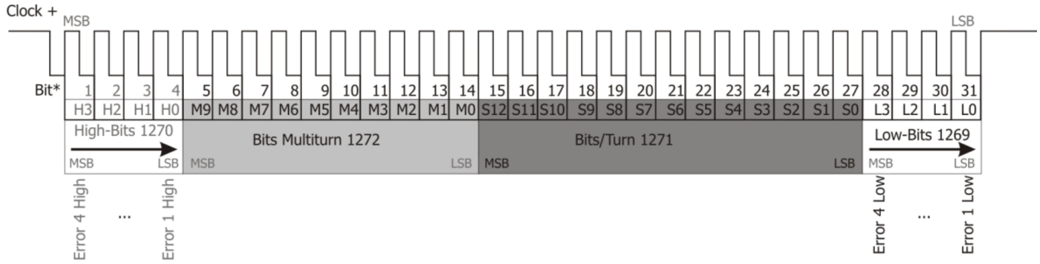
In the case of application encoders, the reference between motor and application encoder is parameterized through the gear factors *EC2 Gear Factor Numerator* **513** and *EC2 Gear Factor Denominator* **514**.

In the case of linear encoders, this value must be set according to chapter 7.3.5.

7.4.9 SSI: error/additional bits

If SSI encoders are used, the available error/additional bits of the encoder can be masked for evaluation. Many encoders use one or more bits for error signaling. In some cases, the bits are also used for transmitting additional information not required for encoder evaluation in the frequency inverter.

Arrangement SSI-Bits (number of individual bits as an example)



Bit*: The bits are shown from left to right corresponding to the time transmission, the most significant bit (MSB) is transmitted first, the least significant bit (LSB) last. The number of bits is shown as an example in the illustration. Multiturn bits are only present in the case of Multiturn encoders. The additional bits "Low bits" are used by many encoder manufacturers with 1 or 3 bits. The additional "High bits" are only used very rarely by encoder manufacturers.

Depending on the function intended by the encoder manufacturer, an error bit "High" or "Low" may trigger an error.

Parameters **1269** *SSI: Error-/Extra Bits (Low)* and **1270** *Error-/Extra Bits (High)* can evaluate up to eight error bits each. SSI error MSBits is used for the definition the most significant bits, and SSI error LSBits is used for the less significant bits.

To determine the total data width, the two parameters must always be defined. This definition is also required if no evaluation is to take place. In this case, all bits must be masked as "Don't care" with an "X" in the string.

If no error bits or other bits are present ("empty string"), a dash "" must be parameterized. The input always begins with the MSB.

The following values are permissible:

H: When the bit is "High", error F172A or F172B will be triggered.

L: When the bit is "Low", error F172A or F172B will be triggered.

X: No error will be triggered for the bit, regardless of its status.

-: Number of bits = 0 (use in this case only).

Lowercase letters can be used alternatively in the entry. This parameter cannot be entered by means of KP500. Other values cannot be entered.

Special case: Number of bits = 0:

SSI additional bits in the High range are not used by many encoder manufacturers. In these cases set the parameter to value "-" (dash).

Parameters	Settings			
No.	**Description**	**Min.**	**Max.**	**Default**
1269	SSI: Error-/Extra-Bits (Low)	Special, see text		„-"
1270	SSI: Error-/Extra-Bits (High)			„-"

Due to the shifting of the usable data by the error/additional bits, the number of error/additional bits must always be specified correctly.

7.4.9.1 Example 1

Additional bits (High)	Multiturn bits	Singleturn bits	Additional bits (Low)
0	8	16	Total 1 to be evaluated. "High" is an error situation.

SSI: Error-/Extra-Bits (High) **1270** = "-"

Bits Multiturn. **1272** = 8

Bits/Turn **1271** = 16

SSI: Error-/Extra-Bits (Low) **1269** = "-"

7.4.9.2 Example 2

Additional bits (High)	Multiturn bits	Singleturn bits	Additional bits (Low)
0	12	16	Total 4, the second one is to be evaluated. "Low" is an error situation.

SSI: Error-/Extra-Bits (High) **1270** = "-"

Bits Multiturn. **1272** = 12

Bits/Turn **1271** = 16

SSI: Error-/Extra-Bits (Low) **1269**= "XLXX"

7.4.9.3 Example 3

Additional bits (High)	Multiturn bits	Singleturn bits	Additional bits (Low)
Total 2, the first one is to be evaluated. "High" is an error situation.	8	16	Total 4, the second one is to be evaluated. "Low" is an error situation.

SSI: Error-/Extra-Bits (High) **1270** = "HX"

Bits Multiturn. **1272** = 8

Bits/Turn **1271** = 16

SSI: Error-/Extra-Bits (Low) **1269**= "XLXX"

7.4.9.4 Example 4

Additional bits (High)	Multiturn bits	Singleturn bits	Additional bits (Low)
0	8	16	Encoder has 4 toggle bits all of which are to be ignored.

SSI: Error-/Extra-Bits (High) **1270** = "-"

Bits Multiturn. **1272** = 8

Bits/Turn **1271** = 16

SSI: Error-/Extra-Bits (Low) **1269**= "XXXX"

7.4.10 SSI: Sampling interval

SSI frequency encoders often use a sampling rate in the millisecond range. In order for the evaluation in the device to work correctly, the sampling rate of the SSI absolute value encoder must be set up. If the sampling rate of the encoder cannot be adjusted, use the next higher, available setting. The parameter value is adjusted as a multiplier of 125 us.

Not all steps from 0 to 240 are available. The selection list limits the available options to reasonable settings.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Default**
1268	SSI: Sampling interval	0	240	0

NOTICE

For a good positioning behavior, the sampling rate is to be less than 1 ms.

In the case of higher sampling rates, unwanted high system vibration or even machine damage may occur, if the speed and position controller are configured improperly.

In the case of high sampling rates (> 2 ms), reduce the dynamics of the system via the speed controller and the position controller.

Positioning accuracy will be lower in the case of high sampling rates. For precise applications, use encoders with low sampling rates.

7.4.11 Gear factor speed sensor 3

If the speed sensor is coupled to the motor via one or more gears, the transmission ratio between the motor and the encoder must be configured via *EC3 Gear Factor Numerator* **1473** and *EC3 Gear Factor Denominator* **1474**.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Default**
1473	EC3 Gear Factor Numerator	-300.00	300.00	1.00
1474	EC3 Gear Factor Denominator	0.01	300.00	1.00

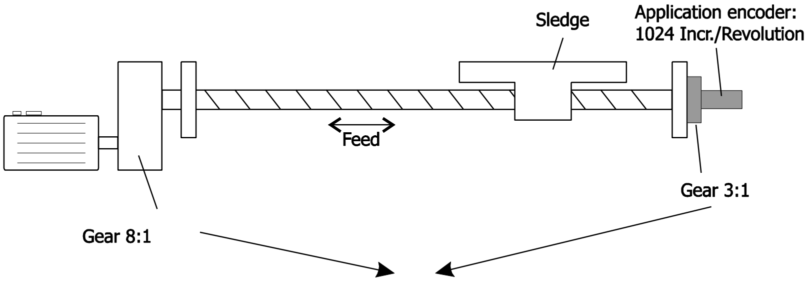
$$\frac{\text{Revolution s of the Motor shaft}}{\text{Revolution s of the EC3 encoder shaft}} = \frac{EC\,3\,Gear\,Factor\,Numerator\,\mathbf{1473}}{EC\,3\,Gear\,Factor\,Denominatır\,\mathbf{1474}}$$

Gear factors *EC3 Gear Factor Numerator* **1473** and *EC3 Gear Factor Denominator* **1474** must always be referred to the motor.

In the case of linear encoders, this value must be set according to chapter 7.3.5.

Example

On a linear axis, the motor is flange-connected via a gear (transmission ratio 8:1) and the application connector is flange-connected via a second gear (transmission ratio 3:1).



EC3 Gear Factor Numerator **1473**
EC3 Gear Factor Denominator **1474**

1 motor revolution = 1/8 turn on output side
= 1/8x3 encoder turn

$$\frac{EC\,3\,Gear\,Factor\,Numerator\,\mathbf{1473}}{EC\,3\,Gear\,Factor\,Denominatır\,\mathbf{1474}} = \frac{\text{Revolution s of the Motor shaft}}{\text{Revolution s of the EC3 encoder shaft}} = \frac{8}{3}$$

7.4.12 Warning Dig. Encoder

Via parameter *Warning Dig. Encoder* **1274**, the current warning status of EnDat 2.1 encoders is displayed. This information can be used for analyzing and eliminating application problems. Parameter *Warning Dig. Encoder* **1274** shows the current warning with an abbreviation. For evaluation via field bus, parameter *Warning Dig. Encoder* **1273** with the warning value in hexadecimal representation can be used. By addition of the values, several warnings can be displayed simultaneously.

EnDat 2.1 warnings			
Abbreviation in *Warning Dig. Encoder* **1274**	**Bit code** *Warning Dig. Encoder* **1273**		**Meaning**
	Bit	**Value**	
Fcoll	0	0x0001	Frequency collision
Temp	1	0x0002	Temperature exceeded
Illum	2	0x0004	Control reserve lighting
Batt	3	0x0008	Battery status
Ref	4	0x0010	Reference point

Warnings which are present at the same time are represented by the bit combination or mathematical addition.

Present warnings can be displayed via the application warning mask in Bit 9.

7.5 Instructions on speed-controlled configurations ("Not x40")

In the case of speed-controlled configurations, an encoder is typically installed. Normally, this encoder is connected to the motor.

An internal format (referred to as 16/16) is used for speed control. The 16 less significant bits represent the position angle on a motor revolution, the 16 more significant bits represent the number of motor revolutions.

If absolute value encoders are used, the absolute value encoder notation is converted to the internal notation. This is why, for proper function, the parameters of the absolute value encoder must be entered in accordance with the data sheet. In the case of other parameterizations, unwanted malfunction of the drive might occur.

7.6 Instructions on positioning (configuration x40)

7.6.1 Settings for electronic gear in configuration x40

If the function "electronic gear" of the positioning in ANG (configuration x40) is used in a slave, synchronization via SYNC or RxPDO1 must be set via systembus. Please check the following settings:

Use of RxPDO A Master Identifier must correspond to the Slave Identifier.	
Master	**Slave**
925 *TxPDO1 Identifier* **926** *TxPDO2 Identifier* **927** *TxPDO3 Identifier*	**924** *RxPDO1 Identifier*
930 *TxPDO1 Function* **932** *TxPDO2 Function* **934** *TxPDO3 Function*	**936** *RxPDO1 Function* = 1 - controlled by SYNC (recommended)
	1180 *Operation mode* = 1- RxPDO

Use of SYNC The Master Sync Identifier must correspond to the Slave Sync Identifier (e.g. 0 → Predefined Set 0x80 = 128).	
Master	**Slave**
	936 *RxPDO1 Function* = 1 - controlled by SYNC (recommended)
918 *Sync Identifier* **919** *Sync Time*	**918** *Sync Identifier* **1180** *Operation mode*= 10-SYNC

Operation mode **1180** ensures synchronization of the operating systems of different devices and must be set up in configuration x40 in one of the two ways described.

RxPDO1 Function **936** should be set to "`1 – controlled by SYNC`" in order to synchronize the master position with the OS in the slave. Although this setting is optional, BONFIGLIOLI VECTRON recommends setting this parameter accordingly.

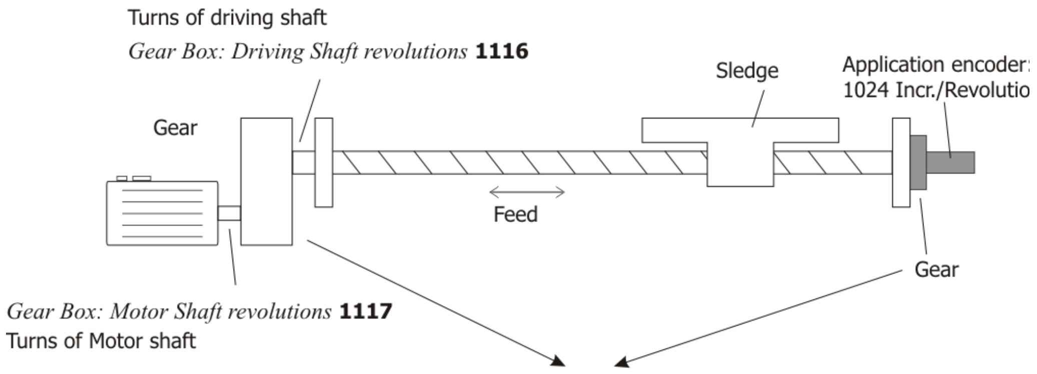
If positioning (configuration x40) and an absolute value encoder are used, a distinction is made for parameterization between "motor encoders" and "application encoders".

The motor encoder is always needed for speed control and can also be used for position control in the case of no-slip systems.

An application encoder for position control is used in systems where slip may occur for slip compensation. This encoder is also often referred to as an "External encoder" or "Synchronous encoder".

With the module, the following configurations are possible:

System slip	Motor type	Configuration
no-slip system, **high speed precision:** absolute value encoder at motor for speed control and position control	synchronous servomotor & asynchronous motor	540 & 240
no-slip system, **low speed precision:** absolute value encoder as application encoder for position control of motor model with speed control	synchronous servomotor	640
slipping system, **high speed precision:** absolute value encoder as application encoder for position control HTL encoder (ASM) or Resolver (PMSM) as motor encoder for speed control	synchronous servomotor & asynchronous motor	540 & 240
slipping system, **low speed precision:** absolute value encoder as application encoder for position control motor model for speed control	synchronous servomotor & asynchronous motor	640 & 440



EC Gear Factor Numerator

An internal format (referred to as 16/16) is used for speed control and calculation of the positioning trajectory. The 16 less significant bits represent the position angle on a motor revolution, the 16 more significant bits represent the number of motor revolutions.

The positioning offers the user so-called "user units" (abbreviation [u]), which enable adjustment to any application via the reference system. In this way, the resolution of the smallest unit for positioning can be parameterized (e.g. 1 mm, 4 mm, 0.01 °, etc.).

For more information on the reference system, refer to the application manual "Positioning".

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Default**
1115	Feed constant	1 u/U	2^{31}-1 u/U	65536 u/U
1116	Gear Box: Driving Shaft Revolutions	1	65 535	1
1117	Gear Box: Motor Shaft Revolutions	1	65 535	1

For application encoders, a gear transmission between the application encoder and motor must be parameterized via a gear factor.

The conversions between the different reference systems are done automatically, the user sets the target values in user units referred to the distance.

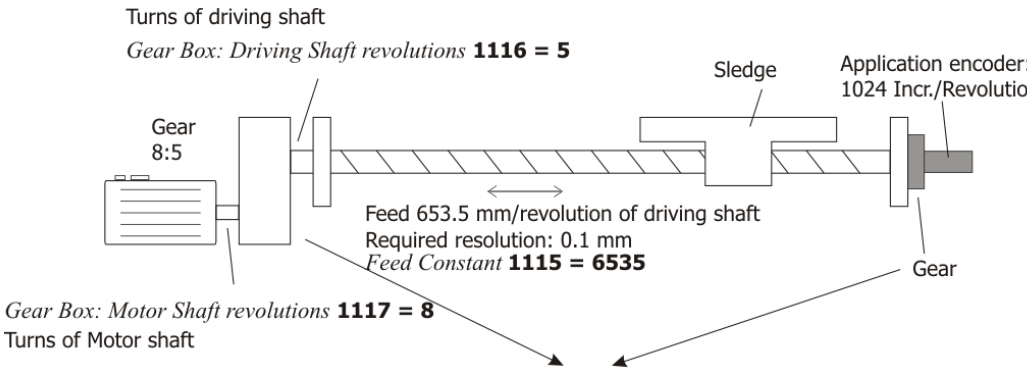
7.6.2 Example

For parameterization of a linear slide, the following properties are known:

Motor gear ratio: 8:5

Application encoder gear ratio: 7:3

Feed rate of linear axis: 635.5 mm/revolution of the output shaft


Turns of driving shaft
Gear Box: Driving Shaft revolutions **1116 = 5**
Sledge
Application encoder:
1024 Incr./Revolutio
Gear
8:5
Feed 653.5 mm/revolution of driving shaft
Required resolution: 0.1 mm
Feed Constant **1115 = 6535**
Gear
Gear Box: Motor Shaft revolutions **1117 = 8**
Turns of Motor shaft

EC3 Gear Factor Numerator

This results in the following parameterization:

Feed constant **1115 =** 6535 rev

Gear shaft turns **1116** = 5

Gear motor turns **1117** = 8

EC3 Gear Factor Numerator **1473** = 24

EC3 Gear Factor Denominator **1474** = 35

In order to move by 1 mm, a positioning order of 10 u must be executed.

In the case of linear systems, the feed constant is typically specified in the data sheet. If this value is unknown, it must be determined empirically. For empirical determination of the feed constant, refer to application manual "Positioning".

7.6.3 Homing

When it comes to positioning, homing may be required or recommended, depending on the application. If no absolute value encoder is used, homing to a known point (e.g. reference cam or limit switch) will typically be performed first upon restoration of mains supply.

When an absolute value encoder is used, homing during operation is unwanted in many situations. If homing is not to be performed during operation, you can set *Operation mode* **1228** = "`10 – No homing`".

By using different frequency inverter data sets, you can configure a setup mode, including homing, and a normal operation mode.

7.7 Actual speed source

The rotary encoder is selected via *Actual Speed Source* **766**. In the default setting, speed sensor 1 is used as the source of actual speed.

Actual speed source 766	Function
1 - Speed sensor 1	The actual speed source is speed sensor 1 of the basic device.
2 - Speed sensor 2	The actual speed source is speed sensor 2 (Resolver or TTL at X410) of the Automation Interface EM-AUT-31.
3 - Motor model	The actual speed source is the motor model of the frequency inverter.
10 - Speed sensor 3	The actual speed source is speed sensor 3 (X412) of the Automation Interface EM-AUT-31.

Setting "`3-Motor model`" is visible and available in configurations 4xx and 6xx only.

7.8 Actual position source

In positioning applications (configurations x40), the actual position source must be set. This is done via *Actual Position Source* **1141**. In the basic setting, the actual value source of the speed control is used as the actual position source.

Actual Position Source 1141	Function
like **766** *Actual speed source*	The actual speed source is the actual position source at the same time (Default).
Encoder 1	The actual position source is speed sensor 1 of the basic device.
Encoder 2	The actual position source is speed sensor 2 (Resolver or TTL at X410) of the Automation Interface EM-AUT-31.
Encoder 3	The actual position source is speed sensor 3 (X412) of the Automation Interface EM-AUT-31.

8 Ethernet/IP communication

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

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, the CIP-defined output assemblies 20/21/22/24. For AC Drives input assemblies 70/71/72/74 are available.

In addition, the vendor-specific 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 control "*interpolated position mode*" and "*cyclic synchronous position mode*".

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

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).

I/O

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

- Polled
- Change of state
- Cyclic

Polled

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

Change of state

With the "`Change of state`" setting, the PLC sends OUT data whenever the OUT data value has changed. The inverter sends IN data whenever the IN data value has changed. To minimize sending repeated messages in short time intervals due to small fluctuations (i.e. speed), the Ethernet/IP parameters **P.34** (*COS Mask 0*, 32 Bit) … **P.39** (*COS Mask 5*, 32 Bit) **of the PLC** can be used to define a bit mask that **excludes** the selected bits while evaluating a value for changes. Every bit of the *COS Mask* corresponds to a bit in the inverter OUT data area. If a *COS Mask* bit is set to "`1`" this bit causes sending OUT data on change. If a *COS Mask* bit is set to "`0`" this bit has no effect on sending OUT data.

For further details and an example, refer to Chapter "Change of State (COS) Mask, Parameters 34 to 39".

Cyclic

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

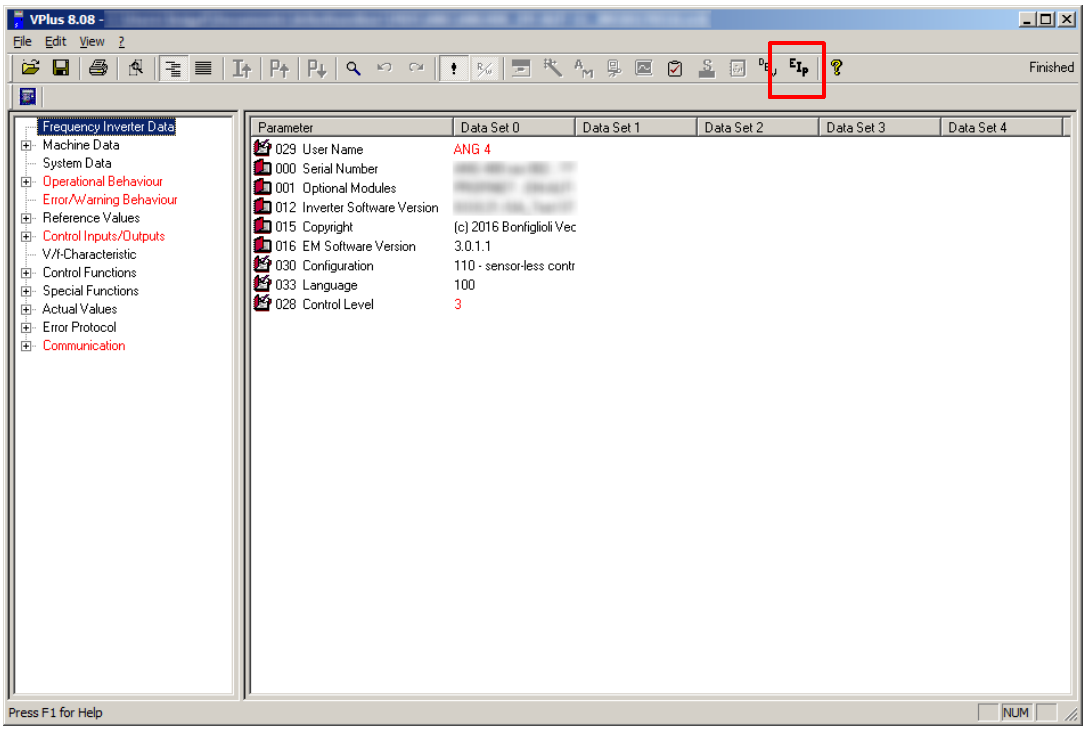
I/O-Connections supported in ANG:

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

8.1 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 are created with the Bonfiglioli Vectron inverter configuration utility VPlus using the "*Create Ethernet/IP EDS File*" function. There are two different methods available. If you are connected to an inverter by PC and service adapter KP232 or Ethernet, an EDS file can be created online. If a vcb-file is available for the inverter, then an EDS file can be created offline.



- Frequency Inverter Data
- Machine Data
- System Data
- Operational Behaviour
- Error/Warning Behaviour
- Reference Values
- Control Inputs/Outputs
- V/f-Characteristic
- Control Functions
- Special Functions
- Actual Values
- Error Protocol
- Communication

Parameter	Data Set 0	Data Set 1	Data Set 2	Data Set 3	Data Set 4
029 User Name	ANG 4				
000 Serial Number	[redacted]				
001 Optional Modules	[redacted]				
012 Inverter Software Version	[redacted]				
015 Copyright	(c) 2016 Bonfiglioli Vec				
016 EM Software Version	3.0.1.1				
030 Configuration	110 - sensor-less contr				
033 Language	100				
028 Control Level	3				

Figure 8-1: Creating EDS files

Online

The "*Create Ethernet/IP EDS File*" function can be started from the Inverter Manager window:

- select an inverter in the inverter tree
- select menu Edit\Inverter Menu\Create Ethernet/IP EDS File

or

- select an inverter in the inverter tree
- click the "*Create Ethernet/IP EDS File*" button on the Inverter Manager toolbar

The "*Create Ethernet/IP EDS File*" function can be started from the Parameter window:

- select menu Edit\Create Ethernet/IP EDS File

or

- click the "*Create Ethernet/IP EDS File*" button on the Parameter window toolbar

Offline

The "*Create Ethernet/IP EDS File*" function can be started from the Parameter window:

- select menu Edit\Create Ethernet/IP EDS File

or

- click the "*Create Ethernet/IP EDS File*" button on the Parameter window tool bar

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.

NOTICE

Risk of component damage!

- When changing an EDS filename, the file extension MUST remain "eds".

ANG****.tpl template files are stored in folder VPlus\EIP\ANG.

- NEVER delete these files.

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).

8.2 Supported configurations

ANG inverters support different types of control and reference values:

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

Motion control configurations are set when parameter *Configuration* **30** = `x40` (in this example `240`). To use the full functionality of the Motion Control Interface Parameter *Local/Remote* **412** = `1-Control via State machine` must be set.

The inverter's behavior with respect to *control word / status word* and *modes of operation / modes of operation display* varies in the two different types of configuration.

Standard:

Necessary settings:

- *Configuration* **30** ≠ `x40`.
- *Local/Remote* **412** = `2 - Control via Remote-Contacts`

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:

- Analogue input
- Fixed values from parameters
- Target velocity

Percentage reference value for technology controller or Torque control:

- Analogue input
- Fixed values from parameters

Please refer to chapter "Configurations without motion control" for the control without Positioning functionality.

Positioning via contacts (or remote contacts):

Necessary settings:

- *Configuration* **30** = `x40`.
- *Local/Remote* **412** = `0 - Control via Contacts` or `2 - Control via Remote-Contacts`

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

Please refer also to the application manual "Positioning".

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

Necessary settings:

- *Configuration* **30** = `x40`.
- *Local/Remote* **412** = `1 - Control via 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 this manual in Chapters "Motion Control Interface (MCI)" and "Motion control configurations".

8.3 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)

8.4 Ethernet/IP overview

8.4.1 Class 0x01, Identity Object

Instance	Attribute	Name	Service	Data Type	Value
0	1	Revision	get	UINT	1
1	1	VendorID	get	UINT	**1202** ***Bonfiglioli*** ***Vectron MDS 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 0x01
	5	Status	get	WORD	current device state
	6	Serial Number	get	UDINT	
	7	Product Name	get	SHORT_STRING	
	8	State	get	USINT	0 – nonexistent 1 – device self testing 2 – Standby 3 – operational 4 – major recoverable fault 5 – major unrecoverable fault
	9	Configuration consistency check	get	UINT	

8.4.2 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	2
1	1	Status	get	DWORD	see below
	2	Configuration Capability	get	DWORD	0x00000014

Instance	Attribute	Name	Service	Data Type	Value
					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 IPmulticast packets)	get/set1)	USINT	see below
	9	Mcast Config - Alloc Control(Multicast addressallocation control word) - Reserved - Num Mcast(No. of IP multicast addreses to allocate forEtherNet/IP) - Mcast Start Addr(Starting multicastaddress from which tobegin allocation)	get/set1)	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.

8.4.3 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	3
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.

Bit(s)	Name	Value
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-XX-XX", starting with the first octet.

8.4.4 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).

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
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

8.4.5 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 e.g. setting the input/output assembly selection.

Instance	Attribute	Name	Service	Data Type	Value
0	2	Max Instance	get	UINT	129
	8	Parameter Class Descriptor	get	WORD	5 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 don't 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

8.4.5.1 Input Assembly Selection, Parameter 1

Instance	Attribute	Name	Service	Data Type	Value
1	1	Param1 InputAssemblySelection	get/set	UINT	70 – Assembly 70 71 – Assembly 71 72 – Assembly 72 74 – Assembly 74 101 – Assembly 101

Parameter **1** "*InputAssemblySelection*" defines the used input assembly. The object content of the different assemblies is described in Chapter "I/O Assemblies".

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 please refer to Chapter "Vendor Specific Assemblies 100/101".

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

8.4.5.2 Input Assembly Length, Parameter 2

Instance	Attribute	Name	Service	Data Type	Value
2	1	Param2 InputAssemblyLength	get	UINT	Depends on selected input assembly (and variable mapping assembly 101) 1 … 24 bytes

Parameter **2** "*InputAssemblyLength*" contains the length of the currently set input assembly.

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

Instance	Attribute	Name	Service	Data Type	Value
3	1	Param3 InputObject1	get/set	UINT	See list
4	1	Param4 InputObject2	get/set	UINT	
5	1	Param5 InputObject3	get/set	UINT	
6	1	Param6 InputObject4	get/set	UINT	
7	1	Param7 InputObject5	get/set	UINT	
8	1	Param8 InputObject6	get/set	UINT	
9	1	Param9 InputObject7	get/set	UINT	
10	1	Param10 InputObject8	get/set	UINT	
11	1	Param11 InputObject9	get/set	UINT	
12	1	Param12 InputObject10	get/set	UINT	
13	1	Param13 InputObject11	get/set	UINT	
14	1	Param14 InputObject12	get/set	UINT	

Parameter **3** *InputObject1* to Parameter **14** *InputObject12*, define the content of vendor specific Input Assembly 101. A choice list is available for these parameters.

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

Choice list for Parameters **3** to **14**:

```
–  0 "- no object mapped here",
–  41 "0x3001 (2B) Digital In actual values",
–  42 "0x3002 (2B) Digital Out actual values",
–  44 "0x3004 (2B) Boolean Mux",
–  47 "0x3007 (2B) Percentage actual value",
–  48 "0x3008 (2B) Percentage actual value2",
–  49 "0x3011 (2B) Act. value word1 (P.1415)",
–  50 "0x3012 (2B) Act. value word2 (P.1416)",
–  51 "0x3021 (4B) Act. value long1 (P.1417)",
–  52 "0x3022 (4B) Act. value long2 (P.1418)",
–  76 "0x5FF0 (1B) active motion block (ttr)",
–  77 "0x5FF1 (1B) motion block to resume (ttr)",
–  81 "0x6041 (2B) statusword",
–  83 "0x6043 (2B) velocity demand",
–  84 "0x6044 (2B) control effort",
–  94 "0x6061 (1B) modes of operation display",
–  95 "0x6064 (4B) position actual value",
–  100 "0x606C (4B) velocity actual value",
–  106 "0x6077 (2B) Torque actual value",
–  107 "0x6078 (2B) Current actual value",
–  108 "0x6079 (4B) DC link voltage",
–  126 "0x60F4 (4B) following error value";
```

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 please refer to Chapter "Vendor Specific Assemblies 100/101".

8.4.5.4 Output Assembly Selection, Parameter 15

Instance	Attribute	Name	Service	Data Type	Value
15	1	Param15 OutputAssemblySelection	get/set	UINT	20 – Assembly 20 21 – Assembly 21 22 – Assembly 22 24 – Assembly 24 100 – Assembly 100

Parameter **15** "*OutputAssemblySelection*" defines the used output assembly. The object content of the different assemblies is described in Chapter "I/O Assemblies".

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 please refer to Chapter "Vendor Specific Assemblies 100/101".

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

8.4.5.5 Output Assembly Length, Parameter 16

Instance	Attribute	Name	Service	Data Type	Value
16	1	Param16 OutputAssemblyLength	get	UINT	Depends on selected input assembly (and variable mapping assembly 100) 1 … 24 bytes

Parameter 16 "*OutputAssemblyLength*" contains the length of the currently set output assembly.

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

Instance	Attribute	Name	Service	Data Type	Value
17	1	Param17 OutputObject1	get/set	UINT	See list
18	1	Param18 OutputObject2	get/set	UINT	
19	1	Param19 InputObject3	get/set	UINT	
20	1	Param20 OutputObject4	get/set	UINT	
21	1	Param21 InputObject5	get/set	UINT	
22	1	Param22 OutputObject6	get/set	UINT	
23	1	Param23 OutputObject7	get/set	UINT	
24	1	Param24 OutputObject8	get/set	UINT	
25	1	Param25 OutputObject9	get/set	UINT	
26	1	Param126 OutputObject10	get/set	UINT	
27	1	Param127 OutputObject11	get/set	UINT	
28	1	Param28 OutputObject12	get/set	UINT	

Parameter **17** *OutputObject1* to Parameter **28** *OutputObject12* define the content of vendor-specific Output Assembly 100. A choice list is available for these parameters.

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

Choice list for Parameters **17** to **28**:

- `0 "- no object mapped here",`
- `43 "0x3003 (1B) Digital Out set values",`
- `45 "0x3005 (2B) Boolean Demux",`
- `46 "0x3006 (2B) Percentage set value",`
- `53 "0x3111 (2B) Ref. value word1 (S.762)",`
- `54 "0x3112 (2B) Ref. value word2 (S.763)",`
- `55 "0x3121 (4B) Ref. value long1 (S.764)",`
- `56 "0x3122 (4B) Ref. value long2 (S.765)",`
- `57 "0x5F10_1 (2B) Numerator",`
- `58 "0x5F10_2 (2B) Denominator",`
- `80 "0x6040 (2B) controlword",`

- 82 "0x6042 (2B) target velocity",
- 93 "0x6060 (1B) modes of operation",
- 105 "0x6071 (2B) Target torque",
- 109 "0x607A (4B) target position",
- 111 "0x6081 (4B) profile velocity",
- 112 "0x6083 (4B) profile acceleration",
- 113 "0x6084 (4B) profile deceleration",
- 114 "0x6085 (4B) quick stop deceleration",
- 128 "0x60FF (4B) target velocity";

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 please refer to Chapter "Vendor Specific Assemblies 100/101".

8.4.5.7 Reserved Parameters 29 - 32

Parameters **29** to **32** have no function and are reserved for future use.

8.4.5.8 Ethernet/IP Abort Behavior, Parameter 33

Instance	Attribute	Name	Service	Data Type	Value
33	1	Param33 Ethernet/IP Abort Behavior	get	USINT	See list

Parameter **33** "*Ethernet/IP Abort Behavior*" defines inverter behavior in the event of bus errors.

This has the same effect as writing Inverter Parameter **388** "*Error Behavior*".

Choice list for parameter **33**:

- 0 no reaction
- 1 fault
- 2 stop (without fault message)
- 3 emergency stop (without fault message)
- 4 stop + fault
- 5 emergency stop + fault

Settings 2, 3, 4 and 5 are only executed as described by the inverter, if Parameter *Local/Remote* **412** is set to "1 - Remote State Machine". With all other settings of **P.412,** the settings 2, 3, 4 and 5 trigger a fault reaction as in choice 1.

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

Instance	Attribute	Name	Service	Data Type	Value
34	1	Param34 COS Mask 0	get/set	DWORD	0x00000000 to 0xFFFFFFFF (Default = 0xFFFFFFFF)
35	1	Param35 COS Mask 1	get/set	DWORD	
36	1	Param36 COS Mask 2	get/set	DWORD	
37	1	Param37 COS Mask 3	get/set	DWORD	
38	1	Param38 COS Mask 4	get/set	DWORD	
39	1	Param39 COS Mask 5	get/set	DWORD	

Parameters **34** "*COS Mask 0*" to **39** "*COS Mask 5*" define a bit mask for the COS function. These can be used to exclude small fluctuations of actual values (i.e. small speed changes) in the evaluation of whether a value has changed, and therefore reduce the number of transmissions.

In the case of input data communication type *Change-Of-State* (COS), the *COS Mask* defines the bits of the *Input Assembly* that are taken into account for the COS function. Every *Input Assembly* consists of a specific number of bytes. Every bit in the *Input Assembly* bytes corresponds to one bit in the *COS Mask*.

If the corresponding bit in the *COS Mask* is set to "1", then this bit in the *Input Assembly* is used for the COS function.

If the corresponding bit in the *COS Mask* is set to "0", then this bit in the *Input Assembly* has no effect for the COS function. A change of this bit in the *Input Assembly* does not trigger the sending of the *Input Assembly*.

The maximum length of the *Input Assembly* is 24 bytes = 192 bits. The six *COS Masks* have 32 bits each and also 192 bits in total.

The relationship between *COS Mask* and *Input Assembly* is defined as:

Input Assembly Byte No.	**COS Mask**	**Instance (Parameter)**
0 – 3	COS Mask 0	34
4 – 7	COS Mask 1	35
8 – 11	COS Mask 2	36
12 – 15	COS Mask 3	37
16 – 19	COS Mask 4	38
20 – 23	COS Mask 5	39

Example:

Input Assembly byte **0** and low byte of *COS Mask* **0**.

	COS Mask 0 / Byte 0 / Bit 0 – 7						
Bit	**7**	**6**	**5**	**4**	**3**	**2**	**1**
	1	1	1	1	1	**0**	**0**
	n	**n**	**n**	**n**	**n**	x	X
Bit	7	6	5	4	3	2	1
	Input Assembly / Byte 0 / Bit 0 – 7						

The low byte of *COS Mask* **0** is set to 0xF8. Any change in bits 0 – 2 of the *Input Assembly* has no effect. A change in bits 3 – 7 triggers the sending of the *Input Assembly*.

8.4.5.10 Reserved Parameter 40

Instance	**Attribute**	**Name**	**Service**	**Data Type**	**Value**
40	1	Param40	-	-	reserved for future use

Parameter **40** has no function and is reserved for future use.

8.4.5.11 CANopen Objects, Parameters 41 – 129

Parameters **41** to **129** are CANopen objects. Some of these objects can be mapped to Input Assembly 101 or Output Assembly 100.

See Chapter "Input Assembly Mapping for Vendor Specific Input Assembly 101, Parameters 3 to 14" and Chapter "Output Assembly Mapping for Vendor specific Output Assembly 100, Parameters 17 to 28".

The use of these parameters is described in Chapter 9.9 "CANopen objects".

All CANopen objects can be accessed by explicit message.

The use of CANopen objects with vendor specific Input Assembly 101 and Output Assembly 100 provides CANopen functionality (e.g. motion control functions) to the Ethernet/IP system.

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

"*Interpolated position mode*" requires equidistant timing for output/input assemblies, which is not supported by Ethernet/IP.

Instance	Attribute	Name	Service	Data Type
41	1	Param41 CANopen_3001 Digital in actual value	get	WORD
42	1	Param42 CANopen_3002 Digital out actual value	get	WORD
43	1	Param43 CANopen_3003 Digital out set values	get/set	USINT
44	1	Param44 CANopen_3004 Boolean Mux	get	WORD
45	1	Param45 CANopen_3005 Boolean Demux	get/set	WORD
46	1	Param46 CANopen_3006 Percentage set value	get/set	INT
47	1	Param47 CANopen_3007 Percentage actual value	get	INT
48	1	Param48 CANopen_3008 Percentage actual value2	get	INT
49	1	Param49 CANopen_3011 Act. Value Word 1 p.1415	get	INT
50	1	Param50 CANopen_3012 Act. Value Word 2 p.1416	get	INT
51	1	Param51 CANopen_3021 Act. Value Long 1 p.1417	get	DINT
52	1	Param52 CANopen_3022 Act. Value Long 2 p.1418	get	DINT
53	1	Param53 CANopen_3111 Ref. Value Word 1 s.762	get/set	INT
54	1	Param54 CANopen_3112 Ref. Value Word 2 s.763	get/set	INT
55	1	Param55 CANopen_3121 Ref. Value Long 1 s.764	get/set	DINT
56	1	Param56 CANopen_3122 Ref. Value Long 2 s.765	get/set	DINT
57	1	Param57 CANopen_5F10 Gear Factor Numerator	get/set	INT
58	1	Param58 CANopen_5F10 Gear Factor Denominator	get/set	UINT
59	1	Param59 CANopen_5F10 Resync on gear factor change	get/set	INT16
60	1	Param60 CANopen_5F11_1 Phasing 1 Offset	get/set	DINT
61	1	Param61 CANopen_5F11_2 Phasing 1 Speed	get/set	UDINT
62	1	Param62 CANopen_5F11_3 Phasing 1 Acceleration	get/set	UDINT
63	1	Param63 CANopen_5F12_1 Phasing 2 Offset	get/set	DINT
64	1	Param64 CANopen_5F12_2 Phasing 2 Speed	get/set	UDINT
65	1	Param65 CANopen_5F12_3 Phasing 2 Acceleration	get/set	UDINT
66	1	Param66 CANopen_5F13_1 Phasing 3 Offset	get/set	DINT
67	1	Param67 CANopen_5F13_2 Phasing 3 Speed	get/set	UDINT
68	1	Param68 CANopen_5F13_3 Phasing 3 Acceleration	get/set	UDINT
69	1	Param69 CANopen_5F14_1 Phasing 4 Offset	get/set	DINT
70	1	Param70 CANopen_5F14_2 Phasing 4 Speed	get/set	UDINT

Instance	Attribute	Name	Service	Data Type
71	1	Param71 CANopen_5F14_3 Phasing 4 Acceleration	get/set	UDINT
72	1	Param72 CANopen_5F15 In Gear Threshold	get/set	UDINT
73	1	Param73 CANopen_5F16 In Gear Time	get/set	UINT
74	1	Param74 CANopen_5F17_1 Position Controller Time constant	get/set	INT
75	1	Param75 CANopen_5F17_2 Position Controller Limitation	get/set	DINT
76	1	Param76 CANopen_5FF0 active motion block	get	SINT
77	1	Param77 CANopen_5FF1 motion block to resume	get	SINT
78	1	Param78 CANopen_6007 abort connection option code	get/set	INT
79	1	Param79 CANopen_603F error code	get	UINT
80	1	Param80 CANopen_6040 controlword	get/set	WORD
81	1	Param81 CANopen_6041 statusword	get	WORD
82	1	Param82 CANopen_6042 target velocity	get/set	INT
83	1	Param83 CANopen_6043 velocity demand	get	INT
84	1	Param84 CANopen_6044 control effort	get	INT
85	1	Param85 CANopen_6046_1 velocity min amount	get/set	UDINT
86	1	Param86 CANopen_6046_2 velocity max amount	get/set	UDINT
87	1	Param87 CANopen_6048_1 velocity acceleration delta speed	get/set	UDINT
88	1	Param88 CANopen_6048_2 velocity acceleration delta time	get	UINT
89	1	Param89 CANopen_6049_1 velocity deceleration delta speed	get/set	UDINT
90	1	Param90 CANopen_6049_2 velocity deceleration delta time	get/set	UINT
91	1	Param91 CANopen_604A_1 velocity quick stop delta speed	get/set	UDINT
92	1	Param92 CANopen_604A_2 velocity quick stop delta time	get/set	UINT
93	1	Param93 CANopen_6060 modes of operation	get/set	SINT
94	1	Param94 CANopen_6061 modes of operation display	get	SINT
95	1	Param95 CANopen_6064 position actual value	get	DINT
96	1	Param96 CANopen_6065 following error window	get/set	UDINT
97	1	Param97 CANopen_6066 following error time out	get/set	UINT
98	1	Param98 CANopen_6067 position window	get/set	UDINT

Instance	Attribute	Name	Service	Data Type
99	1	Param99 CANopen_position window time	get/set	UINT
100	1	Param100 CANopen_606C velocity actual value	get	DINT
101	1	Param101 CANopen_606D velocity window	get/set	UINT
102	1	Param102 CANopen_606E velocity window time	get/set	UINT
103	1	Param103 CANopen_606F velocity threshold	get/set	UINT
104	1	Param104 CANopen_6070 velocity threshold time	get/set	UINT
105	1	Param105 CANopen_6071 target torque	get/set	INT
106	1	Param106 CANopen_6077 torque actual value	get	SINT
107	1	Param107 CANopen_6078 current actual value	get	INT
108	1	Param108 CANopen_6079 DC link voltage	get	UDINT
109	1	Param109 CANopen_607A target position	get/set	DINT
110	1	Param110 CANopen_607C home offset	get/set	DINT
111	1	Param111 CANopen_6081 profile velocity	get/set	UDINT
112	1	Param112 CANopen_6083 profile acceleration	get/set	UDINT
113	1	Param113 CANopen_6084 profile deceleration	get/set	UDINT
114	1	Param114 CANopen_6085 quick stop deceleration	get/set	UDINT
115	1	Param115 CANopen_6086 motion profile type	get/set	INT
116	1	Param116 CANopen_6087 torque slope	get/set	UDINT
117	1	Param117 CANopen_6091_1 gear ratio motor revolutions	get/set	UDINT
118	1	Param118 CANopen_6091_2 gear ratio shaft revolutions	get/set	UDINT
119	1	Param119 CANopen_ 6092_1 feed constant feed	get/set	UDINT
120	1	Param120 CANopen_6092_2 feed constant shaft revolutions	get/set	UDINT
121	1	Param121 CANopen_6098 homing method	get/set	SINT
122	1	Param122 CANopen_6099_1 homing speeds speed during search for switch	get/set	UDINT
123	1	Param123 CANopen_6099_2 homing speeds speed during search for zero	get/set	UDINT
124	1	Param124 CANopen_609A homing acceleration	get/set	UDINT
126	1	Param126 CANopen_60F4 following error value	get	DINT
127	1	Param127 CANopen_ 60F8 max slippage	get/set	DINT

Instance	Attribute	Name	Service	Data Type
128	1	Param128 CANopen_60FF target velocity	get/set	DINT
129	1	Param128 CANopen_5F18 M/S-Synchronization offset	get/set	DINT

8.4.6 Class 0x28, Motor Data Object

The values of the *Motor Data Object* attributes are read from internal inverter parameters.

They are **always** read from data set 1, even if data sets **1** to **4** have different values.

Instance	Attribute	Name	Service	Data Type	Value
1	3	MotorType	get	USINT	0 – Non-Standard 3 – PM Synchronous 7 – Squirrel Cage Induction
	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]

8.4.7 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. Please refer to Chapter "I/O Assemblies" for correct handling of attributes and necessary basic settings of inverter parameters.

Instance	Attribute	Name	Service	Data Type	Value
1	3	Run1	get/set	BOOL	1 – Run Forward (=Clockwise ?)
	4	Run2	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

8.4.8 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. Please refer to Chapter "I/O Assemblies" for correct handling of attributes and necessary basic settings of inverter parameters.

Instance	Attribute	Name	Service	Data Type	Value
1	3	AtReference	get	BOOL	1 – Drive actual at reference
	4	NetRef	get/set	BOOL	0 – Set reference not EIP control (local reference) 1 – Set Reference at EIP control (PLC)
	6	DriveMode	get	USINT	0 – Vendor specific 1 – Open Loop Speed 2 – Closed Loop Speed 3 – Torque Control 4 – Process Control 5 – Position Control
	7	SpeedActual	get	INT	xxxx [RPM]
	8	SpeedRef	get/set	INT	xxxx [RPM]
	11	TorqueActual	get	INT	xxxx [Nm]
	12	TorqueRef	get/set	INT	xxxx [Nm]
	13	ProcessActual	get	INT	xxxx [%]
	14	ProcessRef	get/set	INT	xxxx [%]
	29	RefFromNet	get	BOOL	0 – Local torque/speed reference 1 – Network torque/speed reference

EIP = Ethernet/IP

8.5 Operating behavior in the case of bus connection failure

The operating behavior in the case of failure of the Ethernet/IP systems can be parameterized. The required behavior is set via parameter *Bus Error Behaviour* **388.**

Bus Error Behaviour 388	Function
0 - no reaction	Operating point is maintained.
1 - Error	"Fault" status will be activated immediately. **Factory setting.**
2 - Stop	Device state machine processes control command "Disable voltage" and switches to "switch on disabled" status.
3 - Quick stop	Control command "Quick stop" and switch to "switch on disabled" status.
4 - Shutdown + Error	Control command "Disable operation" and switch to "Error" status once the drive has been shut down.
5 - Quick stop + Error	Control command "Quick stop" and switch to "Error" status once the drive has been shut down.

The parameter settings *Bus Error Behaviour* **388** = 2…5 are evaluated depending on the setting of the parameter *Local/Remote* **412.**

The described reactions are only available whenever parameter *LocalRemote* **412** is set to 1 (remote state machine). Otherwise setting paramete*r Error Behavior* **388** to values 2 … 5 always results in the device state machine changing to state "fault" immediately.

The error and warning behavior of the frequency inverter can be parameterized in various ways. For details about possible faults, refer to Chapter 18.4 "Error messages".

8.6 Access to Inverter Parameters (Classes 0x64 … 0x73)

All inverter parameters are accessible by explicit message. Inverter parameters are addressed by parameter number (**0 …. 1599**) and data set number (**0, 1 … 9**).

The necessary values for class/instance/attribute for R/W access by explicit message is calculated by the algorithm:

- class = (parameter number / 100) + 100
- attribute = (parameter number MODULO 100) + 1
- instance = data set + 1

Example: P.520 ***fixed percentage 1*****, data set 3 (data type = int, 2 bytes)**

- class = (520 / 100) + 100 = 5 + 100 = 105
- attribute = (520 MODULO 100) +1 = 20 + 1 = 21
- instance = 3 + 1 = 4

Refer to the parameter list of the respective configuration in the standard operating instructions to see the parameters that can be set. The parameter list states whether a parameter is data set change-over capable (data set = 1 to 4) or only exists once (data set = 0).

The parameter list also provides information about the display format of a parameter and its type (int/uint/long/string). For values with decimal places, the decimal point is not transmitted.

Typically, parameters are addressed via data set = **0, 1** to **4**. Written values are thus stored to the controller RAM and automatically to the EEPROM. If values are written frequently, no entry shall be made into the EEPROM, as the latter only possesses a limited number of admissible writing cycles (approx. 1 million cycles).

NOTICE

Risk of component damage!

If a maximum number of write cycles for the EEPROM is exceeded, it will be destroyed.

The values are entered automatically in the EEPROM of the controller. However, only a limited number of write cycles is permissible for the EEPROM (approx. 1 million cycles).

- If cyclic writing of data is required, use the RAM.

This mechanism is activated by the target data set being increased by five in the specification of the data set.

EEPROM	RAM
Entry into data set 0	Data set = 5
Entry into data set 1	Data set = 6
Entry into data set 2	Data set = 7
Entry into data set 3	Data set = 8
Entry into data set 4	Data set = 9

If the values of a data set change-over capable parameter are to be set to the same value in all data sets, the parameter can be written via data set 0.

8.7 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:

- Parameter 1 (Param1) "*InputAssemblySelection*" for input values to the PLC
    - choice list for input assemblies: 70 (default)/71/72/74/101
- Parameter 15 (Param15) "*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 VPlus control software.

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

VPlus

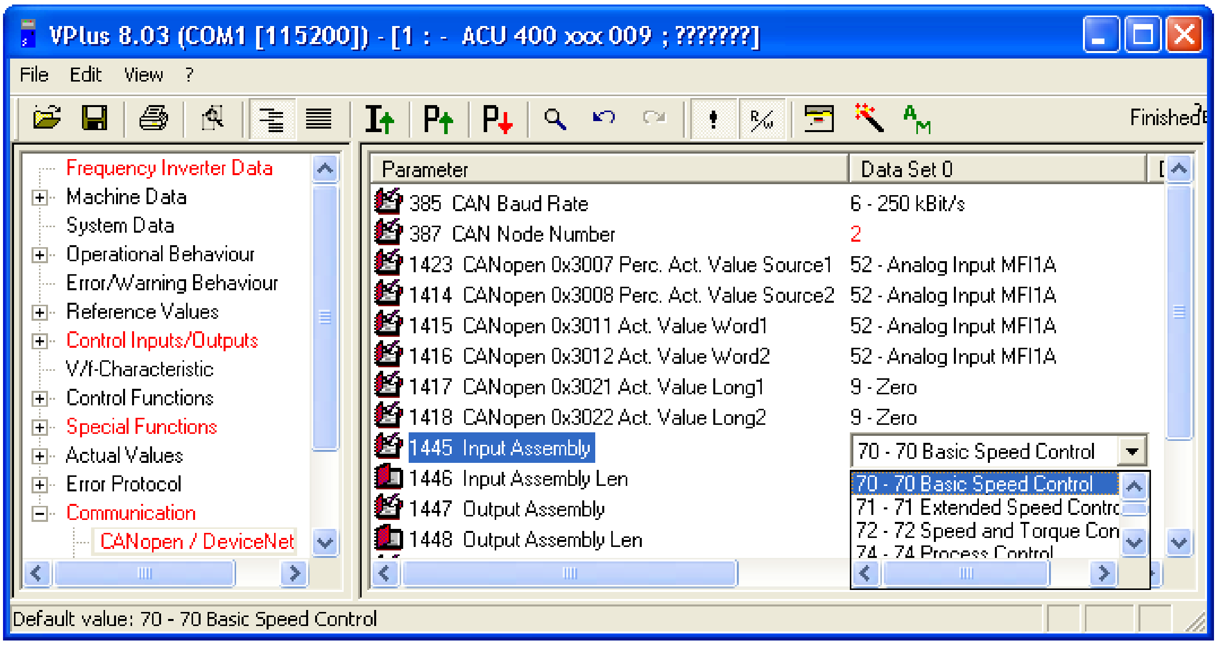
The assembly selection is carried out by writing the following inverter parameters:

- *Input Assembly Selection* **P.1445**
- *Output Assembly Selection* **P.1447**

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

- *Input Assembly Length* **P.1446**
- *Output Assembly Length* **P.1448**

Input Assembly

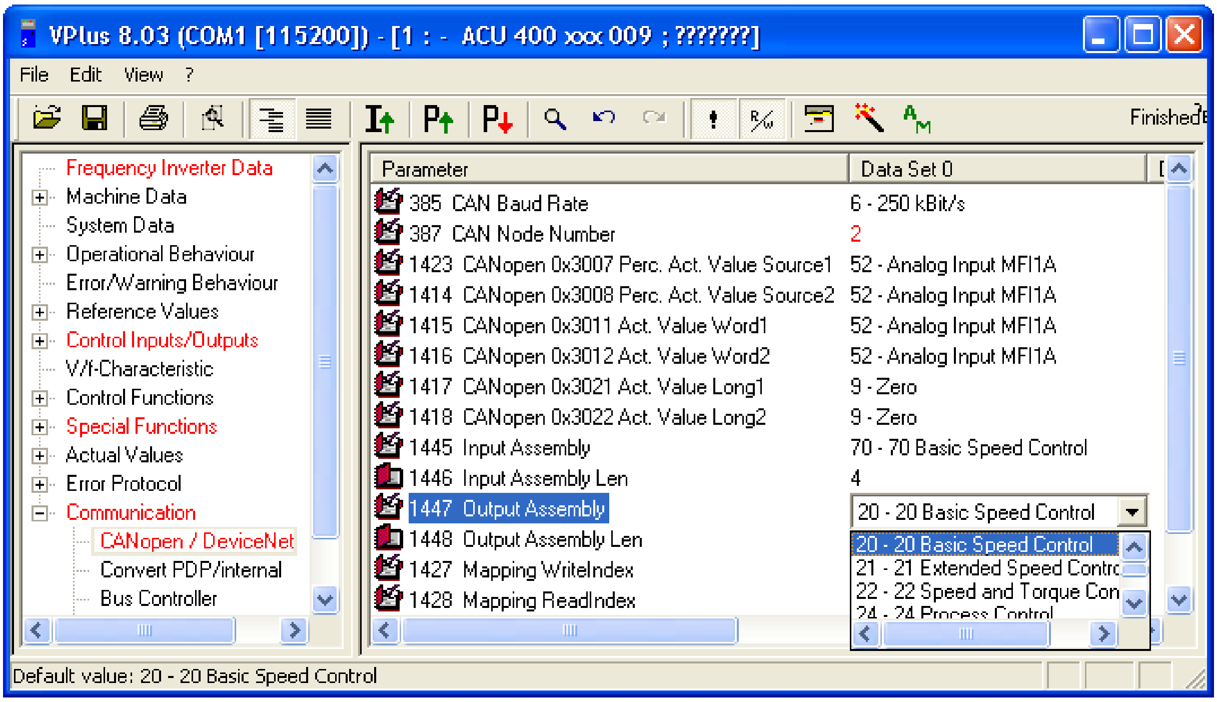


Parameter	Data Set 0
385 CAN Baud Rate	6 - 250 kBit/s
387 CAN Node Number	2
1423 CANopen 0x3007 Perc. Act. Value Source1	52 - Analog Input MFI1A
1414 CANopen 0x3008 Perc. Act. Value Source2	52 - Analog Input MFI1A
1415 CANopen 0x3011 Act. Value Word1	52 - Analog Input MFI1A
1416 CANopen 0x3012 Act. Value Word2	52 - Analog Input MFI1A
1417 CANopen 0x3021 Act. Value Long1	9 - Zero
1418 CANopen 0x3022 Act. Value Long2	9 - Zero
1445 Input Assembly	70 - 70 Basic Speed Control
1446 Input Assembly Len	
1447 Output Assembly	
1448 Output Assembly Len	

70 - 70 Basic Speed Control
71 - 71 Extended Speed Contrc
72 - 72 Speed and Torque Con
74 - 74 Process Control

Default value: 70 - 70 Basic Speed Control

Output Assembly



Parameter	Data Set 0
385 CAN Baud Rate	6 - 250 kBit/s
387 CAN Node Number	2
1423 CANopen 0x3007 Perc. Act. Value Source1	52 - Analog Input MFI1A
1414 CANopen 0x3008 Perc. Act. Value Source2	52 - Analog Input MFI1A
1415 CANopen 0x3011 Act. Value Word1	52 - Analog Input MFI1A
1416 CANopen 0x3012 Act. Value Word2	52 - Analog Input MFI1A
1417 CANopen 0x3021 Act. Value Long1	9 - Zero
1418 CANopen 0x3022 Act. Value Long2	9 - Zero
1445 Input Assembly	70 - 70 Basic Speed Control
1446 Input Assembly Len	4
1447 Output Assembly	20 - 20 Basic Speed Control
1448 Output Assembly Len	
1427 Mapping WriteIndex	
1428 Mapping ReadIndex	

20 - 20 Basic Speed Control
21 - 21 Extended Speed Contrc
22 - 22 Speed and Torque Con
24 - 24 Process Control

Default value: 20 - 20 Basic Speed Control

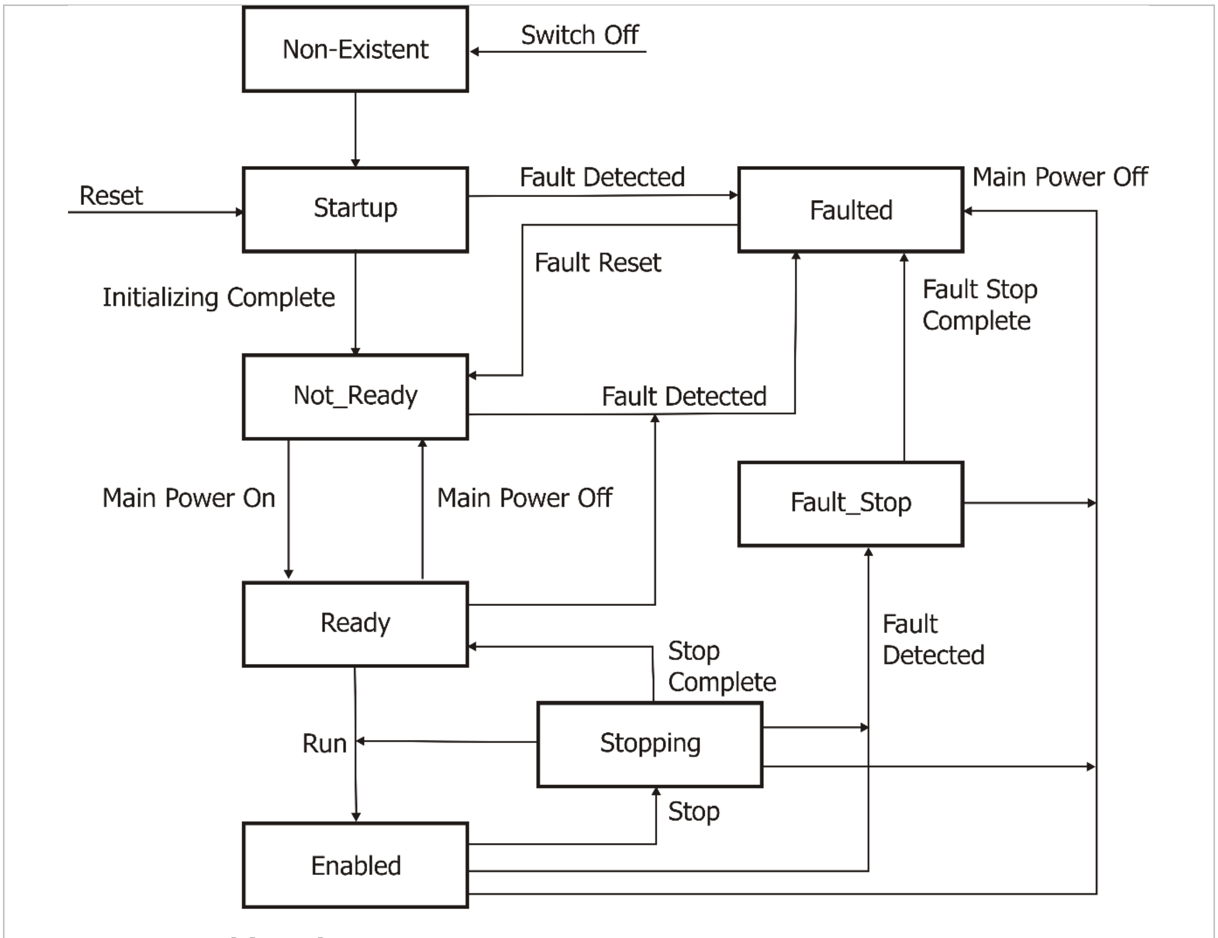
8.7.1 CIP Assemblies 20/21/22/24 + 70/71/72/74

The described functionality of CIP Assemblies requires the following inverter settings:

- Parameter *Local/Remote* **412** = `1` (Remote State Machine)

Parameter **412** is automatically set to `1` whenever *NetCtrl* (Class 0x29, Instance 1, Attribute 5) is set to `1`.

- Digital input STOA and STOB set to `high` (+24V)
- Digital input *Start Clockwise* OR *Start Anticlockwise* set to `high` (+24V) or directly set to `On` (**P.68**/**P.69**)



State Transition Diagram

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

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

8.7.1.1 Output Assembly 20

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]**

(written to inverter speed reference Line Setpoint, see **P.434**)

Speed Reference is written to the inverter reference frequency (*Reference Frequency RAM*, source no. 5, see Parameter *Ramp Setpoint* **434**). The reference frequency is calculated from *Speed Reference* [RPM] and Parameter *No. of Pole Pairs* **373** in the active data set.

The data type of *Speed Reference* is INT. Setting *Speed Reference* to negative values causes the drive to run anticlockwise on *Run Forward* set to 1.

The DeviceNet objects *NetCtrl* (Class 0x29, Instance 1, Attribute 5) and *NetRef* (Class 0x2A, Instance 1, Attribute 4) **must** be set by explicit message from the PLC Program in order to set the inverter to the appropriate control mode.

NetCtrl sets Inverter Parameter *Local/Remote* **412** = 1 "Control via Statemachine" (RAM only)

NetRef sets Inverter Parameter *Ramp Setpoint* **434** = 2 "Line Setpoint" (RAM only)

This can be achieved by sending explicit messages to these objects individually, or setting both objects together by one explicit message, writing to *Class 4* (Assembly Object), *Instance 21* (Assembly 21), *Attribute 3* (Assembly Data).

For a description and example of how to send explicit messages from the PLC Program, please refer to Chapter Sending Explicit Messages from the PLC Program.

8.7.1.2 Output 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

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]**

(written to inverter speed reference Line Setpoint, see **P.434**)

Speed Reference is written to the inverter reference frequency (*Reference Frequency RAM*, source no. 5, see Parameter *Ramp Setpoint* **434**). The reference frequency is calculated from *Speed Reference* [RPM] and Parameter *No. of Pole Pairs* **373** in the active data set.

The data type of *Speed Reference* is INT. Setting *Speed Reference* to negative values causes the drive to run anticlockwise whenever *Run Forward* is set to 1. Otherwise, setting *Speed Reference* to negative values causes the drive to run clockwise whenever *Run Reverse* is set to 1.

- *Net Ctrl* = 1 inverter controlled by net
- *Net Ctrl* = 0 inverter controlled by local control
- *Net Ctrl* = 1 sets Parameter *LocalRemote* **412** to 1 - Remote Statemachine in all data sets.
- *Net Ctrl* = 0 restores Parameter *LocalRemote* **412** to values from before *Net Ctrl* = 0 → 1
- *Net Ref* = 1 speed reference from net
- *Net Ref* = 0 speed reference from local control
- *Net Ref* = 1 sets Parameter *Ramp Setpoint* **434** to 2 - Line Setpoint in all data sets.
- *Net Ref* = 0 restores Parameter *Ramp Setpoint* **434** to values from before *Net Ref* 0 → 1.

8.7.1.3 Output Assembly 22

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]**

Speed Reference is written to the inverter reference frequency (*Reference Frequency RAM*, source no. 5, see Parameter *Ramp Setpoint* **434**). The reference frequency is calculated from *Speed Reference* [RPM] and Parameter *No. of Pole Pairs* **373** in the active data set.

The data type of *Speed Reference* is INT. Setting *Speed Reference* to negative values causes the drive to run anticlockwise whenever *Run Forward* is set to 1.

***Torque Reference* = nnnn [Nm]**

Torque Reference is written to the inverter reference percentage (*Reference Percentage RAM*, source no. 67, see Parameter *S. Ref. Ramp Line Value* **138**). The *Reference Percentage* is calculated from *Torque Reference* [Nm], Parameter *Rated speed* **372** and Parameter *Rated Mech. Power* **376** in the active data set.

The DeviceNet objects *NetCtrl* (Class 0x29, Instance 1, Attribute 5) and *NetRef* (Class 0x2A, Instance 1, Attribute 4) **must** be set by explicit message from the PLC Program in order to set the inverter to the appropriate control mode.

NetCtrl sets Inverter Parameter *Local/Remote* **412** = 1 "Control via Statemachine" (RAM only)

NetRef sets Inverter Parameter *Ramp Setpoint* **434** = 2 "Line Setpoint" (RAM only)

This can be achieved by sending explicit messages to these objects individually, or setting both objects together by one explicit message, writing to *Class 4* (Assembly Object), *Instance 21* (Assembly 21), *Attribute 3* (Assembly Data).

For a description and example of how to send explicit messages from the PLC Program, please refer to Chapter Sending Explicit Messages from the PLC Program.

8.7.1.4 Output Assembly 24

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]**

Speed Reference is written to the inverter reference frequency (*Reference Frequency RAM*, source no. 5, see Parameter *Ramp Setpoint* **434**). The reference frequency is calculated from *Speed Reference* [RPM] and Parameter *No. of Pole Pairs* **373** in the active data set.

The data type of *Speed Reference* is INT. Setting *Speed Reference* to negative values causes the drive to run anticlockwise whenever *Run Forward* is set to 1.

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

Process Reference is written to the inverter reference percentage (*Reference Percentage RAM*, source no. 67, see Parameter *S. Ref. Ramp Line Value* **138**).

The DeviceNet objects *NetCtrl* (Class 0x29, Instance 1, Attribute 5) and *NetRef* (Class 0x2A, Instance 1, Attribute 4) **must** be set by explicit message from the PLC Program in order to set the inverter to the appropriate control mode.

NetCtrl sets Inverter Parameter *Local/Remote* **412** = 1 "Control via Statemachine" (RAM only)

NetRef sets Inverter Parameter *Ramp Setpoint* **434** = 2 "Line Setpoint" (RAM only)

This can be achieved by sending explicit messages to these objects individually, or setting both objects together by one explicit message, writing to *Class 4* (Assembly Object), *Instance 21* (Assembly 21), *Attribute 3* (Assembly Data).

For a description and example of how to send explicit messages from the PLC Program, please refer to Chapter Sending Explicit Messages from the PLC Program.

8.7.1.5 Input Assembly 70

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.

8.7.1.6 Input Assembly 71

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 DeviceNet

Ctrl From Net = 0 inverter controlled by local control

Ref From Net = 1 speed reference from DeviceNet

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.

8.7.1.7 Input Assembly 72

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]**

8.7.1.8 Input Assembly 74

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 [%]**

8.7.2 Vendor Specific Assemblies 100/101

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

ℹ The functions connected to these objects are the same as with CANopen. For detailed information about available CANopen objects please refer to Chapter "CANopen objects".

Both assemblies (Output Assembly 100 and Input Assembly 101) support a maximum number of 12 objects with a maximum number of 24 bytes I/O data.

The selection of these vendor specific assemblies is performed in the same way as for CIP assemblies.

ℹ These assemblies are only settable if an IO-Connection is **not** established.

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 VPlus. The relevant parameters are located under Communication – CANopen / Ethernet/IP:

- **P.1445** Input Assembly = **101**
- **P.1446** Input Assembly Len = *Input assembly length*
- **P.1447** Output Assembly = **100**
- **P.1448** Output Assembly Len = *Output assembly length*

8.7.2.1 Object Mapping - Vendor Specific Output Assembly 100

With VPlus / VTable

When using the inverter utility VTable, Parameter **1429** Indices 1 to 24 offer a selection list where one of the listed objects can be chosen for mapping. Choosing "0 - no object mapped here" marks the end of the mapped objects.

See below for the object selection list.

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

All mapping settings unequal to "- no object mapped here" following the first Index after "- no object mapped here" are ignored.

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

Mapping example:

Index 1 - 0x6040 (2B) controlword

Index 2 - 0x6042 (2B) target velocity

Index 3 - no object mapped here ➔ 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 VPlus . The ignored objects are automatically overwritten with "no object mapped here", however this is only visible after the next screen update.

Object Selection List

The values displayed in the selection list are:

- 0 "- no object mapped here",
- 43 "0x3003 (1B) Digital Out set values",
- 45 "0x3005 (2B) Boolean Demux",
- 46 "0x3006 (2B) Percentage set value",
- 53 "0x3111 (2B) Ref. value word1 (S.762)",
- 54 "0x3112 (2B) Ref. value word2 (S.763)",
- 55 "0x3121 (4B) Ref. value long1 (S.764)",
- 56 "0x3122 (4B) Ref. value long2 (S.765)",
- 57 "0x5F10_1 (2B) Numerator",
- 58 "0x5F10_2 (2B) Denominator",
- 80 "0x6040 (2B) controlword",
- 82 "0x6042 (2B) target velocity",
- 93 "0x6060 (1B) modes of operation",
- 105 "0x6071 (2B) Target torque",
- 109 "0x607A (4B) target position",
- 111 "0x6081 (4B) profile velocity",
- 112 "0x6083 (4B) profile acceleration",
- 113 "0x6084 (4B) profile deceleration",
- 114 "0x6085 (4B) quick stop deceleration",
- 128 "0x60FF (4B) target velocity";

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

8.7.2.2 Object Mapping - Vendor Specific Input Assembly 101

With VPlus / VTable

When using the inverter utility VTable, Parameter **1430** Indices 1 to 24 offer a choice box where one of the listed objects can be chosen for mapping. Choosing "0 – no object mapped here" marks the end of the mapped objects.

See below for the object selection list.

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

All mapping settings unequal to "– no object mapped here" following the first Index after "– no object mapped here" are ignored.

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

Mapping example:

Index 1 - 0x6041 (2B) statusword

Index 2 - 0x6044 (2B) control effort

Index 3 - no object mapped here ➔ 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 VPlus . The ignored objects are automatically overwritten with "no object mapped here", however this is only visible after the next screen update.

Object Selection List

The values displayed in the selection list are:

- 0 "- no object mapped here",
- 41 "0x3001 (2B) Digital In actual values",
- 42 "0x3002 (2B) Digital Out actual values",
- 44 "0x3004 (2B) Boolean Mux",
- 47 "0x3007 (2B) Percentage actual value",
- 48 "0x3008 (2B) Percentage actual value2",
- 49 "0x3011 (2B) Act. value word1 (P.1415)",
- 50 "0x3012 (2B) Act. value word2 (P.1416)",
- 51 "0x3021 (4B) Act. value long1 (P.1417)",
- 52 "0x3022 (4B) Act. value long2 (P.1418)",
- 76 "0x5FF0 (1B) active motion block (ttr)",
- 77 "0x5FF1 (1B) motion block to resume (ttr)",
- 81 "0x6041 (2B) statusword",
- 83 "0x6043 (2B) velocity demand",
- 84 "0x6044 (2B) control effort",
- 94 "0x6061 (1B) modes of operation display",
- 95 "0x6064 (4B) position actual value",
- 100 "0x606C (4B) velocity actual value",
- 106 "0x6077 (2B) Torque actual value",
- 107 "0x6078 (2B) Current actual value",
- 108 "0x6079 (4B) DC link voltage",

```
    –  126 "0x60F4 (4B) following error value";
```

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

8.7.3 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 EM-AUT-21 module Ethernet/IP. Connect the inverter to the network and assign the IP Address (refer to Chapter 6.2 "Network configuration for Ethernet/IP").

The required I/O Assemblies must first be selected with VPlus (see Chapter I/O Assemblies).

If vendor specific Assemblies 100/101 are selected, then check/select the mapped objects with VPlus-VTable (see Chapter Vendor Specific Assemblies 100/101).

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). The necessary EDS files are created by the VPlus software tool (to do this a vcb-file of the inverter must be available or VPlus must be connected to the inverter). Please refer to Chapter "How to create EDS Files".
- 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 as 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.

The size for the Vendor Specific Assemblies **100/101** can be selected (all others are fixed).

- 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.

8.8 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 as described in Chapter 8.7.3.

STEP2

- Define the required messages (see Chapter 8.8.1 "How to create a MESSAGE Tag").

STEP3

- Add the appropriate code to the PLC MainProgram (see Chapter 8.8.2 "PLC Program Code Examples")

8.8.1 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 (see Chapter 8.7.1.2)
Attribute:	3,	Data value

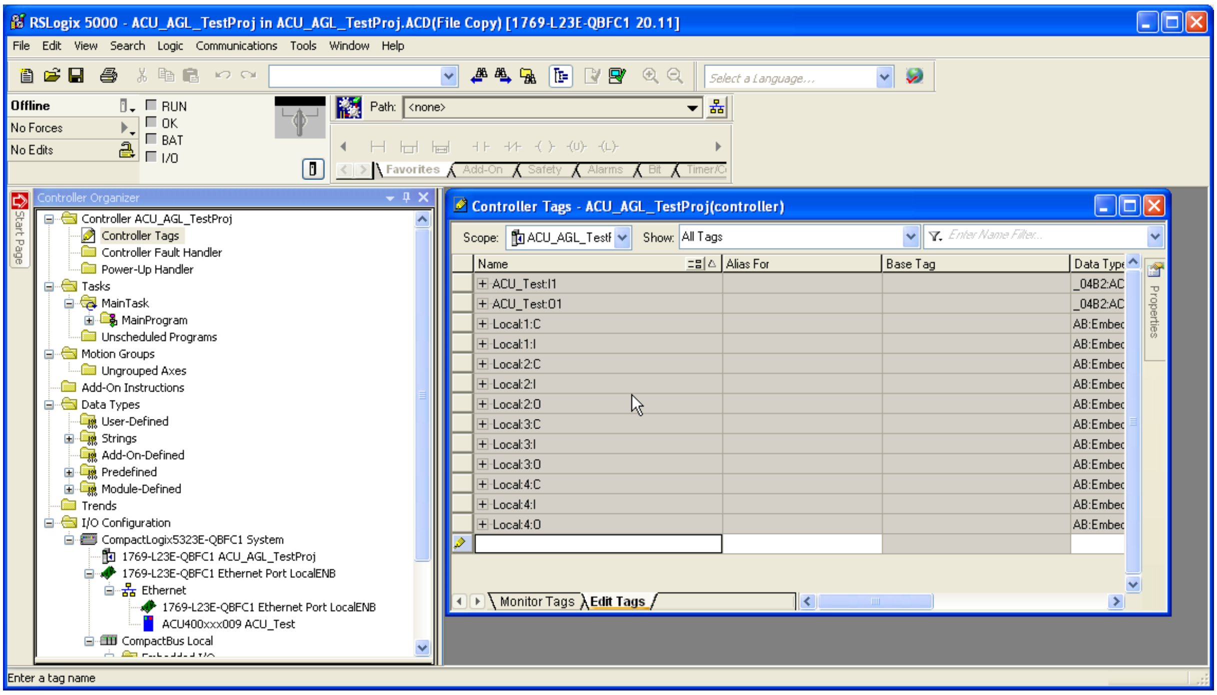
The following four data bytes are predefined for Assembly 21.

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 (see Chapters 8.7.1.1, 8.7.1.3 and 8.7.1.4) 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:



RSLogix 5000 - ACU_AGL_TestProj in ACU_AGL_TestProj.ACD(File Copy) [1769-L23E-QBFC1 20.11]

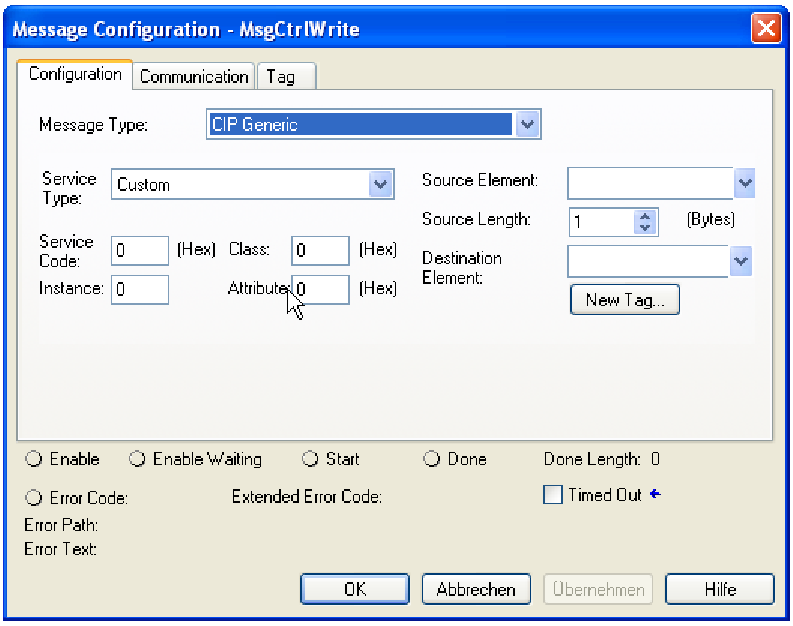
Controller Tags - ACU_AGL_TestProj(controller)

Scope: ACU_AGL_Testf Show: All Tags

Name	Alias For	Base Tag	Data Type
+ ACU_Test:I1			_04B2:AC
+ ACU_Test:O1			_04B2:AC
+ Local:1:C			AB:Embec
+ Local:1:I			AB:Embec
+ Local:2:C			AB:Embec
+ Local:2:I			AB:Embec
+ Local:2:O			AB:Embec
+ Local:3:C			AB:Embec
+ Local:3:I			AB:Embec
+ Local:3:O			AB:Embec
+ Local:4:C			AB:Embec
+ Local:4:I			AB:Embec
+ Local:4:O			AB:Embec

Enter a tag name

- 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:



Message Configuration - MsgCtrlWrite

Configuration | Communication | Tag

Message Type: CIP Generic

Service Type: Custom

Service Code: 0 (Hex) Class: 0 (Hex)

Instance: 0 Attribute: 0 (Hex)

Source Element:

Source Length: 1 (Bytes)

Destination Element:

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

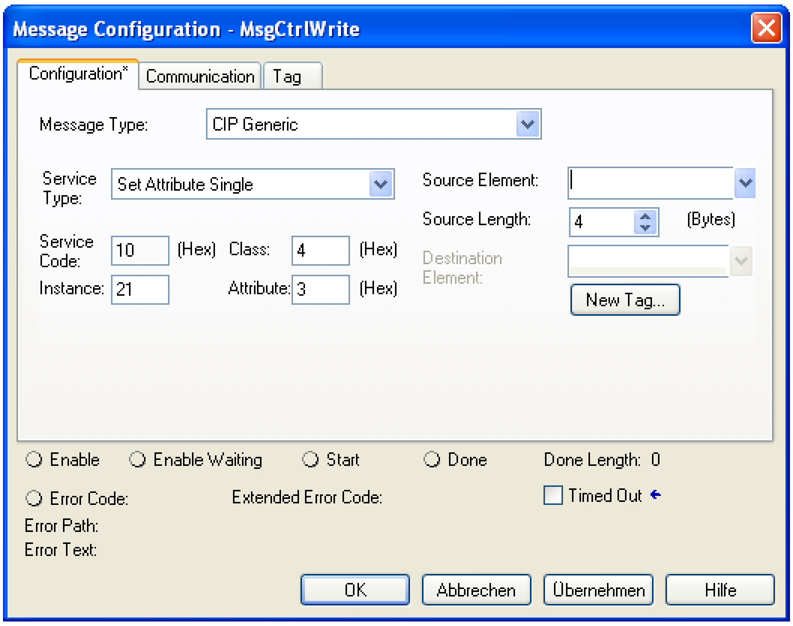
Error Path:

Error Text:

- 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`.



Message Configuration - MsgCtrlWrite

Configuration* | Communication | Tag

Message Type: CIP Generic

Service Type: Set Attribute Single

Service Code: 10 (Hex) Class: 4 (Hex)

Instance: 21 Attribute: 3 (Hex)

Source Element:

Source Length: 4 (Bytes)

Destination Element:

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

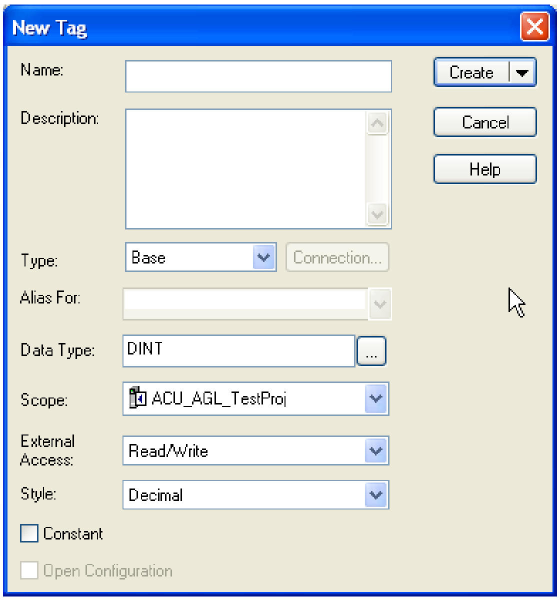
○ Error Code: Extended Error Code: □ Timed Out

Error Path:

Error Text:

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:



New Tag

Name:

Description:

Type: Base

Alias For:

Data Type: DINT

Scope: ACU_AGL_TestProj

External Access: Read/Write

Style: Decimal

□ Constant

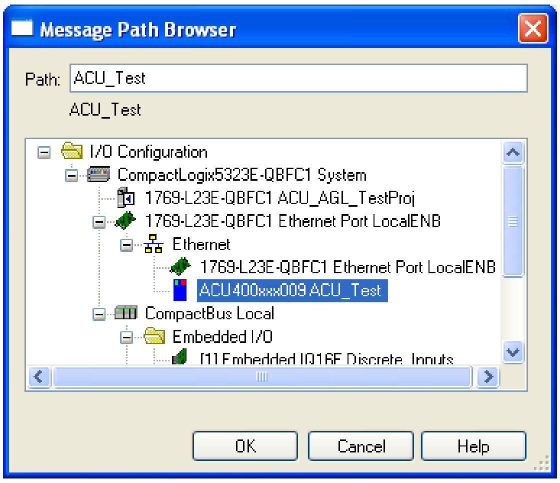
□ Open Configuration

- 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:



Path: ACU_Test

ACU_Test

- I/O Configuration
  - CompactLogix5323E-QBFC1 System
    - 1769-L23E-QBFC1 ACU_AGL_TestProj
    - 1769-L23E-QBFC1 Ethernet Port LocalENB
      - Ethernet
        - 1769-L23E-QBFC1 Ethernet Port LocalENB
        - ACU400xxx009 ACU_Test
    - CompactBus Local
      - Embedded I/O
        - [1] Embedded IQ16F Discrete_Inputs

- 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.

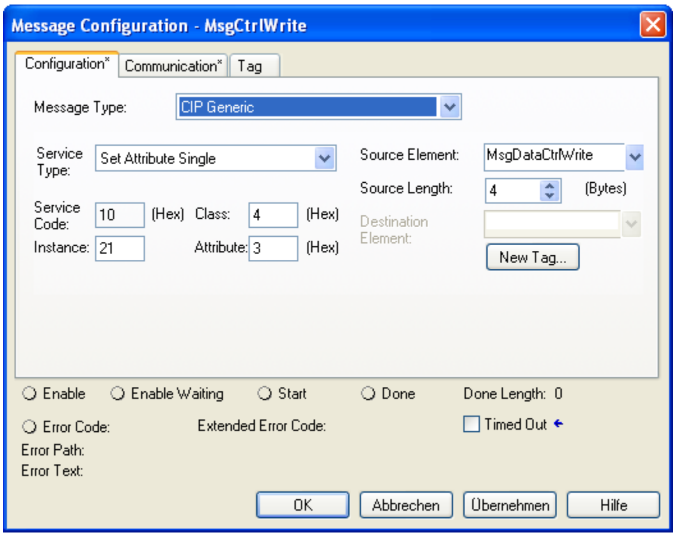
8.8.2 PLC Program Code Examples

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:



Message Type: CIP Generic

Service Type: Set Attribute Single

Service Code: 10 (Hex) Class: 4 (Hex)

Instance: 21 Attribute: 3 (Hex)

Source Element: MsgDataCtrlWrite

Source Length: 4 (Bytes)

Destination Element:

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path:

Error Text:

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

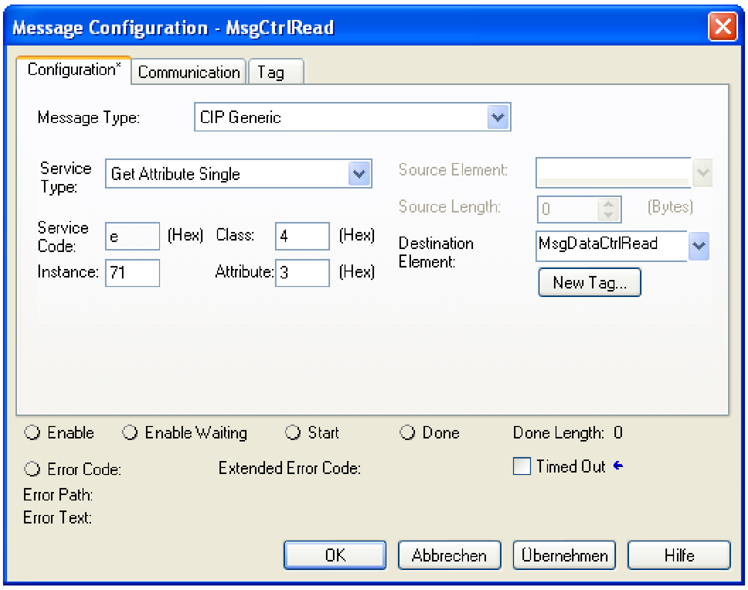
<u>Example 2</u>: Reading back *NetCtrl* and *NetRef* via Input Assembly *71*.

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:



Message Configuration - MsgCtrlRead

Message Type: CIP Generic

Service Type: Get Attribute Single

Service Code: e (Hex) Class: 4 (Hex)

Instance: 71 Attribute: 3 (Hex)

Source Element:

Source Length: 0 (Bytes)

Destination Element: MsgDataCtrlRead

○ Enable ○ Enable Waiting ○ Start ○ Done Done Length: 0

○ Error Code: Extended Error Code: □ Timed Out

Error Path:

Error Text:

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.

8.9 Diagnosis of Ethernet/IP state

The Ethernet/IP State is displayed via the Inverter actual value Parameter *Node-State* **1290**.

9 CANopen

The Automation Interface EM-AUT allows using CANopen® as Field bus.

This chapter describes the CANopen® interface.

WARNING

Communication interrupted

With the Selection "CANopen" the functionality "EthernetIP" is completely deactivated, an running EthernetIP communication is interrupted.

- Simultaneous operation of Systembus and EthernetIP® is possible.
- Simultaneous operation of Systembus and CANopen® is possible.
- Simultaneous operation of EthernetIP® and CANopen® is **not** possible.

- Exercise caution when changing the communication protocol.

This chapter contains the description of the CANopen structure.

The objects which can be used for CANopen® are described in chapter 9.9 "CANopen objects".

9.1 Bus termination

The bus necessary on the phase in the physically first and last subscriber can be activated via the DIP switch S2 on the EM-AUT expansion module.

- Set S2 to ON (Left position) for passive termination.

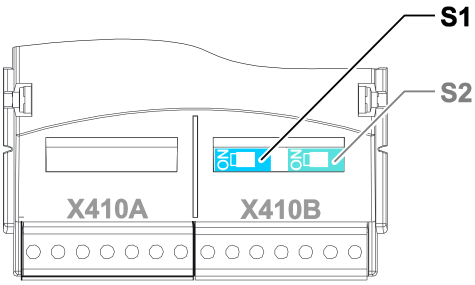
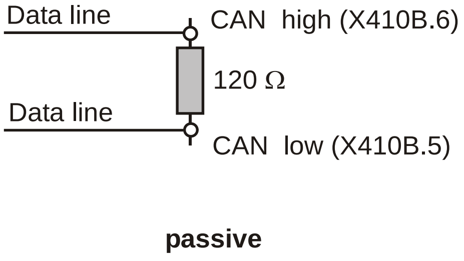


Figure 9-1: Bus termination

By default, the bus termination is set to OFF (switch in right position).



9.2 Cables

For the bus line, use twisted cable with harness shield (**no foil shield**).

NOTICE

Control and communication cables must be kept physically separate from the power cables. The braided shield of the communication cable is to be connected to ground (PE) on both sides on a large area and with good conductivity.

9.3 Control terminal X410B

The systembus is connected via three sockets of the plug **X410B** on the EM-AUT expansion module.

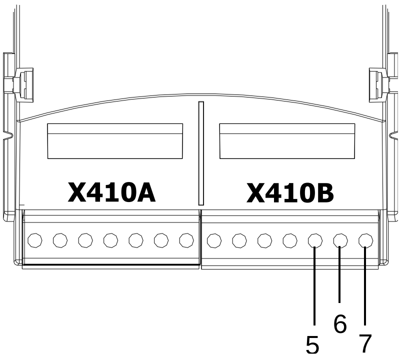


Figure 9-2: Bus termination

Control terminal X410B		
Terminal	**Input/output**	**Description**
X410B.5	CAN-Low	CAN-Low
X410B.6	CAN-High	CAN-High
X410B.7	$\mathrm{GND_{CAN}}$	CAN-GND

NOTICE

For the connection of two or more devices CAN Low, CAN High and $\mathrm{GND_{CAN}}$ must be connected.

Without $\mathrm{GND_{CAN}}$ connection telegram faults or telegram interruptions can occur.

9.4 Baud rate setting/line lengths

The transmission speed of the CANopen® interface can be set via the parameter *CAN Baud rate* **385**.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Default**
385	CAN Baud rate	1	8	6

The transmission rate is a function of a variety of application-specific parameters. The line length of the communication network limits the transmission speed due to the signal propagation time of the CANopen® protocols.

CANopen® interface		
Operation mode	**Function**	**max. Line length**
1 -10 kBaud	Transmission rate 10 kBaud	5000 meter
2 -20 kBaud	Transmission rate 20 kBaud	2500 meter
3 -50 kBaud	Transmission rate 50 kBaud	1000 meter
4 -100 kBaud	Transmission rate 100 kBaud	500 meter
5 -125 kBaud	Transmission rate 125 kBaud	500 meter
6 -250 kBaud	Transmission rate 250 kBaud	250 meter
7 -500 kBaud	Transmission rate 500 kBaud	100 meter
8 -1000 kBaud	Transmission rate 1000 kBaud	25 meter

Changing the baud rate causes a restart of the CANopen® system (NOT a reset of the inverter).

9.5 Setting the node number

The CANopen® protocol supports a maximum of 127 nodes in a communication network. Each frequency inverter is assigned a node ID, which may only exist once in the system, for its unambiguous identification. The node number is set with parameter *CAN Node Number* 387.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Default**
387	CAN Node Number	-1	127	-1

The factory setting *CAN Node Number* **387** = -1 means that the CANopen® interface has been **deactivated**.

The value *Can Node number* **387** = 0 is not allowed and cannot be set.

Changing the node number causes a restart of the CANopen® system (NOT a reset of the inverter).

9.6 Operating behavior in the case of bus connection failure

The operating behavior in the case of failure of the CANopen® systems can be parameterized. The required behavior can be set via in parameter *Bus Error Behaviour* **388** or via Object 0x6007 abort connection option code.

For the description of the inverter's functional behavior, see chapter 9.9.5.1 “0x6007/0 Abort Connection option code”.

Bus Error Behaviour 388	Function
0 -No Reaction	Operating point is maintained
1 -Error	“Fault” status will be activated immediately. **Factory setting**.
2 -Stop	Control command “Disable voltage” and switch to “switch on disabled” status.
3 -Quick-Stop	Control command “Quick stop” and switch to “switch on disabled d” status.
4 -Ramp-Stop + Error	Control command “Disable operation” and switch to “Error” status once the drive has been shut down.
5 -Quick-Stop + Error	Control command “Quick stop” and switch to “Error” status once the drive has been shut down.

NOTICE

The parameter settings *Bus Error Behaviour* **388** = 2 … 5 are evaluated depending of parameter *Local/Remote* **412**. This is described in detail in chapter 9.9.5.1 “0x6007/0 Abort Connection option code”.

Parameter *Bus Error Behaviour* **388** corresponds to the device profile object 0x6007 abort connection option code.

There are numerous options of parameterizing the fault and warning behavior of the frequency inverter. For details about possible faults, refer to Chapter 18.4 “Error messages”.

NOTICE

The disconnection of a connector or another contact loss can only be detected safely via set up timeout monitoring.

9.7 CANopen Overview

CANopen® is used in a wide range of applications and is an especially favoured communication system for motion control applications. The CANopen® based standard DS402 “drives and motion control” describes and defines the necessary objects and functions for motion control systems.

The CANopen® standard DS301 describes the basic communication functions in principle. This chapter will give a short overview of the different functions based on DS301. Detailed information on the CAN physical layer and CANopen® DS301 functions can be found in the respective literature (e. g. "Controller Area Network" by Prof. Dr.-Ing. K. Etschberger) and standards published by CAN-in-Automation CiA® (www.can-cia.org).

Every CANopen® device contains an object dictionary with all supported objects. The objects can be divided into the two main groups – communication objects and application objects. The objects are addressed by their index 0xnnnn (16 bit) and sub-index 0xnn (8 bit).

The different functions defined by CANopen® (NMT, SDO, SYNC, PDO, Emergency) use fixed identifier ranges. These identifier ranges are defined by the "Predefined Connection Set". By default every function uses an identifier calculated as the base number plus node-ID (node–ID set by parameter *CAN node number* **387**.

9.7.1 Communication Objects

The communication objects are located in the index range 0x1nnn. They describe the communication behaviour of a CANopen® device. Some of the communication objects comprise device information

(e. g. manufacturer's vendor-id or inverter serial number). With the help of communication objects the application objects for device control are mapped to the PDO messages.

9.7.2 Application Objects

Application objects are divided in two groups. Index range 0x2000 to 0x5FFF is reserved for manufacturer-specific objects, and index range 0x6nnn is reserved for specific device profile objects. The specific device profile objects 0x6nnn are defined by CANopen® DS402 "drive and motion control". They are used for controlling device functions (Start/Stop, speed, positioning functions).

9.7.3 SDO Function

The SDO (Service Data Objects) messages are used for reading and writing the objects located in the object dictionary. Objects with up to four bytes of data are transferred with an expedited SDO transfer that uses one request and one response message. Access to objects with more than four bytes of data is accomplished by a segmented domain transfer.

In chapter 9.9.3 "Manufacturer objects (0x2nnn)" the necessary messages for reading/writing objects with expedited transfer are described in detail. Access to communication, manufacturer and device profile specific objects with up to four bytes of data is accomplished in the same way. The only difference is in the index and sub-index number.

The inverter supports one server SDO. This server SDO is accessed by the client SDO on the PLC side. An SDO message always has a COB-ID followed by 8 data bytes.

SDO-message:

COB-ID	0	1	2	3	4	5	6	7
COB-ID	command specifier (cs)	index		sub-index	data	data	data	data
	nn	LSB	MSB					

Default Identifiers (COB-ID):

TxSDO	0x600 (=1536) + Node-ID
RxSDO	0x580 (=1408) + Node-ID

Depending on the transfer direction and the amount of data bytes, different command specifiers are used.

The error codes of failed SDO accesses are listed in chapter 6.3.2.

9.7.3.1 Read Access

Client → Server, Upload Request

COB-ID	0	1	2	3	4	5	6	7
0x600 + Node-ID	cs	index		sub- index	data	data	data	data
	0x40	LSB	MSB		00	00	00	00

Server → Client, Upload Response

COB-ID	0	1	2	3	4	5	6	7
0x580 + Node-ID	cs	index		sub- index	data	data	data	data
	0x4x	LSB	MSB		data01	data02	data03	data04

The amount of valid data bytes is coded in the response of the command specifier.

Amount of data bytes	**1**	**2**	**3**	**4**
Command specifier (cs)	0x4F	0x4B	0x47	0x43

9.7.3.2 Write Access

Client → Server, Download Request

COB-ID	0	1	2	3	4	5	6	7
0x600 + Node-ID	cs	index		sub- index	data	data	data	data
	0x2x	LSB	MSB		data01	data02	data03	data04

Server → Client, Download Response

COB-ID	0	1	2	3	4	5	6	7
0x580 + Node-ID	cs	index		sub- index	data	data	data	data
	0x60	LSB	MSB		00	00	00	00

The amount of valid data bytes must be coded in the request of the command specifier.

Amount of data bytes	**1**	**2**	**3**	**4**
Command specifier	0x2F	0x2B	0x27	0x23

NOTICE

Using Write accesses for parameters (objects 0x2nnn = index), the sub-index is used to define the Write access into EEPROM or RAM. Please refer to chapter 9.9.3.1 "Handling of data sets/cyclic writing".

9.7.3.3 Error codes

If an error occurs in reading or writing, the server SDO of the frequency inverter replies with the SDO abort message. This message contains the index/subindex and appropriate error code.

Server → Client, Abort SDO Transfer

COB-ID	0	1	2	3	4	5	6	7
0x580 + Node-ID	cs	index		sub- index	abort code low		abort code high	
	0x80	LSB	MSB	LSB	MSB	LSB	MSB	00

The fault codes are listed in chapter 6.3.2 "SDO Error code table".

9.7.3.4 Segmented Transfer

For data lengths > 4 Bytes the so called Segmented Transfer is used – the expedited Transfer only supports lengths up to 4 Bytes.

In the first "Initiate" Telegram the overall amount of used data of the following sequence telegrams is defined.

In the following telegrams 7 data byes per telegram are send until the amount of data bytes to be transmitted was reached. The sequences of the Segmented Transfer are separated by a toggle bit in the command specifier for the request and the reply telegram. A "Continue" Bit marks the last telegram.

9.7.3.5 Reading „Segmented Transfer"

When Reading a regular Read access via command specifier 0x40 is executed. The response contains the command specifier 0x41 that marks the requirement of Segmented Transfer for this object. The following requests alternate with command specifiers 0x60 and 0x70 until all data bytes were transmitted. In the last segment the command specifier (bits 1…3) contains the amount of not used data bytes in that last segment.

The resulting request and response telegrams are shown in the following sequence.

The Command Specifier have the following setup:

Initiate Upload Command Specifier:

Request:	Bit	7	6	5	4	3	2	1	0
		ccs			0	0	0	0	0
Response:	Bit	7	6	5	4	3	2	1	0
		scs			0	n		e	s

Segment Upload Command Specifier:

Request:	Bit	7	6	5	4	3	2	1	0
		ccs			t	0	0	0	0
Response:	Bit	7	6	5	4	3	2	1	0
		scs			t	n			c

Abbreviation	**Description**	**Values**
ccs	Client command Specifier	2 = Initiate upload request 3 = Upload segment request
scs	Server command Specifier	2 = Initiate upload response 0 = Upload segment response
n	Only valid if e =1 AND s = 1, in all other cases n = 0.	If valid: Amount of data bytes, that contain no useful data
e	Transfer type	0 = Normal (Segmented) Transfer 1 = Expedited Transfer (see chapter 9.7.3.1)
s	Size indicator	0 = Data frame size is displayed 1 = Data frame size is not displayed
t	Toggle bit, toggled with each Segment change	0 = First and odd segments 1 = Second and even segments
c	Continue bit, marks following segments	0 = Further segments follow. 1 = This was the last segment.

The following sequence of telegrams results:

Initiate SDO Upload

		COB-ID	0	1	2	3	4	5	6	7
Request	Client → Server	0x600 + Node-ID	cs	Index		Subidx	Data			
			0x40	LSB	MSB		00	00	00	00
Response	Server → Client	0x580 + Node-ID	cs	Index		Subidx	Data			
			0x41	LSB	MSB		LSB	…	…	MSB

Segment Upload, first and odd segments

		COB-ID	0	1	2	3	4	5	6	7
Request	Client → Server	0x600 + Node-ID	cs	Data						
			0x60	00	00	00	00	00	00	00
Response	Server → Client	0x580 + Node-ID	cs	Data						
			0x00	LSB	…	…	…	…	…	MSB

Segment Upload, second and even segments

		COB-ID	0	1	2	3	4	5	6	7
Request	Client → Server	0x600 + Node-ID	cs	Data						
			0x70	00	00	00	00	00	00	00
Response	Server → Client	0x580 + Node-ID	cs	Data						
			0x10	LSB	…	…	…	…	…	MSB

Segment Upload, last segment

		COB-ID	0	1	2	3	4	5	6	7
Request	Client → Server	0x600 + Node-ID	cs	Data						
			0x60 or **0x70**	00	00	00	00	00	00	00
Response	Server → Client	0x580 + Node-ID	cs	Data						
			0xnn	LSB	…	…	…	…	…	MSB

9.7.3.6 Writing Segmented Transfer

The first telegram to write is executed via Command Specifier 0x21. The amount of entered data bytes in the data area defines the amount of data bytes to be transmitted in the following segment transfers. The following segments are controlled via Command Specifier 0x00 and 0x10 in toggling order until all data were transmitted. The last segment contains in the Command specifier (Bit 1…3) the amount of not used data bytes in the last telegram.

The resulting request and response telegrams are shown in the following sequence.

The Command Specifier have the following setup:

Initiate Download Command Specifier:

Request:	Bit	7	6	5	4	3	2	1	0
		ccs			0	n		e	s
Response:	Bit	7	6	5	4	3	2	1	0
		scs			0				

Download SDO Segment Command Specifier:

Request:	Bit	7	6	5	4	3	2	1	0
		ccs			t	n			c
Response:	Bit	7	6	5	4	3	2	1	0
		scs			t	0	0	0	0

Abbreviation	Description	Values
ccs	Client command Specifier	1 = Initiate download request 0 = Download sequence request
scs	Server command Specifier	3 = Initiate download request 1 = Download sequence response
n	Only valid if e =1 AND s = 1, in all other cases n = 0.	If valid: Amount of data bytes, that contain no useful data
e	Transfer type	0 = Normal (Segmented) Transfer 1 = Expedited Transfer (see chapter 9.7.3.1)
s	Size indicator	0 = Data frame size is displayed 1 = Data frame size is not displayed
t	Toggle bit, toggled with each Segment change	0 = First and odd segments 1 = Second and even segments
c	Continue bit, marks following segments	0 = Further segments follow. 1 = This was the last segment.

Initiate SDO Upload

		COB-ID	0	1	2	3	4	5	6	7
Request	Client → Server	0x600 + Node-ID	cs	Index		Subidx	Data			
			0x21	LSB	MSB		LSB	…	…	MSB
Response	Server → Client	0x580 + Node-ID	cs	Index		Subidx	Data			
			0x41	LSB	MSB		00	00	00	00

Segment Upload, first and odd segments

		COB-ID	0	1	2	3	4	5	6	7
Request	Client → Server	0x600 + Node-ID	cs	Data						
			0x00	00	00	00	00	00	00	00
Response	Server → Client	0x580 + Node-ID	cs	Data						
			0x20	00	00	00	00	00	00	00

Segment Upload, second and even segments

		COB-ID	0	1	2	3	4	5	6	7
Request	Client → Server	0x600 + Node-ID	cs	Data						
			0x10	00	00	00	00	00	00	00
Response	Server → Client	0x580 + Node-ID	cs	Data						
			0x30	00	00	00	00	00	00	00

Segment Upload, last segment

		COB-ID	0	1	2	3	4	5	6	7
Request	Client → Server	0x600 + Node-ID	cs	Data						
			0xnn	00	00	00	00	00	00	00
Response	Server → Client	0x580 + Node-ID	cs	Data						
			0x10 oder 0x20	LSB	…	…	…	…	…	MSB

9.7.4 PDO Function

PDO (Process Data Objects) messages contain up to 8 bytes of process data. Using communication objects (communication/mapping parameters) the process data objects are mapped to Rx/Tx-PDOs. The frequency inverters support three RxPDOs (PLC → frequency inverter) and three TxPDOs (frequency inverter → PLC).

Process data objects are linked directly to the functions of the frequency inverter.

PDO-message:

Byte	0	1	2	3	4	5	6	7
	data	data	data	data	data	data	data	data

The number of bytes is 1 … 8 and depends on the mapped objects. The bytes are arranged in the Intel format.

Byte	0	1	2	3	4	5
	16 bit object		32 bit object			
	LSB	MSB	LSB	…	…	MSB

Default Identifiers:

	Decimal	Hexadecimal
TxPDO1	384 + Node-ID	0x180 + Node-ID
RxPDO1	512 + Node-ID	0x200 + Node-ID
TxPDO2	640 + Node-ID	0x280 + Node-ID
RxPDO2	798 + Node-ID	0x300 + Node-ID
TxPDO3	896 + Node-ID	0x380 + Node-ID
RxPDO3	1024 + Node-ID	0x400 + Node-ID

9.7.5 Emergency Function

If there is a communication error or an error in the frequency inverter, the frequency inverter will send an error message. The error message contains the relevant error information. Once the error is acknowledged (error reset), an error message is sent, with the data bytes reset to zero.

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
0x80 (=128)+ Node-ID	EEC	EEC	ER				MEC	MEC

EEC: Emergency Error Code according to DS301
ER: Emergency Register Code according to DS301
MEC: Manufacturer Error Code

The Manufacturer Error Code corresponds to the inverter Fault codes that are described in the Operating Instructions and in this documentation in chapter "0x1014/0 COB-ID Emergency Message".

Additional information is described in chapter "0x1014/0 COB-ID Emergency Message".

9.7.6 SYNC (synchronous transmission)

The SYNC message has two meanings.

The SYNC message is necessary for Rx/TxPDO with transmission type synchronous. The SYNC message synchronizes the different devices to communicate with data from the same (defined) time. As soon as the SYNC telegram is received, the data of all devices are "frozen" and then exchanged during the following data telegrams.

The RxPDO telegrams are collected until a SYNC telegram is received. With the reception of the SYNC telegram the data are transferred internally to the application parameters.

TxPDOs defined as synchronous send the actual application data on SYNC reception.

Additionally the SYNC mechanism can be used to synchronize the operating systems (OS) of different drives. This is useful when the electronic gear is used to enhance the performance of the application. The synchronization of the operating systems is described in chapter 6.3 "OS Synchronization".

The SYNC message is a message with no data or with one byte data which is ignored. The default Identifier = 0x80 (=128).

COB-ID	Byte 0
0x80 (=128)	SYNC

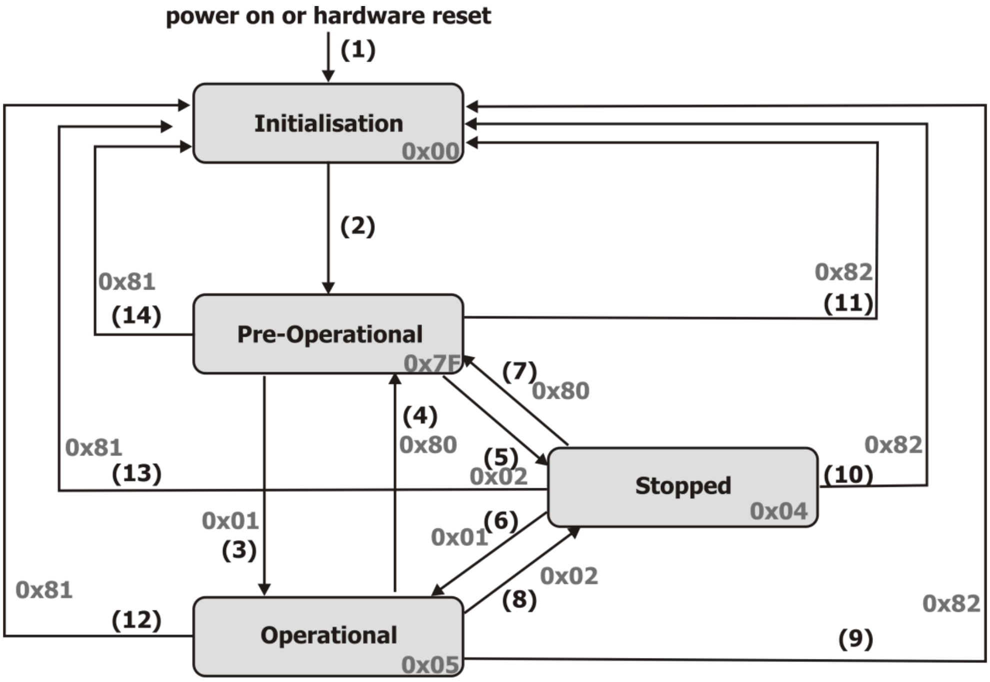
9.7.7 NMT Functions

The NMT (= Network Management) functions describe the NMT Statemachine and NMT error saving functions.

The NMT state machine is controlled by NMT commands. The error control functions guarding and heartbeat are set up by associated communication objects and controlled by special protocols.

The NMT-State is displayed via the actual value parameter *Node-State* **1290**.

NMT state machine



A change of NMT-State may also be triggered by a communication (Bus-off, Guarding, etc.). The behaviour of the NMT state machine in such a case is described in Chapter 9.8.9 “0x1029/n Error Behavior”.

transition	NMT command
(1)	At power on NMT state Initialisation is entered autonomously
(2)	NMT state Initialisation finished → NMT state Pre-Operational entered automatically, device sends Boot-Up message
(3)	Start Remote Node
(4), (7)	Enter Pre-Operational
(5), (8)	Stop Remote Node
(6)	Start Remote Node
(9), (10), (11)	Reset Node. Communication objects 0x1nnn and application objects 0x6nnn are reset.
(12), (13), (14)	Reset Communication. Communication objects 0x1nnn are reset.

In state transition (2) Initialisation ➔ Pre-Operational the device sends the Boot-Up message.

9.7.7.1 Boot-Up message

Identifier	Byte 0
0x700 (=1792) + Node-ID	0

The Boot-Up message is sent automatically when the device is powered on or reset (i.e. fault reset). This helps the PLC recognizing to switch on a device (i.e. after a power failure and recovery) reliable during operation without Nodeguarding.

If the inverter is switched on after the PLC, the PLC can use this boot-up message to begin the initialization. The boot-up message signals the PLC, that the inverter is ready for the PLC to communicate. Using a NMT telegram "Reset Node" or "Reset Communication" forces a Reset of the node communication and results in a Boot-Up message.

Please refer also to chapter 9.7.8 "Guarding".

9.7.7.2 NMT commands

	Byte 0	Byte 1
Identifier	Command Specifier	Node-ID
0	cs	id

id = 0	command addressed to **all** devices
id = 1…0x7F (=127)	command addressed to device with Node-ID = id
cs: 1	Start Remote Node
2	Stop Remote Node
0x80 (=128)	Enter Pre-Operational
0x81 (=129)	Reset Node
0x82 (=130)	Reset Communication

NMT states and active communication objects:

	Pre-Operational	**Operational**	**Stopped**
PDO		X	
SDO	X	X	
SYNC	X	X	
Emergency	X	X	
Node control + NMT error control *****	X	X	X

***** NMT commands + Guarding/Heartbeat function

9.7.8 Guarding

Guarding response:

The inverter responds to every guarding request of the PLC. This is used by some PLCs when powering on to search for available devices. This response is done always independent of the settings of objects 0x100C/0 *Guard Time* and 0x100D/0 *Lifetime Factor.*

Guarding activation:

The Guarding is set whenever objects 0x100C/0 *Guard Time* and 0x100D/0 *Lifetime Factor* are both unequal to zero. The resulting guarding time is *Guard Time* **x** *Lifetime Factor*. Guarding is activated after setting the objects and on reception of the first guarding request.

Guarding fault behaviour:

If the inverter does not receive a guarding request within the specified guarding time a guarding event is triggered. The inverter's reaction to this guarding event is defined by objects 0x6007 *abort connection option code* and 0x1029 *error behaviour*.

Guarding sequence:

The PLC sends via a RTR (Remote Transmission Request) a guarding request with Identifier 0x700 (= 1792) + Node-ID (no data bytes). This remote frame is answered by the inverter with the same Identifier and one data byte. The data byte contains a toggle bit and the NMT state of the inverter.

PLC:

Identifier
0x700 (=1792)+ Node-ID RTR

Inverter:

	Byte 0							
Identifier	NMT state + toggle bit							
0x700 + Node-ID	7	6	5	4	3	2	1	0
	t	NMT state						

t: Toggle bit toggled on each transmission (first transmission t = 0)

NMT state:		
	0	Boot-Up
	4	Stopped
	5	Operational
	0x7F (=127)	Pre-Operational

9.7.9 Heartbeat

The heartbeat uses the producer/consumer method. The inverter as heartbeat consumer can monitor up to three heartbeat producers. The inverter can also send the heartbeat message (as heartbeat producer). The heartbeat contains the NMT state of the producer.

The heartbeat consumer function is set by object 0x1016/n *Consumer Heartbeat Time*. After setting the object the Monitoring of the heartbeat message(s) starts with reception of the first heartbeat message.

If the inverter does not receive a producer heartbeat message within the specified consumer heartbeat time, a heartbeat event is triggered. The reaction to this heartbeat event is defined by objects 0x6007 *abort connection option code* and 0x1029 *error behaviour*.

The heartbeat producer function is set by object 0x1017 *Producer Heartbeat Time*. If object 0x1017 *Producer Heartbeat Time* is set unequal to zero the inverter sends a heartbeat message periodically.

Heartbeat message structure:

	Byte 0							
Identifier	NMT state							
0x700 (=1792) + Node-ID	7	6	5	4	3	2	1	0
	r	NMT state						

r: reserved (always 0)

NMT state:		
	0	Boot-Up
	4	Stopped
	5	Operational
	127	Pre-Operational

9.8 Special CANopen objects

CANopen® uses for Communication several objects that are not used with Ethernet I/P®. These are described in this chapter.

For easier usage, the objects are summarized in a table in each paragraph. This table is marked additionally by color.

Orange color	= Rd Only object
Green color	= Read and Write object
Blue color	= Write only object

Abbreviations used

Access:	Access type
r/w:	R/W
ro:	Rd Only
wo:	Write only
Map:	Mapping
Def.-Val:	Default value

The examples show some typical data telegrams that can be watched or used with a CAN analysis tool. The order of the examples consider the standard CANopen® Format: Lowest Byte left, Highest Byte right.

The headings are displayed in the format *Index/Subindex Object name*.

The objects which can be used for either Ethernet I/P® or CANopen® are described in chapter 9.9 "CANopen objects" to prevent double descriptions in this manual.

9.8.1 0x1005/0 COB-ID SYNC Message

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1005	0	COB-ID SYNC Message	Unsigned 32	r/w	No	0

Object 0x1005 *COB-ID SYNC message* defines the identifier for the SYNC message as well as whether the CANopen® device generates the SYNC.

The default value of this object is 128 (identifier = 128, SYNC message not generated).

Object 0x1005/0				
Bit 31	Bit 30	Bit 29	Bit 11 … 28	Bit 0 … 10
X	gen	frame	0	11 bit CAN-ID

Bit 31: X = don't care

Bit 30: 0 = SYNC message not generated
1 = SYNC message generated

Bit 29: 0 = 11 bit ID
1 = 29 bit ID **NOT ALLOWED**

Bit 0 … 10: 11 bit CAN-ID

9.8.2 0x1006/0 Communication Cycle Period

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1006	0	Communication Cycle Period	Unsigned 32	r/w	No	0

The *communication cycle period* is the time distance between two consecutive SYNC messages. The SYNC message is used by the inverter for synchronisation of the motion control system to the SYNC message. This is especially important for the interpolated position mode.

The value for *communication cycle period* is given in multiples of micro seconds.

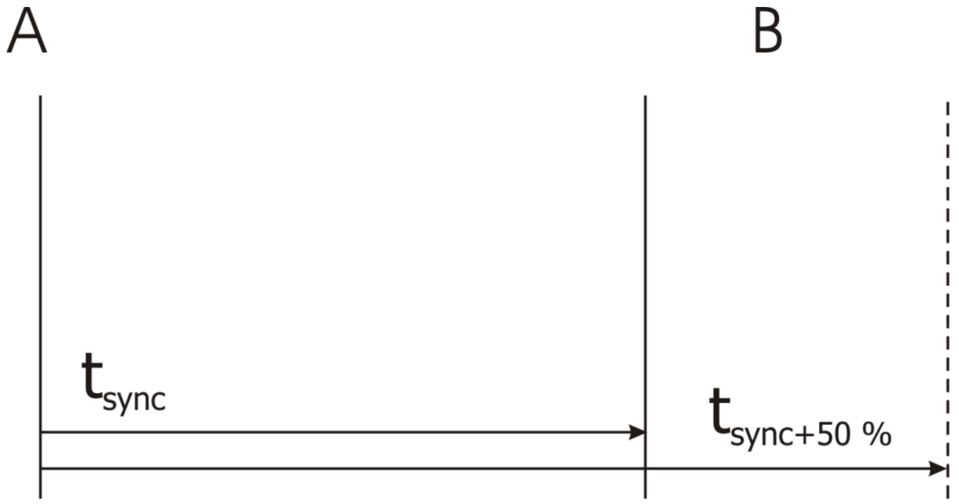
Values smaller than 20000 (20 ms) are allowed.

The synchronization of the inverter to an external clock has to be met under the condition, that at least one RxPDO or TxPDO is defined as synchronous object and is activated. The definitions of the TxPDO / RxPDO objects can be changed via objects 0x1400 / 0x1800.

The inverter can only process the SYNC mechanism in multiples of milliseconds. For this reason the allowed values for object 0x1006/0 *communication cycle period* are multiples of milliseconds.

E.g.: 0x1006/0 = 4000 = 4 ms

If the *communication cycle period* is NOT set (0x1006/0 = 0), the inverter measures the time distance between the SYNC messages over the first 11 messages. Please note, that the monitoring function is deactivated for setting "0". The measurement is solely for internal uses of the frequency inverter. The time must not change after the measurement.



9.8.3 0x1007/0 Synchronous window length

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1007	0	Synchronous window length	Unsigned 32	r/w	No	See Text

Synchronous window length is the time span after a SYNC message in which the inverter is supposed to update its data from receive PDOs and to send transmit PDOs. If either of these actions is not possible in the specified time an emergency message is sent and all remaining synchronous PDOs are discarded until the next SYNC message.

The value for *synchronous window length* is given in multiples of micro seconds.

E.g.: 0x1007/0 = 2000 = 2 ms

If object 0x1007/0 *synchronous window length* is not set (= zero), then this monitoring function is not active.

To avoid unnecessary bus load, the emergency message is sent once only. The next emergency message concerning this problem will be sent after the successful processing of all synchronous PDOs within the *synchronous window length* and a new violation of *synchronous window length*.

9.8.4 0x100C/0 Guard Time

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x100C	0	Guard time	Unsigned 16	r/w	No	0

The response monitoring time is calculated by the multiplication of the objects g*uard time* and *lifetime factor*. Object 0x100C/0 defines the *guard time* in units of one millisecond. *Guard time* = 0 deactivates the guarding function.

If the response monitoring time is exceeded, the node reacts as defined by the setting of object 0x6007 *abort connection option code*.

9.8.5 0x100D/0 Lifetime Factor

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x100D	0	Lifetime factor	Unsigned 8	r/w	No	0

The object "Lifetime Factor" is the multiplier for *guard time. Lifetime factor* = 0 deactivates the guarding function.

9.8.6 0x1014/0 COB-ID Emergency Message

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1014	0	COB-ID Emergency Message	Unsigned32	r/w	No	See text

The identifier and thus the definition of the priority for the emergency message can be set with object 0x1014/0.

The default value of the identifier is 128 + Node ID (valid).

Object 0x1014/0				
Bit 31	Bit 30	Bit 29	Bit 11 ... 28	Bit 0 … 10
valid	0	frame	0	11 bit CAN-ID

Bit 31: 0 = EMCY existent / valid
1 = EMCY non-existent / not valid

Bit 29: 0 = 11 Bit ID
1 = 29 Bit ID **NOT ALLOWED**

Bit 0 … 10: 11 bit CAN-ID

The emergency message is transmitted with the emergency message COB-ID and comprises eight bytes. This object is generated in individual cases and the fault acknowledgement signaled by an emergency message with the data contents equal to zero. The contents are coded according to the following table:

Emergency Message	
Byte	**Contents**
0	Low-byte error code (0x603F)
1	High-byte error code (0x603F)
2	Error register (0x1001)
3	**0**
4	**0**
5	**0**
6	Low-byte, internal error code
7	High-byte, internal error code

Bytes 0, 1 and 2 have a fixed definition within the emergency object. Bytes 6 and 7 are used product-specifically on the basis of the specification.

9.8.7 0x1016/n Consumer Heartbeat Time

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1016	0	Highest sub-index supported	Unsigned8	ro	No	3
	1	Consumer Heartbeat Time 1	Unsigned32	r/w	No	See text
	2	Consumer Heartbeat Time 2	Unsigned32	r/w	No	See text
	3	Consumer Heartbeat Time 3	Unsigned32	r/w	No	See text

Up to three heartbeat producers can be monitored with object 0x1016/n (controlled via sub-indexes n = 1 ... 3). Setting "Consumer Heartbeat Time" = 0 means no monitoring.

Node ID identifies the device to be monitored. The *Heartbeat Time* states the maximum time in milliseconds between two heartbeat messages of the heartbeat producer to be monitored. If this time is exceeded, the monitoring node reacts as defined by the setting of object 0x6007 *abort connection option code*.

Value of the consumer heartbeat time		
Bit 24 to Bit 31	Bit 16 to Bit 23	Bits 0 to Bit 15
not used	Node ID	Heartbeat Time

9.8.8 0x1017/0 Producer Heartbeat Time

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1017	0	Producer Heartbeat Time	Unsigned16	r/w	No	0 ms

The time for the transmission of a heartbeat object is set with object 0x1017/0. The setting "Producer Heartbeat Time" = 0 means that no heartbeat object is transmitted.

9.8.9 0x1029/n Error Behavior

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1029	0	Highest sub-index supported	Unsigned8	ro	No	1
	1	Communication error	Unsigned8	r/w	No	0

The Error Behavior object defines the behavior of the NMT state machine in the event of a communication error (BusOff, Guarding, Heartbeat, SYNC, RxPDO-length).

Value	Function
0	Change to NMT state Pre-Operational (default) (only if currently in NMT state Operational)
1	No change of NMT state
2	Change to NMT state Stopped

9.8.10 0x1200/n SDO Server Parameter

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1200	0	Highest sub-index supported	Unsigned8	ro	No	2
	1	COB-ID client → server (Rx)	Unsigned32	ro	No	See text
	2	COB-ID server → client (Tx)	Unsigned32	ro	No	See text

Object 0x1200 defines the SDO server parameters. The values are read-only and pre defined according to the device node address.

COB-ID client → server (Rx) = 1536 + node address

COB-ID server → client (Tx) = 1408 + node address

Object 0x1200/1, 2				
Bit 31	Bit 30	Bit 29	Bit 11 … 28	Bit 0 … 10
valid	0	frame	0	11 bit CAN-ID

Bit 31: **0** = SDO existent / valid

Bit 29: **0** = 11 Bit ID

Bit 0 … 10: 11 bit CAN-ID

9.8.11 0x1400/n, 0x1401/n, 0x1402/n RxPDO Communication Parameters

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1400 0x1401 0x1402	0	Highest sub-index supported	Unsigned8	ro	No	2
	1	COB ID	Unsigned32	rw	No	See text
	2	Transmission type	Unsigned8	rw	No	See text
	3	Inhibit time	Unsigned16	rw	No	See text
	4	-	-	-	-	-
	5	Event time	Unsigned16	rw	No	See text

RxPDO Communication parameters:

0x1400/n RxPDO1	COB-ID Default value: 0x200 (=512) +Node ID
0x1401/n RxPDO2	COB-ID Default value: 0x300 (=768) +Node ID
0x1402/n RxPDO3	COB-ID Default value: 0x400 (=1024) +Node ID

These communication parameters define the COB-ID and transmission type used by the RxPDOs. Only sub-index 1,2 and 5 are used for RxPDOs. The default setting for the used COB-ID depends on the Node ID and can be changed. The default value for transmission type is 255 (event driven) and can also be changed (see table).

Object 0x1400/0x1401/0x1402 COB-ID				
Bit 31	Bit 30	Bit 29	Bit 11 ... 28	Bit 0 … 10
valid	**0**	frame	**0**	11 bit CAN-ID

Bit 31: 0 = PDO existent/valid
1 = PDO non-existent/not valid

Bit 29: 0 = 11 Bit ID
1 = 29 Bit ID **NOT ALLOWED**

Bit 0 … 10: 11 bit CAN-ID

RxPDO1 factory setting = valid

RxPDO2/3 factory setting = not valid

Object 0x1400/0x1401/0x1402 transmission type		
value	**meaning**	**description**
0	synchronous	Update RxPDO data on **each** SYNC
1 … 240	synchronous	Update RxPDO data on **each** SYNC
~~241 … 251~~	reserved	Value not allowed
~~252~~	synchronous/RTR	Value not allowed
~~253~~	asynchronous/RTR	Value not allowed
254	asynchronous	Event driven (manufacturer specific)
255	asynchronous	Event driven (profile specific) **default value**

Values 254 & 255 are handled identically. Update RxPDO data on each Rx.

Inhibit time:

The inhibit time for RxPDO is without function. Values can be entered, but are without further function.

Event time:

The event time is used as monitoring function for RxPDO’s. If during the set time no RxPDO is received, one of the following faults is triggered:

202A Fault RxPDO1

202B Fault RxPDO2

202C Fault RxPDO3

9.8.12 0x1800/n, 0x1801/n, 0x1802/n TxPDO Communication Parameters

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1800 0x1801 0x1802	0	Highest sub-index supported	Unsigned8	ro	No	5
	1	COB ID	Unsigned32	rw	No	See text
	2	Transmission type	Unsigned8	rw	No	255
	3	Inhibit time	Unsigned16	rw	No	See text
	4	-	-	-	-	-
	5	Event time	Unsigned16	rw	No	See text

TxPDO Communication parameters:

0x1800/n TxPDO1	COB-ID Default value: 0x180 (=384) +Node ID
0x1801/n TxPDO2	COB-ID Default value: 0x280 (=640) +Node ID
0x1802/n TxPDO3	COB-ID Default value: 0x380 (=896) +Node ID

These communication parameters define the COB-ID and transmission type used by the TxPDOs. The default setting for the COB-ID depends on the Node ID and can be changed. The default value for the transmission type is 255 (event driven) and can also be changed (see table).

Object 0x1800/0x1801//1802 COB-ID				
Bit 31	Bit 30	Bit 29	Bit 11 ... 28	Bit 0 … 10
valid	0	frame	0	11 bit CAN-ID

Bit 31: 0 = PDO existent / valid
1 = PDO non-existent / not valid

Bit 29: 0 = 11 Bit ID
1 = 29 Bit ID **NOT ALLOWED**

Bit 0 … 10: 11 bit CAN-ID

TxPDO1 factory setting = valid
TxPDO2/3 factory setting = not valid

Object 0x1800/0x1801/0x1802 transmission type		
value	**meaning**	**description**
0	Synchronous	Update TxPDO data and send on SYNC **only** when data has changed
1 … 240	Synchronous	Update TxPDO data and send on **each** "n" SYNC
241 … 251	Reserved	Value not allowed
252	synchronous/RTR	Update TxPDO data on SYNC and send **on following** RTR
253	asynchronous/RTR	Update TxPDO data and send on RTR
254	asynchronous	Event driven (manufacturer specific)
255	asynchronous	Event driven (profile specific) **default value**

Values 254 + 255 are handled identically. Send TxPDO on data change or event time.

Inhibit time: The inhibit time is the minimum time distance between two consecutive TxPDOs for asynchronous TxPDOs. During the inhibit time, the TxPDO is not send again. Therefore a value change occurring in this time is send earliest after the inhibit time has elapsed. The value range is 0…65535.

The inhibit time is set in hundreds of microseconds, e. g. a value of 300 is 300 *100 us = 30 ms.

The device internal time resolution for the inhibit time is in milliseconds, the last digit is always converted to "0". An inhibit time value = 37 is truncated to 30 [3.7 ms → 3 ms].

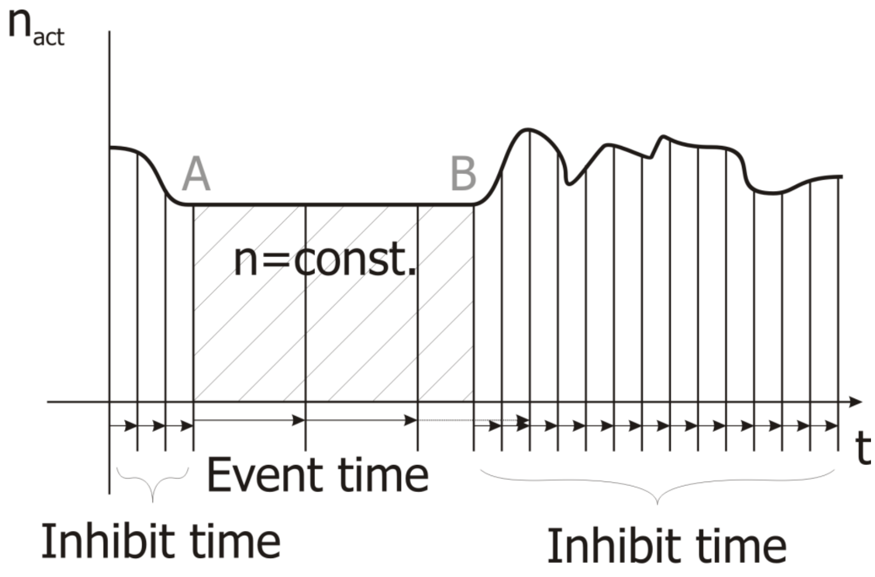
Values less than 10 are interpreted as 0.

Event time: The event time is the time distance between two consecutive TxPDOs whenever the TxPDO data has not changed (cycle time). If the inhibit time is set to zero the TxPDO is only sent on a change of the TxPDO's data. The value range is 0…65535.

The event time is set in milliseconds, e.g. a value of 2000 = 2000 ms.

Example Event time & Inhibt time:

The actual speed value is transferred via TxPDO. The value is updated after the inhibit time has elapsed. At time A, the value remains constant. During this time, the value is updated after the Event time has elapsed. At time B, the value changes and is transmitted via TxPDO. The value changes again frequently and is only updated after the inhibit time has elapsed



Sub index 4:

Sub-index 4 is included for compatibility reasons. An SDO read/write access to sub index 4 results in an SDO abort.

9.8.13 0x3000/0 SYNC Jitter

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3000	0	SYNC Jitter	Unsigned16	rw	No	See Text

DS301 does not include an object for monitoring the jittering of the SYNC message. ANG inverters monitor SYNC message jittering with object 0x3000/0 *SYNC Jitter* (given in multiples of micro seconds).

If the SYNC message is received outside the time defined by:

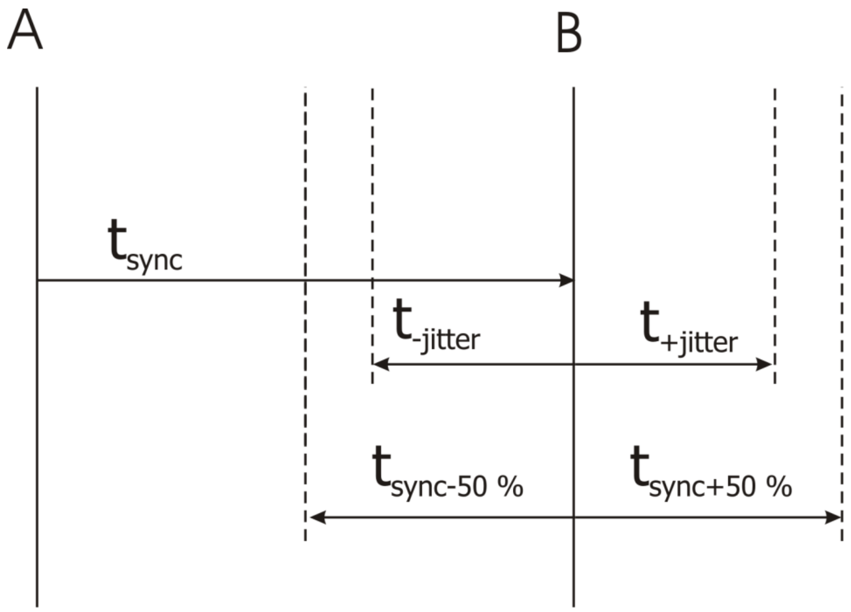
0x1006/0 communication cycle period* +/- 0x3000/0 *SYNC Jitter

a communication error event is triggered.

The value for object 0x3000/0 *SYNC Jitter* depends on the CANopen® master's capability for time accuracy. The value range is 0 … 17.000 (µs) and is in addition internally restricted to 50% of the *communication cycle period* (object 0x1006/0 or the measured value).

If object 0x3000/0 *SYNC Jitter* is set to 0, there is no monitoring of the SYNC message time.

If object 0x3000/0 *SYNC Jitter* is set to ≠ 0 then monitoring of the SYNC message time is active. The jitter monitoring is independent of how the communication cycle period is determined (either set with object 0x1006/0 or by measuring).



Last SYNC “A” Expected SYNC “B”
SYNC jitter SYNC jitter
Internal limit Internal limit

9.9 CANopen objects

This Chapter "CANopen Objects" and the following Chapters "Motion Control Interface (MCI)"and "Control" are only relevant when using Ethernet/IP manufacturer specific assemblies 100/101.

Some parameters available to the user are not described in the following subchapters. They can be used to customize specific functions in application. Please contact the manufacturer for details.

9.9.1 Objects tabular overview

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

Access type			
Rd Only	The PLC can only read data from the frequency inverter.		
R/W	The PLC is granted unlimited access (reading and writing) to the frequency inverter data.		
Data type			
Unsigned32	32 Bit value:	0…2^{32}-1 0…0xFFFF FFFF	
Unsigned16	16 Bit value:	0…2^{16}-1 0…0x FFFF	(0…65535)
Unsigned8	8 Bit value:	0…2^{8}-1 0…0xFF	(0…255)
Integer32	Signed 32 Bit value:	-2^{31}…2^{31}-1 0x8000 0000…0x7FFF FFFF	
Integer16	Signed 16 Bit value:	2^{15}…2^{15}-1 0x8000…0x7FFF	(-32768…32767)
Integer8	Signed 8 Bit value: -	2^{7}…2^{7}-1 0x80…0x7F	(-128…127)
Visible string	String up to 99 characters long. Transmission via Segmented Transfer.		
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.

9.9.1.1 Communication objects

The communication objects listed in the table below do not support PDO mapping.

Index	SubIndex	Name	SDO Acc.	Data type
0x1000	0	Device type	Rd Only	Unsigned32
0x1001	0	Error register	Rd Only	Unsigned8
0x1008	0	Manufacturer device name	Rd Only	Visible string
0x1009	0	Manufacturer hardware version	Rd Only	Visible string
0x100A	0	Manufacturer software version	Rd Only	Visible string
0x1010		Store parameters		
	0	Highest Sub-index supported	Rd Only	Unsigned8
	1	Save all parameters	R/W	Unsigned32
	2	Save communication parameters	R/W	Unsigned32
	3	Save application parameters	R/W	Unsigned32

Index	SubIndex	Name	SDO Acc.	Data type
0x1011		Restore default parameters		
	0	Highest Sub-index supported	Rd Only	Unsigned8
	1	Restore all default parameters	R/W	Unsigned32
	2	Restore communication default parameters	R/W	Unsigned32
	3	Restore application default parameters	R/W	Unsigned32
0x1018		Identity object		
	0	Highest Sub-index supported	Rd Only	Unsigned8
	1	Vendor ID	Rd Only	Unsigned32
	2	Product code	Rd Only	Unsigned32
	3	Revision number	Rd Only	Unsigned32
	4	Serial number	Rd Only	Unsigned32
0x1600		RxPDO1 mapping parameter		
	0	No. of mapped objects	R/W	Unsigned8
	1	1. mapped obj.	R/W	Unsigned32
	2	2. mapped obj.	R/W	Unsigned32
	3	3. mapped obj.	R/W	Unsigned32
	4	4. mapped obj.	R/W	Unsigned32
	5	5. mapped obj.	R/W	Unsigned32
	6	6. mapped obj.	R/W	Unsigned32
	7	7. mapped obj.	R/W	Unsigned32
	8	8. mapped obj.	R/W	Unsigned32
0x1601		RxPDO2 mapping parameter		
	0	No. of mapped objects	R/W	Unsigned8
	1	1. mapped obj.	R/W	Unsigned32
	2	2. mapped obj.	R/W	Unsigned32
	3	3. mapped obj.	R/W	Unsigned32
	4	4. mapped obj.	R/W	Unsigned32
	5	5. mapped obj.	R/W	Unsigned32
	6	6. mapped obj.	R/W	Unsigned32
	7	7. mapped obj.	R/W	Unsigned32
	8	8. mapped obj.	R/W	Unsigned32
0x1602		RxPDO3 mapping parameter		
	0	No. of mapped objects	R/W	Unsigned8
	1	1. mapped obj.	R/W	Unsigned32
	2	2. mapped obj.	R/W	Unsigned32
	3	3. mapped obj.	R/W	Unsigned32
	4	4. mapped obj.	R/W	Unsigned32
	5	5. mapped obj.	R/W	Unsigned32
	6	6. mapped obj.	R/W	Unsigned32
	7	7. mapped obj.	R/W	Unsigned32
	8	8. mapped obj.	R/W	Unsigned32
0x1A00		TxPDO1 mapping parameter		
	0	No. of mapped objects	R/W	Unsigned8
	1	1. mapped obj.	R/W	Unsigned32
	2	2. mapped obj.	R/W	Unsigned32
	3	3. mapped obj.	R/W	Unsigned32
	4	4. mapped obj.	R/W	Unsigned32
	5	5. mapped obj.	R/W	Unsigned32
	6	6. mapped obj.	R/W	Unsigned32
	7	7. mapped obj.	R/W	Unsigned32
	8	8. mapped obj.	R/W	Unsigned32
0x1A01		TxPDO2 mapping parameter		
	0	No. of mapped objects	R/W	Unsigned8
	1	1. mapped obj.	R/W	Unsigned32
	2	2. mapped obj.	R/W	Unsigned32
	3	3. mapped obj.	R/W	Unsigned32
	4	4. mapped obj.	R/W	Unsigned32
	5	5. mapped obj.	R/W	Unsigned32
	6	6. mapped obj.	R/W	Unsigned32
	7	7. mapped obj.	R/W	Unsigned32
	8	8. mapped obj.	R/W	Unsigned32

Index	SubIndex	Name	SDO Acc.	Data type
0x1A02		TxPDO3 mapping parameter		
	0	No. of mapped objects	R/W	Unsigned8
	1	1. mapped obj.	R/W	Unsigned32
	2	2. mapped obj.	R/W	Unsigned32
	3	3. mapped obj.	R/W	Unsigned32
	4	4. mapped obj.	R/W	Unsigned32
	5	5. mapped obj.	R/W	Unsigned32
	6	6. mapped obj.	R/W	Unsigned32
	7	7. mapped obj.	R/W	Unsigned32
	8	8. mapped obj.	R/W	Unsigned32

9.9.1.2 Manufacturer objects

Index	SubIndex	Name	SDO Acc.	Data type	PDO-mapping	Default	Min…Max
0x2nnn	0, 1, … 9	Manufacturer specific Direct access to inverter parameters R/W access by SDO transfer only					
		Please refer to chapter	9.9.3.1 "Handling of data sets/cyclic writing of the parameters"				
0x3001	0	Digital In actual values	Rd Only	Unsigned16	Tx	-	-
0x3002	0	Digital Out actual values	Rd Only	Unsigned16	Tx	-	-
0x3003	0	Digital Out set values	R/W	Unsigned16	Rx	0	0…0x1F
0x3004	0	Boolean Mux	Rd Only	Unsigned16	Tx	-	-
0x3005	0	Boolean Demux	R/W	Unsigned16	Rx	0	0…0xFFFF
0x3006	0	Percentage set value	R/W	Unsigned16	Rx	0	0x8AD0… 0x7530
0x3007	0	Percentage actual value 1	Rd Only	Unsigned16	Tx	-	-
0x3008	0	Percentage actual value 2	Rd Only	Unsigned16	Tx	-	-
0x3011	0	Act. value Word 1	Rd Only	Unsigned16	Tx	-	-
0x3012	0	Act. value Word 2	Rd Only	Unsigned16	Tx	-	-
0x3021	0	Act. value Long 1	Rd Only	Unsigned32	Tx	-	-
0x3022	0	Act. value Long 2	Rd Only	Unsigned32	Tx	-	-
0x3111	0	Ref. value Word 1	R/W	Unsigned16	Rx	0	0…0xFFFF
0x3112	0	Ref. value Word 2	R/W	Unsigned16	Rx	0	0…0xFFFF
0x3121	0	Ref. value Long 1	R/W	Unsigned32	Rx	0	0… 0xFFFF.FFFF
0x3122	0	Ref. value Long 2	R/W	Unsigned32	Rx	0	0… 0xFFFF.FFFF
0x5F10		Gear factor [9)]					
	0	Highest sub-index supported	Rd Only	Unsigned8	No	-	-
P.1123	1	Numerator	R/W	Integer16	Rx	1	1…0x7FFF
P.1124	2	Denominator	R/W	Unsigned16	Rx	1	1…0xFFFF
P.1142	3	Resync on Change	R/W	Integer16	No	1	0…1
0x5F11		Phasing 1 [9)]					
	0	Highest sub-index supported	Rd Only	Unsigned8	No	-	-
P.1125 DS1	1	Offset	R/W	Integer32	No	0x0001.0000	0x8000.0001 … 0x7FFF.FFFF
P.1126 DS1	2	Speed	R/W	Unsigned32	No	0x0005.0000	1… 0x7FFF.FFFF

Index	SubIndex	Name	SDO Acc.	Data type	PDO-mapping	Default	Min…Max
P.1127 DS1	3	Acceleration	R/W	Unsigned32	No	0x0005.0000	1… 0x7FFF.FFFF
0x5F12		Phasing 2 [g)]					
	0	Highest sub-index supported	Rd Only	Unsigned8	No	-	-
P.1125 DS2	1	Offset	R/W	Integer32	No	0x0001.0000	0x8000.0001 … 0x7FFF.FFFF
P.1126 DS2	2	Speed	R/W	Unsigned32	No	0x0005.0000	1… 0x7FFF.FFFF
P.1127 DS2	3	Acceleration	R/W	Unsigned32	No	0x0005.0000	1… 0x7FFF.FFFF
0x5F13		Phasing 3 [g)]					
	0	Highest sub-index supported	Rd Only	Unsigned8	No	-	-
P.1125 DS3	1	Offset	R/W	Integer32	No	0x0001.0000	0x8000.0001 … 0x7FFF.FFFF
P.1126 DS3	2	Speed	R/W	Unsigned32	No	0x0005.0000	1… 0x7FFF.FFFF
P.1127 DS3	3	Acceleration	R/W	Unsigned32	No	0x0005.0000	1… 0x7FFF.FFFF
0x5F14		Phasing 4 [g)]					
	0	Highest sub-index supported	Rd Only	Unsigned8	No	-	-
P.1125 DS4	1	Offset	R/W	Integer32	No	0x0001.0000	0x8000.0001 … 0x7FFF.FFFF
P.1126 DS4	2	Speed	R/W	Unsigned32	No	0x0005.0000	1… 0x7FFF.FFFF
P.1127 DS4	3	Acceleration	R/W	Unsigned32	No	0x0005.0000	1… 0x7FFF.FFFF
0x5F15 **P.1168**	0	In Gear Threshold	R/W	Unsigned32	No	0	0…0x7FFF.FFFF
0x5F16 **P.1169**	0	In Gear Time	R/W	Unsigned16	No	10	1…0xFFFF
	0	Highest sub-index supported	Rd Only	Unsigned8	No	-	-
P.1104	1	Time Constant [ms]	R/W	Integer32	No	10,00	1,00…300,00
P.1118	2	Limitation	R/W	Unsigned32	No	327680	0… 0x7FFF.FFFF
0x5F18 **P.1284**	0	Master Synchronization Offset [g)]	R/W	Integer32	No	0	0x8000.0001 … 0x7FFF.FFFF
0x5FF0	0	Active motion block [t)]	Rd Only	Unsigned8	Tx	-	-
0x5FF1	0	Motion block to resume [t)]	Rd Only	Unsigned8	Tx	-	-

v) Velocity Mode only: This Object is only used in Velocity mode [rpm].
u) Profile Velocity Mode only: This Object is only used in Profile Velocity mode [u/s].
h) Homing Mode only: This Object is only used in Homing mode.
p) Profile Position Mode only: This Object is only used in Profile Position mode.
g) Electronic gear: slave Mode only: This Object is only used in Electronic Gear mode.
t) Table travel record mode only: This Object is only used in Table travel record mode.

9.9.1.3 Device profile objects

Index	SubIndex	Designation	SDO Access	Data type	PDO-mapping	Default setting	Min…Max
0x6007 **P.388**	0	Abort connection option code	R/W	Integer16	No	1	-2…3
0x603F	0	Error code	Rd Only	Unsigned16	No	-	-
0x6040 **P.410**	0	Control word	R/W	Unsigned16	Rx	-	-
0x6041 **P.411**	0	Status word	Read/only	Unsigned16	Tx	-	-
0x6042	0	vl target velocity v)	R/W	Interger16	Rx	0	-32768…32767
0x6043	0	vl velocity demand	Rd Only	Integer16	Tx	-	-
0x6044	0	vl velocity actual value	Rd Only	Integer16	Tx	-	-
0x6046		vl velocity min max amount					
	0	Highest sub-index supported	Rd Only	Unsigned8	No	-	-
P.418	1	vl velocity min amount	R/W	Unsigned32	No	0	0…32767
P.419	2	vl velocity max amount	R/W	Unsigned32	No	32767	0…32767
0x6048		vl velocity acceleration v)				-	-
	0	Highest sub-index supported	Rd Only	Unsigned8	No		
P.420	1	Delta speed	R/W	Unsigned32	No	150	1…32767
P.422	2	Delta time	R/W	Unsigned16	No	1	1…65535
0x6049		vl velocity deceleration v)					
	0	Highest sub-index supported	Rd Only	Unsigned8	No	-	-
P.421	1	Delta speed	R/W	Unsigned32	No	150	1…32767
P.423	2	Delta time	R/W	Unsigned16	No	1	1…65535
0x604A		vl velocity quick stop v)					
	0	Highest sub-index supported	Rd Only	Unsigned8	No	-	-
P.424	1	Delta speed	R/W	Unsigned32	No	150	1…32767
P.425	2	Delta time	R/W	Unsigned16	No	1	1…65535
0x6060	0	Modes of operation	Write only	Integer8	Rx	2	-3…9
0x6061	0	Modes of operation display	Rd Only	Integer8	Tx	2	-
0x6064 **P.1108**	0	Position actual value	Rd Only	Integer32	Tx	-	0x8000.0001 … 0x7FFF.FFFF
0x6065 **P.1105**	0	Following error window	R/W	Unsigned32	No	0xFFFF.FFFF	0… 0xFFFF.FFFF
0x6066 **P.1119**	0	Following error time out	R/W	Unsigned16	No	10	0…65535
0x6067 **P.1165**	0	Position window	R/W	Unsigned32	No	0xFFFF.FFFF	0… 0xFFFF.FFFF
0x6068 **P.1166**	0	Position window time	R/W	Unsigned16	No	10	0…65535
0x606C	0	Velocity Actual value u)	Read	Integer32	Tx		
0x606D **P.1276**	0	Velocity Window u)	R/W	Unsigned16	No	1000	0…65535

Index	SubIndex	Designation	SDO Access	Data type	PDO-mapping	Default setting	Min…Max
0x606E **P.1277**	0	Velocity Window Time [u]	R/W	Unsigned16	No	0	0…65535
0x606F **P.1278**	0	Velocity Threshold [u]	R/W	Unsigned16	No	100	0…65535
0x6070 **P.1279**	0	Velocity Threshold Time [u]	R/W	Unsigned16	No	0	0…65535
0x6071	0	Target torque	R/W	Integer16	Rx		
0x6077 **P.224**	0	Torque actual value	Rd Only	Integer16	Tx		
0x6078 **P.211**	0	Current actual value	Rd Only	Integer16	Tx		
0x6079 **P.222**	0	DC link circuit voltage	Rd Only	Integer32	Tx		
0x607A **P.1202**	0	Target position [p]	R/W	Integer32	Rx	0	0x8000.0001 … 0x7FFF.FFFF
0x607C **P.1131**	0	Home offset [h]	R/W	Integer32	No	0	0x8000.0001 … 0x7FFF.FFFF
0x6081	0	Profile velocity [p] [u]	R/W	Unsigned32	Rx	0x0005.0000	1… 0x7FFF.FFFF
0x6083	0	Profile acceleration [p] [u]	R/W	Unsigned32	Rx	0x0005.0000	1… 0x7FFF.FFFF
0x6084	0	Profile deceleration [p] [u]	R/W	Unsigned32	Rx	0x0005.0000	1… 0x7FFF.FFFF
0x6085 **P.1179**	0	Quick stop deceleration [h] [p] [t] [u]	R/W	Unsigned32	No	0x000A.0000	1… 0x7FFF.FFFF
0x6086	0	Motion profile type [u]	R/W	Integer16	No	3	0…3
0x6091		Gear ratio					
	0	Highest sub-index supported	Rd Only	Unsigned8	No	-	-
P.1116	1	Motor revolutions	R/W	Unsigned32	No	1	1…65535
P.1117	2	(Driving) Shaft revolutions	R/W	Unsigned32	No	1	1…65535
0x6092		Feed constant					
	0	Highest sub-index supported	Rd Only	Unsigned8	No	-	-
P.1115	1	Feed	R/W	Unsigned32	No	0x0001.0000	1… 0x7FFF.FFFF
	2	(Driving) Shaft revolutions	R/W	Unsigned32	No	1	1
0x6098 **P.1130**	0	Homing method [h]	R/W	Integer8	No	0	0…35
0x6099		Homing speeds [h] [l]					
	0	Highest sub-index supported	Rd Only	Unsigned8	No	-	-
P.1132	1	Speed during search for switch	R/W	Unsigned32	No	0x0005.0000	1… 0x7FFF.FFFF
P.1133	2	Speed during search for zero	R/W	Unsigned32	No	0x0002.0000	1… 0x7FFF.FFFF

Index	SubIndex	Designation	SDO Access	Data type	PDO-mapping	Default setting	Min…Max
0x609A **P.1134**	0	Homing acceleration [h)]	R/W	Unsigned32	No	0x0005.0000	1… 0x7FFF.FFFF
0x60F4 **P.1109**	0	Following error actual value	Rd Only	Integer32	Tx		
0x60F8 **P.1275**	0	Max Slippage [u)]	R/W	Integer32	No	0	
0x60FF	0	Target velocity [u)]	R/W	Integer32	Rx		
0x6502	0	Supported drive modes	Rd Only	Unsigned32	No	0x0202	0x0202

v) Velocity Mode: This Object is used in Velocity mode [rpm].
u) Profile Velocity Mode: This Object is used in Profile Velocity mode [u/s].
h) Homing Mode: This Object is used in Homing mode.
p) Profile Position Mode: This Object is used in Profile Position mode.
t) Table travel record mode: This Object is used in Table travel record mode.
l) Move away from limit switch Mode: This Object is used in Move away from limit switch mode.

The Modes "*Homing*", "*Profile Position*", "*Profile Velocity*", "*Table travel record*" and "*Move away from Limit Switch*" require a configuration capable of Positioning. Check chapter

Motion control configurations" for details.

The representations of CANopen® objects and parameters may be different (refer to relevant object description).

NOTICE

Unexpected behavior!

Some of the above listed CANopen® objects have corresponding inverter parameters.

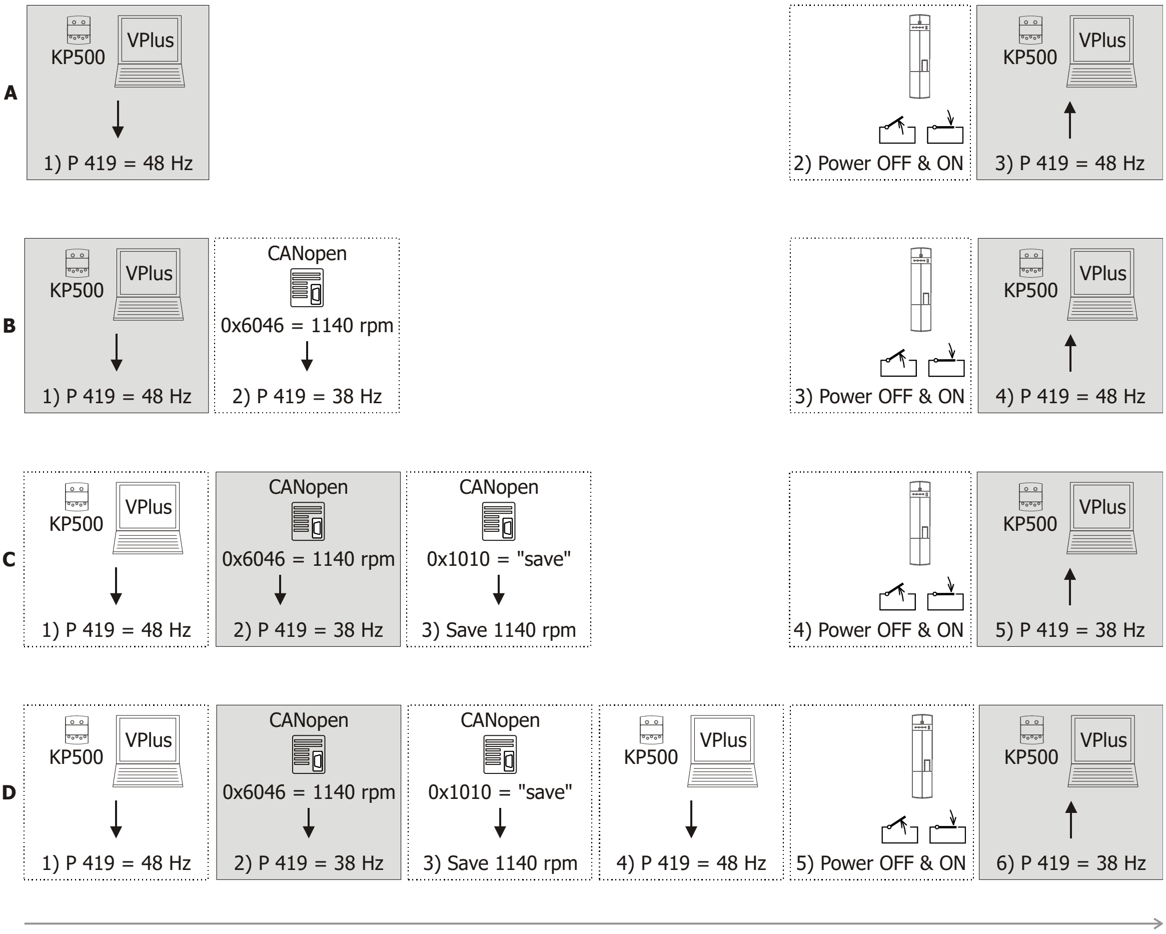
These objects are handled in a special way.

If one of these CANopen® objects has been written by SDO followed by a "save" command (see object 0x1010), the written value is stored to non-volatile memory of the inverter. After the next power-on of the inverter these CANopen® object values are restored again and overwrite the inverter parameter values.

- Be careful when using this method. If a CANopen® object was written and saved and then the corresponding inverter parameter was set by e. g. VPlus or KP500, the next power-on cycle overwrites the value set by VPlus with the value stored by the "save" command.

For more information on VPlus, see chapter 17.

Effect of the "save" command (Object 0x1010) (Example of sequence of parameter entries and object entries)



Sequence

A Value of a parameter is set via KP500 or VPlus. No "save" command.

1) Setting of *Maximum Frequency* **419** = 48 Hz at KP500 or in VPlus.
2) Power OFF and ON.
3) The value of KP500/VPlus is active (48 Hz).

B No "save" command. The value of the CANopen® object is overwritten.

1) Setting of *Maximum Frequency* **419** = 48 Hz at KP500 or in VPlus.
2) Setting of CANopen® object 0x6046 = 1140 rpm* (equivalent to 38 Hz).
3) Power OFF and ON.
4) Parameter value of KP500/VPlus overwrites the value of the CANopen® object. The value of KP500/VPlus is active (48 Hz).

C "Save" command. The value of the CANopen® object is stored.

1) Setting of *Maximum Frequency* **419** = 48 Hz at KP500 or in VPlus.
2) Setting of CANopen® object 0x6046 = 1140 rpm* (equivalent to 38 Hz).
3) "Save" command via CANopen® object 0x1010.
4) Power OFF and ON.
5) The value of CANopen® object 0x6046 is active (38 Hz).

D "Save" command. The value of the CANopen® object is stored – even if the corresponding parameter value has been changed after the "save" command.

1) Setting of *Maximum Frequency* **419** = 48 Hz at KP500 or in VPlus.
2) Setting of CANopen® object 0x6046 = 1140 rpm***** (equivalent to 38 Hz).
3) "Save" command via CANopen® object 0x1010.
4) Setting of *Maximum Frequency* **419** = 48 Hz at KP500 or in VPlus.
5) Power OFF and ON.
6) Value of CANopen® object 0x6046 overwrites the parameter value. The value of CANopen® object 0x6046 is active (38 Hz).

* Internal conversion to a frequency value taking into account the *No. of Pole Pairs* **373**. In this example the number of pole pairs is two (four-pole machine).

NOTICE

Implausible results!

For some frequency parameters, which are calculated from CANopen® DS402 objects, it is necessary to enter the number of pole pairs, e. g. for calculation of the deceleration or acceleration parameters. These calculations use the number of pole pairs from Dataset 1. If the number of pole pairs in the datasets is different, the result of the calculation may be implausible to the user.

- For this reason, we recommend writing the frequency inverter parameters via the SDO channel with objects 0x2nnn (manufacturer) and **not using** the CANopen® DS402 objects. In this way, inconsistencies are avoided.

CANopen® DS402 objects with corresponding frequency inverter parameters are indicated in this manual.

9.9.2 Communication Objects (0x1nnn)

Communication objects *0x1nnn* contain all parameters for CANopen communication.

To facilitate the overview, the objects are summarized by a table in each paragraph. This table is marked additionally by color.

Orange color	= Rd Only object
Green color	= Read and Write object
Blue color	= Write only object

Abbreviations used

Access:	Access type
r/w:	R/W
ro:	Rd Only
wo:	Write only
Map:	Mapping
Def.-Val:	Default value

The headings of the following chapters are displayed in the format *Index/Subindex Object name*.

9.9.2.10x1000/0 Device Type

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1000	0	Device Type	Unsigned 32	ro	No	0

The device identification is carried out upon network startup. The information about the device type and functionality (type) is defined by the CANopen® DS402 standard.

Object 0x1000/0					
Additional Information				Device Profile Number	
Mode Bits		Type			
31	24	23	16	15	0

The standard device profile "*Drives and Motion Control*" used by the frequency inverter is shown as device profile number **402**. The other information specifies the device functionality of the frequency inverter.

Device Profile Number	= 402	drives and motion control
Type	= 42	servo drive
Mode bits	= 0	unused

9.9.2.20x1001/0 Error Register

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1001	0	Error Register	Unsigned 8	ro	No	0

Object 0x1001/0 is the error register for internal frequency inverter errors. Status "no error" (0x1001/0 = 0) or "Error" (0x1001/0 ≠ 0) is displayed.

Detailed information on the device error can be read via VPlus, Parameter *Current error* **259** (see Chapter 18.4 "Error messages").

In case of an error, the PLC can evaluate detailed information via the Emergency Message (see Chapter "" and 9.9.5.2 "0x603F/0 Error code").

Object 0x1001/0	
Bit	
0	Generic error
1	Current
2	Voltage
3	Temperature
4	Communication error
5	Device profile specific error
6	Reserved
7	Manufacturer specific error

9.9.2.30x1008/0 Manufacturer Device Name

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1008	0	Manufacturer Device name	Visible string	ro	No	See Text

The device name is displayed as a sequence of ASCII characters.

Example: "ACTIVE NEXT GENERATION"

9.9.2.40x1009/0 Manufacturer Hardware Version

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1009	0	Manufacturer Hardware version	Visible string	ro	No	See Text

The device version is displayed as a sequence of ASCII characters.

Example: "ANG 400 512 344"

9.9.2.50x100A/0 Manufacturer Software Version

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x100A	0	Manufacturer Software version	Visible string	ro	No	See Text

The software version is displayed as a sequence of ASCII characters.

Example: 8.0.5

9.9.2.60x1010/n Store Parameters

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1010	0	Highest sub-index supported	Unsigned8	ro	No	3
	1	Store all parameters	Unsigned32	r/w	No	See text
	2	Store communication parameters	Unsigned32	r/w	No	See text
	3	Store application parameters	Unsigned32	r/w	No	See text

With object 0x1010/n parameter/object settings can be stored to non-volatile memory. This object supports three sub-indexes with various functions.

Writing of "save" in 0x1010/3 saves all application parameters (0x6nnn) in the non-volatile memory.

Specification for writing the "save" instruction

LSB			MSB
"s"	**"a"**	**"v"**	**"e"**
0x73	0x61	0x76	0x65

Writing of values other than "`save`" will result in cancellation of SDO. The store command is **not** processed.

9.9.2.70x1011/n Restore default Parameters

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1011	0	Highest sub-index supported	Unsigned8	ro	No	3
	1	Restore all parameters	Unsigned32	r/w	No	See text
	2	Restore communication parameters	Unsigned32	r/w	No	See text
	3	Restore application parameters	Unsigned32	r/w	No	See text

With object 0x1011/n, you can reset parameters/objects to the default values. This object supports three sub-indexes with different functions.

Writing of "load" in 0x1011/3 restores all application parameters (0x6nnn).

Specification for writing the "load" instruction

LSB			MSB
"l"	"o"	"a"	"d"
0x6C	0x6F	0x61	0x64

Writing of values other than "load" will result in cancellation of SDO. The restore defaults command is **not** processed.

9.9.2.80x1018/n Identity Object

The object *identity* provides information about the device manufacturer and the device.

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1018	0	Highest Sub-index supported	Unsigned8	ro	No	4
	1	Vendor ID	Unsigned32	ro	No	See text
	2	Product code	Unsigned32	ro	No	See text
	3	Revision number	Unsigned32	ro	No	See text
	4	Serial number	Unsigned32	ro	No	See text

"*Vendor ID*" "`0xD5`" refers to manufacturer **BONFIGLIOLI VECTRON MDS GmbH**.

This "*Vendor ID*" is assigned by the CiA (CAN in Automation) CANopen users' organization.

Product code:

shows the type ID of the frequency inverter.

Revision number:

shows the revision level of the CANopen® system of the frequency inverter.

Serial number:

shows the serial number of the frequency inverter.

9.9.2.90x1600/n, 0x1601/n, 0x1602/n, RxPDO Mapping Parameter

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1600 0x1601 0x1602	0	Number of mapped objects	Unsigned8	rw	No	2
	1	1st mapped obj.	Unsigned32	rw	No	See text
	2	2nd mapped obj.	Unsigned8	rw	No	See text
	3	3rd mapped obj.	Unsigned8	rw	No	See text
	4	4th mapped obj.	Unsigned8	rw	No	See text
	5	5th mapped obj.	Unsigned8	rw	No	See text
	6	6th mapped obj.	Unsigned8	rw	No	See text
	7	7th mapped obj.	Unsigned8	rw	No	See text
	8	8th mapped obj.	Unsigned8	rw	No	See text

RxPDO Mapping parameters:

0x1600/n RxPDO1

0x1601/n RxPDO2

0x1602/n RxPDO3

0x1600/0 = 0 = no object mapped

0x1600/0 = 1 … 8 = 1 … 8 mapped objects

Mapping entry:

MSB			LSB
Object index		Subindex	Length (no. of bits)
High byte	Low byte	si	ll

Examples:

Mapping of 0x6040/0 *Control word* (unsigned16 = 10_{hex}) to 1st mapped object in RxPDO1:

0x1600/1 = 0x60400010

Refer to chapter 9.9.1 for a tabular overview of all objects and their corresponding data types.

Default mapping

RxPDO1	0x1600/0	0x1600/1	0x1600/2	0x1600/3…8
	2	0x6040 Control word	0x6042 *vl* target velocity	0x00000000
RxPDO2	0x1601/0	0x1601/1…8		
	1	No mapping		
RxPDO3	0x1602/0	0x1602/1…8		
	1	No mapping		

9.9.2.10 0x1A00/n, 0x1A01/n, 0x1A02/n, TxPDO Mapping Parameter

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x1A00 0x1A01 0x1A02	0	Number of mapped objects	Unsigned8	rw	No	2
	1	1st mapped obj.	Unsigned32	rw	No	See text
	2	2nd mapped obj.	Unsigned32	rw	No	See text
	3	3rd mapped obj.	Unsigned32	rw	No	See text
	4	4th mapped obj.	Unsigned32	rw	No	See text
	5	5th mapped obj.	Unsigned32	rw	No	See text
	6	6th mapped obj.	Unsigned32	rw	No	See text
	7	7th mapped obj.	Unsigned32	rw	No	See text
	8	8th mapped obj.	Unsigned32	rw	No	See text

TxPDO Mapping parameters:

- 0x1A00/n TxPDO1
- 0x1A01/n TxPDO2
- 0x1A02/n TxPDO3
- 0x1A00/0 = 0 = no object mapped
- 0x1A00/0 = 1 … 8 = 1 … 8 mapped objects

Mapping entry:

MSB			LSB
Object index		Subindex	Length (no. of bits)
High byte	Low byte	si	ll

Examples:

Mapping of 0x6041/0 *Statusword* (unsigned16) to "1st mapped obj." in TxPDO1:

0x1A00/1 = 0x60410010

Mapping of 0x6064/0 *Position actual value* (integer32) to "2nd mapped obj." in TxPDO1:

0x1A00/2 = 0x60640020

Default mapping

TxPDO1	0x1A00/0	0x1A00/1	0x1A00/2	0x1A00/3…8
	2	0x6041 Status word	0x6044 *vl* velocity actual value	0x00000000
TxPDO2	0x1A01/0	0x1A01/1…8		
	1	No mapping		
TxPDO3	0x1A02/0	0x1A02/1…8		
	1	No mapping		

The number of objects that can be mapped depends on the length of the object. The maximum number of bytes that can be mapped is 8.

9.9.3 Manufacturer objects (0x2nnn) – Parameter access

For direct write/read access to inverter parameters via the SDO channel, a parameter is addressed via index and sub-index. Index and sub-index are used as follows for accessing the inverter parameters:

Index	=	Parameter number + 0x2000
Sub-index	=	required data set (0, 1 ... 4, 5, 6 ... 9)

Mapping of numerical data is always an integer or long data type. Values with decimal places will be written without decimal point (e.g. value 17.35 will be transmitted as 1735).

9.9.3.1 Handling of data sets/cyclic writing of the parameters

The parameter values are accessed based on the parameter number and the required dataset. There are parameters which only have one value (data set 0), as well as parameters which have four values (data sets 1...4). The latter are used for the data set change-over of a parameter.

If parameters with four data values are set via data set = 0, all four data sets are set to the same transmitted value. A read access with data set = 0 to such parameters is only successful if all four data sets are set to the same value. If this is not the case, an error will be signaled.

NOTICE

Risk of component damage!

If a maximum number of write cycles for the EEPROM is exceeded, it will be destroyed.

The values are entered automatically in the EEPROM of the controller. However, only a limited number of write cycles is permissible for the EEPROM (approx. 1 million cycles).

- If cyclic writing of data is required, use the RAM.

In the RAM, the data is not protected against loss of power. Once power supply is disrupted, the data must be written again.

This mechanism is activated by the target data set being increased by five in the specification of the data set.

Writing on virtual dataset in RAM

Parameter	EEPROM	RAM
Data set 0	0	5
Data set 1	1	6
Data set 2	2	7
Data set 3	3	8
Data set 4	4	9

9.9.3.2 Handling of index parameters/cyclic writing

NOTICE

Risk of component damage!

If a maximum number of write cycles for the EEPROM is exceeded, it will be destroyed.

The values are entered automatically in the EEPROM of the controller. However, only a limited number of write cycles is permissible for the EEPROM (approx. 1 million cycles).

- If cyclic writing of data is required, use the RAM.

Index parameters are used for various frequency inverter functions. Here, 16 or 32 indexes are used instead of the 4 data sets. For each function, the individual indexes are addressed separately via an index access parameter. Via the indexing parameter, you can select if the data is to be written to EEPROM or RAM.

Function	Parameter	Index range		Index access parameter
		Write EEPROM and Read	Write RAM	
Positioning	**1202** *Target Position / Distance* **1203** *Speed* **1204** *Acceleration* **1205** *Ramp Rise time* **1206** *Deceleration* **1207** *Ramp Fall time* **1208** *Motion mode* **1209** *Touch-Probe-Window* **1210** *Touch-Probe-Error:Next Motion Block* **1211** *No. of Repetitions* **1212** *Delay* **1213** *Delay: Next Motion Block* **1214** *Event 1* **1215** *Event1: Next Motion Block* **1216** *Event 2* **1217** *Event2: Next Motion Block* **1218** *Digital Signal 1* **1219** *Digital Signal 2* **1247** *Digital Signal 1* **1248** *Digital Signal 2* **1260** *Interrupt-Event 1* **1261** *Int.-Event 1: Eval.-Mode* **1262** *Int.-Event 1: Next Motion Block* **1263** *Interrupt- Event 2* **1264** *Int.-Event 2: Eval.-Mode* **1265** *Int.-Event 2: Next Motion Block*	0[1)]; 1…32	33[1)]; 34…65	**1200** Write **1201** Read
PLC Function (Function table)	**1343** *FT-instruction* **1344** *FT-input 1* **1345** *FT-input 2* **1346** *FT-input 3* **1347** *FT-input 4* **1348** *FT-Parameter 1* **1349** *FT-Parameter 2* **1350** *FT-target output 1* **1351** *FT- target output 2* **1352** *FT-commentary*	0[1)]; 1…32	33[1)]; 34…65	**1341** Write **1342** Read

Multiplexer	**1252** *Mux Input*	0[1)]; 1…16	17[1)]; 18…33	**1250** Write **1251** Read
CANopen Multiplexer	**1422** *CANopen Mux Input*	0[1)]; 1…16	17[1)]; 18…33	**1420** Write **1421** Read

1) If the index access parameter is set = 0, all indexes are accessed to write into EEPROM. Selection 17 and 33 respectively, write all indexes into RAM.

In the RAM, the data is not protected against loss of power. Once the power supply is interrupted, the data must be written again.

This mechanism is activated by the target data set being increased by five in the specification of the data set.

Example Writing an index parameter

Typically an index parameter is written during commissioning or regularly at simple positioning applications.

Writing Parameter *Target Position / Distance* **1202** (Type long), in Index 1 into RAM (→index **34** for write access) with Parameter value 30000.

Index = 1200 + 0x2000 = 0x24B0, Value (int) = 34 = 0x0022

Index = 1202 + 0x2000 = 0x24B2, Value (long) = 30000 = 0x0000 7530

If various parameters of an index are to be edited, it will be sufficient to set index access via parameter **1200** once at the beginning.

Example Reading an index parameter

In order to read an index parameter, you will have to set the index parameter to the relevant index first, then you can read the parameter.

Reading Parameter *Target Position / Distance* **1202** (Type long), in Index **1** with Parameter value 123000.

Index = 1201 + 0x2000 = 0x24B1, Value (int) = 1 = 0x0001

Index = 1202 + 0x2000 = 0x24B2, Value (long) = 123000 = 0x0001 E078

If various parameters in an index are to be edited, it will be sufficient to set index access via parameter **1201** once at the beginning.

9.9.4 Manufacturer objects (0x3000 … 0x5FF1)

In addition to the device profile objects the following manufacturer specific objects are implemented.

9.9.4.10x3001/0 Digital In actual value

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3001	0	Digital In actual value	Unsigned16	ro	Tx	

Object 0x3001 *Digital In actual value* shows – like parameter *Digital inputs* **250** – the current state of the digital inputs and multifunction input **1** (when parameter *Operation mode* **452** is set to "3 – Digital input").

9.9.4.20x3002/0 Digital Out actual value

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3002	0	Digital Out actual value	Unsigned16	ro	Tx	

Object 0x3001 *Digital Out actual value* shows – like parameter *Digital outputs* **254** – the current state of the digital outputs and multifunction output **1** (when parameter *Operation mode* **550** is set to "1 – Digital"). The selection of digital outputs depends on the expansion modules installed as an option.

9.9.4.30x3003/0 Digital Out set values

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3003	0	Digital Out set values	Unsigned8	rw	Rx	0

Via object 0x3003, five digital parameter sources are available requiring assignment of digital sources.

Object 0x3003			
Bit	Source number	Source name	Operation mode Digital output
0	810	Obj 0x3003 Digout 1	90/190
1	811	Obj 0x3003 Digout 2	91/191
2	812	Obj 0x3003 Digout 3	92/192
3	813	Obj 0x3003 Digout 4	93/193
4	814	Obj 0x3003 Digout 5	94/194

The values of object 0x3003 range from 0 to 31.

No.	Object	Min.	Max.
0x3003/0	Digital Out set values	0	31 (= 0x1F)

Digital outputs use these sources as operation modes 90 … 94 with source names Obj 0x3003 DigOut 1 … 5 and, inverted, as operation modes 190 … 194 with source names inv. Obj 0x3003 DigOut 1 … 5 (e.g. parameter *Op. Mode Digital In-/Output 1* **530**). These object bits can be mapped to the output as required.

Example:

Function	Parameter no.	Choice list (excerpt)	
Op. Mode Digital Output 3	532	0 -	OFF
		1 -	Ready or Standby Signal
		2 -	Run Signal
		…	
		43 -	External Fan
		90 -	Obj 0x3003 Digout 1
		91 -	Obj 0x3003 Digout 2
		92 -	Obj 0x3003 Digout 3
		93 -	Obj 0x3003 Digout 4
		94 -	Obj 0x3003 Digout 5
		…	
		143 -	inv. External Fan
		190 -	inv. Obj 0x3003 Digout 1
		191 -	inv. Obj 0x3003 Digout 2
		192 -	inv. Obj 0x3003 Digout 3
		193 -	inv. Obj 0x3003 Digout 4
		194 -	inv. Obj 0x3003 Digout 5
		…	

Sources 810…814 with source names Obj 0x3003 DigOut 1 … 5 can be selected directly via the selection list for parameters. This can be used, for example, for direct setting of Boolean inputs.

9.9.4.40x3004/0 Boolean Mux

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3004	0	Boolean Mux	Unsigned16	ro	Tx	

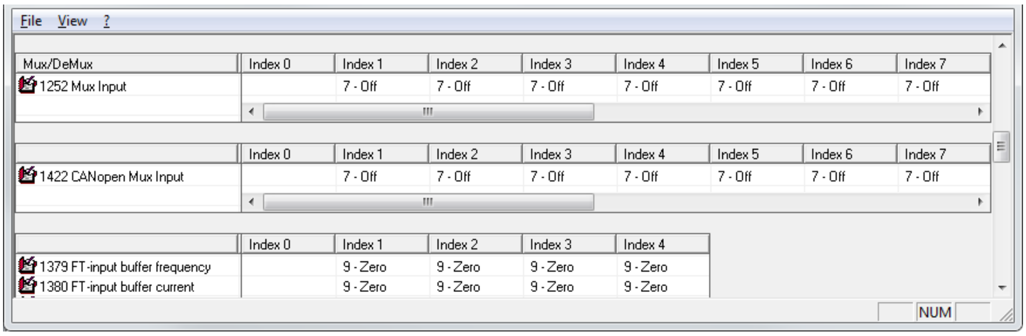
Via object 0x3004, up to 16 packed Boolean values can be read. Each bit in 16-bit object 0x3004 shows the actual value of the assigned Boolean source.

Bit numbers 0 … 15 correspond to index numbers 1 … 16!

The sources of the 16 bits can be selected from a choice list of index parameter *CANopen Mux Input* **1422**. Parameters **1420** and **1421** are write and read parameters which must be set before writing/reading of parameter **1422**.

By using VTable this process is easier and laid out more clearly.

For writing and reading index parameters refer to chapter 9.9.3.2 "Handling of index parameters/cyclic writing".



Mux/DeMux	Index 0	Index 1	Index 2	Index 3	Index 4	Index 5	Index 6	Index 7
1252 Mux Input		7 - Off	7 - Off	7 - Off	7 - Off	7 - Off	7 - Off	7 - Off

	Index 0	Index 1	Index 2	Index 3	Index 4	Index 5	Index 6	Index 7
1422 CANopen Mux Input		7 - Off	7 - Off	7 - Off	7 - Off	7 - Off	7 - Off	7 - Off

	Index 0	Index 1	Index 2	Index 3	Index 4
1379 FT-input buffer frequency		9 - Zero	9 - Zero	9 - Zero	9 - Zero
1380 FT-input buffer current		9 - Zero	9 - Zero	9 - Zero	9 - Zero

Default value is `7 – Off`.

9.9.4.50x3005/0 Boolean DeMux

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3005	0	Boolean DeMux	Unsigned16	rw	Rx	0

Via object 0x3005, up to 16 packed Boolean values can be written. These values are available as sources and can be selected via a choice list for parameters as objects `832…847` with source names `Obj 0x3005 Demux Out 1…16`.

Obj. 0x3005		
Bit no.	**Source No.**	**Source name**
0	832	Obj. 0x3005 Demux Out 1
1	833	Obj. 0x3005 Demux Out 2
2	834	Obj. 0x3005 Demux Out 3
3	835	Obj. 0x3005 Demux Out 4
4	836	Obj. 0x3005 Demux Out 5
5	837	Obj. 0x3005 Demux Out 6
6	838	Obj. 0x3005 Demux Out 7
7	839	Obj. 0x3005 Demux Out 8
8	840	Obj. 0x3005 Demux Out 9
9	841	Obj. 0x3005 Demux Out 10
10	842	Obj. 0x3005 Demux Out 11
11	843	Obj. 0x3005 Demux Out 12
12	844	Obj. 0x3005 Demux Out 13
13	845	Obj. 0x3005 Demux Out 14
14	846	Obj. 0x3005 Demux Out 15
15	847	Obj. 0x3005 Demux Out 16

9.9.4.60x3006/0 Percentage set value

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3006	0	Percentage set value	Unsigned16	rw	Rx	0

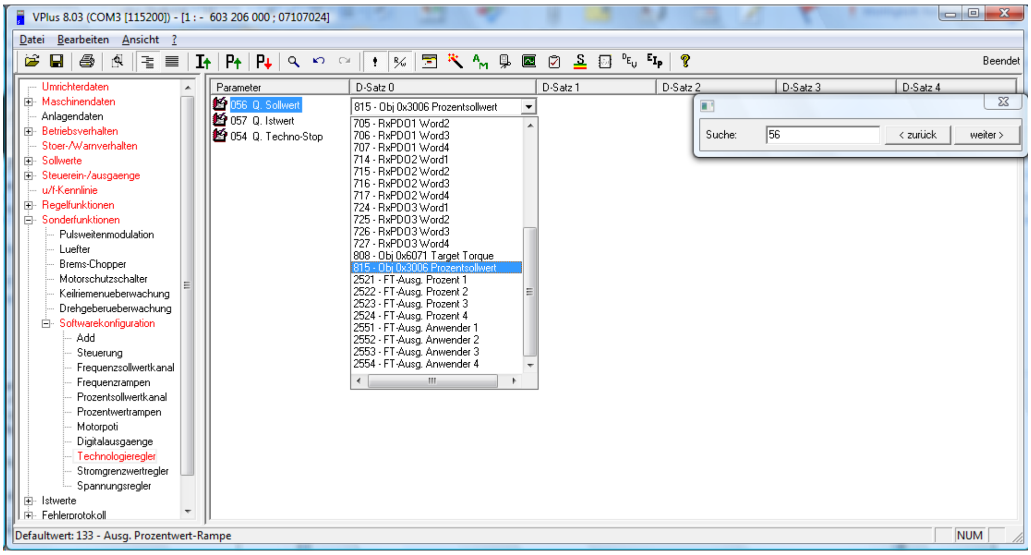
Via object 0x3006, you can write a percentage source – e.g. parameter *S. reference* **056**.

The value of object 0x3006 is available as a source and can be selected as `815` – `Obj 0x3006 Reference Percentage` via a choice list for parameters.

The values of object 0x3006 range from `-30000` to `30000` (equivalent to percentage range from -300.00 % to 300.00 %)

No.	Object	Min.	Max.
0x3006/0	Percentage set value	-30000 (= 0x8AD0)	30000 (= 0x7530)

Example: Technology controller parameter *S. Reference Value* **056**.



Parameter	D-Satz 0	D-Satz 1	D-Satz 2	D-Satz 3	D-Satz 4
056 Q. Sollwert	815 - Obj 0x3006 Prozentsollwert				
057 Q. Istwert					
054 Q. Techno-Stop					

705 - RxPDO1 Word2
706 - RxPDO1 Word3
707 - RxPDO1 Word4
714 - RxPDO2 Word1
715 - RxPDO2 Word2
716 - RxPDO2 Word3
717 - RxPDO2 Word4
724 - RxPDO3 Word1
725 - RxPDO3 Word2
726 - RxPDO3 Word3
727 - RxPDO3 Word4
808 - Obj 0x6071 Target Torque
815 - Obj 0x3006 Prozentsollwert
2521 - FT-Ausg. Prozent 1
2522 - FT-Ausg. Prozent 2
2523 - FT-Ausg. Prozent 3
2524 - FT-Ausg. Prozent 4
2551 - FT-Ausg. Anwender 1
2552 - FT-Ausg. Anwender 2
2553 - FT-Ausg. Anwender 3
2554 - FT-Ausg. Anwender 4

Suche: 56

Defaultwert: 133 - Ausg. Prozentwert-Rampe

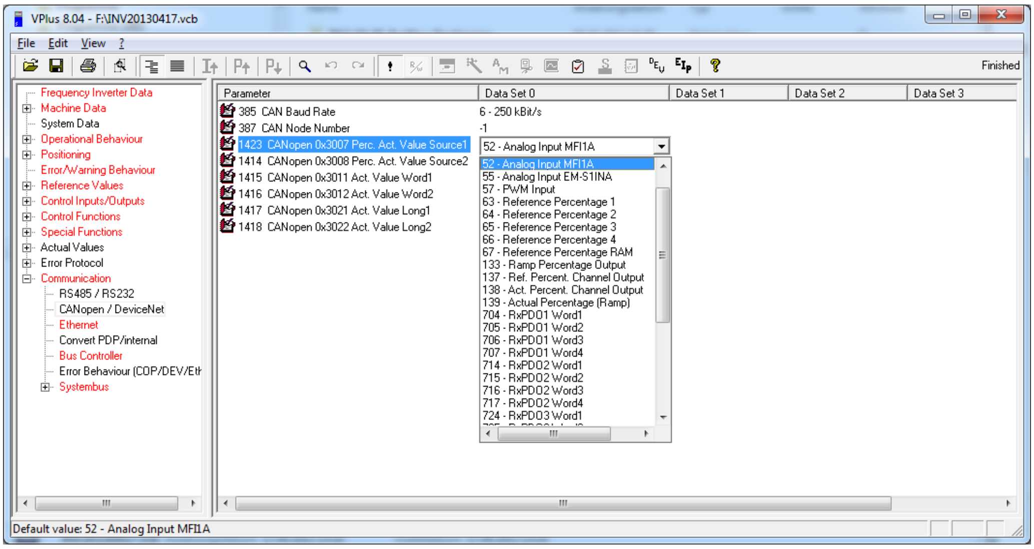
The percentage value is scaled as `percent * 100` (e.g. `5678` represents `56.78%`).

9.9.4.7 0x3007/0 Percentage Actual Value Source 1

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3007	0	Percentage Actual Value Source 1	Unsigned16	ro	Tx	

Object 0x3007 shows the actual value of the percentage source which can be selected via parameter *CANopen Percentage Actual Value Source 1* **1423**.

Default setting: `52 – Analog input MF1IA`.



Parameter	Data Set 0	Data Set 1	Data Set 2	Data Set 3
385 CAN Baud Rate	6 - 250 kBit/s			
387 CAN Node Number	-1			
1423 CANopen 0x3007 Perc. Act. Value Source1	52 - Analog Input MFI1A			
1414 CANopen 0x3008 Perc. Act. Value Source2				
1415 CANopen 0x3011 Act. Value Word1				
1416 CANopen 0x3012 Act. Value Word2				
1417 CANopen 0x3021 Act. Value Long1				
1418 CANopen 0x3022 Act. Value Long2				

52 - Analog Input MFI1A
55 - Analog Input EM-S1INA
57 - PWM Input
63 - Reference Percentage 1
64 - Reference Percentage 2
65 - Reference Percentage 3
66 - Reference Percentage 4
67 - Reference Percentage RAM
133 - Ramp Percentage Output
137 - Ref. Percent. Channel Output
138 - Act. Percent. Channel Output
139 - Actual Percentage (Ramp)
704 - RxPDO1 Word1
705 - RxPDO1 Word2
706 - RxPDO1 Word3
707 - RxPDO1 Word4
714 - RxPDO2 Word1
715 - RxPDO2 Word2
716 - RxPDO2 Word3
717 - RxPDO2 Word4
724 - RxPDO3 Word1

Default value: 52 - Analog Input MFI1A

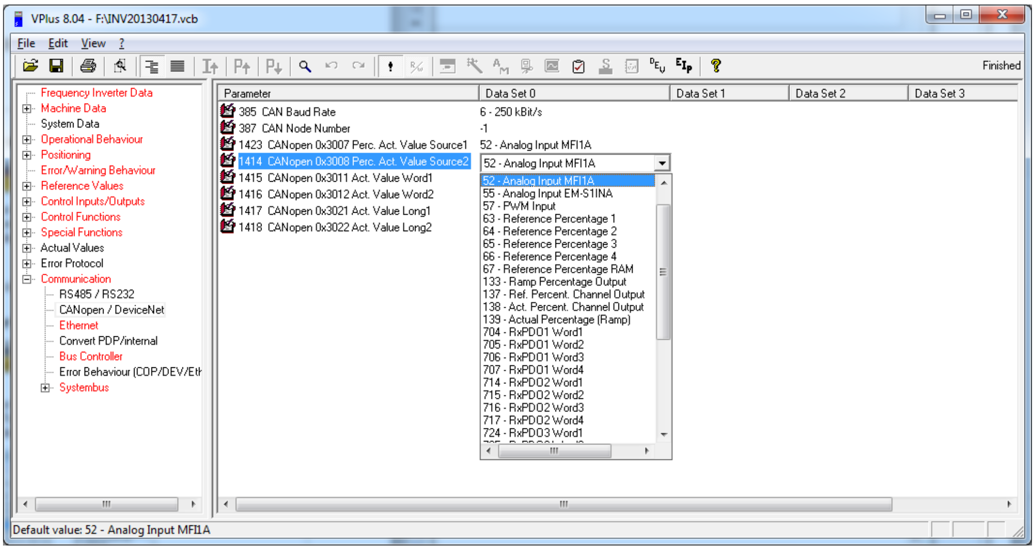
The percentage value is scaled as `percent * 100` (e.g. `5678` represents `56.78%`).

9.9.4.8 0x3008/0 Percentage Actual Value Source 2

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3008	0	Percentage Actual Value Source 2	Unsigned16	ro	Tx	

Object 0x3008 shows the actual value of the percentage source which can be selected via parameter *CANopen Percentage Actual Value Source 2* **1414**.

Default setting: `52 – Analog input MF1IA`.



Parameter	Data Set 0	Data Set 1	Data Set 2	Data Set 3
385 CAN Baud Rate	6 - 250 kBit/s			
387 CAN Node Number	-1			
1423 CANopen 0x3007 Perc. Act. Value Source1	52 - Analog Input MFI1A			
1414 CANopen 0x3008 Perc. Act. Value Source2	52 - Analog Input MFI1A			
1415 CANopen 0x3011 Act. Value Word1				
1416 CANopen 0x3012 Act. Value Word2				
1417 CANopen 0x3021 Act. Value Long1				
1418 CANopen 0x3022 Act. Value Long2				

52 - Analog Input MFI1A
55 - Analog Input EM-S1INA
57 - PWM Input
63 - Reference Percentage 1
64 - Reference Percentage 2
65 - Reference Percentage 3
66 - Reference Percentage 4
67 - Reference Percentage RAM
133 - Ramp Percentage Output
137 - Ref. Percent. Channel Output
138 - Act. Percent. Channel Output
139 - Actual Percentage (Ramp)
704 - RxPDO1 Word1
705 - RxPDO1 Word2
706 - RxPDO1 Word3
707 - RxPDO1 Word4
714 - RxPDO2 Word1
715 - RxPDO2 Word2
716 - RxPDO2 Word3
717 - RxPDO2 Word4
724 - RxPDO3 Word1

Default value: 52 - Analog Input MFI1A

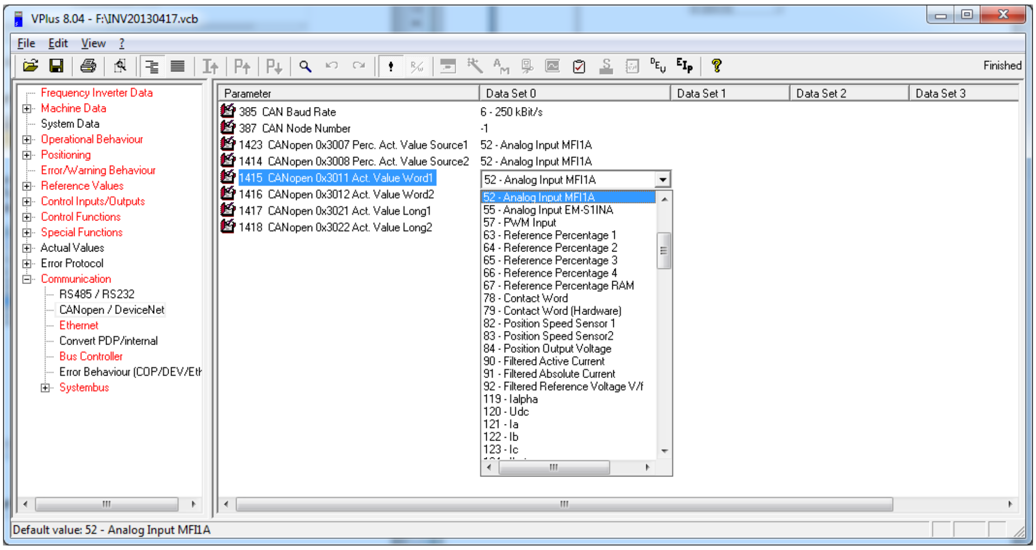
The percentage value is scaled as `percent * 100` (e.g. `5678` represents `56.78%`).

9.9.4.9 0x3011/0 Actual Value Word 1

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3011	0	Actual Value Word 1	Unsigned16	ro	Tx	

Object 0x3011 shows the actual value of the word source which can be selected via parameter *CANopen 0x3011 Act. Value Word 1* **1415.**

Default setting: `52 – Analog input MF1IA`.



Parameter	Data Set 0	Data Set 1	Data Set 2	Data Set 3
385 CAN Baud Rate	6 - 250 kBit/s			
387 CAN Node Number	-1			
1423 CANopen 0x3007 Perc. Act. Value Source1	52 - Analog Input MFI1A			
1414 CANopen 0x3008 Perc. Act. Value Source2	52 - Analog Input MFI1A			
1415 CANopen 0x3011 Act. Value Word1	52 - Analog Input MFI1A			
1416 CANopen 0x3012 Act. Value Word2				
1417 CANopen 0x3021 Act. Value Long1				
1418 CANopen 0x3022 Act. Value Long2				

52 - Analog Input MFI1A
55 - Analog Input EM-S1INA
57 - PWM Input
63 - Reference Percentage 1
64 - Reference Percentage 2
65 - Reference Percentage 3
66 - Reference Percentage 4
67 - Reference Percentage RAM
78 - Contact Word
79 - Contact Word (Hardware)
82 - Position Speed Sensor 1
83 - Position Speed Sensor2
84 - Position Output Voltage
90 - Filtered Active Current
91 - Filtered Absolute Current
92 - Filtered Reference Voltage V/f
119 - Ialpha
120 - Udc
121 - Ia
122 - Ib
123 - Ic

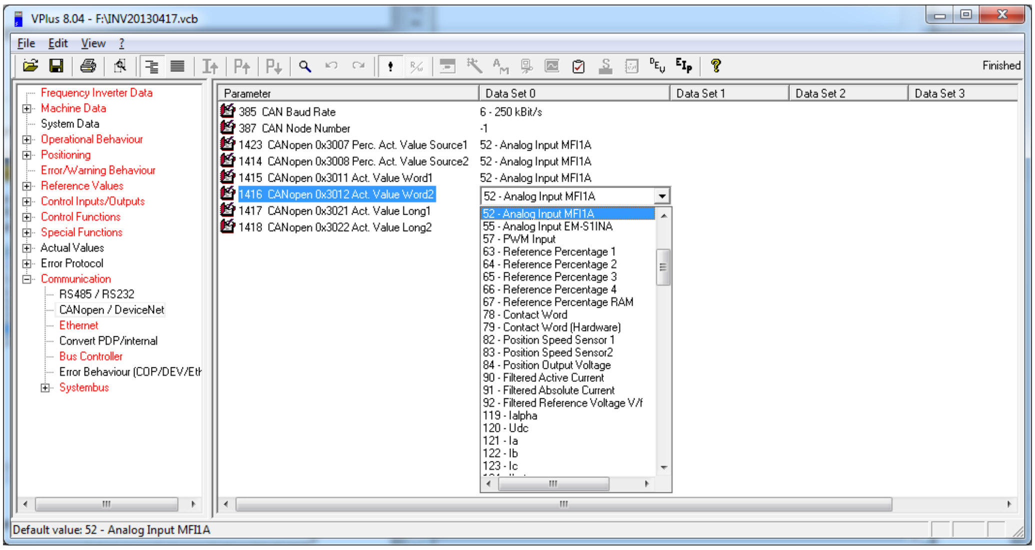
Default value: 52 - Analog Input MFI1A

9.9.4.10 0x3012/0 Actual Value Word 2

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3012	0	Actual Value Word 2	Unsigned16	ro	Tx	

Object 0x3012 shows the actual value of the word source which can be selected via parameter *CANopen 0x3012 Act. Value Word 2* **1416.**

Default setting: `52 – Analog input MF1IA.`



Parameter	Data Set 0	Data Set 1	Data Set 2	Data Set 3
385 CAN Baud Rate	6 - 250 kBit/s			
387 CAN Node Number	-1			
1423 CANopen 0x3007 Perc. Act. Value Source1	52 - Analog Input MFI1A			
1414 CANopen 0x3008 Perc. Act. Value Source2	52 - Analog Input MFI1A			
1415 CANopen 0x3011 Act. Value Word1	52 - Analog Input MFI1A			
1416 CANopen 0x3012 Act. Value Word2	52 - Analog Input MFI1A			
1417 CANopen 0x3021 Act. Value Long1				
1418 CANopen 0x3022 Act. Value Long2				

52 - Analog Input MFI1A
55 - Analog Input EM-S1INA
57 - PWM Input
63 - Reference Percentage 1
64 - Reference Percentage 2
65 - Reference Percentage 3
66 - Reference Percentage 4
67 - Reference Percentage RAM
78 - Contact Word
79 - Contact Word (Hardware)
82 - Position Speed Sensor 1
83 - Position Speed Sensor2
84 - Position Output Voltage
90 - Filtered Active Current
91 - Filtered Absolute Current
92 - Filtered Reference Voltage V/f
119 - Ialpha
120 - Udc
121 - Ia
122 - Ib
123 - Ic

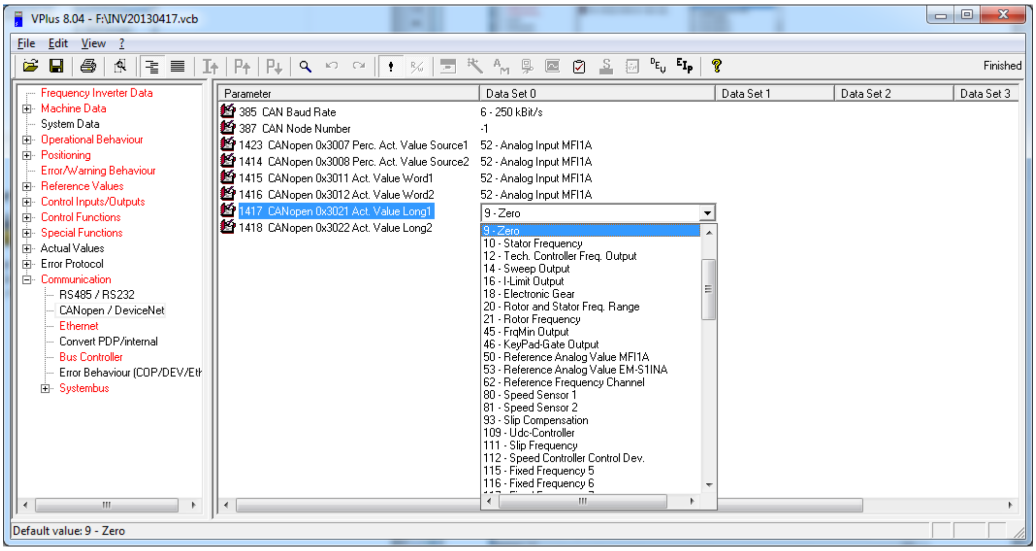
Default value: 52 - Analog Input MFI1A

9.9.4.11 0x3021/0 Actual Value Long 1

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3021	0	Actual Value Long 1	Unsigned32	ro	Tx	

Object 0x3021 shows the actual value of the long source which can be selected via parameter *CANopen 0x3021 Act. Value Long 1* **1417**.

Default setting: `9-zero`.



Parameter	Data Set 0	Data Set 1	Data Set 2	Data Set 3
385 CAN Baud Rate	6 - 250 kBit/s			
387 CAN Node Number	-1			
1423 CANopen 0x3007 Perc. Act. Value Source1	52 - Analog Input MFI1A			
1414 CANopen 0x3008 Perc. Act. Value Source2	52 - Analog Input MFI1A			
1415 CANopen 0x3011 Act. Value Word1	52 - Analog Input MFI1A			
1416 CANopen 0x3012 Act. Value Word2	52 - Analog Input MFI1A			
1417 CANopen 0x3021 Act. Value Long1	9 - Zero			
1418 CANopen 0x3022 Act. Value Long2				

9 - Zero
10 - Stator Frequency
12 - Tech. Controller Freq. Output
14 - Sweep Output
16 - I-Limit Output
18 - Electronic Gear
20 - Rotor and Stator Freq. Range
21 - Rotor Frequency
45 - FrqMin Output
46 - KeyPad-Gate Output
50 - Reference Analog Value MFI1A
53 - Reference Analog Value EM-S1INA
62 - Reference Frequency Channel
80 - Speed Sensor 1
81 - Speed Sensor 2
93 - Slip Compensation
109 - Udc-Controller
111 - Slip Frequency
112 - Speed Controller Control Dev.
115 - Fixed Frequency 5
116 - Fixed Frequency 6

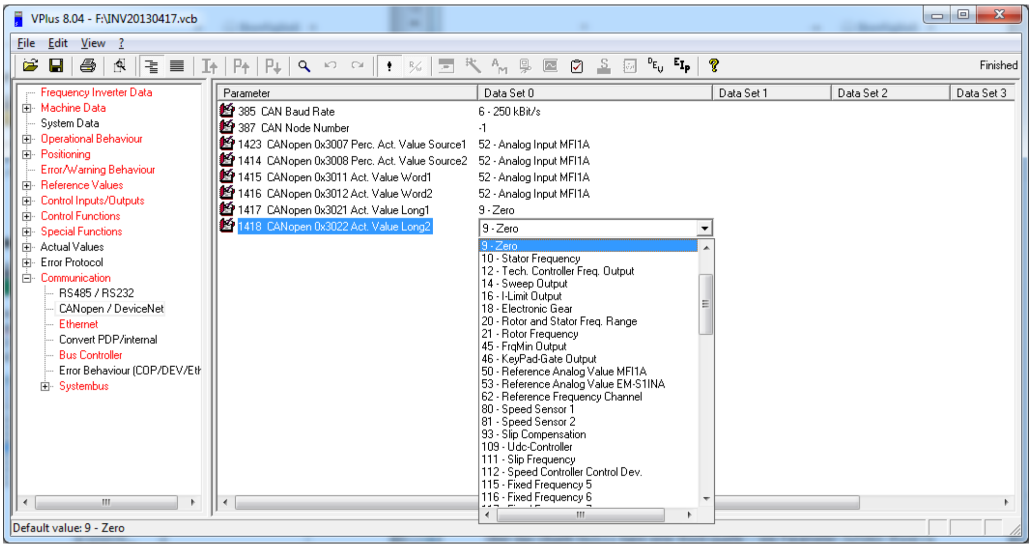
Default value: 9 - Zero

9.9.4.12 0x3022/0 Actual Value Long 2

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3022	0	Actual Value Long 2	Unsigned32	ro	Tx	

Object 0x3022 shows the actual value of the long source which can be selected via parameter *CANopen 0x3022 Act. Value Long 2* **1418**.

Default setting: `9-zero`.



Parameter	Data Set 0	Data Set 1	Data Set 2	Data Set 3
385 CAN Baud Rate	6 - 250 kBit/s			
387 CAN Node Number	-1			
1423 CANopen 0x3007 Perc. Act. Value Source1	52 - Analog Input MFI1A			
1414 CANopen 0x3008 Perc. Act. Value Source2	52 - Analog Input MFI1A			
1415 CANopen 0x3011 Act. Value Word1	52 - Analog Input MFI1A			
1416 CANopen 0x3012 Act. Value Word2	52 - Analog Input MFI1A			
1417 CANopen 0x3021 Act. Value Long1	9 - Zero			
1418 CANopen 0x3022 Act. Value Long2	9 - Zero			

9 - Zero
10 - Stator Frequency
12 - Tech. Controller Freq. Output
14 - Sweep Output
16 - I-Limit Output
18 - Electronic Gear
20 - Rotor and Stator Freq. Range
21 - Rotor Frequency
45 - FrqMin Output
46 - KeyPad-Gate Output
50 - Reference Analog Value MFI1A
53 - Reference Analog Value EM-S1INA
62 - Reference Frequency Channel
80 - Speed Sensor 1
81 - Speed Sensor 2
93 - Slip Compensation
109 - Udc-Controller
111 - Slip Frequency
112 - Speed Controller Control Dev.
115 - Fixed Frequency 5
116 - Fixed Frequency 6

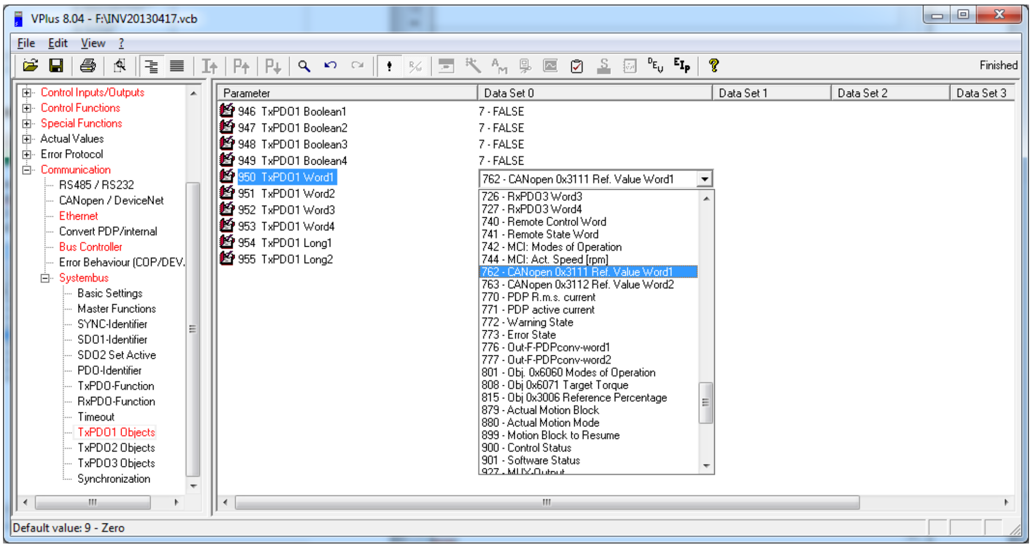
Default value: 9 - Zero

9.9.4.13 0x3111/0 Ref. Value Word 1

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3111	0	Ref. Value Word 1	Unsigned16	rw	Rx	0

Via object 0x3111, you can write a word source – e.g. parameter *TxPDO1 Word 1* **950** of Systembus.

The value of object 0x3111 is available as a source and can be selected as `762 – CANopen 0x3111 Ref. Value` via a choice list for parameters.



Parameter	Data Set 0	Data Set 1	Data Set 2	Data Set 3
946 TxPDO1 Boolean1	7 - FALSE			
947 TxPDO1 Boolean2	7 - FALSE			
948 TxPDO1 Boolean3	7 - FALSE			
949 TxPDO1 Boolean4	7 - FALSE			
950 TxPDO1 Word1	762 - CANopen 0x3111 Ref. Value Word1			
951 TxPDO1 Word2				
952 TxPDO1 Word3				
953 TxPDO1 Word4				
954 TxPDO1 Long1				
955 TxPDO1 Long2				

726 - RxPDO3 Word3
727 - RxPDO3 Word4
740 - Remote Control Word
741 - Remote State Word
742 - MCI: Modes of Operation
744 - MCI: Act. Speed [rpm]
762 - CANopen 0x3111 Ref. Value Word1
763 - CANopen 0x3112 Ref. Value Word2
770 - PDP R.m.s. current
771 - PDP active current
772 - Warning State
773 - Error State
776 - Out-F-PDPconv-word1
777 - Out-F-PDPconv-word2
801 - Obj. 0x6060 Modes of Operation
808 - Obj 0x6071 Target Torque
815 - Obj 0x3006 Reference Percentage
879 - Actual Motion Block
880 - Actual Motion Mode
899 - Motion Block to Resume
900 - Control Status
901 - Software Status
927 - MUX-Output

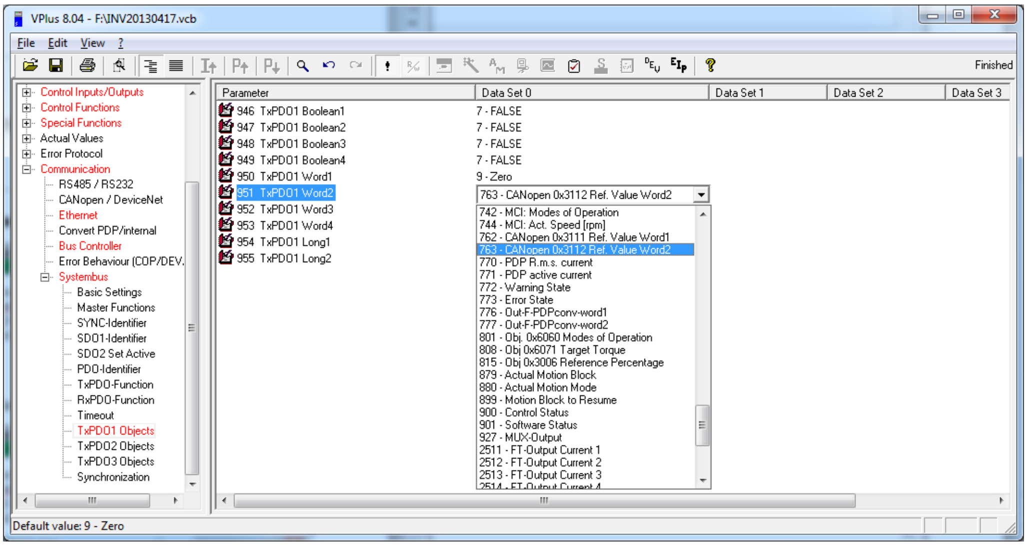
Default value: 9 - Zero

9.9.4.14 0x3112/0 Ref. Value Word 2

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3112	0	Ref. Value Word 2	Unsigned16	rw	Rx	0

Via object 0x3112, you can write a word source – e.g. parameter *TxPDO1 Word 2* **951** of Systembus.

The value of object 0x3112 is available as a source and can be selected as `763 – CANopen 0x3112 Ref. Value` via a choice list for parameters.



Parameter	Data Set 0	Data Set 1	Data Set 2	Data Set 3
946 TxPDO1 Boolean1	7 - FALSE			
947 TxPDO1 Boolean2	7 - FALSE			
948 TxPDO1 Boolean3	7 - FALSE			
949 TxPDO1 Boolean4	7 - FALSE			
950 TxPDO1 Word1	9 - Zero			
951 TxPDO1 Word2	763 - CANopen 0x3112 Ref. Value Word2			
952 TxPDO1 Word3				
953 TxPDO1 Word4				
954 TxPDO1 Long1				
955 TxPDO1 Long2				

742 - MCI: Modes of Operation
744 - MCI: Act. Speed [rpm]
762 - CANopen 0x3111 Ref. Value Word1
763 - CANopen 0x3112 Ref. Value Word2
770 - PDP R.m.s. current
771 - PDP active current
772 - Warning State
773 - Error State
776 - Out-F-PDPconv-word1
777 - Out-F-PDPconv-word2
801 - Obj. 0x6060 Modes of Operation
808 - Obj 0x6071 Target Torque
815 - Obj 0x3006 Reference Percentage
879 - Actual Motion Block
880 - Actual Motion Mode
899 - Motion Block to Resume
900 - Control Status
901 - Software Status
927 - MUX-Output
2511 - FT-Output Current 1
2512 - FT-Output Current 2
2513 - FT-Output Current 3
2514 - FT-Output Current 4

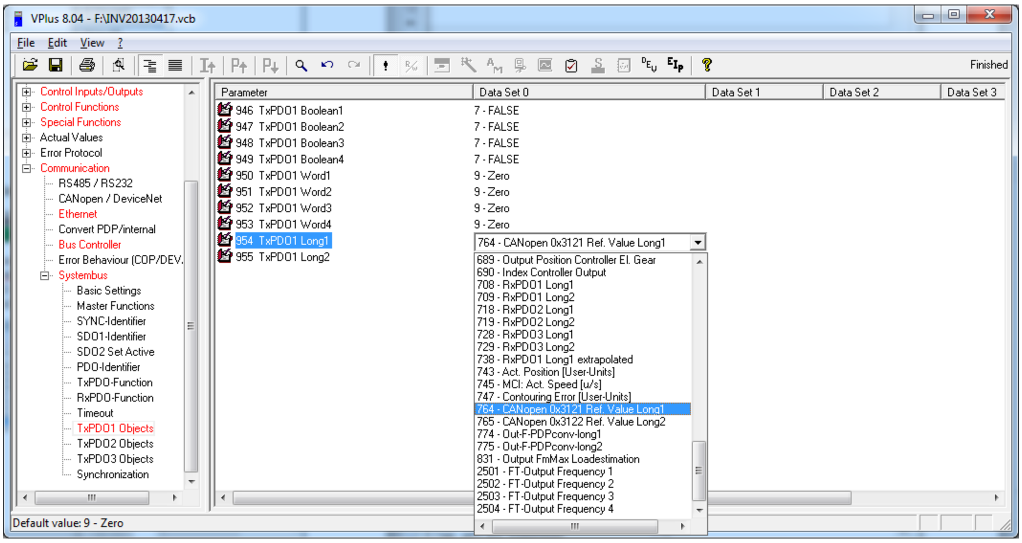
Default value: 9 - Zero

9.9.4.15 0x3121/0 Ref. Value Long 1

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3121	0	Ref. Value Long 1	Unsigned32	rw	Rx	0

Via object 0x3121, you can write a long source – e.g. parameter *TxPDO1 Long 1* **954** of Systembus.

The value of object 0x3121 is available as a source and can be selected as `764 – CANopen 0x3121 Ref. Value` via a choice list for parameters.



Parameter	Data Set 0	Data Set 1	Data Set 2	Data Set 3
946 TxPDO1 Boolean1	7 - FALSE			
947 TxPDO1 Boolean2	7 - FALSE			
948 TxPDO1 Boolean3	7 - FALSE			
949 TxPDO1 Boolean4	7 - FALSE			
950 TxPDO1 Word1	9 - Zero			
951 TxPDO1 Word2	9 - Zero			
952 TxPDO1 Word3	9 - Zero			
953 TxPDO1 Word4	9 - Zero			
954 TxPDO1 Long1	764 - CANopen 0x3121 Ref. Value Long1			
955 TxPDO1 Long2				

689 - Output Position Controller El. Gear
690 - Index Controller Output
708 - RxPDO1 Long1
709 - RxPDO1 Long2
718 - RxPDO2 Long1
719 - RxPDO2 Long2
728 - RxPDO3 Long1
729 - RxPDO3 Long2
738 - RxPDO1 Long1 extrapolated
743 - Act. Position [User-Units]
745 - MCI: Act. Speed [u/s]
747 - Contouring Error [User-Units]
764 - CANopen 0x3121 Ref. Value Long1
765 - CANopen 0x3122 Ref. Value Long2
774 - Out-F-PDPconv-long1
775 - Out-F-PDPconv-long2
831 - Output FmMax Loadestimation
2501 - FT-Output Frequency 1
2502 - FT-Output Frequency 2
2503 - FT-Output Frequency 3
2504 - FT-Output Frequency 4

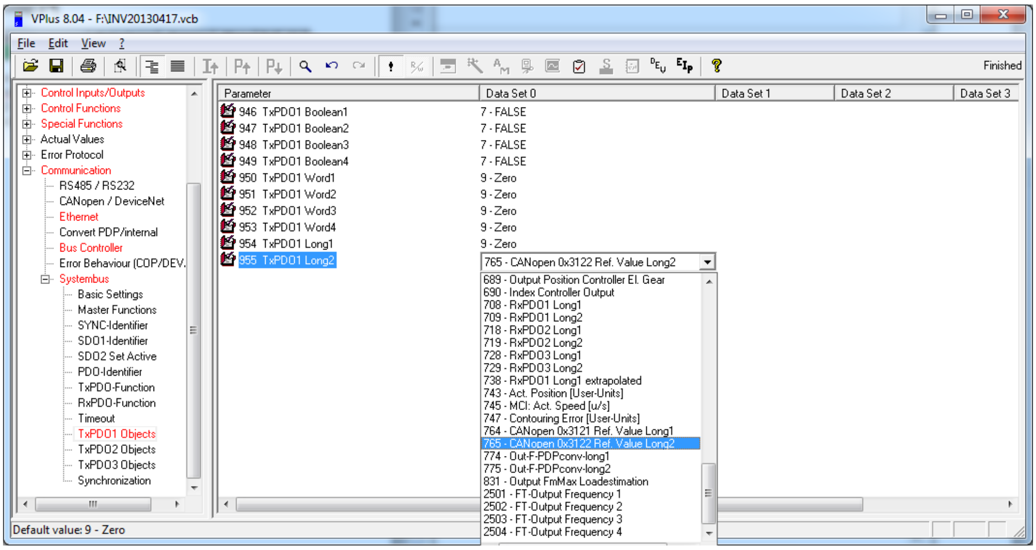
Default value: 9 - Zero

9.9.4.16 0x3122/0 Ref. Value Long 2

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x3122	0	Ref. Value Long 2	Unsigned32	rw	Rx	0

Via object 0x3122, you can write a long source – e.g. parameter *TxPDO1 Long 2* **955** of Systembus.

The value of object 0x3122 is available as a source and can be selected as `765 – CANopen 0x3122 Ref. Value` via a choice list for parameters.



- Control Inputs/Outputs
- Control Functions
- Special Functions
- Actual Values
- Error Protocol
- Communication
  - RS485 / RS232
  - CANopen / DeviceNet
  - Ethernet
  - Convert PDP/internal
  - Bus Controller
  - Error Behaviour (COP/DEV.
  - Systembus
    - Basic Settings
    - Master Functions
    - SYNC-Identifier
    - SDO1-Identifier
    - SDO2 Set Active
    - PDO-Identifier
    - TxPDO-Function
    - RxPDO-Function
    - Timeout
    - TxPDO1 Objects
    - TxPDO2 Objects
    - TxPDO3 Objects
    - Synchronization

Parameter	Data Set 0	Data Set 1	Data Set 2	Data Set 3
946 TxPDO1 Boolean1	7 - FALSE			
947 TxPDO1 Boolean2	7 - FALSE			
948 TxPDO1 Boolean3	7 - FALSE			
949 TxPDO1 Boolean4	7 - FALSE			
950 TxPDO1 Word1	9 - Zero			
951 TxPDO1 Word2	9 - Zero			
952 TxPDO1 Word3	9 - Zero			
953 TxPDO1 Word4	9 - Zero			
954 TxPDO1 Long1	9 - Zero			
955 TxPDO1 Long2	765 - CANopen 0x3122 Ref. Value Long2			

689 - Output Position Controller El. Gear
690 - Index Controller Output
708 - RxPDO1 Long1
709 - RxPDO1 Long2
718 - RxPDO2 Long1
719 - RxPDO2 Long2
728 - RxPDO3 Long1
729 - RxPDO3 Long2
738 - RxPDO1 Long1 extrapolated
743 - Act. Position [User-Units]
745 - MCI: Act. Speed [u/s]
747 - Contouring Error [User-Units]
764 - CANopen 0x3121 Ref. Value Long1
765 - CANopen 0x3122 Ref. Value Long2
774 - Out-F-PDPconv-long1
775 - Out-F-PDPconv-long2
831 - Output FmMax Loadestimation
2501 - FT-Output Frequency 1
2502 - FT-Output Frequency 2
2503 - FT-Output Frequency 3
2504 - FT-Output Frequency 4

Default value: 9 - Zero

9.9.4.17 0x5F10/n Gear factor

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x5F10	0	Highest sub-index supported	Unsigned8	ro	No	3
	1	Numerator	Integer16	rw	Rx	1
	2	Denominator	Unsigned16	rw	Rx	1
	3	Resync on change	Integer16	rw	No	1

Object works in:	Object does not work in:
– Motion Control: ○ **Electronic Gear: Slave** ○ **Table Travel Record mode (Electronic Gear operation)**	– Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Profile Velocity mode** ○ **Homing mode** ○ **Move away from Limit Switch** ○ **Non motion Control** (conf. ≠ x40)

Object 0x5F10 *active motion block* is available in *Electronic Gear: Slave* mode in motion control configurations (**P.30** = x40). The *Electronic Gear: Slave* mode is activated by object 0x6060 *modes of operation* set to **-3**.

With the Gear factors (numerator and denominator) a multiplier for the master speed can be set up. The Slave speed results in:

$$v_{Slave} = v_{Master} \times \frac{Numerator\ \mathtt{0x5F10/1}}{Denominator\ \mathtt{0x5F10/2}}$$

Limitation of acceleration when the gear factor is changed is effected via Object 0x5F10/3 *Gear Factor: Resync on change*. The slave is resynchronized with the master when the gear factor has changed. This function avoids sudden speed changes.

Alternatively the parameters **1123**, **1124** and **1142** can be used instead of the Objects.

Usage of the Objects will write the parameters in RAM (data set 5).

0x5F10/3 *Gear Factor: Resync on change*		Function
0 -	Off	Resynchronization is switched off.
1 -	On	The slave is resynchronized with the master frequency when the gear factor has changed. The drive adjusts to the new frequency. The acceleration ramps set in Object 0x6083 *Profile Acceleration* is considered.

Object	Parameter	
0x5F10/1 Gear factor Numerator	**1123**	*Gear Factor Numerator*
0x5F10/2 Gear factor Denominator	**1124**	*Gear Factor Denominator*
0x5F10/3 Gear factor Resync on change	**1142**	*Resync. on Change of Gear-Factor*

9.9.4.18 0x5F11/n…0x5F14/n Phasing 1…4

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x5F11	0	Highest sub-index supported	Unsigned8	ro	No	3
	1	Offset	Integer32	rw	No	0x1 0000
	2	Speed	Unsigned32	rw	No	0x5 0000
	3	Acceleration	Unsigned32	rw	No	0x5 0000

Object works in:	Object does not work in:
– Motion Control: ○ **Electronic Gear: Slave** ○ **Table Travel Record mode (Electronic Gear operation)**	– Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Profile Velocity mode** ○ **Homing mode** ○ **Table Travel Record mode** ○ **Move away from Limit Switch** – **Non motion Control** (conf. ≠ `x40`)

Objects 0x5F11 *Phasing 1*, 0x5F12 *Phasing 2,* 0x5F13 *Phasing 3* and 0x5F14 *Phasing 4 active motion block* is available in *Electronic Gear: Slave* mode in motion control configurations (**P.30** = `x40`). The *table travel record* mode is activated by object 0x6060 *modes of operation* set to **-3**.

For better readability in the following section Object **0x5F11** is used. For Objects **0x5F12**, **0x5F13** und **0x5F14** the descriptions apply correspondingly.

With the phasing function, the slave position is offset from the received position of the master by the value entered in 0x5F11/1 *Phasing 1: Offset*.

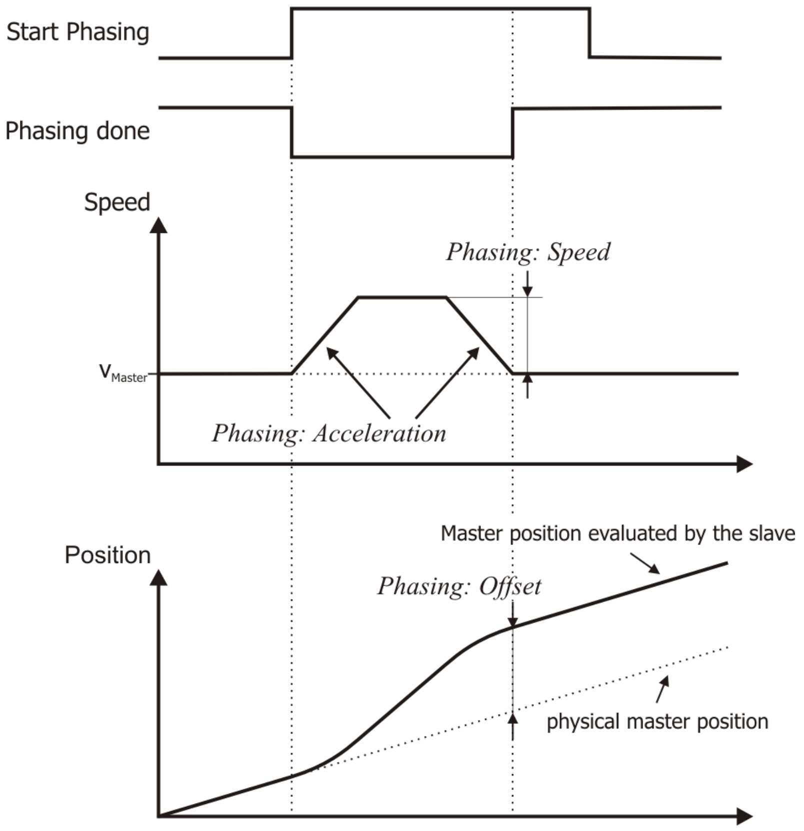
The function is started via Bit 9 of the Control Word. After start, 0x5F11/2 *Phasing 1: Speed* and 0x5F11/3 *Phasing 1: Acceleration* are used until the slave position is offset from the master position by *Phasing 1: Offset*.

During Phasing the Status word bit 8 "*Phasing Done*" is set to "`Low`". As soon as the Phasing is finished or cancelled, the Bit is set to "`High`". After first switch-on (or after a device reset) the "Phasing Done" bit is also "`Low`".

The values of Objects 0x5F11/n…0x5F14/n are limited as follows:

Object		Setting	
No.	**Object**	**Min.**	**Max.**
0x5F11/1 0x5F12/1 0x5F13/1 0x5F14/1	Phasing: Offset	-2147483647 (= 0x8000 0001)	2147483647 (= 0x7FFF FFFF)

0x5F11/2 0x5F12/2 0x5F13/2 0x5F14/2	Phasing: Speed	1	2147483647 (= 0x7FFF FFFF)
0x5F11/3 0x5F12/3 0x5F13/3 0x5F14/3	Phasing: Acceleration	1	2147483647 (= 0x7FFF FFFF)



Via Objects 0x5F11, 0x5F12, 0x5F13 and 0x5F14 four different Phasing profiles can be created. The Phasing Profile is selected via Control word bits 12 and 13.

Phasing select		**Phasing Profile**
Bit 13	Bit 12	
0	0	1 (0x5F11)
0	1	2 (0x5F12)
1	0	3 (0x5F13)
1	1	4 (0x5F14)

Alternatively the parameters **1125**, **1126** and **1127** can be used instead of the Objects. The 4 data sets of the parameters correspond to the 4 Objects. Usage of the Objects will write the parameters in RAM (data set 6…9).

Object		**Parameter**	
0x5F11/1 0x5F12/1 0x5F13/1 0x5F14/1	Phasing 1: Offset Phasing 2: Offset Phasing 3: Offset Phasing 4: Offset	**1125.1** **1125.2** **1125.3** **1125.4**	*Phasing: Offset*
0x5F11/2 0x5F12/2 0x5F13/2 0x5F14/2	Phasing 1: Speed Phasing 2: Speed Phasing 3: Speed Phasing 4: Speed	**1126.1** **1126.2** **1126.3** **1126.4**	*Phasing: Speed*

Object		Parameter	
0x5F11/3 0x5F12/3 0x5F13/3 0x5F14/3	Phasing 1: Acceleration Phasing 2: Acceleration Phasing 3: Acceleration Phasing 4: Acceleration	**1127.1** **1127.2** **1127.3** **1127.4**	*Phasing: Acceleration*

9.9.4.19 0x5F15/0 In Gear Threshold

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x5F15	0	In Gear Threshold	Unsigned32	rw	No	0

Object works in:	Object does not work in:
– Motion Control: ○ **Electronic Gear: Slave** ○ **Table Travel Record mode**	– Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Profile Velocity mode** ○ **Homing mode** ○ **Move away from Limit Switch** – **Non motion Control** (conf. ≠ x40)

The Status Word Bit 10 "In Gear" is set if the relative deviation between master- and slave-position is lower than the value of 0x5F15/0 *In Gear Threshold* for at least 0x5F16/0 *In Gear Time* .

When 0x5F15/0 In Gear Threshold is set to zero, the signal "in gear" is set as soon as the drive reaches the master speed.

The signals "In Gear" are reset in the following occurrences:

- The relative deviation between master and slave position exceeds the value of 0x5F15/0 In Gear Threshold.
- The speed of the master drive exceeds the value of *Maximum Speed* *.

*) *Maximum speed* refers to either 0x6046/2 *vl velocity max amount* or *Maximum frequency* **419**. It is set either via 0x6046/2 *vl velocity max amount* [rpm] or *Maximum frequency* **419** [Hz]. *Maximum frequency* **419** is usually set up during motor commissioning.

The value range of des Object 0x5F15/0 is limited as follows:

Object		Setting	
No.	**Object**	**Min.**	**Max.**
0x5F15/0	In Gear Threshold	0	2147483647 (= 0x7FFF FFFF)

Alternatively parameter *"In-Gear"-Threshold* **1168** can be used instead of the Object 0x5F15/0 *In Gear Threshold*.

Object		Parameter	
0x5F15/0	In Gear Threshold	**1168**	*"In-Gear"-Threshold*

9.9.4.20 0x5F16/0 In Gear Time

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x5F16	0	In Gear Time	Unsigned16	rw	No	10

Object works in: – Motion Control: ○ **Electronic Gear: Slave** ○ **Table Travel Record mode**	Object does not work in: – Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Profile Velocity mode** ○ **Homing mode** ○ **Move away from Limit Switch** – **Non motion Control** (conf. ≠ x40)

The Status Word Bit 10 "In Gear" is set if the relative deviation between master- and slave-position is lower than the value of 0x5F15/0 *In Gear Threshold* for at least 0x5F16/0 *In Gear Time* .

If parameter 0x5F15/0 *In Gear Threshold* is set to the value zero the signal "In Gear" is set when the drive attains the master speed.

The signals "In Gear" are reset in the following occurrences:

- The relative deviation between master- and slave-position exceeds the value of 0x5F15/0 In Gear Threshold.
- The speed of the master drive exceeds the value of Maximum Speed *.

*) *Maximum speed* refers to either 0x6046/2 *vl velocity max amount* or *Maximum frequency* **419**. It is set either via 0x6046/2 *vl velocity max amount* [rpm] or *Maximum frequency* **419** [Hz]. *Maximum frequency* **419** is usually set up during motor commissioning.

The Position Controller (0x5F17 *Position Controller*) can cause a higher overall speed than *Maximum speed*. However the Position Controller does not affect the Signal "In Gear".

The value range of des Object 0x5F16/0 is limited as follows:

Object		Setting	
No.	**Object**	**Min.**	**Max.**
	In Gear Time [ms]	1	65535 (= 0xFFFF)

Alternatively parameter *"In-Gear"-Time* **1169** can be used instead of the Object 0x5F16/0 *In Gear Time*.

Object		Parameter	
0x5F16/0	In Gear Time	**1169**	*In Gear-Time*

9.9.4.21 0x5F17/n Position Controller

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x5F17	0	Highest sub-index supported	Unsigned8	ro	No	2
	1	Time Constant	Integer32	rw	No	10,00 ms
	2	Limitation	Unsigned32	rw	No	327680

Object works in: – Motion Control: ○ **All modes**	Object does not work in: – **Non motion Control** (conf. ≠ x40)

The position controller evaluates the positioning operation (target/actual position) and tries to control the drive such that it comes as close as possible to the specifications. For this purpose, an additional frequency is calculated for compensation of position deviations. By setting the corresponding parameter, this frequency can be limited. The parameter settings of the position controller determine how quick and to what extent position deviations are to be compensated.

Via the *Position Controller:Time Constant* , you can define the maximum time in which the position deviation is to be compensated.

Via parameter *Position Controller:Limitation*, you can define to which value the speed is limited for compensation of the position deviation.

NOTICE

Risk of component damage!

The Output of the Position Controller is not limited by 0x6046/2 *vl velocity max amount* (or *Maximum frequency* **419**). The *Maximum speed** limits the value of the Motion Profile generation. Caused by the addition of the Profile generator reference speed and the output of the Position Controller higher frequencies than *Maximum speed** can occur.

- Observe all relevant settings carefully.

*Maximum speed** and *Limitation* **1118** must be set for fitting values during the commissioning.

Chapter **18.4** contains conversion formulas between Hz, rpm and u/s.

Bonfiglioli Vectron recommends:

- Set *Maximum speed** to 90 % of the mechanical rated speed and the *Limitation* **1118** of the Position Controller to the value corresponding to 10 % of the Maximum frequency.

*) *Maximum speed* refers to either 0x6046/2 *vl velocity max amount* or *Maximum frequency* **419**. It is set either via 0x6046/2 *vl velocity max amount* [rpm] or *Maximum frequency* **419** [Hz]. *Maximum frequency* **419** is usually set up during motor commissioning.

The values of Objects 0x5F17/n are limited as follows:

Object		Setting	
No.	**Object**	**Min.**	**Max.**
0x5F17/1	Position Controller: Time Constant	1,00 ms	300,00 ms
0x5F17/2	Position Controller: Limitation	0	2147483647 (= 0x7FFF FFFF)

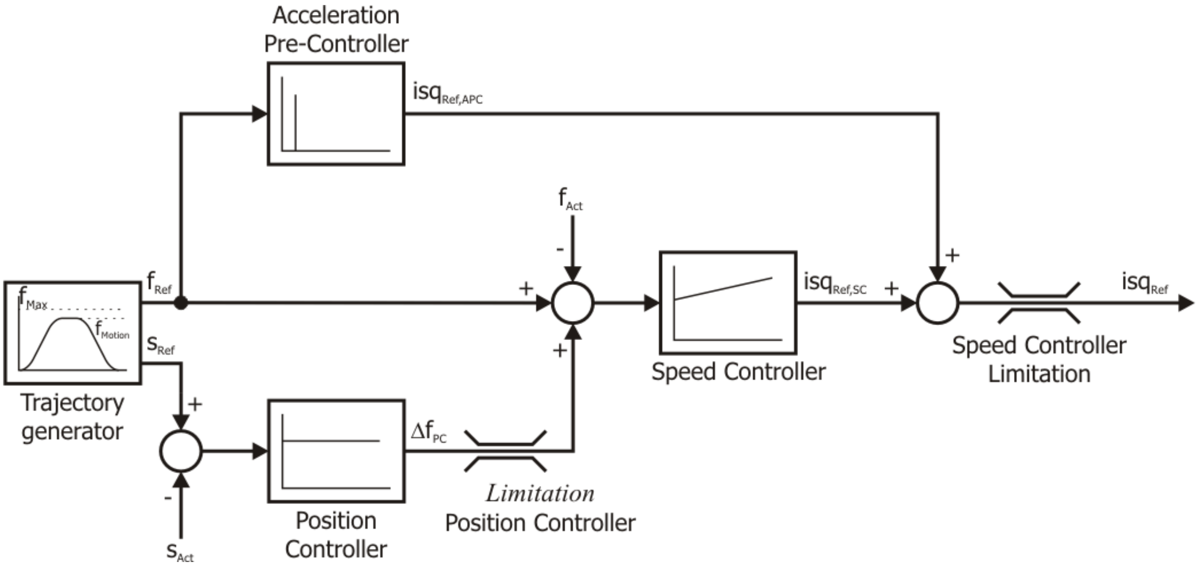
Alternatively the parameters **1104** and **1118** can be used instead of the Objects.

Object		Parameter	
0x5F17/1	Position Controller: Time Constant	**1104**	*Time Constant*
0x5F17/2	Position Controller: Limitation	**1118**	*Limitation*

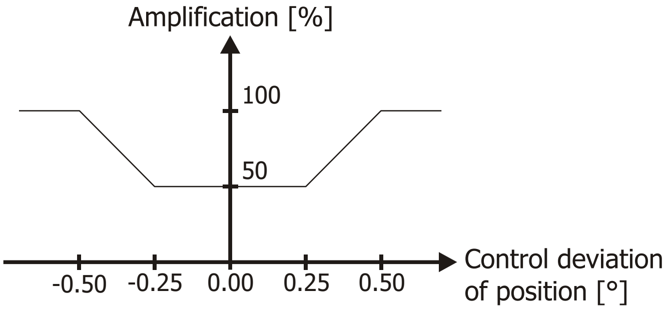
Example:

Position deviates by 1 motor shaft revolution, time constant is set to 1 ms. The position controller will increase the motor frequency by 1000 Hz in order to compensate the position deviation. Parameter *Limitation* **1118** must be set accordingly.

Controller block diagram



In order to avoid oscillations of the drive while it is at standstill, amplification is reduced to 50 % of the parameterized value for small position deviations



The following behavior may indicate that the controller parameters are not configured properly:

- drive is very loud
- drive vibrates
- frequent contouring errors
- inexact control

For the setting options of other control parameters, e.g. speed controller and acceleration pilot control, refer to the operating instructions of the frequency inverter.

Optimize the settings in actual operating conditions, as control parameters for speed controller and acceleration pilot control depend on actual load. Optimize with different load types to obtain a good control behavior in all situations.

9.9.4.22 0x5F18/0 M/S Synchronization Offset

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x5F18	0	M/S Synchronization Offset	Integer32	rw	No	0

Object works in:	Object does not work in:
– Motion Control: ○ **Table Travel record mode**	– Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Profile Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** ○ **Non motion Control** (conf. ≠ x40)

The M/S Synchronization Offset can be used in the electronic gear functionality to align the Slave drive to the absolute position of the master drive.

Comply with chapter 12.4.7.2 "Master/Slave Position Correction".

When using this functionality master drive and slave drive have to use the same mechanical characteristics (i.e. gear transmission ratios) and use the same reference system.

The values of Object 0x5F18/0 are limited as follows:

Object		Setting	
No.	**Object**	**Min.**	**Max.**
0x5F18/0	M/S Synchronization Offset	-2147483647 (= 0x8000 0001)	2147483647 (= 0x7FFF FFFF)

Alternatively parameter **1284** can be used instead of the Objects.

Object		Parameter	
0x5F18/0	M/S Synchronization Offset	**1284**	*M/S Synchronization Offset*

9.9.4.23 0x5FF0/0 Active motion block

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x5FF0	0	Active motion block	Unsigned8	ro	Tx	

Object works in:	Object does not work in:
– Motion Control: ○ **Table Travel record mode**	– Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Profile Velocity mode** ○ **Homing mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave** – **Non motion Control** (conf. ≠ x40)

Object 0x5FF0 *active motion block* is only available in *table travel record* mode in motion control configurations (**P.30** = x40). The *table travel record* mode is activated by object 0x6060 *modes of operation* set to **-1**.

The data of *active motion block* displays the active motion block number in *table travel record* mode. The Object refers to parameter *Actual motion block* **1246**. Refer to the application manual "Positioning" for the usage of the motion blocks.

9.9.4.24 0x5FF1/0 Motion block to resume

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x5FF1	0	Motion block to resume	Unsigned8	ro	Tx	

Object works in:	Object does not work in:
– Motion Control: ○ **Table Travel record mode**	– Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Profile Velocity mode** ○ **Homing mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave** – **Non motion Control** (conf. ≠ x40)

Object 0x5FF1 *motion block to resume* is only available in *table travel record* mode in motion control configurations (**P.30** = x40). The *table travel record* mode is activated by object 0x6060 *modes of operation* set to **-1**.

The data of *motion block to resume* displays the motion block to resume number in *table travel record* mode. The Object refers to parameter *Motion block to resume* **1249**. Refer to the application manual "Positioning" for the usage of the motion blocks.

9.9.5 Device Profile Objects (0x6nnn)

9.9.5.10x6007/0 Abort Connection option code

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6007	0	Abort Connection option code	Integer16	rw	No	1

Object *abort connection 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 *Local/Remote* **412**, the response of the setting of object 0x6007 will change as shown in the following table.

Object 0x6007/0		
Operation mode	**Function with "Control via Statemachine"**	**Function in "Other control"**
0 - No reaction	Operating point is maintained.	Operating point is maintained.
1 - Error	"Fault" status will be activated immediately. **Factory setting**.	The controller (state machine) switches to "Fault" state immediately.
2 - Stop	Control command "Disable voltage" and switch to "switch on disabled" status.	
3 - Quick-Stop	Control command "Quick stop" and switch to "switch on disabled d" status.	
-1 - Ramp-Stop + (Minus 1) Error	Control command "Disable operation" and switch to "Error" status once the drive has been shut down.	
-2 - Quick-Stop + (Minus 2) Error	Control command "Quick stop" and switch to "Error" status once the drive has been shut down.	

NOTICE

Object *abort connection option code* corresponds to frequency inverter parameter *Bus Error Behavior* **388**.

The parameter settings *Bus Error Behavior* **388** = -2…3 are evaluated depending on parameter *Local/Remote* **412**.

No.	Object	Min.	Max.
0x6007/0	Abort Connection option code	-2 (=0xFFFE)	3

Bus Error behavior 388	0x6007
0	0
1	1
2	2
3	3
4	-1
5	-2

Writing of parameters *Bus Error Behavior* **388** and writing of object 0x6007 have the same effect.

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. When the frequency inverter is switched on the next time, the value for 0x6007 set before will be activated again and overwrite the setting of Parameter *Bus Error Behavior* **388**.

For details about possible faults, refer to Chapter 18.4 "Error messages".

9.9.5.20x603F/0 Error code

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x603F	0	Error code	Unsigned16	ro	No	

In object *error code*, the last error that has occurred will be saved.

According to CANopen® DS402, a great number of possible error messages is specified. The following list shows the relation between the error code displayed by the frequency inverter on the control panel and the error saved in object *error code*.

Error reports				
Device Error		**DS402 Error code**		**Meaning**
F00	xx	00	00	No error has occurred
Overload				
F01	xx	23	10	Frequency inverter was overloaded
Case				
F02	xx	42	10	Case temperature outside the temperature limits
Inside				
F03	xx	41	10	Inside temperature outside of temperature limits
Motor connection				
F04	xx	43	10	Motor temperature too high or sensor defective
Output current				
F05	xx	23	40	Motor phase current above current limit
DC link voltage				
F07	xx	32	10	DC link voltage outside the voltage range

Electronic voltage				
F08	xx	51	11	Electronic voltage outside the voltage range
Motor connection				
F13	xx	23	30	Earth fault on frequency inverter output
Generic error				
Fyy	xx	10	00	Other error messages

If CANopen® DS402 *error code* 1000 = generic-error occurs, the error code can be read via parameter *Current error* **260** (unsigned16). Parameter *Current error* **260** contains the error code in internal product-specific format.

For the error code assignment table of the relevant messages, refer to the operating instructions.

In the emergency message, the error code of the frequency inverter is transmitted in bytes 4 ... 7 and the DS402 error code in bytes 0 and 1.

9.9.5.3 0x6040/0 Control word

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6040	0	Control word	Unsigned16	rw	Rx	0

Object 0x6040/0 *controlword* is relevant to the frequency inverter if parameter *Local/Remote* **412** is set to "1 - Control via statemachine". Object 0x6040/0 *controlword* is linked to the internal parameter *Control word* **410**. When using CANopen® use object 0x6040/0 *controlword* instead of parameter *Control word* **410**.

Control word (Control word)	
Bit	
0	Switch on
1	Enable voltage
2	Quick stop (Low active)
3	Enable operation
4	Operation mode specific
5	Operation mode specific
6	Operation mode specific
7	Fault reset
8	Halt
9	Operation mode specific
10	Reserved
11	Manufacturer specific
12	Manufacturer specific
13	Manufacturer specific
14	Manufacturer specific
15	Manufacturer specific

Bits 4, 5, 6 and 9 … 15 are used in motion control configurations (**P.30** = `x40`) only.

See chapter 12 "Control of frequency inverter" and 18.1 "Control Word/Status word Overview".

9.9.5.4 0x6041/0 Status word

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6041	0	Status word	Unsigned16	ro	Tx	

Object 0x6041/0 *statusword* shows the current state of the frequency inverter.

Object 0x6041/0 *statusword* is linked to the internal parameter *Status word* **411**.

When using CANopen® use object 0x6041/0 *statusword* instead of parameter *Status word* **411**.

Bits 8, 12, 13 and 14 are used in motion control configurations (**P.30** = `x40`) only.

See chapter 12 "Control of frequency inverter" and 18.1 "Control Word/Status word Overview".

Object 0x6041/0 Status word	
Bit	
0	Ready to switch on
1	Enable voltage
2	Switched on
3	Enable operation
4	Operation enabled
5	Operation mode specific
6	Fault
7	Fault reset
8	Voltage enabled
9	Operation mode specific
10	Quick stop (Low active)
11	Manufacturer specific
12	Switch on disabled
13	Manufacturer specific
14	Warning
15	Manufacturer specific Warning2

9.9.5.50x6042/0 *vl* target velocity [rpm]

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6042	0	Vl target velocity	Integer16	rw	Rx	0

Object works in:	Object does not work in:
– Motion Control: ○ **Velocity mode** – Non motion Control (conf. ≠ `x40`)	– Motion Control: ○ **Table Travel record mode** ○ **Profile Velocity mode** ○ **Profile Positioning mode** ○ **Homing mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave**

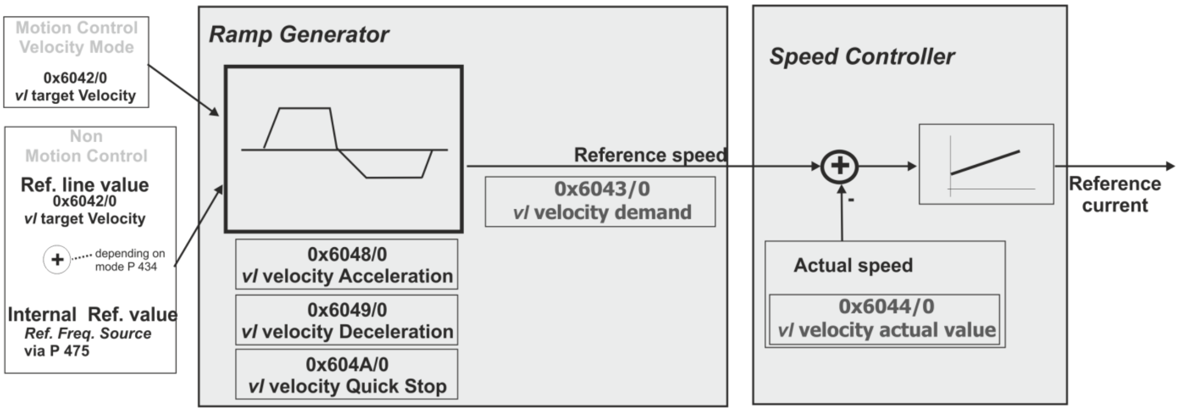
The object *vl target velocity* is the reference speed for the frequency inverter. *vl target velocity* is interpreted as a rotational speed with unit min^{-1}. The internal reference frequency of the frequency inverter is calculated from the target velocity in min^{-1} considering parameter *No. of pole pairs* **373**.

Parameter *No. of pole pairs* **373** has four different data sets. In applications with Motion control (configuration = `x40`), only data set 1 is used.

In applications without Motion Control (configuration ≠ `x40`), more than one motor is connected to the frequency inverter in many cases (only one motor will be active at a time, switched via contactor). These motors can have different numbers of pole pairs. In this case, the entry in parameter *No. of pole pairs* **373** will be different in the four datasets. Upon switching to a motor, object *vl target* velocity must be written at least once in order to enable calculation of the reference frequency of the frequency inverter with the right number of pole pairs.

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x6042	*vl* target velocity	-32768	32767

In Non motion Control configurations (conf. ≠ x40) the *vl target velocity* reference value is product-internally connected to the **Reference line value.** This reference value is combined with the internal reference frequency value from the frequency reference value channel in the input of the ramp function (see chapter 12.3.3 "Reference value / actual value").



9.9.5.60x6043/0 *vl* velocity demand [rpm]

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6043	0	*vl* velocity demand	Integer16	ro	Tx	

Object *vl velocity demand* is the output quantity of the ramp function in unit min^{-1}. The object has the same notation as object *vl target velocity* and can be read as an actual value. For calculation of *vl velocity demand*, parameter *No. of pole pairs* **373** is considered (in the same way as described for object *vl target velocity*).

9.9.5.70x6044/0 *vl* velocity actual value

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6044	0	Vl velocity actual value	Integer16	ro	Tx	

Object *vl velocity actual value* is the current speed of the drive in min^{-1}. The object has the same notation as object *vl target velocity* and can be read as an actual value. For calculation of *vl velocity actual value demand*, parameter *No. of pole pairs* **373** is considered (in the same way as described for object *vl target velocity*).

9.9.5.80x6046/n *vl* velocity min max amount

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6046	0	Highest sub-index supported	Unsigned8	ro	No	2
	1	*vl* velocity min amount (RPM)	Unsigned32	rw	No	See text
	2	*vl* velocity max amount (RPM)	Unsigned32	rw	No	See text

Object works in:	Object does not work in:
– Motion Control: ○ **All modes** – Non motion Control (conf. ≠ x40)	

Object vl velocity min max amount comprises Subindex 01 = vl velocity min amount and Subindex 02 = vl velocity max amount.

The unit of 0x6046/n *vl velocity min max amount* is min^{-1} (positive values only). The value of 0x6046/n *vl velocity min max amount* is converted to a frequency value internally, considering parameter *No. of pole pairs* **373** (in dataset 1).

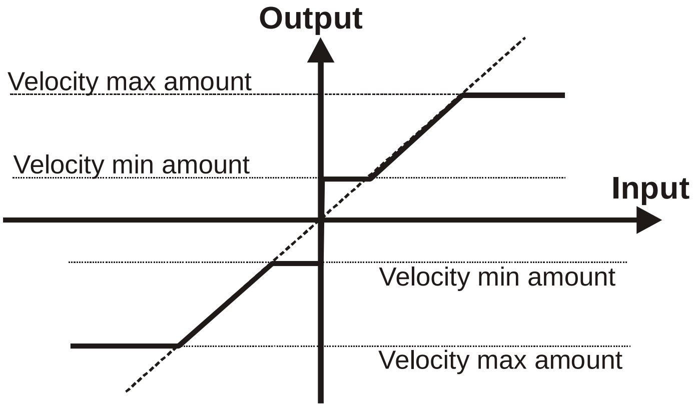
Writing of object 0x6046/1 *vl velocity min amount* will automatically generate a write instruction for parameter *Minimum frequency* **418** to RAM (→ dataset 5, all datasets in RAM only).

Writing of object 0x6046/2 *vl velocity max amount* will automatically generate a write instruction for parameter *Maximum frequency* **419** to RAM (→ dataset 5, all datasets in RAM only).

The default setting depends on the motor settings used.

When the reference value specified with object 0x6042 *vl target velocity* is smaller than object value 0x6046/1 *vl velocity min amount* or greater than 0x6046/2 *vl velocity max amount*, 0x6042 *vl target velocity* is limited to the relevant values.

No.	Object	Min.	Max.
0x6046/1	*vl* velocity min amount (RPM)	1	32767 (= 0x7FFF)
0x6046/2	*vl* velocity max amount (RPM)	1	32767 (= 0x7FFF)



When objects 0x6046/1 or 0x6046/2 are written and a save parameters instruction is generated after that (object 0x1010), the object values in the non-volatile memory will be saved. When the frequency inverter is switched on the next time, the values set before will be activated again and overwrite the settings of parameters *Minimum frequency* **418** and *Maximum frequency* **419**.

In Positioning applications the overall speed can fall below or exceed the limits defined by Minimum and Maximum frequency due to the influence of the Position controller. The output of the Position Controller can be limited by *Limitation* **1118.**

9.9.5.9 0x6048/n *vl* velocity acceleration

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6048	0	Highest sub-index supported	Unsigned8	ro	No	2
	1	Delta speed (min^{-1})	Unsigned32	rw	No	0x96
	2	Delta time (sec)	Unsigned16	rw	No	1

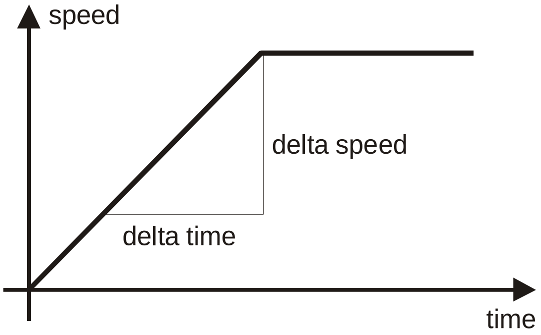
Object works in:	Object does not work in:
– Motion Control: ○ **Velocity mode** – Non motion Control (conf. ≠ x40)	– Motion Control: ○ **Table Travel record mode** ○ **Profile Velocity mode** ○ **Profile Positioning mode** ○ **Homing mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave**

With object 0x6048 *vl velocity acceleration* the change of speed and startup time is set in **velocity mode.** Object 0x6048 *vl velocity acceleration* comprises *delta speed* in min^{-1} and *delta time* in seconds.

The frequency gradient during startup is written to parameters *Acceleration clockwise* **420** and *Acceleration anticlockwise* **422** (dataset 5, all datasets only in RAM). Both parameters are set to the same value. The values of parameters *Acceleration clockwise* **420** and *Acceleration anticlockwise* **422** are converted to a value in unit frequency/second, considering parameter *No. of pole pairs* **373** (in dataset 1).

The gradient is changed internally by the change in objects *delta-time* or *delta-speed*.

No.	Object	Min.	Max.
0x6048/1	Delta speed (RPM)	1	32767 (= 0x7FFF)
0x6048/2	Delta time (sec)	1	65535 (= 0xFFFF)



When objects 0x6048/1 or 0x6048/2 are written and a save parameters instruction is generated after that (object 0x1010), the object values in the non-volatile memory will be saved. When the frequency inverter is switched on the next time, the values set before will be activated again and overwrite the settings of parameters *Acceleration clockwise* **420** and *Acceleration anticlockwise* **422**.

9.9.5.10 0x6049/n *vl* velocity deceleration

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6049	0	Highest sub-index supported	Unsigned8	ro	No	2
	1	Delta speed (min^{-1})	Unsigned32	rw	No	0x96
	2	Delta time (sec)	Unsigned16	rw	No	1

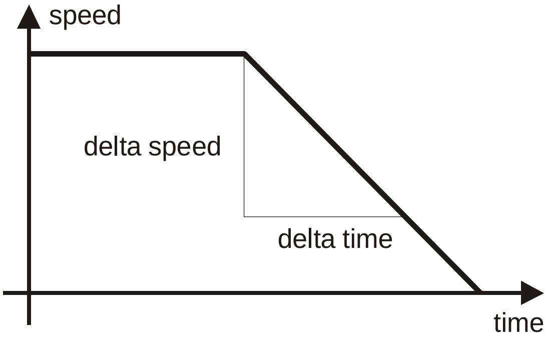
Object works in:	Object does not work in:
– Motion Control: ○ **Velocity mode** – Non motion Control (conf. ≠ x40)	– Motion Control: ○ **Table Travel record mode** ○ **Profile Velocity mode** ○ **Profile Positioning mode** ○ **Homing mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave**

With object 0x6049 *vl velocity deceleration* the change of speed and shutdown time is set. Object 0x6049 *vl velocity deceleration* comprises *delta speed* in min^{-1} and *delta time* in seconds.

The frequency gradient during shutdown is written to parameters *Deceleration clockwise* **421** and *Deceleration anticlockwise* **423** (dataset 5, all datasets only in RAM). Both parameters are set to the same value. The values of parameters *Deceleration clockwise* **421** and *Deceleration* anticlockwise 423 are converted to a value in unit frequency/second, considering parameter *No. of pole pairs* **373** (in dataset 1).

The gradient is changed internally by the change in objects *delta-time* or *delta-speed*.

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x6049/1	Delta speed (RPM)	1	32767 (= 0x7FFF)
0x6049/2	Delta time (sec)	1	65535 (= 0xFFFF)



When objects 0x6049/1 or 0x6049/2 are written and a save parameters instruction is generated after that (object 0x1010), the object values in the non-volatile memory will be saved. When the frequency inverter is switched on the next time, the values set before will be activated again and overwrite the settings of parameters *Deceleration clockwise* **421** and *Deceleration anticlockwise* **423**.

9.9.5.11 0x604A/n *vl* velocity quick stop

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x604A	0	Highest sub-index supported	Unsigned8	ro	No	2
	1	Delta speed (min^{-1})	Unsigned32	rw	No	0x96
	2	Delta time (sec)	Unsigned16	rw	No	1

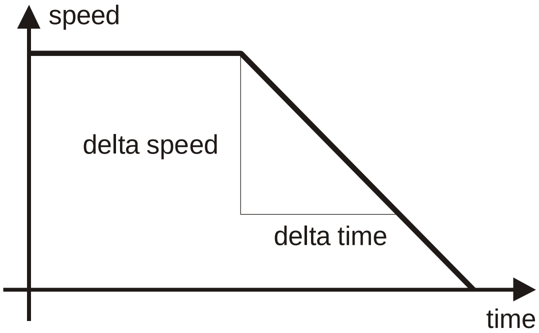
Object works in:	Object does not work in:
– Motion Control: ○ **Velocity mode** – Non motion Control (conf. ≠ x40)	– Motion Control: ○ **Table Travel record mode** ○ **Profile Velocity mode** ○ **Profile Positioning mode** ○ **Homing mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave**

With object 0x604A *vl velocity quick stop*, you can set the quick stop deceleration. Object 0x604A *vl velocity quick stop* comprises speed change in min^{-1} and shutdown time in seconds.

The frequency gradient during shutdown is written to parameters *Emergency Stop Clockwise* **424** and *Emergency Stop Anticlockwise* **425** (dataset 5, all datasets only in RAM). Both parameters are set to the same value. The value of parameters *Emergency Stop Clockwise* **424** and *Emergency Stop Anticlockwise* **425** are converted internally to a value in unit frequency/second, considering parameter *No. of pole pairs* **373** (in dataset 1).

The gradient is changed internally by the change in objects *delta-time* or *delta-speed*.

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x604A/1	Delta speed (rpm)	1	32767 (= 0x7FFF)
0x604A/2	Delta time (sec)	1	65535 (= 0xFFFF)



When objects 0x604A/1 or 0x604A/2 are written and a save parameters instruction is generated after that (object 0x1010), the object values in the non-volatile memory will be saved. When the frequency inverter is switched on the next time, the values set before will be activated again and overwrite the settings of parameters *Emergency Stop Clockwise* **424** and *Emergency Stop Anticlockwise* **425**.

9.9.5.12 0x6060/0 Modes of operation

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6060	0	Modes of operation	Integer8	wo	Rx	2

Object works in:	Object does not work in:
– Motion Control: ○ **All modes**	– **Non motion Control** (conf. ≠ x40)

With object **0x6060** *modes of operation*, the designated operation mode of the inverter is set. Depending on the used configuration of the inverter, there are different choices feasible.

Available values for *modes of operation* with inverter in motion control configuration (**P.30** = x40 and **412** *Local/Remote* = "1 – Control via Statemachine"):

Modes of operation		
Dec. / Hex.		**Mode**
1 0x01	–	Profile position mode
2 0x02	–	Velocity mode **(Default)**
3 0x03	–	Profile velocity mode
6 0x06	–	Homing mode
7 0x07	–	Interpolated position mode - **not supported by Ethernet/IP**
8 0x08	–	Cyclic sync position mode - **not supported by Ethernet/IP**
9 0x09	–	Cyclic sync velocity mode - **not supported by Ethernet/IP**
-1 0xFF	–	Table travel record (manufacturer specific mode)
-2 0xFE	–	Move away from Limit switch (manufacturer specific mode)
-3 0xFD	–	Electronic Gear: Slave (manufacturer specific mode)

Object 0x6060 *modes of operation* is limited as described in the table.

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x6060/0	Modes of operation	-3 | 0xFD	9

Available value for *modes of operation* with inverter in non-motion control configuration (**P.30** = `x40` and **412** *Local/Remote* = "1 – Control via Statemachine")

Modes of operation	
2 –	velocity mode

The inverter in non-motion control configuration ignores all settings other than "**2**". When accessing via SDO, an SDO fault message is generated, that prompts the invalid value.

For further information see chapter 12 "Control of frequency inverter".

9.9.5.13 0x6061/0 Modes of operation display

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6061	0	Modes of operation display	Integer8	ro	Tx	

Object works in: – Motion Control: ○ **All modes**	Object does not work in: – Non motion Control (conf. ≠ `x40`) ○ Value always "`2`"

Object 0x6061 *modes of operation display* acknowledges the previously set value of *modes of operation* by displaying the same value as *modes of operation*.

After setting 0x6060 *modes of operation*, the PLC must wait for this acknowledgement before sending any other command to the inverter.

For further information see chapter 12 "Control of frequency inverter".

9.9.5.14 0x6064/0 Position actual value

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6064	0	Position actual value	Integer32	ro	Tx	

Object works in: – Motion Control: ○ **All modes**	Object does not work in: – **Non motion Control** (conf. ≠ `x40`)

Object 0x6064 *position actual value* represents the actual value of the position measurement device in user units.

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*. The value is the same like stated in parameter *Actual Position* **1108**.

9.9.5.15 0x6065/0 Following error window

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6065	0	Following error window	Unsigned32	rw	No	0xFFFF FFFF

Object works in: – Motion Control: ○ **All modes**	Object does not work in: – **Non motion Control** (conf. ≠ `x40`)

Object 0x6065 *following error window* is used to set the threshold of a device warning when the following error becomes too big.

In the application manual "Positioning", the term "Contouring error" is used instead of the CANopen® term "Following error".

Object 0x6065 *following error window* defines a range of tolerated position values symmetrical to the *position demand value* defined in user units.

The valid value range of object 0x6065/0 *following error window* is 0 … 0x7FFF FFFF (2^{31}-1). Writing a value of 0x8000 0000 (2^{31})… 0xFFFF FFFE (2^{32}-2) results in an SDO abort (value range).

If the value of the *following error window* is set to 0xFFFF FFFF (2^{32}-1) OR 0, the *following error window* is switched off.

The actual following error is displayed in object 0x60F4 Following error actual value.

The warning is triggered if the Following error window was exceeded for the time specified in Object 0x6066 *following error time out*. No device fault is triggered.

ℹ Writing to object *following error window* automatically generates a write command to contouring error parameter *Warning Threshold* **1105** (data set 5, all data sets in RAM only !).

ℹ If object 0x6065/0 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of *Warning Threshold* **1105**.

ℹ The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

9.9.5.16 0x6066/0 Following error time out

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6066	0	Following error time out	Unsigned16	rw	No	0xA (=10)

Object works in:	Object does not work in:
– Motion Control: ○ **All modes**	– **Non motion Control** (conf. ≠ x40)

When a following error (Object 0x6065 *following error window*) occurs longer than the defined value of object 0x6066 *following error time out* given in milliseconds, the corresponding bit in the Status word (bit 13 *following error*) is set to one. No device fault is triggered.

ℹ Writing to object *following error time out* automatically generates a write command to parameter *Contouring Error Time* **1119** (data set 5, all data sets in RAM only !).

ℹ If object 0x6066/0 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of *Contouring Error Time* **1119**.

9.9.5.17 0x6067/0 Position window

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6067	0	Position window	Unsigned32	rw	No	0xFFFF FFFF

Object works in:	Object does not work in:
– Motion Control: ○ **All modes**	– **Non motion Control** (conf. ≠ x40)

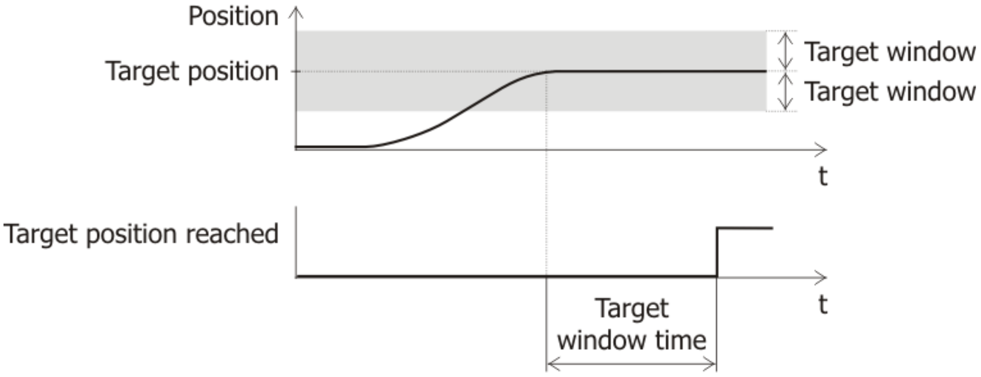
The signal "target position reached" can be changed in accuracy with Object 0x6067 *Position window* for the modes which use Status Word Bit 10 "Target reached" as "Target Position reached" like "Profile Positioning Mode" and "Table Travel Record Mode".

Object 0x6067 *position window* defines a symmetrical range of accepted positions relative to the target position in user units. If the actual value of the position measurement device is within the position window, the target position is regarded as reached. "Target reached" is displayed in Bit 10 of the status word. The actual position must be inside the position window during the time specified in Object 0x6068 *Position window time*.

If the actual position drifts outside the target window or if a new target position is set, the "Target reached" Bit is reset until the position and time conditions are met again.

The valid value range of object 0x6067/0 *position window* is 0 … 0x7FFF FFFF (2^{31}-1). Writing a value of 0x8000 0000 (2^{31})… 0xFFFF FFFE (2^{32}-2) results in an SDO abort (value range).

If the value of *Position window* is set to 0xFFFF FFFF (2^{32}-1) OR 0, the position window control is switched off.



Writing to object *position window* automatically generates a write command to parameter *Target Window* **1165** (data set 5, all data sets in RAM only !).

If object 0x6067/0 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of *Target Window* **1165**.

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

9.9.5.18 0x6068/0 Position window time

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6068	0	Position window time	Unsigned16	rw	No	0xA (=10)

Object works in:	Object does not work in:
– Motion Control: ○ **All modes**	– **Non motion Control** (conf. ≠ x40)

When the actual position is within the *Position window* during the defined *Position window time* (given in milliseconds), then the corresponding bit in the Status word (bit 10 *target reached*) is set to one. This is considered in Modes that use Status Word Bit 10 "Target reached" as "Target Position reached" like "Profile Positioning Mode" and "Table Travel Record Mode".

Writing to object *Position window time* automatically generates a write command to parameter *Target Window Time* **1166** (data set 5, all data sets in RAM only!).

If object 0x6068/0 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of *Target Window Time* **1166**.

9.9.5.19 0x606C/0 Velocity actual value [u/s]

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x606C	0	velocity actual value	Integer32	ro	Tx	

Object works in: – Motion Control: ○ **All modes**	Object does not work in: – **Non motion Control** (conf. ≠ x40)

The actual velocity value in [u/s] is displayed.

9.9.5.20 0x606D/0 Velocity Window

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x606D	0	Velocity Window	Unsigned16	rw	No	1000

Object works in: – Motion Control: ○ **Profile Velocity mode**	Object does not work in: – Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave** – **Non motion Control** (conf. ≠ x40)

Object 0x606D *Velocity window* is used to define the threshold of Bit 10 "Target reached" of the Status word in Profile Velocity mode.

Object 0x606D *Velocity window* defines the symmetric range around the value of Object 0x60FF *Target Velocity* in user units/s.

Bit 10 "Target reached" is set in the Status word when the difference between 0x60FF *Target Velocity* and 0x606C *Velocity Actual value* is smaller than the 0x606D *Velocity Window* for a longer time than 0x606E *Velocity Window Time*.

The value range of Object 0x606D/0 *Velocity Window* is 0 … 65535 u/s.

If the value of 0x606D/0 *Velocity Window* is set to 0, bit 10 "Target reached" of the Status word is only set with the exact equality of actual speed and reference speed. It is recommended to set the value large enough to get a reliable status information of Bit 10 "Target reached".

Writing to object 0x606D/0 *Velocity Window* automatically generates a write command to parameter *Velocity Window* **1276** (data set 5, all data sets in RAM only !).

If object 0x606D/0 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Velocity Window* **1276**.

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

9.9.5.21 0x606E/0 Velocity Window Time

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x606E	0	Velocity Window time	Unsigned16	rw	No	0

Object works in:	Object does not work in:
– Motion Control: ○ **Profile Velocity mode**	– Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave** – **Non motion Control** (conf. ≠ x40)

Object 0x606E *Velocity window Time* defines the time, for which at least reference velocity and Actual velocity must be similar enough to set Bit 10 "Target reached" of the Status word. The similarity ("Hysteresis") is defined via 0x606D *Velocity Window*.

0x606D *Velocity window* defines the symmetric range around the value of Object 0x60FF *Target Velocity* in user units/s.

Bit 10 "Target reached" is set in the Status word when the difference between 0x60FF *Target Velocity* and 0x606C *Velocity Actual value* is smaller than the 0x606D *Velocity window* for a longer time than 0x606E *Velocity Window Time*.

If both conditions are not met at the same time, bit 10 "Target reached" of the Status word is reset.

The value range of Object 0x606E/0 *Velocity Window Time* is 0 … 65535 ms.

Writing to object 0x606E/0 *Velocity Window Time* automatically generates a write command to parameter *Velocity Window Time* **1277** (data set 5, all data sets in RAM only).

If object 0x606D/0 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Velocity Window Time* **1277**.

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

9.9.5.22 0x606F/0 Velocity Threshold

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x606F	0	Velocity Threshold	Unsigned16	rw	No	100

Object works in:	Object does not work in:
– Motion Control: ○ **Profile Velocity mode**	– Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave** – **Non motion Control** (conf. ≠ x40)

Object 0x606F *Velocity Window Threshold* defines a threshold to change Bit 12 "Velocity" of the Status word in Profile Velocity mode. If the absolute value of the Actual Velocity lies for the time given over 0x6070 *Velocity Threshold Time* above the threshold 0x606F *Velocity Threshold*, the bit is reset. If the Actual Velocity falls below the defined threshold of 0x606F *Velocity Threshold*, bit 12 "Velocity" of the Status word is set.

The value range of Object 0x606F/0 *Velocity Window Threshold* is 0 … 65535 u/s.

Writing to object 0x606F/0 *Velocity Window Threshold* automatically generates a write command to parameter *Threshold Window* **1278** (data set 5, all data sets in RAM only !).

If object 0x606F/0 *Velocity Threshold* was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Threshold Window* **1278**.

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

9.9.5.23 0x6070/0 Velocity Threshold Time

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6070	0	Velocity Threshold Time	Unsigned16	rw	No	0

Object works in:	Object does not work in:
– Motion Control: ○ **Profile Velocity mode**	– Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave** – **Non motion Control** (conf. ≠ x40)

If the absolute value of the Actual Velocity lies for the time given over 0x6070 *Velocity Threshold Time* above the threshold 0x606F *Velocity Threshold*, the bit is reset. If the Actual Velocity falls below the defined threshold of 0x606F *Velocity Threshold*, bit 12 "Velocity" of the Status word is set.

The value range of Object 0x6070/0 *Velocity Window Time* is 0 … 65535 ms.

Writing to object 0x606F/0 *Velocity Window Threshold* automatically generates a write command to parameter *Threshold Window Time* **1279** (data set 5, all data sets in RAM only !).

If object 0x606F/0 *Velocity Threshold* was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Threshold Window Time* **1279**.

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

9.9.5.24 0x6071/0 Target Torque

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6071	0	Target Torque	Integer16	rw	Rx	0

The value that can be transmitted via object 0x6071 is selectable as source 808 for various parameters (e.g. *FT input buffer percentage* **1381**).

It is also available as operation mode 95 or inverted operation mode 195 (e.g. for parameter *Reference percentage source* **476**) in configurations with torque control (*Configuration* **30** = x30).

The value 0x3E8 (=1000) is the rated motor torque (100.0 %).

By default, object 0x6071 is not linked to a device function. In order to use object 0x6071, at least one device function must be linked to the object by setting the relevant parameters.

The values of object 0x6071 range from -3000 to 3000 (= -300.0…300.0 %).

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x6071/0	Target Torque	-3000 (= 0xF448)	3000 (= 0x0BB8)

Hexadecimal value 0x6071	Decimal value 0x6071	Percentage of Target Torque
0x03E8	1000	100.0
0x0064	100	10.0
0x0001	1	0.1
0xFF18	-1000	-100.0
0xFF9C	-100	-10.0
0xFFFF	-1	-0.1

9.9.5.25 0x6077/0 Torque actual value

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6077	0	Torque actual value	Integer16	ro	Tx	

Object 0x6077 *Torque actual value* shows the actual torque (see parameter *Torque* **224**).

The value 0x3E8 (=1000) is the rated motor torque (100.0 %). Please also note object 0x6071.

9.9.5.26 0x6078/0 Current actual value

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6078	0	Current actual value	Integer16	ro	Tx	

Object 0x6078 *Current actual value* shows the actual current value (see parameter *R.m.s current* **211**).

The value 0x3E8 (=1000) is the rated motor current (100.0 %). The rated motor current is set during commissioning in parameter *Rated current* **371**.

9.9.5.27 0x6079/0 DC link circuit voltage

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6079	0	DClink circuit voltage	Integer32	ro	Tx	

Object 0x6079 *DC link circuit voltage* shows the actual value of the DC link voltage in mV (see parameter *DC link voltage* **222**).

Value 0x0001 86A0 (=100 000) is equivalent to 100.000 V (three decimal places).

9.9.5.28 0x607A/0 Target position

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x607A	0	Target position	Integer32	rw	Rx	0

Object works in:	Object does not work in:
– Motion Control: ○ **Profile Positioning mode**	– Motion Control: ○ **Velocity mode** ○ **Profile Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave** – Non motion Control (conf. ≠ x40)

Object 0x607A *target position* defines the position (in user units) that the drive should move to in profile position mode.

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

9.9.5.29 0x607C/0 Home offset

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x607C	0	Target position	Integer32	rw	No	0

Object works in:	Object does not work in:
– Motion Control: ○ **Homing mode**	– Motion Control: ○ **Velocity mode** ○ **Profile Velocity mode** ○ **Profile Positioning mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave** – Non motion Control (conf. ≠ x40)

Object 0x607C *home offset* defines the offset between the zero position of the position measurement device found during homing and the zero position of the application. All subsequent movements are in relation to the application zero position.

Writing to object *home offset* automatically generates a write command to parameter *Home Offset* **1131** (data set 5, all data sets in RAM only !).

If object 0x607C/0 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Home Offset* **1131**.

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

9.9.5.30 0x6081/0 Profile velocity [u/s]

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6081	0	Profile velocity	Unsigned32	rw	Rx	0x5 0000

Object works in:	Object does not work in:
– Motion Control: ○ **Profile Velocity mode**	– Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave** – **Non motion Control** (conf. ≠ x40)

Object 0x6081 *profile velocity* is the velocity (in user units per second) at the end of the acceleration ramp in profile position mode.

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

The values of Object 0x6081 are limited to 1 to 0x7FFF FFFF.

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x6081/0	Profile velocity (u/s)	1	2147483647 (= 0x7FFF FFFF)

9.9.5.31 0x6083/0 Profile acceleration

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6083	0	Profile acceleration	Unsigned32	rw	Rx	0x5 0000

Object works in:	Object does not work in:
– Motion Control: ○ **Profile Velocity mode** ○ **Profile Positioning mode** ○ **Electronic Gear: Slave**	– Motion Control: ○ **Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** – **Non motion Control** (conf. ≠ x40)

Object 0x6083 *profile acceleration* is the acceleration in user units per second^2 [u/s²] in profile position mode.

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

The values of Object 0x6083 are limited to 1 to 0x7FFF FFFF (2^{31}-1).

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x6083/0	Profile acceleration (u/s²)	1	2147483647 (= 0x7FFF FFFF)

9.9.5.32 0x6084/0 Profile deceleration

WARNING

Safety functions failure

If **0x6084** is set to inappropriate settings, the safety function may fail!

- Always check the deceleration ramp configuration for appropriate settings.

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6084	0	Profile deceleration	Unsigned32	rw	Rx	0x5 0000

Object works in:	Object does not work in:
– Motion Control: ○ **Profile Velocity mode** ○ **Profile Positioning mode** ○ **Electronic Gear: Slave**	– Motion Control: ○ **Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** – **Non motion Control** (conf. ≠ `x40`)

Object 0x6084 *profile deceleration* is the deceleration in u/s².

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

The values of Object 0x6084 are limited to 1 to 0x7FFF FFFF (2^{31}-1).

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x6084/0	Profile deceleration (u/s²)	1	2147483647 (= 0x7FFF FFFF)

9.9.5.33 0x6085/0 Quick stop deceleration

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6085	0	Quick stop deceleration	Unsigned32	rw	No	0xA 0000

Object works in:	Object does not work in:
– Motion Control: ○ **Profile Positioning mode** ○ **Profile Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave**	– Motion Control: ○ **Velocity mode** ○ ○ **Non motion Control** (conf. ≠ x40)

Object 0x6085 *quick stop deceleration* is the deceleration (in user units per second²) in profile position mode for quick stop mode (Control word bit 2 = 0).

Writing to object *Quick stop deceleration* automatically generates a write command to parameter *Emergency Ramp* **1179** (data set 5, all data sets in RAM only !).

If object 0x6085/0 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory. After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Emergency Ramp* **1179**.

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

The values of Object 0x6085 are limited to 1 to 0x7FFF FFFF (2^{31}-1).

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x6085/0	Quick stop deceleration (u/s²)	1	2147483647 (= 0x7FFF FFFF)

9.9.5.34 0x6086/0 Motion profile type

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6086	0	Motion profile type	Integer16	rw	No	3

Object works in:	Object does not work in:
– Motion Control: ○ **Profile Positioning mode** ○ **Profile Velocity mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave**	– Motion Control: ○ **Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Non motion Control (conf. ≠ x40)**

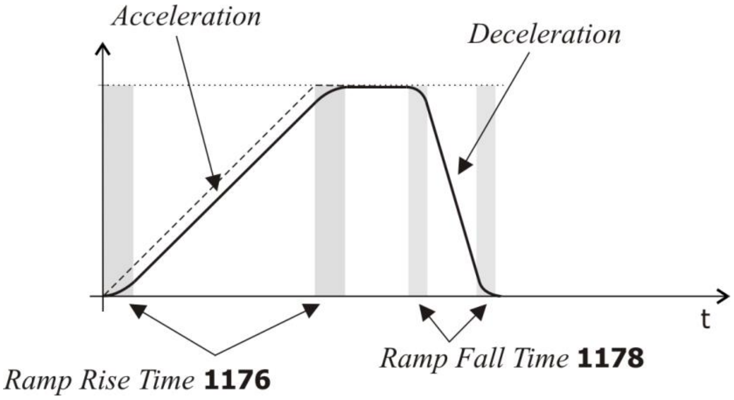
Object 0x6086 *motion profile type* defines the ramp behavior for acceleration/deceleration.

Supported values for *motion profile type*:

- 0 - linear ramp
- 3- jerk limited ramp

In mode 3 "jerk limited ramp", the ramp uses the parameters:

- *Ramp Rise Time* **1176**
- *Ramp Fall Time* **1178**



- The Ramp Rise/Fall time in *Table travel record mode* is defined via parameters **1205** and **1207**.
- The Ramp Rise/Fall time in *Homing mode* is defined via parameter **1135**.
- The Ramp Rise/Fall time in *Velocity mode* and in Non-Motion Control configurations is defined via parameters **430**…**433**.
- The Ramp Rise/Fall times in these modes are independent of the settings of object 0x6086.

9.9.5.35 0x6091/n Gear ratio

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6091	0	Highest sub-index supported	Unsigned8	ro	No	2
	1	Motor shaft revolutions	Unsigned32	rw	No	1
	2	Driving shaft revolutions	Unsigned32	rw	No	1

Object works in:	Object does not work in:
– Motion Control: ○ **All modes**	○ **Non motion Control** (conf. ≠ x40)

Object 0x6091 *gear ratio* defines the ratio of motor shaft revolutions to driving shaft revolutions.

$$\text{Gear ratio} = \frac{\text{0x6091/1 } \textit{motor shaft revolutions}}{\text{0x6091/2 } \textit{driving shaft revolutions}} \mathrel{\hat{=}} \frac{\text{Parameter } \textit{Gear Box : Motor Shaft Revolutions } \mathbf{1117}}{\text{Parameter } \textit{Gear Box : Driving Shaft Revolutions } \mathbf{1116}}$$

Writing to object *motor shaft revolutions* automatically generates a write command to parameter *Gear Box: Motor Shaft Revolutions* **1117** (data set 5, all data sets in RAM only!).

If object 0x6091/1 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of *Gear Box: Motor shaft revolutions* **1117**.

Writing to object *driving shaft revolutions* automatically generates a write command to parameter *Gear Box: Driving Shaft Revolutions* **1116** (data set 5, all data sets in RAM only!).

If object 0x6091/2 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Gear Box: Driving Shaft Revolutions* **1116**.

Alternatively the parameters **1116** and **1117** can be used instead of the Objects.

Object		Parameter	
0x6091/1	Motor Shaft revolutions	**1117**	*Gear Box: Motor Shaft Revolutions*
0x6091/2	Driving Shaft revolutions	**1116**	*Gear Box: Driving Shaft Revolutions*

The values of Objects 0x6091/1 and 6091/2 are limited as follows:

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x6091/1	Motor shaft revolutions	1	65535 (= 0x0000 FFFF)
0x6091/2	Driving shaft revolutions	1	65535 (= 0x0000 FFFF)

9.9.5.36 0x6092/n Feed constant

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6092	0	Highest sub-index supported	Unsigned8	ro	No	2
	1	Feed	Unsigned32	rw	No	0x1 0000
	2	(Driving) shaft revolutions	Unsigned32	rw	No	1

Object works in:	Object does not work in:
– Motion Control: ○ **All modes**	○ **Non motion Control** (conf. ≠ x40)

Object 0x6092 *feed constant* defines the feed (in user units) per driving shaft revolutions.

$$\text{Feed constant} = \frac{\text{0x6092/1 } \textit{feed}}{\text{0x6092/2 } \textit{driving shaft revolutions}} \;\hat{=}\; \frac{\text{Parameter } \textit{Feed Constant } \mathbf{1115}}{1}$$

The allowed value for 0x6092/2 *driving shaft revolutions* is **1** only. Writing values other than 1 results in an SDO abort response.

Writing to object *feed* or *driving shaft revolutions* automatically generates a write command to parameter *Feed Constant* **1115** (data set 5, all data sets in RAM only!).

If object 0x6092/1 or 0x6092/2 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Feed constant* **1115**.

The values of Object 0x6092/1 and 6092/2 are limited as follows:

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x6092/1	Feed	1	65535 (= 0x0000 FFFF)

0x6092/2	(Driving) shaft revolutions	1	1

9.9.5.37 0x6098/0 Homing method

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6098	0	Homing method	Integer8	rw	No	0

Object works in:	Object does not work in:
– Motion Control x40: ○ **Homing mode**	– Motion Control x40: ○ Profile Posi**tioning mode** ○ **Profile Velocity mode** ○ **Velocity mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave** ○ **Non motion Control** (conf. ≠ x40)

Object 0x6098/0 *homing method* determines the method that will be used during homing. For a detailed description of the different homing modes see the application manual "Positioning".

Writing to object *homing method* automatically generates a write command to parameter *Homing Mode* **1130** (data set 5, all data sets in RAM only !).

If object 0x6098/0 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Homing Mode* **1130**.

Homing Method 0x6098/0		Function
0 -	No Homing	**Factory setting.** No homing; the current position value is not changed. The current position value is the value saved upon the last disconnection of the power supply.
1 -	Neg. Limit Switch & Ref. -Signal	Homing to negative HW limit switch with detection of encoder ref. signal.
2 -	Pos. Limit Switch & Ref. -Signal	Homing to positive HW limit switch with detection of encoder ref. signal.
3 -	Pos. Home-Sw., Ref.-Signal left of Edge	Homing to positive home switch with detection of encoder ref. signal. Home position is the first encoder ref. signal to the left of the edge of the home switch signal.
4 -	Pos. Home-Sw., Ref.-Signal right of Edge	Homing to positive home switch with detection of encoder ref. signal. Home position is the first encoder ref. signal to the right of the edge of the home switch signal.
5 -	Neg. Home-Sw., Ref.-Signal right of Edge	Homing to negative home switch with detection of encoder ref. signal. Home position is the first encoder ref. signal to the right of the edge of the home switch signal.
6 -	Neg. Home-Sw.: Ref.-Signal left of Edge	Homing to negative home switch with detection of encoder ref. signal. Home position is the first encoder ref. signal to the left of the edge of the home switch signal.
7 -	Pos. Lim.-Sw., Ref.-Sig. left of left Edge of Home-Sw.	Homing to home switch with detection of encoder ref. signal. Homing direction positive (clockwise). Reversal of direction of rotation when positive HW limit switch is reached. Home position is the first encoder ref. signal to the left or right of the left or right edge of the home switch signal.
8 -	Pos. Lim.-Sw., Ref.-Sig. right of left Edge of Home-Sw.	
9 -	Pos. Lim.-Sw., Ref.-Sig. left of right Edge of Home-Sw.	
10 -	Pos. Lim.-Sw., Ref.-Sig. right of right Edge of Home-Sw.	

Homing Method 0x6098/0		Function
11 -	Neg. Lim.-Sw., Ref.-Sig. right of right Edge of Home-Sw.	Homing to home switch with detection of encoder ref. signal. Homing direction negative (anticlockwise). Reversal of direction of rotation when negative HW limit switch is reached. Home position is the first encoder ref. signal to the left or right of the left or right edge of the home switch signal.
12 -	Neg. Lim.-Sw., Ref.-Sig. left of right Edge of Home-Sw.	
13 -	Neg. Lim.-Sw., Ref.-Sig. right of left Edge of Home-Sw.	
14 -	Neg. Lim.-Sw., Ref.-Sig. left of left Edge of Home-Sw.	
17 … 30: like 1 … 14, but without encoder ref. signal		
17 -	Neg. Limit Switch	Homing to negative HW limit switch.
18 -	Pos. Limit Switch	Homing to positive HW limit switch.
19 -	Pos. Home-Sw., left of Edge	Homing to positive home switch. Home position is at the left of the edge of the home switch signal.
20 -	Pos. Home-Sw., right of Edge	Homing to positive home switch. Home position is at the right of the edge of the home switch signal.
21 -	Neg. Home-Sw., right of Edge	Homing to negative home switch. Home position is at the right of the edge of the home switch signal.
22 -	Neg. Home-Sw., left of Edge	Homing to negative home switch. Home position is at the left of the edge of the home switch signal.
23 -	Pos. Lim.-Sw., left of left Edge of Home-Sw.	Homing to home switch. Homing direction positive (clockwise). Reversal of direction of rotation when positive HW limit switch is reached. Home position is at the left or right of the left or right edge of the home switch signal.
24 -	Pos. Lim.-Sw., right of left Edge of Home-Sw.	
25 -	Pos. Lim.-Sw., left of right Edge of Home-Sw.	
26 -	Pos. Lim.-Sw., right of right Edge of Home-Sw.	
27 -	Neg. Lim.-Sw., right of right Edge of Home-Sw.	Homing to home switch. Homing direction negative (anticlockwise). Reversal of direction of rotation when negative HW limit switch is reached. Home position is at the left or right of the left or right edge of the home switch signal.
28 -	Neg. Lim.-Sw., left of right Edge of Home-Sw.	
29 -	Neg. Lim.-Sw., right of left Edge of Home-Sw.	
30 -	Neg. Lim.-Sw., left of left Edge of Home-Sw.	

33 -	Ref.-Signal left of act. pos.	Home position is the first encoder ref. signal in negative (operation mode 33) or positive (operation mode 34) direction.
34 -	Ref.-Signal right of act. pos.	
35 -	Current Position	Current position is home position. Home offset (Parameter *Home-Offset* **1131**) is taken over as actual position value.

9.9.5.38 0x6099/n Homing speeds

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x6099	0	Highest sub-index supported	Unsigned8	ro	No	2
	1	speed during search for switch	Unsigned32	rw	No	0x5 0000
	2	speed during search for zero	Unsigned32	rw	No	0x2 0000

Object works in:

Motion Control:

- **Homing mode**
- **Move away from Limit Switch**
- **Electronic Gear: Slave** [1)]

Object does not work in:

Motion Control x40:

- **Profile Positioning mode**
- **Profile Velocity mode**
- **Velocity mode**
- **Table Travel record mode**

- **Non motion C**ontrol (conf. ≠ x40)

1) Electronic Gear: Slave uses this object for the Master/Slave Position Correction Function, see chapter 12.4.7.2 "Master/Slave Position Correction".

Object 0x6099/1 *speed during search for switch* defines the speed (in user units per second) during search for switch.

Object 0x6099/2 *speed during search for zero* defines the speed (in user units per second) during search for zero. This speed is also used as reference value in the "*Move away from Limit Switch*" mode.

The values of Object 0x6099/1 and 6099/2 are limited as follows:

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x6099/1	speed during search for switch	1	2147483647 (= 0x7FFF FFFF)
0x6099/2	speed during search for zero	1	2147483647 (= 0x7FFF FFFF)

Writing to object *speed during search for switch* automatically generates a write command to parameter *Fast Speed* **1132** into RAM (data set 5, all data sets in RAM only!).

If object 0x6099/1 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Fast speed* **1133**.

Writing to object *speed during search for zero* automatically generates a write command to parameter *Creep speed* **1133** into RAM (data set 5, all data sets in RAM only!).

If object 0x6099/2 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Creep speed* **1133**.

The dimension of the user units is set via objects 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

9.9.5.39 0x609A/0 Homing acceleration

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x609A	0	Homing acceleration	Unsigned32	rw	No	0x5 0000

Object works in:	Object does not work in:
Motion Control: ○ Homing mo**de** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave**[1)]	Motion Control: ○ Profile P**ositioning mode** ○ **Profile Velocity mode** ○ **Velocity mode** ○ **Table Travel record mode** ○ **Non motion** Control (conf. ≠ x40)

1) Electronic Gear: Slave uses this object for the Master/Slave Position Correction Function, see chapter 12.4.7.2 "Master/Slave Position Correction".

Object 0x609A/0 *homing acceleration* defines acceleration and deceleration (in user units per second^2) during homing.

The set value is also used as reference acceleration and deceleration value in "Move away from Limit Switch" mode.

Writing to object *homing acceleration* automatically generates a write command to parameter *Acceleration* **1134** (data set 5, all data sets in RAM only !).

If object 0x609A/0 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Acceleration* **1134**.

The dimension of the user units is set via objects 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

The values of Object 0x609A/0 are limited as follows:

Parameter		Setting	
No.	**Object**	**Min.**	**Max.**
0x609A/0	Homing acceleration	1	2147483647 (= 0x7FFF FFFF)

9.9.5.40 0x60F4/0 Following error actual value

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x60F4	0	Following error actual value	Integer32	ro	Tx	

Object works in:	Object does not work in:
Motion Control: ○ **All modes**	○ **Non** motion Control (conf. ≠ x40)

In the application manual "Positioning", the term "Contouring error" is used instead of the CANopen® term "Following error".

Object 0x60F4 shows the *following error actual value*. The value is the same like stated in parameter *Actual Contouring error* **1109**.

The allowed following error is defined by object 0x6065 *Following error window*.

The Contouring error can be monitored internally to trigger a device fault if a set threshold was reached. Please refer to the Application manual "Positioning" for details concerning the parameters *Fault reaction* **1120**, *Warning threshold* **1105**, *Error Threshold* **1106** and *Contouring error Time* **1119**.

9.9.5.41 0x60F8/0 Max Slippage [u/s]

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x60F8	0	Max Slippage	Integer32	rw	No	0

Object works in:	Object does not work in:
Motion Control: ○ **Profile Velocity mode**	Motion Control: ○ **Profile Positioning mode** ○ **Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave** ○ **Non motion Control** (conf. ≠ x40)

Object 0x60F8/0 *Max Slippage* can be used to trigger a warning in bit 13 "*maximum slip fault*" in the status word when a too high slip occurs. When the difference of stator frequency and actual speed exceeds the value set in 0x60F8 Max Slippage, Bit 13 "Max Slippage" of the Status word is set, otherwise reset.

Writing to object 0x60F8 *Max Slippage* automatically generates a write command to parameter *Max Slippage* **1275** (data set 5, all data sets in RAM only!).

If object 0x60F8/0 was written and then a save parameters command (object 0x1010) processed, the object value is stored in non-volatile memory.

After the next power on of the inverter, the previously set value is reactivated and overwrites the setting of parameter *Max Slippage* **1275**.

The dimension of the user units is defined by 0x6091 *Gear ratio* and 0x6092 *Feed constant*.

9.9.5.42 0x60FF/0 Target Velocity [u/s]

Index	Sub-index	Meaning	Data type	Access	Map	Def.-Val
0x60F8	0	Target Velocity	Integer32	rw	Rx	0

Object works in:	Object does not work in:
Motion Control: ○ Profile Velocity mode	Motion Control: ○ Profile Positioning mode ○ **Velocity mode** ○ **Homing mode** ○ **Table Travel record mode** ○ **Move away from Limit Switch** ○ **Electronic Gear: Slave** ○ **Non motion Control** (conf. ≠ x40)

Object 0x60FF *Target Velocity* defines the reference velocity in Profile velocity mode.

9.9.5.43 0x6502/0 Supported drive modes

Index	Sub-Index	Meaning	Data Type	Access	Map	Def.-Val
0x6502	0	Supported drive modes	Unsigned32	ro	No	

Object 0x6502 *Supported drive modes* displays the supported 0x6060 Modes of Operation.

The value 0x0000 0202 means:

- Bit 0: "Profile Positioning mode" is supported.
- Bit 1: "Velocity mode" is supported.
- Bit 2: "Profile Velocity mode" is supported.
- Bit 5: "Homing mode" is supported.
- Bit 6: "Interpolated Position mode" is **not supported**.
- Bit 7: "Cyclic Synchronous Positioning mode" is **not supported**.
- Bit 8: "Cyclic Synchronous Velocity mode" is **not supported**.

- Bit 16: Manufacturer mode -1 "Table Travel record mode" is supported.
- Bit 17: Manufacturer mode -2 "Move away from limit switch" is supported.
- Bit 18: Manufacturer mode -3 "Electronic Gear: Slave" is supported.

10 Systembus

This chapter describes the usage of Systembus on the CAN interface.

10.1Baud rate setting/line lengths

The Baud rate settings must be the same in all subscribers. The maximum Baud rate depends on the necessary total cable length of the systembus. The Baud rate is set up via parameter *Baud-Rate* **903**and defines the available cable length.

Operation mode		Function	max. line length
3 -	50 kBaud	Transmission rate 50 kBaud	1000 meters
4 -	100 kBaud	Transmission rate 100 kBaud	800 meters
5 -	125 kBaud	Transmission rate 125 kBaud	500 meters
6 -	250 kBaud	Transmission rate 250 kBaud	250 meters
7 -	500 kBaud	Transmission rate 500 kBaud	100 meters
8 -	1000 kBaud	Transmission rate 1000 kBaud	25 meters

A baud rate under 50 kBaud, as defined according to CANopen, is not sensible for the systembus as the data throughput is too low. The maximum line lengths stated are guidelines. Depending on the number of subscribers, the baud rate is limited. There are the following restrictions:

Up to and including	250 kBit/s:	not more than 64 subscribers
	500 kBit/s:	not more than 28 subscribers
	1000 kBit/s:	not more than 10 subscribers

The bus load must be considered in the projecting phase.

10.2Setting the node address

A maximum of 63 slaves or frequency inverters with systembus can be operated on the systembus. Each frequency inverter is given a node ID, which may only exist once in the system, for its unambiguous identification. The setting of the systembus node ID is done via the parameter *Node-ID* **900**.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
900	Node-ID	-1	63	-1

Thus, the systembus possesses a maximum number of 63 subscribers (Network nodes), plus one frequency inverter as a master.

With the factory setting of parameter *Node-ID* **900** = -1, the systembus is deactivated for this frequency inverter.

If *Node-ID* **900** = 0 is set, the frequency inverter is defined as the master. Only one frequency inverter on the systembus may be defined as the master.

10.3Functional overview

The systembus produces the physical connection between the frequency inverters. Logical communication channels are produced via this physical medium. These channels are defined via the identifiers. As CAN does not possess a subscriber-oriented, but a message-oriented addressing via the identifiers, the logical channels can be displayed via it.

In the basic state (factory setting) the identifiers are set according to the Predefined Connection Set of CANopen. These settings are aimed at one master serving all the channels. In order to be able to build up process data movement via the PDO channels between individual or a number of inverters (transverse movement), the setting of the identifiers in the subscribers has to be adapted.

The exchange of data is done message-oriented. A frequency inverter can transmit and receive a number of messages, identified via various identifiers.

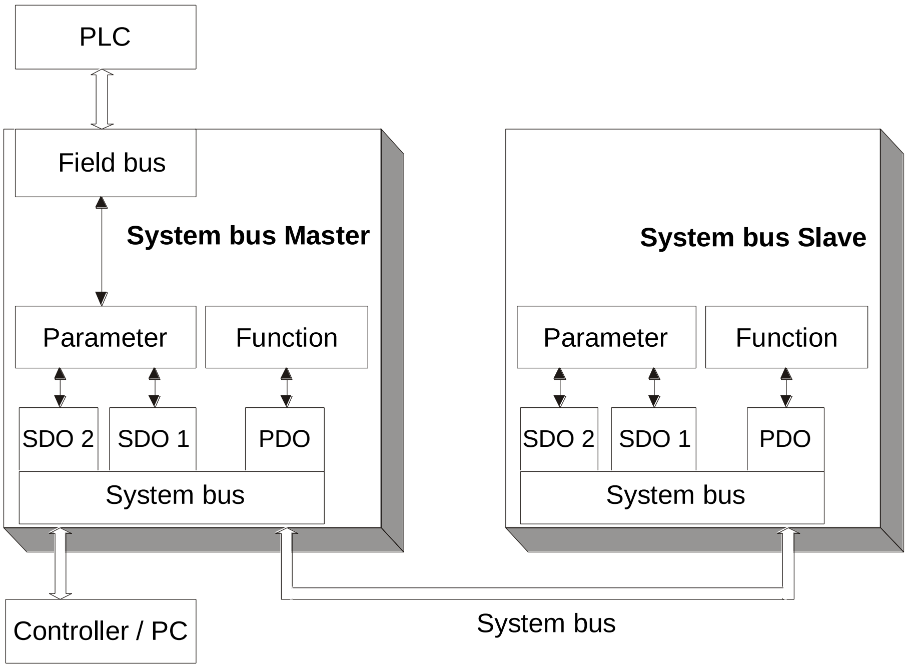
As a special feature, the properties of the CAN bus mean that the messages transmitted by one subscriber can be received by a number of subscribers simultaneously. The error monitoring methods of the CAN bus result in the message being rejected by all recipients and automatically transmitted again if there is a faulty reception in one receiver.

10.4Network management

The network management controls the start of all subscribers to the systembus. Subscribers can be started or stopped individually or jointly. For subscriber recognition in a CAL or CAN open system, the slaves on the systembus generate a starting telegram (boot-up report).

In the event of a fault, the slaves automatically transmit a fault report (emergency message).

For the functions of the network management, the methods and NMT telegrams (network management telegrams) defined according to CAN open (CiA DS 301) are used.



10.4.1 SDO channels (parameter data)

Each frequency inverter possesses two SDO channels for the exchange of parameter data. In a slave device, these are two server SDOs, in a device defined as a master a client SDO and a server SDO. Attention must be paid to the fact that only one master for each SDO channel may exist in a system.

Only one master can initiate by the systembus an exchange of data via its client SDO.

The identifier assignment for the SDO channels (Rx/Tx) is done according to the Predefined Connection Set.

This assignment can be amended by parameterization, in order to solve identifier conflicts in a larger system in which further devices are on the CAN bus alongside the frequency inverters.

NOTICE

If a system in which a frequency inverter works as a master is produced, the identifier allocations for the SDO channel may not be altered.

In this way, an addressing of individual subscribers via the field bus/systembus path of the master frequency inverter is possible.

Parameters are read/written via the SDO channels. With the limitation to the SDO Segment Protocol Expedited, which minimizes the requirements of the parameter exchange, the transmittable data are limited to the uint / int / long types. This permits complete parameterization of the frequency inverters via the systembus, as all the settings and practically all the actual values are displayed via these data types.

10.4.2 PDO channels (process data)

Each frequency inverter possesses three PDO channels (Rx/Tx) for the exchange of process data.

The identifier assignment for the PDO channel (Rx/Tx) is done by default according to the Predefined Connection Set. This assignment corresponds to an alignment to a central master control.

In order to produce the logical channels between the devices (transverse movement) on the systembus, the amendment of the PDO identifiers for Rx/Tx is necessary.

Each PDO channel can be operated with time or SYNC control. In this way, the operation behavior can be set for each PDO channel:

The setting of the operation mode is done via the following parameters:

- *TxPDO1 Function* **930**, *TxPDO2 Function* **932** and *TxPDO3 Function* **934**
- *RxPDO1 Function* **936**, *RxPDO2 Function* **937** and *RxPDO3 Function* **938**

Operation mode	Function
0 - disabled	no exchange of data via the PDO channel (Rx and/or Tx)
1 - time-controlled	Tx-PDOs cyclically transmit according to the time specification Rx-PDOs are read in with Ta = 1 ms and forward the data received to the application
2 - SYNC controlled	Tx-PDOs transmit the data from the application that are then current after the arrival of the SYNC telegram. Rx-PDOs forward the last data received to the application after the arrival of the SYNC telegram.

For synchronous PDOs, the master (PC, PLC or frequency inverter) generates the SYNC telegram. The identifier assignment for the SYNC telegram is done by default according to the Predefined Connection Set. This assignment can be altered by parameterization.

10.5Master functionality

An external control or a frequency inverter defined as a master (node ID = 0) can be used as a master. The fundamental tasks of the master are controlling the start of the network (boot-up sequence), generating the SYNC telegram and evaluating the emergency messages of the slaves.

Further, there can be access to the parameterization of all the frequency inverters on the systembus by means of a field bus connection via the client SDO of the master frequency inverter.

10.5.1 Control boot-up sequence, network management

The Minimum Capability Boot-Up method defined according to CANopen is used for the state control of the nodes.

This method knows the pre-operational, operational and stopped states.

After the initialization phase, all the subscribers are in the pre-operational state. The systembus master transmits the NMT command **Start-Remote-Node**. With this command, individual nodes or all the nodes can be started together. A frequency inverter defined as a master starts **all** the nodes with **one** command. After receipt of the Start Remote Node command, the subscribers change into the Operational state. From this time on, process data exchange via the PDO channels is activated.

A master in the form of a PLC/PC can start the subscribers on the systembus individually and also stop them again.

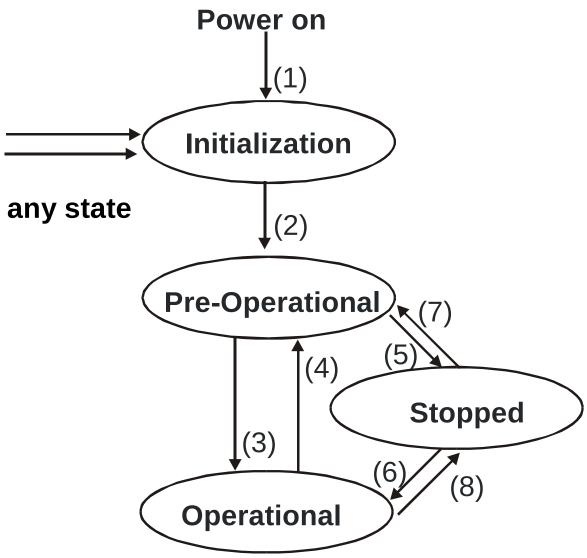
As the slaves on the systembus need different lengths of time to conclude their initialization phases (especially if external components exist alongside the frequency inverters), an adjustable delay for the change to Operational is necessary. The setting is done in a frequency inverter defined as a systembus master via *Boot-Up Delay* **904**.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
904	Boot-up delay	3500 ms	50000 ms	3500 ms

Properties of the states:

State	Properties
Pre-Operational	Parameterization via SDO channel possible Exchange of process data via PDO channel not possible
Operational	Parameterization via SDO channel possible Exchange of process data via PDO channel possible
Stopped	Parameterization via SDO channel not possible Exchange of process data via PDO channel not possible

Start-Remote-Node is cyclically transmitted with the set delay time by a frequency inverter defined as a systembus master, in order to put slaves added with a delay or temporarily separated from the network back into the Operational state.



After Power On and the initialization, the slaves are in the Pre-Operational state.

The transition (2) is automatic. The systembus master (frequency inverter or PLC/PC) triggers the transition (3) to Operational state.

The transitions are controlled via NMT telegrams.

The identifier used for the NMT telegrams is "0" and may only be used by the systembus master for NMT telegrams. The telegram contains two data bytes.

Byte 0	Byte 1
CS (Command Specifier)	Node-ID

Identifier = 0

With the statement of the node ID ≠ 0, the NMT command acts on the subscriber selected via the node ID. If node ID = 0, all the subscribers are addressed. If Node-ID = 0, all nodes are addressed.

Transition	Command	Command Specifier
(3) , (6)	Start Remote Node	1
(4) , (7)	Enter Pre-Operational	128

(5) , (8)	Stop Remote Node	2
-	Reset Node	129
-	Reset Communication	130

A frequency inverter defined as a systembus master only transmits the command "Start Remote Node" with node ID = 0 (for all subscribers). Transmission of the command is done after completion of the initialization phase and the time delay *Boot-Up Delay* **904** following it.

10.5.2 SYNC telegram, generation

If synchronous PDO's have been created on the systembus, the master must send the SYNC telegram cyclically. If a frequency inverter has been defined as a systembus master, the latter must generate the SYNC telegram. The interval for the SYNC telegram of a frequency inverter defined as the systembus master is adjustable. The SYNC telegram is a telegram without data.

The default identifier = 128 according to the Predefined Connection Set.

If a PC or PLC is used as a master, the identifier of the SYNC telegrams can be adapted by parameterization on the frequency inverter.

The identifier of the SYNC telegram must be set identically in all clients on the systembus.

The setting of the identifier of the SYNC telegram is done via parameter *SYNC-Identifier* **918**.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
918	SYNC identifier	0	2047	0

The setting "`0`" results in identifier assignment according to the Predefined Connection Set.

NOTICE

The identifier range `129...191` may not be used as the emergency telegrams can be found there.

The temporal cycle for the SYNCH telegram is set on a frequency inverter defined as the systembus master via parameter *SYNC-Time* **919**.

A setting of 0 ms for the parameter *SYNC-Time* **919** means "no SYNC telegram".

10.5.3 Emergency message, reaction

If a slave on the systembus suffers a fault, it transmits the emergency telegram. The emergency telegram marks the node ID for the identification of the failed node via its identifier and the existing fault message via its data contents (8 bytes).

After a fault has been acknowledged on the slave, the latter again transmits an emergency telegram with the data content zero.

The emergency telegram has the identifier 128 + node ID (= `129 ... 191`)

The systembus master evaluates the emergency telegrams of the slaves. Its reaction to an emergency telegram can be set with *Emergency Reaction* **989**.

Operation mode		Function
0 -	Error	The systembus master receives the emergency telegram and switches-off.
1 -	No Error	The Emergency Telegram is displayed as a warning.
2 -	Ignore	The Emergency Telegram is ignored.

Operation mode - parameter 989 = 0 – Error

Behavior of the systembus master in the case of *Emergency Reaction* **989** = `0 - Error`:

As soon as the systembus master receives an emergency telegram, it also switches to failure mode and reports the failed subscriber on the basis of its ID via the kind of error. Only the subscriber is reported, not the cause of the error.

The fault message on the systembus master via *Type of error* **260** is `21nn` with `nn` **=** node ID (hexadecimal) of the slave where a fault shutdown has occurred.

In addition, the systembus master reports the warning Sysbus (0x2000) via *Warning Status* **270** Bit 13.

If a fault shutdown occurs on a number of slaves, the first slave to transmit its emergency telegram is displayed on the systembus master.

Operation mode - parameter 989 = 1 – No Error

Behavior of systembus master in the case of *Emergency Reaction* **989** = 1 / No Error:

As soon as the systembus master receives an emergency telegram, it reports the warning Sysbus (0x2000) via *Warning status* **270** Bit 13.

In both cases, the Boolean variable *SysbusEmergency* with source number **730** is set to `TRUE` in the systembus master. It can be used in the systembus master and (in transmission via a TxPDO) in the slaves for a defined shutdown.

SysbusEmergency is also set if the systembus master breaks down.

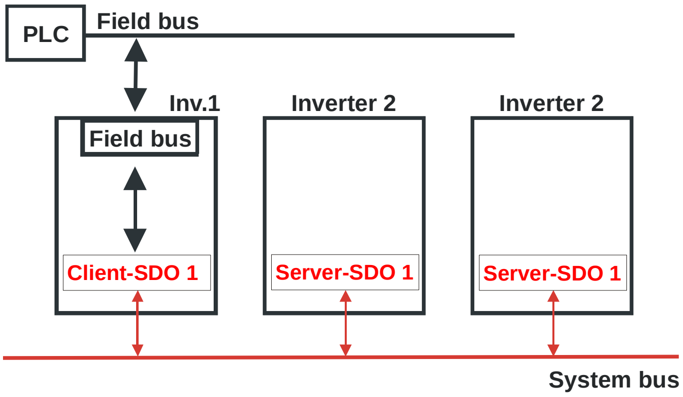
Resetting of *SysbusEmergency* is done with the fault acknowledgment.

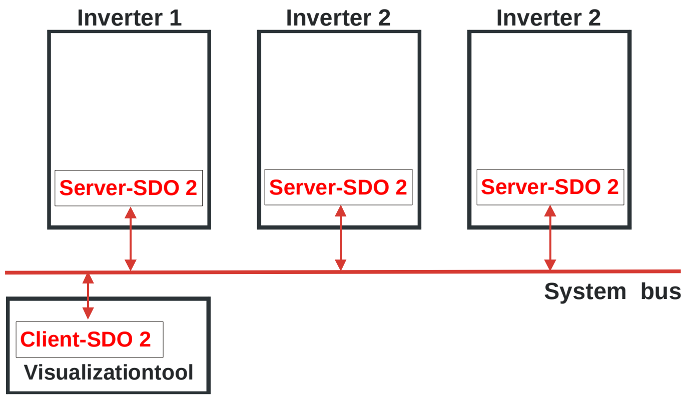
10.5.4 Client SDO (systembus master)

Each subscriber on the systembus can be addressed via the SDO channels. In this way, each subscriber can be addressed and parameterized by one master via its client SDO1. All the parameters of the data types uint/int/long are accessible. String parameters **cannot** be processed. If a frequency inverter has been defined as a systembus master, each subscriber on the systembus in this frequency inverter can be addressed by means of a field bus connection (RS232, RS485, Profibus-DP) via its client SDO1.

The second SDO channel SDO2 of the frequency inverters is planned for the parameterization of the frequency inverters via a visualization tool on the systembus.

The service used is SDO Segment Protocol Expedited according to CANopen. A frequency inverter defined as a systembus master automatically generates the correct telegrams. If the SDO channel is operated via a PLC/PC on the systembus, the telegrams must be generated according to the specification.





10.6Slave functionality

10.6.1 Implement boot-up sequence, network management

10.6.1.1 Boot-up message

After the initialization, each slave on the systembus transmits its boot-up message (heartbeat message).

The boot-up telegram has the identifier `1792 + node ID` and a data byte with contents `= 0x00`.

This telegram is irrelevant if a PLC/PC with CANopen functionality is used as a master. A frequency inverter defined as a systembus master **does not** evaluate the boot-up message.

10.6.1.2 State control

The identifier used for the NMT telegrams is "`0`" and may only be used by the systembus master for NMT telegrams. The telegram contains two data bytes.

Byte 0	Byte 1
CS (Command Specifier)	Node-ID

Identifier = 0

With the statement of the node ID ≠ 0, the NMT command acts on the subscriber selected via the node ID. If node ID = 0, all the subscribers are addressed. If Node-ID = 0, **all** subscribers are addressed.

Transition	Command	Command Specifier
(3),(6)	Start Remote Node	1
(4),(7)	Enter Pre-Operational	128
(5),(8)	Stop Remote Node	2
-	Reset Node	129
-	Reset Communication	130

After a slave has received the command "Start Remote Node", it activates the PDO channels and is ready for the exchange of process data.

The reset node and reset communication command specified according to DS 301 lead to a change to Pre-Operational via Initialization in the frequency inverters. There is a new boot-up message.

10.6.2 Process SYNC telegram

If synchronous PDO's have been created in a frequency inverter, their processing is synchronized with the SYNC telegram. The Sync event can either by a SYNC telegram or a RxPDO telegram and is set up via **1180** *Operation mode* synchronization.

The SYNC telegram is generated by the systembus master and is a telegram without data or 1 byte data. The data byte is ignored.

The identifier is 128 according to the Predefined Connection Set.

If a PC or PLC is used as a master, the identifier of the SYNC telegrams can be adapted by parameterization on the frequency inverter. The identifier of the SYNC telegram must be set identically in all clients on the systembus.

NOTICE

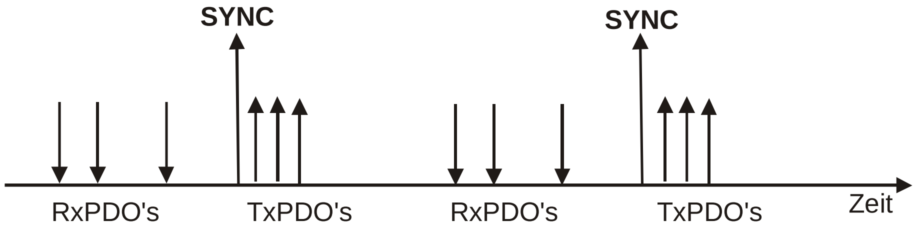
The identifier range `129...191` may not be used as this range is used for the emergency telegrams.

The setting of the identifier of the SYNC telegram is done via parameter *SYNC-Identifier* **918**.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
918	SYNC identifier	0	2047	0

The setting "0" results in identifier assignment according to the Predefined Connection Set.

The data of the Rx-PDO's are forwarded to the application after the arrival of the SYNC telegram. At the same time, the Tx-PDO's with the currently available data from the application are sent.



This method enables pre-occupancy of set points in the systembus subscribers and a synchronous / parallel take-over of the data.

10.6.3 Emergency-Message, fault shutdown

As soon as a fault shutdown occurs in a slave frequency inverter, the emergency telegram is transmitted. The emergency telegram marks the node ID for the identification of the failed node via its identifier and the existing fault message via its data contents (8 bytes).

The emergency telegram has the identifier 128 + node ID.

After a fault acknowledgment, another emergency telegram is transmitted, with the data content (Byte `0...7`) being set to "`0`" this time. This identifies the subscriber's repeated readiness for operation. If a further fault occurs subsequently, it is transmitted in a new emergency telegram.

The acknowledgment sequence is based on the definitions according to CANopen.

Data contents of the emergency telegram:

Emergency telegram		
Byte	**Value**	**Meaning**
0	0x00	low-byte error code
1	0x10	high-byte error code
2	0x80	Error register
3	0x00	-
4	0x00	-
5	0x00	-
6	0xnn	internal error code, low-byte
7	0xmm	internal error code, high-byte

Bytes `0, 1` and `2` are firmly defined and compatible with CANopen.

Bytes `6/7` contain the product specific VECTRON error code.

Error code	= 0x1000	= general error
Error register	= 0x80	= manufacturer-dependent error

The explanation and description of the product-specific VECTRON error code can be found in the annex "Error messages".

10.6.4 Server-SDO1/SDO2

The communication channel for the exchange of parameter data is the SDO channel. Communication works according to the client/server model. The server is the subscriber holding the data (here the frequency inverter), the client the subscriber requesting or wanting to alter the data (PLC, PC or frequency inverter as systembus master).

For the frequency inverter, two server SDO channels have been implemented.

The first SDO channel **SDO1** is used for the parameterization of the PLC/PC as a master or frequency inverter with field bus connection as a systembus master.

The second SDO channel **SDO2** is reserved for a visualization tool for parameterization. An exchange of data can only be implemented by the master via a client SDO.

The SDO channels are stipulated for the server SDO's via identifiers according to the Predefined Connection Set to CANopen. As CANopen only provides for and defines one SDO channel in the Predefined Connection Set, the second SDO channel can be deactivated.

In addition, the number of systembus subscribers and the adjustable node ID are limited to 63.

Identifier assignment according to the Predefined Connection Set:

Identifier Rx-SDO = 1536 + Node-ID	(Node ID = 1 ... 127, Identifier = 1537 ... 1663)
Identifier Tx-SDO = 1408 + Node ID	(Node ID = 1 ... 127, Identifier = 1409 ... 1535)

Identifier assignment for SDO1/SDO2 compatible with the Predefined Connection Set:

Identifier Rx-SDO1 = 1536 + Node ID	(Node ID = 1 ... 63, Identifier = 1537 ... 1599)
Identifier Tx-SDO1 = 1408 + Node ID	(Node ID = 1 ... 63, Identifier = 1409 ... 1471)
Identifier Rx-SDO2 = 1600 + Node ID	(Node ID = 0 ... 63, Identifier = 1600 ... 1663)
Identifier Tx-SDO2 = 1472 + Node ID	(Node ID = 0 ... 63, Identifier = 1472 ... 1535)

This corresponds to the factory settings of the frequency inverters for the SDO's.

The node ID = 0 for SDO2 is the systembus master.

NOTICE

The SDO2 must be deactivated in a CANopen system in order not to generate any compatibility problems.

If a frequency inverter has been defined as the systembus master, the above settings for the SDO1 must be maintained in all the frequency inverters. In this way, access to the parameterization of the frequency inverters via a field bus connection on the master frequency inverter is possible.

The client SDO1 in the master frequency inverter addresses the server SDO1 of the slaves via the above identifiers.

The identifiers for a visualization tool on the second SDO channel SDO2 cannot be changed.

If a PC or a PLC is used as a master, the identifiers of the **Rx/Tx-SDO1** can be adapted by parameterization on the frequency inverter.

NOTICE

Identifiers may only be assigned once, i.e. no double assignments.

The identifier range 129...191 may not be used as the emergency telegrams can be found there.

The setting of the identifiers of the RxSDO1 is done via the parameter *RxSDO1-Identifier* **921**.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
921	RxSDO1 identifier	0	2047	0

The setting of the identifiers of the TxSDO1 is done via parameter number **922**.

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
922	TxSDO1 identifier	0	2047	0

The setting “0” results in identifier assignment according to the Predefined Connection Set.

The second SDO channel can be deactivated via the *SDO2 Set Active* **923**.

Operation mode		Function
0 -	SDO2 deactivated	Communication channel deactivated
1 -	SDO2 activated	Communication channel activated for the visualization tool

The identifier assignment for the second SDO channel is always to the specification:

Identifier Rx-SDO2	= 1600 + Node ID
Identifier Tx-SDO2	= 1472 + Node ID

In this way, firm identifiers via which communication takes place are available for the visualization tool.

10.7Communication channels, SDO1/SDO2

10.7.1 SDO telegram (SDO1/SDO2)

The service used for the exchange of parameter data is **SDO Segment Protocol Expedited**. The data (type uint, int, long) are exchanged in a telegram.

Access to the parameters in the frequency inverters with a statement of parameter number and data set is displayed via the addressing defined for object access pursuant to the specifications of CANopen via Index/Sub-Index.

Index = parameter number / Sub index = data set.

The data to be transmitted have a length of 2 bytes for uint/int and 4 Bytes for long. For simplification and standardization, 4 bytes are always transmitted.

The data are on bytes 4...7 of the SDO telegram.

- uint/int variables are transmitted in bytes 4 and 5 with bytes 6 und 7 = 0.
- long variables are transmitted in bytes 4...7.

Writing parameters:

Client ➔ Server / SDO Download (expedited)

0	**1**	**2**	**3**	**4**	**5**	**6**	**7**
Control byte	Parameter number		Data Set	Data			
0x22	LSB	MSB	0xnn	LSB			MSB
uint/int				LSB	MSB	0x00	0x00
long				LSB	...	...	MSB

Server ➔ Client / Download Response ➔ writing process free of errors

0	1	2	3	4	5	6	7
Control byte	Parameter number		Data Set	Data			
0x60	LSB	MSB	0xnn	0			

Server ➔ Client / Abort SDO Transfer ➔ writing process with error

0	1	2	3	4	5	6	7
Control byte	Parameter number		Data Set	Data			
0x80	LSB	MSB	0xnn	Code	0	0	0

The error code is stated in byte 4 in a faulty reading process. (See table, failure codes).

Control byte 0x22 for the identification "SDO Download expedited" does not consider the bits "s" (data size indicated) and "n" (number of bytes not containing data). If set, they are ignored. The user is responsible for the number of bytes matching the type of data.

Reading parameters:

Client ➔ Server / SDO Upload (expedited)

0	1	2	3	4	5	6	7
Control byte	Parameter number		Data Set	Data			
0x40	LSB	MSB	0xnn	0			

Server ➔ Client /Upload Response ➔ reading process without errors

0	1	2	3	4	5	6	7
Control byte	Parameter number		Data Set	Data			
0x42	LSB	MSB	0xnn	LSB			MSB
uint/int				LSB	MSB	0x00	0x00
Long				LSB	...	...	MSB

Server ➔ Client / Abort SDO Transfer ➔ reading process faulty

0	1	2	3	4	5	6	7
Control byte	Parameter number		Data Set	Data			
0x80	LSB	MSB	0xnn	Code	0	0	0

The error code is stated in byte 4 in a faulty reading process. (See table, failure codes).

Error codes	
Code	**Description**
1	inadmissible parameter value
2	inadmissible data set
3	Parameter not readable
4	Parameter not writeable
5	read error EEPROM
6	write error EEPROM
7	checksum error EEPROM
8	parameter cannot be written while the drive is running
9	values of the data sets differ from one another
10	wrong parameter type
11	unknown parameter
12	BCC error in VECTRON bus protocol
15	unknown error
20	systembus subscriber not available only in access via field bus connection
21	string parameter not admissible only in access via VECTRON bus protocol

Errors marked in the table are generated by the field bus side, not in the Abort SDO Transfer of the systembus.

10.8Process data channels, PDO

This chapter describes the PDO usage of Systembus.

10.8.1 Identifier assignment process data channel

The process channel for the exchange of process data under CANopen and Systembus is the PDO channel. Up to three PDO channels with differing properties can be used in one device.

The PDO channels are defined via identifiers according to the Predefined Connection Set to CANopen:

Identifier 1. Rx-PDO =	512 + Node ID
Identifier 1. Tx-PDO =	384 + Node ID
Identifier 2. Rx-PDO =	768 + Node ID
Identifier 2. Tx-PDO =	640 + Node ID
Identifier 3. Rx-PDO =	1024 + Node ID
Identifier 3. Tx-PDO =	896 + Node ID

This corresponds to the factory settings of the frequency inverters for the Rx/Tx-PDO‘s. This occupancy is aligned to an external master (PLC/PC) serving all the channels.

If the PDO channels are used for a connection of the frequency inverters amongst one another, the identifiers are to be set accordingly by parameterization.

NOTICE

Identifiers may only be assigned once, i.e. no double assignments.

The identifier range `129...191` may not be used as the emergency telegrams can be found there.

Setting of the identifiers of the Rx/TxPDOs:

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
924	RxPDO1 Identifier	0	2047	0
925	TxPDO1 Identifier	0	2047	0
926	RxPDO2 Identifier	0	2047	0
927	TxPDO2 Identifier	0	2047	0
928	RxPDO3 Identifier	0	2047	0
929	TxPDO3 Identifier	0	2047	0

The setting "`0`" results in identifier assignment according to the Predefined Connection Set.

10.8.2 Operation modes process data channel

The sending/receiving behavior can be time-controlled or controlled via a SYNC telegram. The behavior can be parameterized for each PDO channel.

Tx-PDOs can work time-controlled or SYNC-controlled. Time-controlled TxPDO sends its data at the set time intervals. A SYNC-controlled TxPDO will send its data once a SYNC-telegram is received.

RxPDOs in the time controlled setting forward the received data to the application immediately. If an RxPDO has been defined as SYNC controlled, it forwards its received data to the application after the arrival of a SYNC telegram.

Settings TxPDO1/2/3

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
931	TxPDO1 Time	1 ms	50000 ms	8 ms
933	TxPDO2 Time	1 ms	50000 ms	8 ms
935	TxPDO3 Time	1 ms	50000 ms	8 ms

The setting of the operation mode is done via the following parameters:

TxPDO1 Function **930**, *TxPDO2 Function* **932** and *TxPDO3 Function* **934**

Operation mode		Function
0 -	Not Active	No data are sent.
1 -	Controlled by time	In the cycle of the adjusted time interval the data are sent.
2 -	Controlled by SYNC	To arrival of a SYNC telegram the data are sent.

Settings RxPDO1/2/3

The setting of the operation mode is done via the following parameters:

RxPDO1 Function **936**, *RxPDO2 Function* **937** and *RxPDO3 Function* **938**

Operation mode		Function
0 -	Controlled by time	The received data are passed on immediately.
1 -	Controlled by SYNC	After arrival of a SYNC telegram the received data are passed on

In the "controlled by time" operation mode, there is a polling of the received data with the trigger cycle of Ta = 1 ms.

10.8.3 Timeout monitoring process data channel

Each frequency inverter monitors its received data for whether they are updated within a defined time window.

The monitoring is done onto the SYNC telegram and the RxPDO channels.

Monitoring SYNC / RxPDOs

Parameters		Settings		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
939	SYNC timeout	0 ms	60000 ms	0 ms
941	RxPDO1 Timeout	0 ms	60000 ms	0 ms
942	RxPDO2 Timeout	0 ms	60000 ms	0 ms
945	RxPDO3 Timeout	0 ms	60000 ms	0 ms

Setting "0" means no timeout monitoring.

There is only monitoring for the SYNC telegram if at least one RxPDO or one TxPDO channel is defined as SYNC controlled.

If a timeout period is exceeded, the frequency inverter switches to failure mode and reports one of the faults:

F2200	**Systembus Timeout SYNC**
F2201	**Systembus Timeout RxPDO1**
F2202	**Systembus Timeout RxPDO2**
F2203	**Systembus Timeout RxPDO3**

10.8.4 Communication relationships of the process data channels

Regardless of the process data to be transmitted, the communication relationships of the process data channels must be defined. The connection of PDO channels is done via the assignment of the identifiers. The identifiers of Rx-/Tx-PDO must match in each case.

Generally, there are two possibilities:

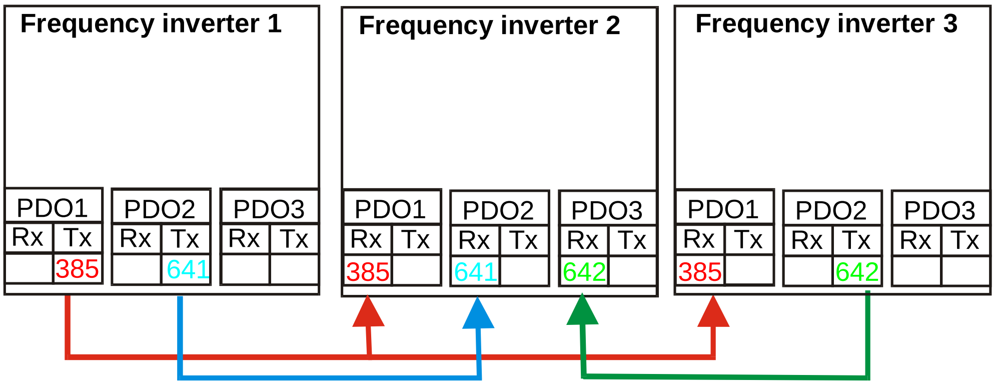
- **one** Rx-PDO to **one** Tx-PDO (one to one)
- connect **several** Rx-PDO's to **one** TxPDO (one to many)

This process is documented in a tabular form via a **communication relationship list**.

Example:

Frequency inverter 1		Frequency inverter 2		Frequency inverter 3	
PDO	**Identifier**	**PDO**	**Identifier**	**PDO**	**Identifier**
TxPDO1	**385**	TxPDO1		TxPDO1	
RxPDO1		**RxPDO1**	**385**	**RxPDO1**	**385**
TxPDO2	**641**	TxPDO2		**TxPDO2**	**642**
RxPDO2		**RxPDO2**	**641**	RxPDO2	
TxPDO3		TxPDO3		TxPDO3	
RxPDO3		**RxPDO3**	**642**	RxPDO3	

All the TxPDOs used must have unique identifiers!!!
The Identifier must be unique in the systembus network.



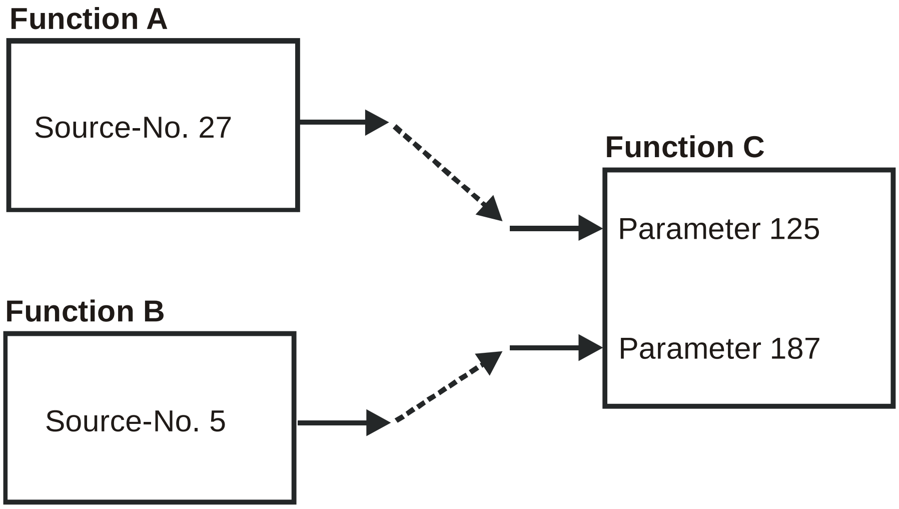
10.8.5 Virtual links

A PDO telegram contains 0 ...8 data bytes according to CANopen. A mapping for any kind of objects can be done in these data bytes.

For the systembus, the PDO telegrams are firmly defined with 8 data bytes. The mapping is not done via mapping parameters as with CANopen, but via the method of sources and links.

Each function provides its output data via a source. These sources are defined via source numbers. The input data of functions are defined via parameters. The link of a data input to a data output is done via the assignment of parameters to source numbers.

Example 1:



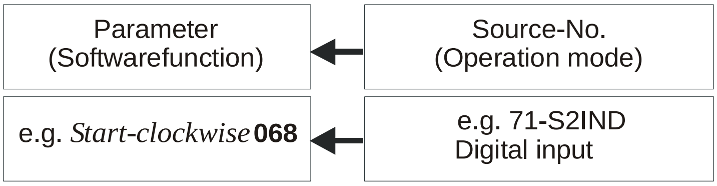
In example 1, the two inputs of function C are linked to the outputs of the functions A and B. The parameterization for this connection is thus:

Function C

Parameter 125 = Source-No. 27

Parameter 187 = Source-No. 5

Example of a virtual connection in VPlus:

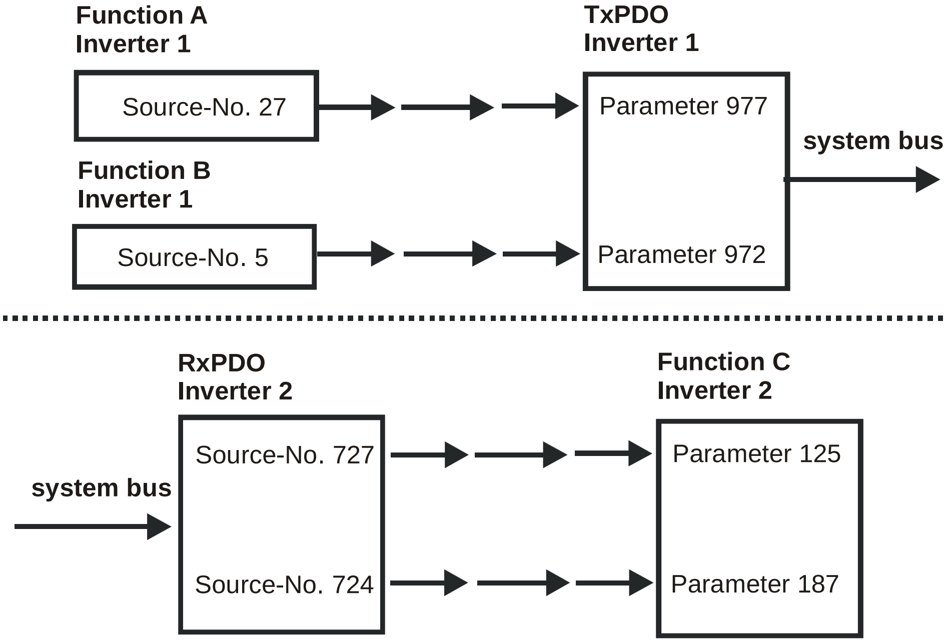


The assignment of the operation modes to the software functions available can be adapted to the application in question.

For more information on VPlus, see chapter 17.

For the systembus, the input data of the TxPDOs are also displayed as input parameters and the output data of the RxPDOs as sources.

Example 2:



Example 2 displays the same situation as Example 1. But now, the functions A and B are in frequency inverter 1 and function C in frequency inverter 2. The connection is done via a TxPDO in frequency inverter 1 and a RxPDO in frequency inverter 2. Thus, the parameterization for this connection is:

Frequency inverter 1

Parameter 977 = Source-No. 27

Parameter 972 = Source-No. 5

Frequency inverter 2

Parameter 125 = Source-No. 727

Parameter 187 = Source-No. 724

As the links with the system used exceed the device limits, they are termed "virtual links".

The virtual links with the possible sources are related to the Rx/TxPDO channels. For this purpose, the eight bytes of the Rx-/TxPDOs are defined structured as inputs and sources. This exists for each of the three PDO channels.

Each transmit PDO and receive PDO can be occupied as follows:

4 Boolean variables

or

4 uint/int variables

or

2 long variables

or

a mixture paying attention to the eight bytes available

Assignment data type / number of bytes:

Assignment	
Data type	**Length**
Boolean	2 Bytes
uint/int	2 Bytes
long	4 Bytes

10.8.5.1 Input parameters of the TxPDOs for data to be transmitted

The listed parameters can be used for determining the data that are to be transported there for each position in the TxPDO telegrams. The setting is done in such a way that a source number is entered for the required data in the parameters.

TxPDO1	**P. No.**	**TxPDO1**	**P. No.**	**TxPDO1**	**P. No.**
Byte	**Boolean input**	**Byte**	**uint/int input**	**Byte**	**long input**
0	**946**	0	**950**	0	
1	Boolean1	1	Word1	1	**954**
2	**947**	2	**951**	2	
3	Boolean2	3	Word2	3	Long1
4	**948**	4	**952**	4	
5	Boolean3	5	Word3	5	**955**
6	**949**	6	**953**	6	
7	Boolean4	7	Word4	7	Long2

TxPDO2	**P. No.**	**TxPDO2**	**P. No.**	**TxPDO2**	**P. No.**
Byte	**Boolean input**	**Byte**	**uint/int input**	**Byte**	**long input**
0	**956**	0	**960**	0	
1	Boolean1	1	Word1	1	**964**
2	**957**	2	**961**	2	
3	Boolean2	3	Word2	3	Long1
4	**958**	4	**962**	4	
5	Boolean3	5	Word3	5	**965**
6	**959**	6	**963**	6	
7	Boolean4	7	Word4	7	Long2

TxPDO3	**P. No.**	**TxPDO3**	**P. No.**	**TxPDO3**	**P. No.**
Byte	**Boolean input**	**Byte**	**uint/int input**	**Byte**	**long input**
0	**966**	0	**972**	0	
1	Boolean1	1	Word1	1	**976**
2	**967**	2	**973**	2	

3	Boolean2	3	Word2	3	Long1
4	**968** Boolean3	4	**974** Word3	4	**977** Long2
5		5		5	
6	**969** Boolean4	6	**975** Word4	6	
7		7		7	

Depending on the selected data information the percentages values are displayed via the uint/int inputs.

With this method, there are up to three possibilities for a meaning of the contents of the individual bytes. Each byte may only be used for one possibility.

To ensure this, the processing of the input links is derived from the setting.

If an input link has been set to the fixed value of zero, it is **not** processed.

The settings for the fixed value zero are:

Source =	7 (FALSE)	for Boolean variables
Source =	9 (0)	for uint, int, long variables

This is, at the same time, the factory setting.

Examples Boolean source

Boolean source	
Source	**Data**
6	TRUE
7	FALSE
70	Contact input 1
71	Contact input 2
72	Contact input 3
161	Run signal
163	Reference value reached
164	Set frequency reached (P. 510)

Examples uint/int source

unit/int source	
Source	**Data**
9	0
63	Reference Percentage 1
64	Reference Percentage 2
52	Percentage MFE1
133	Output percentage ramp
137	Output reference percentage channel
138	Output actual percentage channel
740	Control word
741	Status word

Examples long source

long source	
Source	**Data**
9	0
0	Output frequency ramp
1	Fixed frequency 1
5	Reference line value
62	Output Frequency reference value channel
50	Reference Frequency MFE1

10.8.5.2 Source numbers of the RxPDOs for received data

Equivalent to the input links of the TxPDOs, the received data of the RxPDOs are displayed via sources or source numbers. The sources existing in this way can be used in the frequency inverter via the local input links for the data targets.

RxPDO1 Byte	Source no. Boolean value	RxPDO1 Byte	Source no. uint/int value	RxPDO1 Byte	Source no. long Value
0	**700** Boolean1	0	**704** Word1	0	**708** Long1
1		1		1	
2	**701** Boolean2	2	**705** Word2	2	
3		3		3	
4	**702** Boolean3	4	**706** Word3	4	**709** Long2
5		5		5	
6	**703** Boolean4	6	**707** Word4	6	
7		7		7	

RxPDO2 Byte	Source no. Boolean value	RxPDO2 Byte	Source no. uint/int value	RxPDO2 Byte	Source no. long value
0	**710** Boolean1	0	**714** Word1	0	**718** Long1
1		1		1	
2	**711** Boolean2	2	**715** Word2	2	
3		3		3	
4	**712** Boolean3	4	**716** Word3	4	**719** Long2
5		5		5	
6	**713** Boolean4	6	**717** Word4	6	
7		7		7	

RxPDO3 Byte	Source no. Boolean value	RxPDO3 Byte	Source no. uint/int value	RxPDO3 Byte	Source no. long value
0	**720** Boolean1	0	**724** Word1	0	**728** Long1
1		1		1	
2	**721** Boolean2	2	**725** Word2	2	
3		3		3	
4	**722** Boolean3	4	**726** Word3	4	**729** Long2
5		5		5	
6	**723** Boolean4	6	**727** Word4	6	
7		7		7	

With this method, there are up to three possibilities for a meaning of the contents of the individual bytes. Each byte may only be used for one possibility.

Depending on the selected data information the percentages values are displayed via the uint/int inputs.

10.8.5.3 Examples of virtual links

Example 1:

Frequency inverter 1				Frequency inverter 2		
Source no.	**Input link**	**TxPDO1 Byte**		**RxPDO1 Byte**	**Source - No.**	**Target**
Control word **740**	950	0	→	0	704	Control input, Control word **99**
		1		1		
		2		2		
		3		3		
Output reference	955	4	→	4	709	Ramp input,
		5		5		

frequency channel **62**		6		6		Line set value **137**
		7		7		

Parameter 950 = Source-No. 740

Parameter 955 = Source-No. 62

Parameter 99 = Source-No. 704

Parameter 137 = Source-No. 709

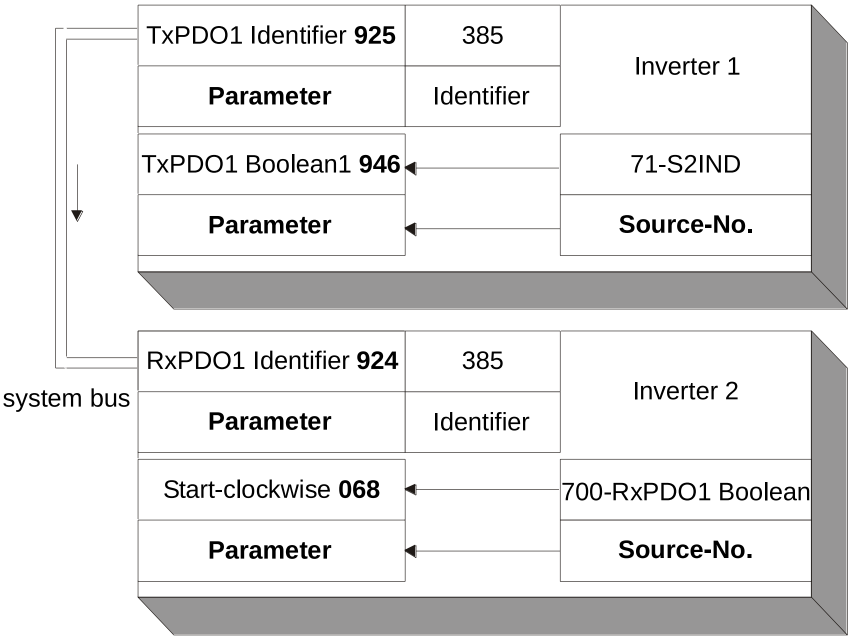
The control word of frequency inverter 1 is linked with the control word of frequency inverter 2. In this way, both frequency inverters can be operated synchronously via the remote control. The output of the reference value channel of frequency inverter 1 is laid onto the output of the ramp of frequency inverter 2. In this way, both frequency inverters have a joint source of reference values and are given reference values in the internal notation.

As an extension, a number of frequency inverters can also exist on the receive side (Rx), these then being supplied with data parallel and simultaneously.

The input link not used in the TxPDO1 of frequency inverter 1 is on ZERO and is thus not served.

Example 2:

Example of a virtual link with transmission via the systembus:



10.9Diagnosis parameters

For the monitoring of the systembus and the display of the internal states, two control parameters are provided. There is a report of the systembus state and a report of the CAN state via two actual value parameters.

The *Node State* **978** parameter gives information about the Pre-Operational, Operational, Stopped state. A PDO transfer is only possible in the Operational state. The state is controlled by the systembus master (PLC / PC / frequency inverter) via NMT telegrams.

The *CAN-State* **979** parameter gives information about the state of the physical layer. If there are transmission errors, the state changes from OKAY to WARNING until the cancellation of the communication with BUS-OFF. After BUS-OFF, the CAN controller is automatically re-initialized and the systembus started again.

If the BUS-OFF state occurs, the frequency inverter breaks down with "**F2210 BUS-OFF**".

After Bus-OFF, the systembus in the frequency inverter is completely reinitialized. There is a new boot-up message from the subscriber and an emergency telegram with the Bus-OFF message is transmitted. The change of state of the subscriber to Operational is done by the Start-Remote-Node telegram cyclically sent by the systembus master.

Actual values of the systembus

No.	Description	Display
978	Node state	1 - Pre-Operational 2 - Operational 3 - Stopped
979	CAN state	1 - OKAY 2 - WARNING 3 - BUS-OFF

10.10 Ancillaries

For the planning of the systembus according to the drive tasks in question, there are ancillaries in the form of tables.

The planning of the systembus is done in three steps:

1 Definition of the communication relationships

2 Production of the virtual links

3 Capacity planning of the systembus

The priority assignment of the identifiers is relevant for the definition of the communication relationships. Data that are to be transmitted with a higher priority must be given low identifiers. This results in the message with the higher priority being transmitted first with a simultaneous access of two subscribers to the bus.

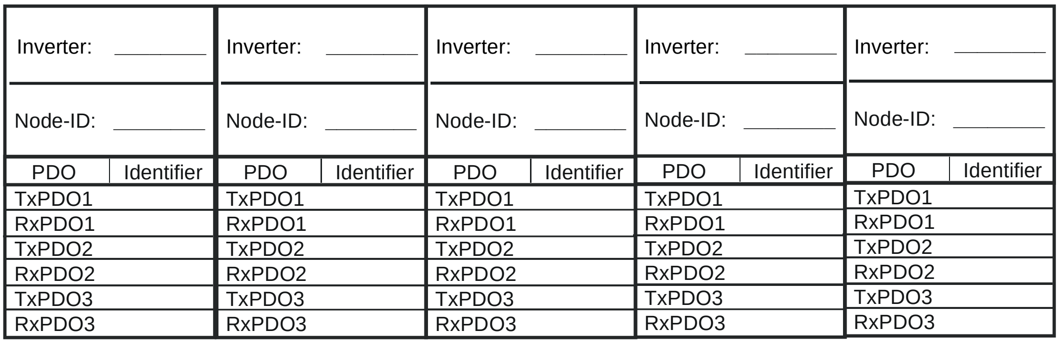
The recommended identifier range for the communication relationships via the PDO channels is 385 ...

The identifiers below 385 are used for the NMT telegrams (boot-up sequence, SYNC telegram) and emergency message.

The identifiers above 1407 are used for the SDO channel for parameterization.

10.10.1 Definition of the communication relationships

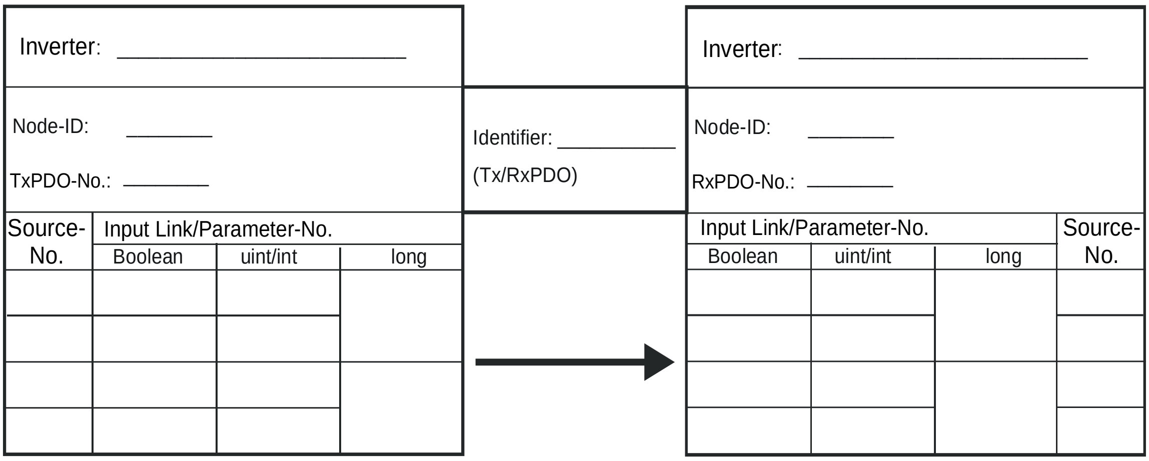
The communication relationships are planned and documented with the help of the table. The table is available as a Microsoft Word document “kbl.doc” upon request.



Inverter: ________		Inverter: ________		Inverter: ________		Inverter: ________		Inverter: ________	
Node-ID: ________		Node-ID: ________		Node-ID: ________		Node-ID: ________		Node-ID: ________	
PDO	Identifier	PDO	Identifier	PDO	Identifier	PDO	Identifier	PDO	Identifier
TxPDO1		TxPDO1		TxPDO1		TxPDO1		TxPDO1	
RxPDO1		RxPDO1		RxPDO1		RxPDO1		RxPDO1	
TxPDO2		TxPDO2		TxPDO2		TxPDO2		TxPDO2	
RxPDO2		RxPDO2		RxPDO2		RxPDO2		RxPDO2	
TxPDO3		TxPDO3		TxPDO3		TxPDO3		TxPDO3	
RxPDO3		RxPDO3		RxPDO3		RxPDO3		RxPDO3	

10.10.2 Production of the virtual links

The virtual links are planned and documented with the help of the table. The table is available as a Microsoft Word document “vvk.doc” upon request.



Inverter: ________________________

Node-ID: ________

TxPDO-No.: ________

Identifier: __________

(Tx/RxPDO)

Inverter: ________________________

Node-ID: ________

RxPDO-No.: ________

Source-No.	Input Link/Parameter-No.		
	Boolean	uint/int	long

Input Link/Parameter-No.			Source-No.
Boolean	uint/int	long	

10.10.3 Capacity planning of the systembus

Each PDO telegram possesses a constant useful data content of 8 Bytes. According to worst case, this results in a maximum telegram length of 140 bits. The maximum telegram run time of the PDOs is thus stipulated via the set baud rate.

Capacity planning

Baud rate [kBaud]	Telegram runtime [μs]
1000	140
500	280
250	560
125	1120
100	1400
50	2800

As a function of the set baud rate and the transmission interval of the TxPDOs selected, the following bus loads results:

Capacity of the systembus

Baud rate / kBaud	Bus load as a function of the transmission for one TxPDO in %									
	1ms	2ms	3ms	4ms	5ms	6ms	7ms	8ms	9ms	10ms
1.000	14	7	4.7	3.5	2.8	2.3	2	1.8	1.6	1.4
500	28	14	9.3	7	5.6	4.7	4	3.5	3.1	2.8
250	56	28	18.7	14	11.2	9.3	8	7	6.2	5.6
125	112	56	37.3	28	22.4	18.7	16	14	12.4	11.2
100	140	70	46.7	35	28	23.3	20	17.5	15.6	14
50	280	140	93.3	70	56	46.7	40	35	31.1	28

NOTICE

Network failure

A bus load >100% means that a telegram cannot be dispatched completely between two transmission times.

- Avoid using such a setting!

This observation must be done for each TxPDO. The sum of all the TxPDOs decides on the entire bus load. The bus load must be designed in such a way that any telegram repetitions for transmission errors are possible without exceeding the bus capacity.

To facilitate capacity planning, a Microsoft Excel file with the name "Load_Systembus.xls" is available.

The capacity planning are planned and documented with the help of the table. The work sheet is available as a Microsoft Excel document "Load_Systembus.xls" on request.

Load systembus			
Baud rate [kBaud]: 50, 100, 125, 250, 500, 1000			**1000**
Frequency inverter	**TxPDO Number**	**Ta [ms]**	**Workload [%]**
1	1	0	0
	2	0	0
	3	0	0
2	1	0	0
	2	0	0
	3	0	0
3	1	0	0
	2	0	0
	3	0	0
4	1	0	0
	2	0	0
	3	0	0
5	1	0	0
	2	0	0
	3	0	0
6	1	0	0
	2	0	0
	3	0	0
7	1	0	0
	2	0	0
	3	0	0
8	1	1	14
	2	1	14
	3	1	14
9	1	1	14
	2	1	14
	3	0	0
10	1	0	0
	2	0	0
	3	0	0
Total workload [%]			70

In the table, the set baud rate is entered from the parameter *Baud Rate* **903**in kBaud. For each frequency inverter, the set time for the transmission interval (e. g. *TxPDO1 Time* **931**) in ms is entered for the TxPDO being used at the time. In the column **Load** the bus load caused by the individual TxPDO appears, under **Total Load** the entire bus load.

For the bus load (Total load) the following limits have been defined:

≤ 80 %	→ **OKAY**
80 ... 90 %	→ **CRITICAL**
> 90 %	→ **NOT POSSIBLE**

11 Motion Control Interface (MCI)

The Motion Control Interface (MCI) is a defined interface of the ANG device for positioning control via Field bus. Typically this interface is used via a Field bus like CANopen. With the Motion Control Interface, the user can carry out a positioning operation via a field bus using a positioning profile typically including the target position, speed, acceleration, deceleration, quick stop and mode-specific information.

The Motion Control interface uses object 0x6060 *Modes of Operation* for switching between the different modes. The supported modes as per CANopen® Standard DS402 are:

1 – Profile Position mode

2 – Velocity mode [rpm]

3 – Profile Velocity mode [u/s]

6 – Homing

7 – Interpolated mode - "Interpolated position mode" requires equidistant timing for output/input assemblies, which is **not supported by Ethernet/IP**.

8 – Cyclic sync position mode - **not supported by Ethernet/IP**

9 – Cyclic sync velocity mode - **not supported by Ethernet/IP**

Bonfiglioli Vectron MDS specific modes

- -1 (or 0xFF) – Table Travel record mode
- -2 (or 0xFE) – Move Away from Limit Switch
- -3 (or 0xFD) – Electronic Gear: Slave (electronic gear as slave)

The actual mode is displayed in 0x6061 *Modes of Operation Display*.

The mode of operation can be switched in any operating state.

It is recommended that a currently active movement be stopped by the PLC first, then to switch 0x6060 Modes of Operation and restart in the new mode.

In order to use the Motion Control Interface, **412** *Local/Remote* = "1 - Control via statemachine" must be set. In configurations without Motion control (*Configuration* **30** ≠ x40), only velocity mode vl is available.

For a description of the positioning parameters please refer to the "Positioning Application manual".

11.1Object and parameter relationships

Depending on the selected object 0x6060 Modes of, various objects and parameters are used. The various objects and parameters must be set specifically for the different modes of operation. Use of "Deceleration" and "Quick Stop" depends on the modes of operation, control commands and behavior in the case of communication errors (see object 0x6007/0 *abort connection option code*).

The following tables provide an overview of the different objects and parameters. The object / parameter mentioned first in a cell will typically be used. If an object is related to a parameter, the parameter will be specified.

Parameters **1292** *Modes of Operation* and following *(***1293**, **1294**, **1295**, **1296** & **1297***)* and **1285** *S.Target velocity pv [u/s]* are used for linking the internal functions to CANopen® objects. Usually, these need not to be changed when using CANopen®.

Mode	Homing	Velocity Mode	Profile Velocity Mode
Modes of Operation 1)2)	**6**	**2**	**3**
Target Position			
Speed	Obj. 0x6099/1 & /2 Homing Speeds → **1132** & **1133**	**1297** *S.Target velocity* 2) Default: 806 - Obj. 0x6042 *vl* Target Velocity	**1285** *S.Target velocity pv [u/s]* 2)

Mode	Homing	Velocity Mode	Profile Velocity Mode
			Default: 816 - Obj. 0x60FF Target Velocity
Limitation[3]	Obj. 0x6046/1 & /2 *vl* velocity min max amount = **418 & 419**	Obj. 0x6046/1 & /2 *vl* velocity min max amount = **418 & 419**	Obj. 0x6046/1 & /2 *vl* velocity min max amount = **418 & 419**
Acceleration	Obj. 0x609A/0 Acceleration → **1134**	Obj. 0x6048/0 *vl* velocity acceleration = **420** (&422)	**1295**, *Acceleration* [2] Default :804 - Obj. 0x6083 Profile Acceleration
Deceleration	Obj. 0x609A/0 Acceleration → **1134**	Obj. 0x6049/0 *vl* velocity deceleration = **421** (& 423)	**1296**, *Deceleration* [2] Default : 805 - Obj. 0x6084 Profile Deceleration
Emergency stop [4] Quick Stop	Obj. 0x6085/0 Quick stop deceleration →**1179** *Emergency Ramp*	Obj. 0x604A/0 Velocity Quick Stop = **424** (& 425)	Obj. 0x6085/0 Quick stop deceleration →**1179** *Emergency Ramp*
Homing Method	Obj. 0x6098/0 Homing method →**1130**		

[1] Modes of Operation is set via **1292** *S.Modes of Operation.*

Default setting: **801** - Obj. 0x6060 Modes of Operation.

[2] Parameters **1285**, **1292**, **1293**, **1294**, **1295**, **1296** & **1297** are used for the connection between CANopen Objects and internal functions. For CANopen, these do not have to be changed.

[3] The limitation results from *Minimum frequency* **418** and *Maximum frequency* **419**. Through *Limitation* **1118** of the position controller in Configuration x40, an increase above the maximum frequency can occur, because the output of the position controller is added to the maximum frequency.

[4] Emergency stop or Deceleration is used depending on the stopping behavior *Mode of operation* **630** or the behavior in the case of communication errors 0x6007/0 *abort connection option code*.

Mode	Profile Positioning mode
Modes of Operation [1,2]	1
Target Position	**1293**, *S.Target Pos.* [2]) Default: 802 - Obj. 0x607A Target Position
Speed	**1294**, *S.Profile Vel.* [2] Default: 803 - Obj. 0x6081 Profile Velocity
Limitation[3]	Obj. 0x6046/1 & /2 Velocity min max amount = **418 & 419**
Acceleration	**1295**, *Acceleration* [2] Default :804 - Obj. 0x6083 Profile Acceleration
Deceleration	**1296**, *Deceleration* [2] Default : 805 - Obj. 0x6084 Profile Deceleration
Emergency stop [4] Quick Stop	Obj. 0x6085/0 Quick stop deceleration → **1179** *Emergency Ramp*
Homing Method	

[1] Modes of Operation is set via **1292** *S.Modes of Operation.*

Default setting: **801** - Obj. 0x6060 Modes of Operation.

[2] Parameters **1285**, **1292**, **1293**, **1294**, **1295**, **1296** & **1297** are used for the connection between CANopen Objects and internal functions. For CANopen, these do not have to be changed.

[3] The limitation results from *Minimum frequency* **418** and *Maximum frequency* **419**. Through *Limitation* **1118** of the position controller in Configuration x40, an increase above the maximum frequency can occur, because the output of the position controller is added to the maximum frequency.

[4] Emergency stop or Deceleration is used depending on the stopping behavior *Mode of operation* **630** or the behavior in the case of communication errors 0x6007/0 *abort connection option code*.

Mode	Table Travel Record mode	Move away from limit switches	Electronic Gear: Slave
Modes of Operation [1,2]	**-1**	**-2**	**-3**
Target Position	**1202** *Target Position*		

Mode	Table Travel Record mode	Move away from limit switches	Electronic Gear: Slave
Speed	**1203** *Target Speed*	Obj. 0x6099/1 & /2 Homing Speeds → **1132** & **1133**	**1285** *S.Target velocity pv [u/s]* [2)] Default: 816 - Obj. 0x60FF Target Velocity
Limitation[3)]	Obj. 0x6046/1 & /2 *v*/ velocity min max amount = **418 & 419**	Obj. 0x6046/1 & /2 *v*/ velocity min max amount = **418 & 419**	Obj. 0x6046/1 & /2 *v*/ velocity min max amount = **418 & 419**
Acceleration	**1204** *Acceleration*	Obj. 0x609A/0 Acceleration → **1134**	**1295**, *Acceleration* [2)] Default :804 - Obj. 0x6083 Profile Acceleration
Deceleration	**1205** *Deceleration*	Obj. 0x609A/0 Acceleration → **1134**	**1296**, *Deceleration* [2)] Default : 805 - Obj. 0x6084 Profile Deceleration
Emergency stop [4)] Quick Stop	Obj. 0x6085/0 Quick stop deceleration → **1179** *Emergency Ramp*	Obj. 0x6085/0 Quick stop deceleration →**1179** *Emergency Ramp*	Obj. 0x6085/0 Quick stop deceleration → **1179** *Emergency Ramp*
Motion Block	Selected via Control Word.		
Gear factor			**1123** *Gear Factor Numerator*; 0x5F10/1 Gearfactor Numerator **1124** *Gear Factor Denominator*; 0x5F10/2 Gearfactor Denominator
Phasing [5)]			**1125** *Phasing: Offset*; 0x5F11/1 Phasing 1 Offs. **1126** *Phasing: Speed*; 0x5F11/2 Phasing 1 Speed **1127** *Phasing: Acceleration* 0x5F11/3 Phasing 1 Acceleration

[1)] Modes of Operation is set via **1292** *S.Modes of Operation.*

Default setting: **801** - Obj. 0x6060 Modes of Operation.

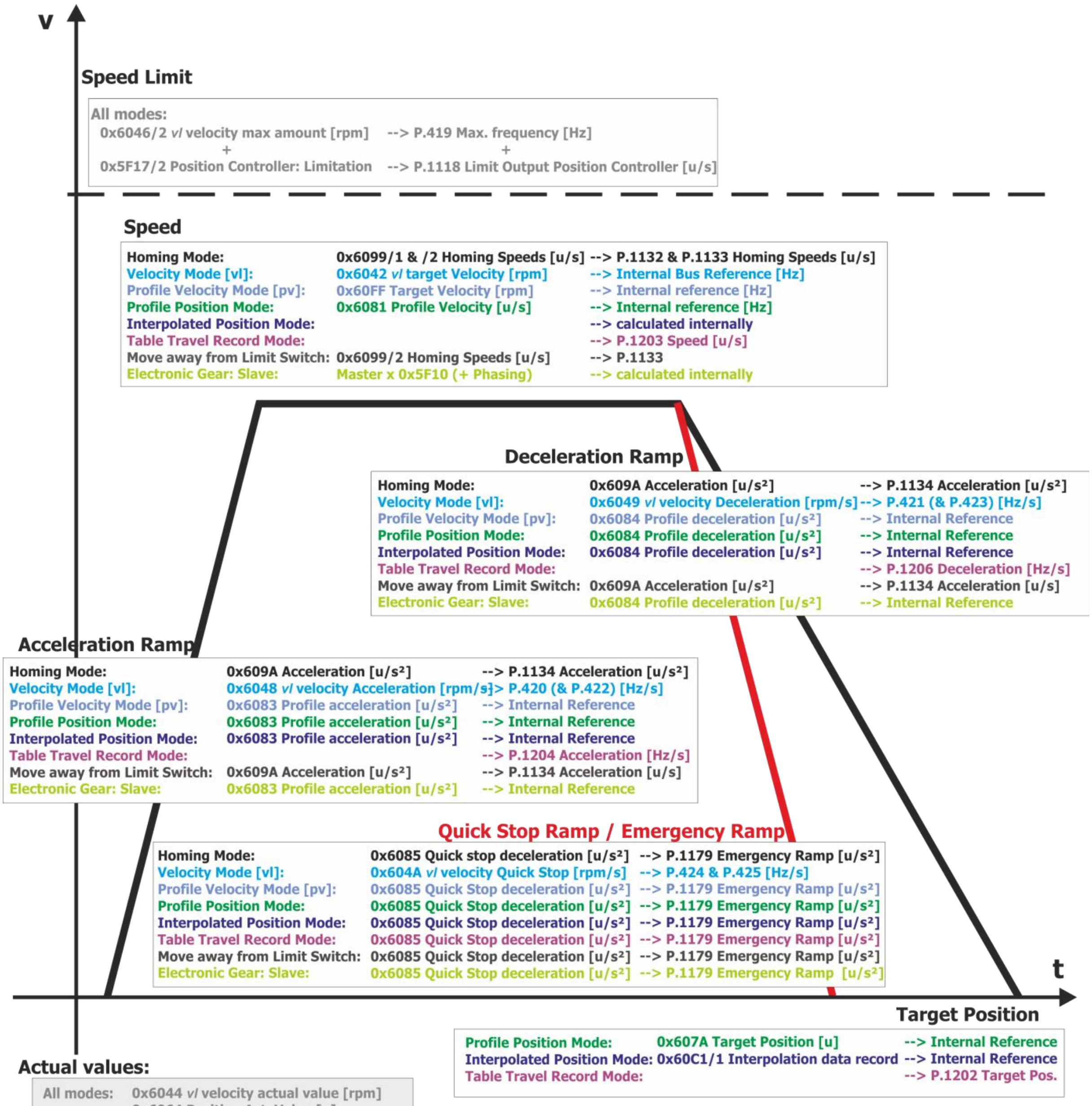
[2)] Parameters **1285**, **1292**, **1293**, **1294**, **1295**, **1296** & **1297** are used for the connection between CANopen Objects and internal functions. For CANopen, these do not have to be changed.

[3)] The limitation results from *Minimum frequency* **418** and *Maximum frequency* **419**. Through *Limitation* **1118** of the position controller in Configuration x40, an increase above the maximum frequency can occur, because the output of the position controller is added to the maximum frequency.

[4)] Emergency stop or Deceleration is used depending on the stopping behavior *Mode of operation* **630** or the behavior in the case of communication errors 0x6007/0 *abort connection option code*.
abort connection option code.

[5)] Phasing is available with 4 profiles in objects 0x5F11…0x5F14.

Relationships between objects, parameters and conversions



Speed Limit

All modes:
0x6046/2 *vl* velocity max amount [rpm] --> P.419 Max. frequency [Hz]
+ +
0x5F17/2 Position Controller: Limitation --> P.1118 Limit Output Position Controller [u/s]

Speed

Homing Mode:	0x6099/1 & /2 Homing Speeds [u/s]	--> P.1132 & P.1133 Homing Speeds [u/s]
Velocity Mode [vl]:	0x6042 *vl* target Velocity [rpm]	--> Internal Bus Reference [Hz]
Profile Velocity Mode [pv]:	0x60FF Target Velocity [rpm]	--> Internal reference [Hz]
Profile Position Mode:	0x6081 Profile Velocity [u/s]	--> Internal reference [Hz]
Interpolated Position Mode:		--> calculated internally
Table Travel Record Mode:		--> P.1203 Speed [u/s]
Move away from Limit Switch:	0x6099/2 Homing Speeds [u/s]	--> P.1133
Electronic Gear: Slave:	Master x 0x5F10 (+ Phasing)	--> calculated internally

Deceleration Ramp

Homing Mode:	0x609A Acceleration [u/s²]	--> P.1134 Acceleration [u/s²]
Velocity Mode [vl]:	0x6049 *vl* velocity Deceleration [rpm/s]	--> P.421 (& P.423) [Hz/s]
Profile Velocity Mode [pv]:	0x6084 Profile deceleration [u/s²]	--> Internal Reference
Profile Position Mode:	0x6084 Profile deceleration [u/s²]	--> Internal Reference
Interpolated Position Mode:	0x6084 Profile deceleration [u/s²]	--> Internal Reference
Table Travel Record Mode:		--> P.1206 Deceleration [Hz/s]
Move away from Limit Switch:	0x609A Acceleration [u/s²]	--> P.1134 Acceleration [u/s]
Electronic Gear: Slave:	0x6084 Profile deceleration [u/s²]	--> Internal Reference

Acceleration Ramp

Homing Mode:	0x609A Acceleration [u/s²]	--> P.1134 Acceleration [u/s²]
Velocity Mode [vl]:	0x6048 *vl* velocity Acceleration [rpm/s]	--> P.420 (& P.422) [Hz/s]
Profile Velocity Mode [pv]:	0x6083 Profile acceleration [u/s²]	--> Internal Reference
Profile Position Mode:	0x6083 Profile acceleration [u/s²]	--> Internal Reference
Interpolated Position Mode:	0x6083 Profile acceleration [u/s²]	--> Internal Reference
Table Travel Record Mode:		--> P.1204 Acceleration [Hz/s]
Move away from Limit Switch:	0x609A Acceleration [u/s²]	--> P.1134 Acceleration [u/s]
Electronic Gear: Slave:	0x6083 Profile acceleration [u/s²]	--> Internal Reference

Quick Stop Ramp / Emergency Ramp

Homing Mode:	0x6085 Quick stop deceleration [u/s²]	--> P.1179 Emergency Ramp [u/s²]
Velocity Mode [vl]:	0x604A *vl* velocity Quick Stop [rpm/s]	--> P.424 & P.425 [Hz/s]
Profile Velocity Mode [pv]:	0x6085 Quick Stop deceleration [u/s²]	--> P.1179 Emergency Ramp [u/s²]
Profile Position Mode:	0x6085 Quick Stop deceleration [u/s²]	--> P.1179 Emergency Ramp [u/s²]
Interpolated Position Mode:	0x6085 Quick Stop deceleration [u/s²]	--> P.1179 Emergency Ramp [u/s²]
Table Travel Record Mode:	0x6085 Quick Stop deceleration [u/s²]	--> P.1179 Emergency Ramp [u/s²]
Move away from Limit Switch:	0x6085 Quick Stop deceleration [u/s²]	--> P.1179 Emergency Ramp [u/s²]
Electronic Gear: Slave:	0x6085 Quick Stop deceleration [u/s²]	--> P.1179 Emergency Ramp [u/s²]

Target Position

Profile Position Mode:	0x607A Target Position [u]	--> Internal Reference
Interpolated Position Mode:	0x60C1/1 Interpolation data record	--> Internal Reference
Table Travel Record Mode:		--> P.1202 Target Pos.

Actual values:

All modes:	0x6044 *vl* velocity actual value [rpm]
	0x6064 Position Act. Value [u]
	0x606C Velocity Act. Value [u/s]
	0x60F4 Following error act. Value [u]
	0x6077 Torque Actual Value [x.x %]
	0x6078 Current Actual Value [x.x %]
	0x6079 DC link circuit voltage [x.xxx V]

Special Objects:

All modes:	0x6060 Modes of Operation	
	0x6061 Modes of Operation Display	
	0x6040 Control Word	--> P.410 Control word
	0x6042 Status Word	<-- P.411 Status word
Velocity [pv]:	0x606D Velocity Window	--> P.1276 Velocity Window
	0x606E Velocity Wind. Time	--> P.1277 Velocity Window Time
	0x606F Velocity Threshold	--> P.1278 Velocity Threshold
	0x6070 Velocity Th. Time	--> P.1279 Velocity Threshold Time
	0x60F8 Max. Slippage	--> P.1275 Max Slippage
Homing Mode:	0x6098 Homing method	--> P.1130 homing type
Table Travel Record Mode:		--> Motion Block selection via Control word Bits 11…15
Electronic Gear: Slave:	0x5F10 Gear Factor	--> P.1123, P.1124, P.1142 Gear Factor
	0x5F11…0x5F14 Phasing	--> P.1125, P.1126, P.1127 Phasing

Velocity [vl] → Velocity mode [rpm]
Velocity [pv] → Profile Velocity mode [u/s]

The graphical overview shows the most important objects which are used. Other objects are available in the different modes; for additional information, refer to the descriptions of the objects and modes.

The Motion Control Interface (MCI) is a defined interface of the ANG device for position control. This interface is typically used in combination with a field bus such as CANopen.

11.2Functions of the Motion Control Interface (MCI)

Via the Motion Control Interface, numerous positioning functions can be addressed by a PLC directly.

11.2.1 Reference system

In many modes, the Motion Control Interface uses user units [u]. These user units [u] result from the conversion of the gear factor parameters **1115**, **1116**, **1117** and *No. of pole pairs* **373**.

Conversion between "user units" [u] and frequencies [Hz]

$$f\left[\mathrm{Hz}\right] = v\left[\frac{u}{s}\right] \cdot \frac{\mathit{No.of\ pole\ pairs}\ \mathbf{373} \cdot \mathit{Gear\,Box : Driving\,shaft revolutions}\ \mathbf{1116}}{\mathit{Feed\,Constant}\mathbf{1115}\ \frac{[u]}{U} \cdot \mathit{Gear\ Box : Motor\,shaft revolutions}\ \mathbf{1117}}$$

$$v\left[\frac{u}{s}\right] = f\left[\mathrm{Hz}\right] \cdot \frac{\mathit{Feed\ Constant}\ \mathbf{1115}\ \frac{[u]}{U} \cdot \mathit{Gear\ Box\ : Motor\ shaft\ revolutions}\ \mathbf{1117}}{\mathit{No.of\ \ pole\ pairs}\ \mathbf{373} \cdot \mathit{Gear\ Box\ : Driving\ shaft\ revolutions}\ \mathbf{1116}\ \cdot}$$

Feed Constant **1115** ≙ 0x6092/1 feed

Gear Box: Motor shaft revolutions **1116** ≙ 0x6091/1 motor shaft revolutions

Gear Box: Driving shaft revolutions **1117** ≙ 0x6091/2 driving shaft revolutions

The same formulas can be used for the conversion from acceleration a [Hz/s] to a [u/s²] and vice versa. Replace in the formulas the velocities f[Hz] and v [u/s] with a [Hz/s] and a [Hz/s²].

For more details about the reference system, refer to the "Positioning" application manual.

11.2.2 Homing

When the drive is started, a defined starting position must be specified for absolute positioning modes. In a homing operation, the point of reference of the positioning operation is determined. All positioning data relates to this point of reference. Once the homing operation is started, the drive moves until it reaches a home switch or limit switch and stops there. The limit switches limit the motion path. The direction of movement (search direction) at the start of the homing operation is defined by the homing mode. Additional the reaching of a limit switch will change the direction of the drive (dependent on the homing mode). The limit switches can also be used as the point of reference.

Relative positioning and velocity operations are possible without homing.

Homing can be started:

- via a digital input
- by a control word via systembus or field bus [1)]
- automatically before the start of a motion block positioning operation

[1)] Expansion module with systembus or field bus interface required

When using an Absolute Encoder, a Homing after power on is not necessary. This is defined by parameter *Operation Mode* **1220**.

Further details of the Homing functions are described in the application manual "Positioning".

Start position after homing

After homing:

P.1185 = -1 → Drive remains in "coast to stop" position

P.1185 ≠ -1 → Drive is moved actively to set position.

Flying homing

The Flying homing can be used to update the reference position during a running motion. This function is described in the application manual "Positioning".

11.2.3 Position Controller

The position controller evaluates the positioning operation (target/actual position) and tries to control the drive such that it comes as close as possible to the specifications.

Further details of the Position controller are explained in chapter 9.9.4.21 "0x5F17/n Position Controller" with a description of Object 0x5F17.

11.2.4 Move away from Hardware limit switches

When a hardware limit switch is triggered, an error message will be triggered depending on the settings of parameter *Fault reaction* **1143** and the relevant direction of rotation will be disabled.

After an error reset, it is possible to move in the direction that is still enabled. Generally, any mode of operation can be used for clearing, as long as the travel command has the enabled direction.

As long as the limit switch is triggered, the limit switch warning in the status word and actual value parameters *Warnings* **269**, *Warnings Application* **273** and *Controller status* **275** will remain. Once the limit switch is cleared, the warning will be deleted in the status word and actual value parameters.

For simple clearing of the limit switches, you can use mode "-2 Move away from limit switch" (see Chapter 12.4.6 "Move away from limit switch mode ").

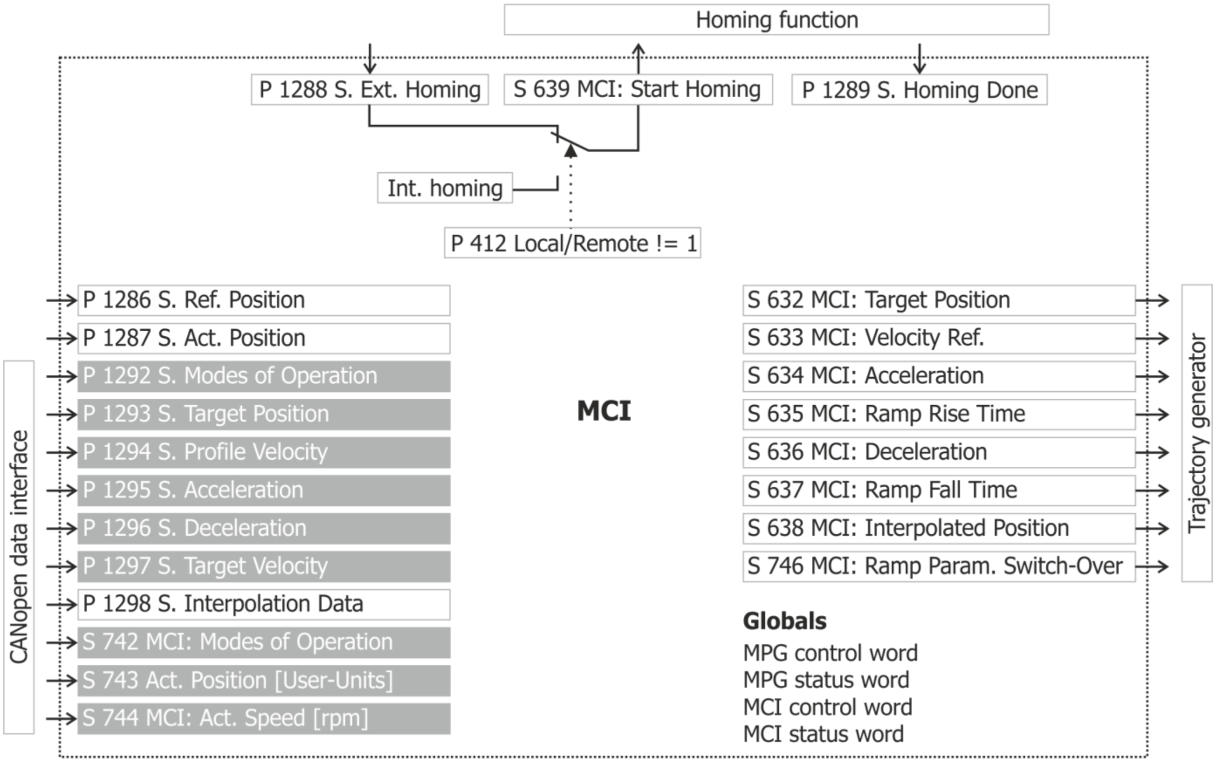
For more details on positioning functions refer to the "Positioning" application manual.

11.3Motion Control Interface for Experts

With the Motion Control Interface, the user can edit the sources which the Motion Control Interface accesses. By default, the sources are set to CANopen®. Experienced users can change these to Systembus sources, for example.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Default setting**
1292	S. Modes of Operation	Selection		801 – Obj. 0x6060 Modes of Operation
1293	S. Target Position	Selection		802 – Obj. 0x607A Target Position
1294	S. Profile Velocity 1)	Selection		803 – Obj. 0x6081 Profile Velocity
1295	S. Acceleration 2)	Selection		804 – Obj. 0x6083 Profile Acceleration
1296	S. Deceleration 3)	Selection		805 – Obj. 0x6084 Profile Deceleration
1297	S. Target Velocity [rpm]	Selection		806 – Obj. 0x6042 *vl* target Velocity
1299	S. Special Function Generator	Selection		9-Zero
1285	S. Target Velocity pv [u/s]	Selection		816 – Obj. 0x60FF Target Velocity

The following graph shows the parameters (P) and sources (S) which are used for defining the Motion Control Interface. For CANopen®, the settings don't have to be changed. The source outputs are linked to the Trajectory generator by default and also need not be changed in standard applications.



11.4Motion Control Override

The Motion Control Override feature can be used for specifying a travel profile via serial communication (VABus or Modbus). This enables testing of a travel profile in the VPlus user software for Windows when the controller has not yet been completely programmed. This function can also be used as a simulation mode.

For more information on VPlus, see chapter 17.

The Function Motion Control Override **does not support** the following modes:

– Interpolated Mode
– Cyclic Synchronous Position Mode
– Cyclic Synchronous Velocity Mode

Parameter		Setting		
No.	Description	Min.	Max.	Factory setting
1454	Override Modes Of Operation	Selection		0
1455	Override Target Position	-2^{31}-1…2^{31}-1 u		-1 u
1456	Override Profile Velocity	-1…2^{31}-1 u/s		-1 u/s
1457	Override Acceleration	-1…2^{31}-1 u/s²		-1 u/s²
1458	Override Deceleration	-1…2^{31}-1 u/s²		-1 u/s²
1459	Override Target Velocity vl [rpm]	-32768…32767 rpm		-1 rpm
1460	Override Target Velocity pv [u/s]	-2^{31}-1…2^{31}-1 u/s		-1 u/s

Based on the standard settings of the Motion Control Interface (Parameters **1292**…**1297**) the following cross reference results between Override Parameters and CANopen®Objects:

1454 *Override Modes Of Operation*	**or**	0x6060 Modes of Operation
1455 *Override Target Position*	**or**	0x607A vl target Position
1456 *Override Profile Velocity*	**or**	0x6081 Profile Velocity
1457 *Override Acceleration*	**or**	0x6083 Profile Acceleration
1458 *Override Deceleration*	**or**	0x6084 Profile Deceleration
1459 *Override Target Velocity vl [rpm]*	**or**	0x6042 vl target Velocity
1460 *Override Target Velocity pv [u/s]*	**or**	0x60FF Target Velocity

With the default settings "-1" in parameters **1455**…**1460** and "0" in parameter **1454** *Override Modes Of Operation*, the values of the Motion Control from the links of parameters **1292**…**1297** are used. If the parameter settings deviate from the factory settings, the value of the relevant parameter will be used. It is possible to define certain ranges of the trajectory via the override function and other values via the Motion Control Interface.

The target position "-1 u" cannot be used as target position, because **1455** *Override Target Position = -1* deactivates the Override Function.

12 Control of frequency inverter

The frequency inverter can generally be controlled via three operation modes. The operation modes can be selected via the data set switchable parameter *Local/Remote* **412**.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
412	Local/Remote	0	44	44

For operation with Field bus, only the settings 0, 1 and 2 are relevant. The other settings refer to the control option via the KP500 control unit.

Operation mode		Function
0 -	Control contacts (chapter 12.1)	The Start and Stop commands as well as the direction of rotation are controlled via digital signals.
1 -	Control via state machine (chapters 12.2, 12.3, 12.4)	The frequency inverter is controlled via the control word. Only in this control mode are the motion control functions supported by *Control word* and *modes of operation* as defined with CANopen® DS402.
2 -	Control via remote contacts (chapter 12.1)	The Start and Stop commands as well as the direction of rotation are controlled via virtual digital signals of the control word.

Parameter *Local/Remote* **412** is dataset switchable, i.e. you can switch between the different operation modes by selecting another data set. For example, a frequency inverter can be controlled via the bus, and emergency mode can be activated locally when the bus master fails. This switch-over is also identified by the status word (remote bit).

Data set switching can be effected locally via control contacts at the digital inputs of the frequency inverter or via the bus. For data set switching via the bus, parameter *Data set selection* **414** is used.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
414	Data set selection	0	4	0

With *Data set selection* **414** = 0, data set switching via contact inputs will be active.

If *Data set selection* **414** is set to 1, 2, 3 or 4, the selected data set is activated and data set switching via the contact inputs is deactivated.

If *Data set selection* **414** is set to 5, data set switching via contact inputs will be active if the frequency inverter is not enabled.

Via parameter *Active data set* **249**, the currently selected data set can be read. *Active data set* **249**, indicates the active data set (value 1, 2, 3 or 4). This is independent of whether the data set switching was done via contact inputs or *Data set selection* **414**.

12.1Control via contacts/remote contacts

In operation mode "Control via contacts" or "Control via remote contacts" (Parameter *Local/Remote* **412** = 0 or 2), the frequency inverter is controlled directly via digital inputs S1IND (STOA and STOB), S2IND to S9IND or via the individual bits of the virtual digital signals in the control word. The function of these inputs is described in the frequency inverter operating instructions.

Control word (Local/Remote 412 = 2)	
Bit	
0	S1IND (=STOA and STOB)
1	S2IND
2	S3IND
3	S4IND
4	S5IND
5	S6IND
6	MF1ID
7	MF4ID
8	REMOTE-1
9	REMOTE-2
10	MF3ID
11	MF2ID
12	-
13	-
14	-
15	-

Status word	
Bit	
0	Ready to switch on
1	Switched on
2	Operation enabled
3	Fault
4	Voltage enabled
5	Quick stop (Low active)
6	Switch on disabled
7	Warning
8	-
9	Remote
10	Target reached
11	Internal limit active
12	-
13	-
14	-
15	Warning 2

If operation mode "Control via remote contacts" is used, controller release must be turned on at STOA (Terminal X210A.3) and STOB (Terminal X210B.2) **and** Bit 0 of the control word must be set in order to be able to start the drive.

Operation modes "Control via contracts" and "Control via remote contacts" support operation mode "Speed vl" (*modes of operation* = "velocity mode").

ANG frequency inverters support an external 24 V power supply for the frequency inverter control electronics. Even when mains voltage is disconnected, communication between the controller (PLC) and the frequency inverter will be possible.

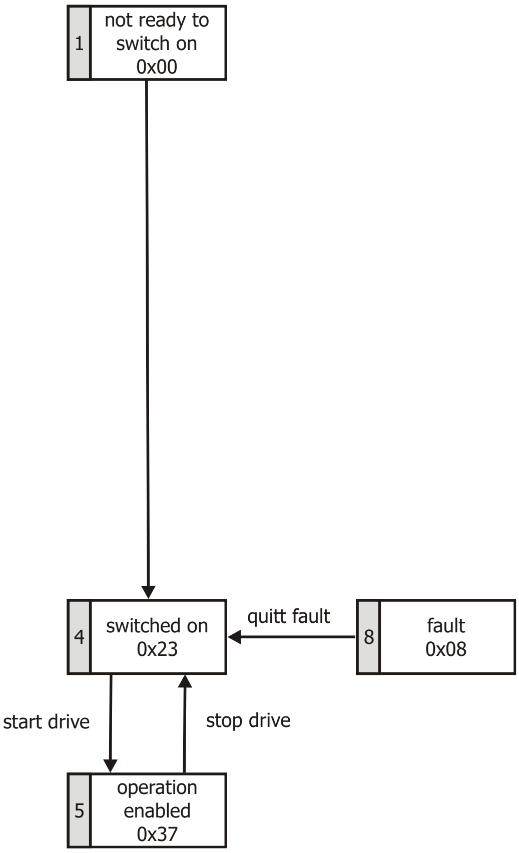
Bit 4 "Voltage enabled" of the *Status word* displays the current state of the mains power supply.

Bit 4 "Voltage enabled" = 0 signals "no mains voltage" and drive start is disabled.

Bit 4 "Voltage enabled" = 1" signals "mains voltage switched on" and drive start is enabled.

12.1.1 Device State machine

State machine:



Status word	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
Switched on	1	0	0	0	1	1
Operation enabled	1	1	0	1	1	1
Fault	x	x	1	x	x	x

"X" means any value.

Bit 7 "**Warning"** can display a device-internal warning message at any time. The current warning is evaluated by reading the warning status with parameter *Warnings* **270.**

Bit 10 "**Target reached**" is set when the specified reference value is reached. In the special case of power failure regulation, the bit is also set when the power failure regulation reaches the frequency 0 Hz (see frequency inverter Operating Instructions).

For "target reached" a hysteresis (tolerance range) applies. It can be set via parameter *Max. control deviation* **549** (see frequency inverter operating instructions).

Bit 11 "**Internal limit value active"** indicates that an internal limit is active. This may be the current limit, the torque limit or the overvoltage control. All of these limit functions will result in the reference value being left or not reached.

Bit 15 "**Warning 2**" signals a critical operating state which will result in a fault switch-off of the frequency inverter within a short time. This bit is set if there is a delayed warning relating to the motor temperature, heat sink/inside temperature, Ixt monitoring or mains phase failure.

12.2Control via state machine

In the operation mode "Control via state machine (*Local/Remote* **412** = 1), the frequency inverter is controlled via the *Control word* of the state machine.

Transition 4 to status "Operation enabled" is only possible:

– if, in a configuration for Motion Control (parameter *Configuration* **30** = x40), the controller release is set via STOA and STOB,

– if, in other configurations (parameter *Configuration* **30** ≠ x40) the controller release is set via STOA and STOB and if one of the digital inputs S2IND or S3IND is set. (Typically: S2IND = Start clockwise/S3IND = Start anticlockwise)

Object *0x6040 Control word* is applicable to the frequency inverter if parameter *Local/Remote* **412** is set to "`1 - Control via statemachine`".

Control word (Local/Remote 412 = 1)		Status word	
Bit		**Bit**	
0	Switch on	**0**	Ready to switch on
1	Enable voltage	**1**	Switched on
2	Quick stop (Low active)	**2**	Operation enabled
3	Enable operation	**3**	Fault
4	Operation mode specific	**4**	Voltage enabled
5	Operation mode specific	**5**	Quick stop (Low active)
6	Operation mode specific	**6**	Switch on disabled
7	Fault reset	**7**	Warning
8	Halt	**8**	Manufacturer specific
9	Operation mode specific	**9**	Remote
10	-	**10**	Target reached
11	Manufacturer specific	**11**	Internal limit active
12	Manufacturer specific	**12**	Operation mode specific
13	Manufacturer specific	**13**	Operation mode specific
14	Manufacturer specific	**14**	Manufacturer specific
15	Manufacturer specific	**15**	Manufacturer specific Warning 2

Bits 9 … 15 are used depending on the configuration and on 0x6060 *Mode of Operation*.

Control word bits 4, 5, 6 *operation mode specific* and bit 8 *halt* are used in motion control configurations (Parameter *Configuration* **30** = `x40`) only.

Bit 14 is not used.

Status word bits 12 and 13 o*peration mode specific* are used in motion control configurations (Parameter *Configuration* **30** = `x40`) only.

ANG frequency inverters support an external 24 V power supply for the frequency inverter control electronics. Even when mains voltage is disconnected, communication between the controller (PLC) and the frequency inverter is still possible.

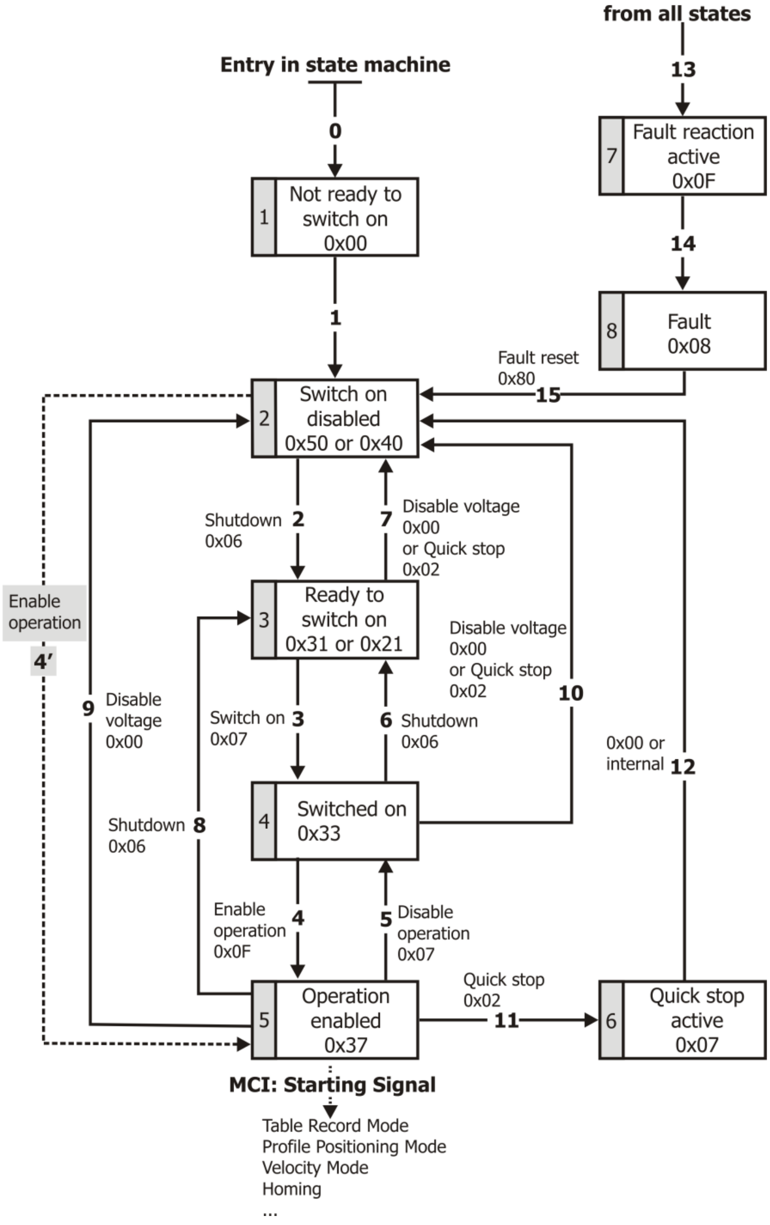
Bit 4 "Voltage enabled" of the *Status word* displays the current state of the mains power supply.

Bit 4 "Voltage enabled" = 0 signals "no mains voltage" and drive start is disabled.

Bit 4 "Voltage enabled" = 1" signals "mains voltage switched on" and drive start is enabled..

12.2.1 State machine diagram

State machine:



Control word:

The device control commands are triggered by the following bit patterns in the *Control word*:

Control word						
Command	Bit 7 Fault reset	Bit 3 Enable operation	Bit 2 Quick stop (Low active	Bit 1 Enable voltage	Bit 0 Switch on	Transitions
Shutdown	X	X	**1**	**1**	**0**	2, 6, 8
Switch on	X	**0**	**1**	**1**	**1**	3
Enable operation	X	**1**	**1**	**1**	**1**	4
Disable voltage	X	X	X	0	X	7, 9, 10, 12
Quick stop (Low active)	X	X	**0**	**1**	X	7, 10, 11
Disable operation	X	**0**	**1**	**1**	**1**	5
Fault reset	**0 → 1**	x	x	x	x	15

"X" means any value.

Transition 3 (command "Switch on" [0x07]) is only processed if bit 4 "Voltage enabled" of the Status word is set.

Transition 4 (Command "Enable operation" [0xF]) will only be processed if the release is set via the hardware contacts STO.

If the hardware release via STO is not set, the frequency inverter will remain in status "Switched On" [0x33] until the hardware release via STO is present.

In status "Operation enabled" [0x37], the device will switch to status "Switched On" [0x33] internally once the hardware release via STO is reset.

In configurations **with** Motion Control (parameter *Configuration* **30** = x40), the following must be noted:

– Transition **4'** is **not** available.

– In status "5-Operation enabled [0x37]" an additional start signal must be provided via bits from the "High Byte" of the control word in order to start a movement of the motor. For a description of the start signal for this "Motion Control Interface" (MCI), refer to chapter 11. Object 0x6060 Modes of Operation is available for switching to other MCI modes.

– Digital inputs (STOA and STOB) must be set. *Start clockwise* and *Start anticlockwise* have no function in these configurations.

In configurations **without** Motion Control (parameter *Configuration* **30** ≠ x40), the following must be noted:

– Transition **4'** will only be processed if Bit 4 "Voltage enabled" of the status word is set. This feature is downward-compatible with older software versions.

– The frequency inverter can only be controlled if the logic operation is true. The logic inputs for Start Clockwise and Start anticlockwise can be connected directly with "On" or "Off" (parameter *Start Clockwise* **68** and *Start Anticlockwise* **69**).

– Digital inputs (STOA and STOB) must be set.

– This results in: Release: (= STOA and STOB) **AND** (Start clockwise **OR** Start Anticlockwise)

Status word:

The status word indicates the operating status.

Status word						
State	**Bit 6**	**Bit 5**	**Bit 3**	**Bit 2**	**Bit 1**	**Bit 0**
	Switch on disabled	**Quick stop (Low active)**	**Fault**	**Operation enabled**	**Switched on**	**Ready to switch on**
Switch on disabled	**1**	X	**0**	**0**	**0**	**0**
Ready to switch on	**0**	**1**	**0**	**0**	**0**	**1**
Switched on	**0**	**1**	**0**	**0**	**1**	**1**
Operation enabled	**0**	**1**	**0**	**1**	**1**	**1**
Quick stop active (Low active)	**0**	**0**	**0**	**1**	**1**	**1**
Fault reaction active	**0**	X	**1**	**1**	**1**	**1**
Fault	**0**	X	**1**	**0**	**0**	**0**

"X" means any value.

Bit 7 **"Warning"** can be set at any time. It reports an internal device warning. The cause of the warning is evaluated by reading the warning status with parameter *Warnings* **270**.

Bit 9 "**Remote"** is set if the operation mode is set to "Control via state machine" (*Local/Remote* **412** = 1) **and** controller release is turned on.

Bit 10 "**Target reached**" is set when the specified reference value is reached.

In configurations without Motion Control (parameter *Configuration* **30** ≠ x40) "target reached" refers to the reference speed from Vl target the reference speed (*vl* target speed) 0x6042. In the special case of power failure regulation, the bit is also set when the power failure regulation reaches the frequency 0 Hz (see frequency inverter operating instructions).

For "target reached", there is a hysteresis (tolerance range) which can be set via the parameter *Max. control deviation* **549** (see frequency inverter Operating Instructions).

Bit 11 "**Internal limit value active"** indicates that an internal limit is active. This may be the current limit, the torque limit or the overvoltage control. All functions will result in the reference value being left or not reached.

Bit 15 "**Warning 2**" signals a critical operating state which will result in a fault switch-off of the frequency inverter within a short time. This bit is set if there is a delayed warning relating to the motor temperature, heat sink/inside temperature, Ixt monitoring or mains phase failure.

12.3Configurations without motion control

In configurations without Motion Control (*Configuration* **30** ≠ x40) object 0x6060 *modes of operation* is set permanently to "2 - velocity mode" (velocity mode vl) Object 0x6061 *modes of operation display* will also be "2 - velocity mode" (velocity mode vl). These settings cannot be changed.

Relevant objects:

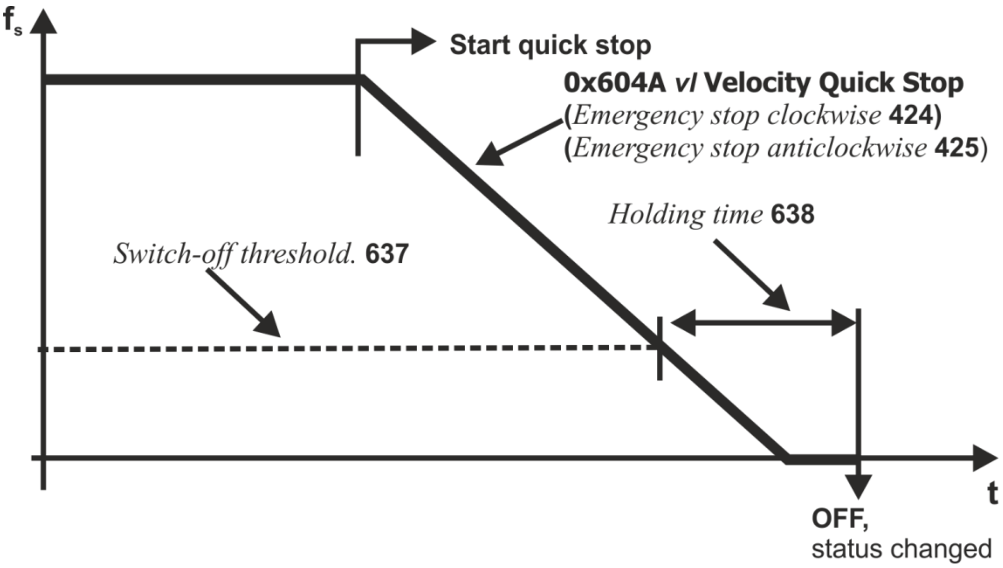
0x6040	Control word
0x6041	Status word
0x6042	*vl* target velocity
0x6043	Velocity demand
0x6044	*vl* velocity actual value
0x6046	*vl* velocity min max amount
0x6048	*vl* velocity acceleration
0x6049	*vl* velocity deceleration
0x604A	*vl* Velocity quick stop

The Ramp Rise/Fall times are set up via parameters **430**…**433**.

12.3.1 Behavior in the case of a quick stop

In quick stop, the parameters *Switch-off threshold* **637** (percent of parameter *Maximum Frequency* **419**) and *Holding time* **638** (holding time after falling short of the switch-off threshold) are relevant. In the case of a quick stop, the drive is stopped via emergency stop ramps.

The emergency stop ramps are set up via Object 0x604A *vl Velocity Quick Stop* or parameters *Emergency Stop Clockwise* **424** and *Emergency Stop Anticlockwise* **425.**



If frequency/speed reaches the value zero during the switch-off time, the drive continues to be supplied with current until the switch-off time has elapsed. This ensures that the drive is at a standstill when the state changes.

The quick stop behavior is only relevant for configurations without Motion Control (parameter *Configuration* **30** ≠ x40).

12.3.2 Behavior in the case of transition 5 (disable operation)

The behavior in transition 5 of the statemachine from "`Operation enabled`" to "`Switched On`" can be configured via parameter *State transition 5* **392**.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
392	State transition 5	0	2	2

Operation mode		Function
0 -	Coast to stop	Immediate transition from "`Operation enabled`" to "`Switched On`", drive coasts to a standstill
1 -	DC brake	Activation of DC brake, at the end of DC deceleration, there is the change from "`Operation enabled`" to "`Switched On`"
2 -	Ramp	Transition at normal ramp, when the drive has come to a standstill, there is the change from "`Operation enabled`" to "`Switched On`"

Setting 1 "`Direct current brake`" is only possible with applications with U/f characteristic control (e.g. configuration 110). Other configurations do not support this operation mode.

If the frequency inverter is operated with a configuration which does not support the operation mode Direct Current Brake (e.g. configuration 210, field-oriented control), value "`1`" cannot be used.

In this case, the operation mode is not offered in the selection menus of the control unit KP500 and the control software VPlus.

For more information on VPlus, see chapter 17.

By default, *State transition 5* **392** is set to operation mode "2 – Ramp" For configurations with torque control, the default value is "0 – coasting".

If the configuration is changed, the value set for *State transition 5* **392** is also changed, if necessary.

The behavior in transition 5 is only relevant for configurations without Motion Control (parameter *Configuration* **30** ≠ x40).

If *State Transition 5* **392** was triggered with "1 – DC brake", a new control word will only be accepted after completion of the transition process. The change of state from "Operation enabled" to "Switched On" is done after the *Braking time* **632** parameterized for the DC brake has elapsed.

If parameter *State Transition* **392** = "2 – Ramp" is set, the control word can be set to "Operation enabled" again, while the drive is decelerating. In this way, the drive accelerates to its set reference value again and remains in the state "Operation enabled".

The change of state from "Operation enabled" to "Switched On" is done after the value has dropped below the set switch-off threshold and the set holding time has elapsed (equivalent to the behavior in the case of a quick stop). In this context, parameters *Switch-off threshold* **637** (percentage of parameter *Maximum frequency* **419**) and *Holding time* **638** (holding time after passing of threshold) are relevant.

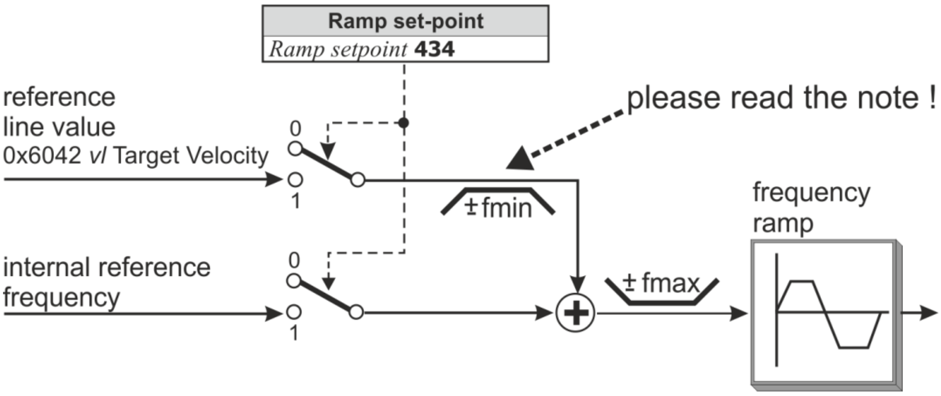
12.3.3 Reference value / actual value

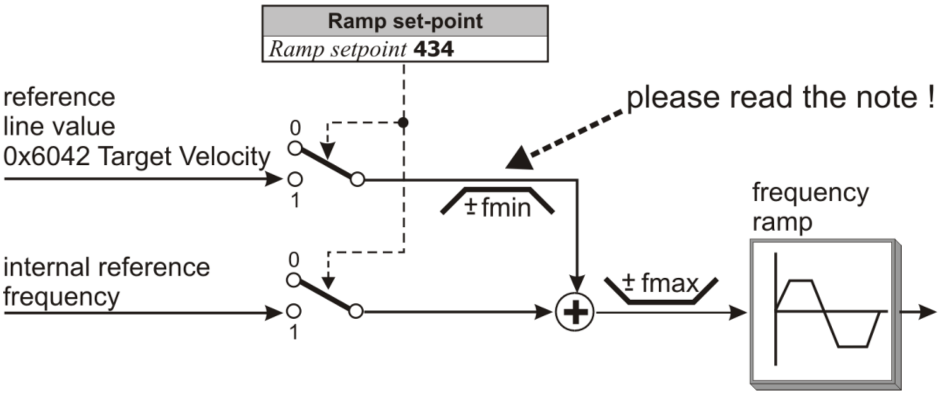
The controller (PLC) can define the reference frequency for the frequency inverter via object 0x6042/0 *vl target velocity* in the used RxPDO and receive the actual value via object 0x6044/0 *vl velocity actual value* in the used TxPDO.

The use of the reference/actual value channel depends on the set configuration (control method). The actual value is generated according to the control method used.

The reference value in object 0x6042/0 *vl target velocity* and the actual value in object 0x6044/0 *vl velocity actual value* are interpreted in the notation RPM. Conversion into a frequency (reference value), or from a frequency (actual value) is carried out in the frequency inverter.

The reference value for the frequency inverter from object 0x6042/0 *vl target velocity* is supplied via the reference line value. This reference value is combined with the internal reference value from the reference frequency value channel in the input of the ramp function. Reference frequency value channel: see operating instructions.





The internal reference value from the reference frequency value channel and the reference line value can be fed to the ramp individually or as an added variable. Setting is carried out via the data set change-over capable parameter *Ramp setpoint* **434**.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
434	Ramp setpoint	1	3	3

Operation mode		Function
1 -	Internal reference frequency value	Reference value from the sources of the reference frequency value channel
2 -	Reference line value	Reference value via a communication interface
3 -	Internal reference frequency value + reference line value	Sum of internal reference frequency value and reference line value

This function is only relevant for non motion control configurations (Parameter *Configuration* **30** ≠ x40)

If *Ramp setpoint* **434** = 2 (reference line value only), this reference line value is limited to *Minimum Frequency* **418** (fmin).

The sign in front of fmin with reference value = 0 is derived from the sign in front of the last reference line value which was not 0.

After Mains On, the reference line value is limited to +fmin.

For *Ramp set-point* **434** = 3, the sign in front of the overall reference value results from the sum of the internal reference frequency value and the reference line value.

The reference values can be read out from the frequency inverter with the help of the KP500 control unit or VPlus operating software. For more information on VPlus, see chapter 17.

Actual values		
Parameter	**Contents**	**Format**
Internal reference frequency **228**	Internal reference value from the reference frequency channel	xxx.xx Hz
Reference bus frequency **282**	Reference line value from the Fieldbus	xxx.xx Hz
Reference ramp frequency **283**	= sum of internal reference frequency + reference line value	xxx.xx Hz

12.3.4 Sequence example

In configurations without Motion Control (*Configuration* **30** ≠ x40), the PLC must send the correct sequence:

1	Control word =	0x0000	Disable voltage
3	Control word =	0x0006	Shutdown
4	Control word =	0x0007	Switch On
5	Control word =	0x000F	Enable Operation

OR

1	Control word =	0x0000	Disable voltage
5	Control word =	0x000F	Enable Operation

In configurations without Motion Control (*Configuration* **30** ≠ x40), the second (shortened) sequence can be used, because transition **4‘** is available in these configurations.

12.4Motion control configurations

WARNING

Dangerous state due to new mode!

If 0x6060 *modes of operation* is changed during operation (control word = 0xnnnF), a dangerous state may occur in the new mode.

- Before changing 0x6060 *modes of operation*, check the status word (e.g. for state 0xnn33).

Motion Control Definition

For the full function of the Motion Control Interface, you will have to set *Local/Remote* **412** = "1-Control via state machine". In all other operation modes of parameter *Local/Remote* **412**, there are major restrictions. The descriptions in this chapter and of all objects used are based on the setting *Local/Remote* **412** = "1-Control via state machine".

The usage of Positioning for setting *Local/Remote* **412** ≠ 1 is described in the "Positioning" application manual.

The function of the state machine describes the basic operating behavior of the frequency inverter in configurations with position control (*Configuration* **30** = x40). The objects *control word* and *status word* described above support the bits marked as operation mode specific.

These bits and bit "Target reached" has different meanings in the different position control operation modes – defined by 0x6060 *modes of operation*. The following chapters describe the application of the operation mode specific bits in the *control word* and *status word*, depending on the different position control operation modes. Default value: 0x6060 *modes of operation* = 2 – velocity mode.

Basic functions:

The state machine must be set to "operation enabled", before the position command can be issued via the operation mode specific bits of the *control word*.

Once a *mode of operation* has been set by the PLC, no commands will be accepted for this operation mode until this operation mode is displayed in the *modes of operation display* object.

The bits in the *control word* and *status word* marked as operation mode specific are only supported in configurations with position control (*Configuration* **30** = x40).

12.4.1 Velocity mode [rpm]

Velocity mode [rpm] can be selected via object 0x6060/0 *Modes of operation* = **2.**

In velocity mode the "*operation mode specific*" bits of the *Control word* control the ramp generator (RFG – Ramp Function Generator). The block diagram illustrates the function.

Relevant objects:

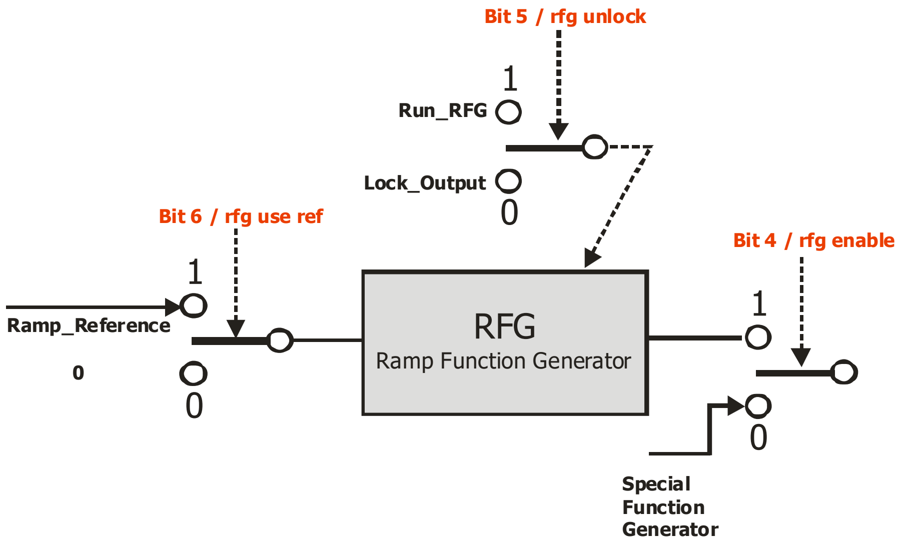
0x6040	Control word
0x6041	Status word
0x6042	*vl* target velocity
0x6043	Velocity demand
0x6044	*vl* velocity actual value

0x6046	*vl* velocity min max amount
0x6048	*vl* velocity acceleration
0x6049	*vl* velocity deceleration
0x604A	Velocity quick stop
0x6060	Modes of operation
0x6061	Modes of operation display

The Ramp Rise/Fall times are set up via parameters **430**…**433**.

Control word *Modes of operation* =2			**Status word**	
Bit			**Bit**	
0	Switch on		**0**	Ready to switch on
1	Enable voltage		**1**	Switched on
2	Quick stop (Low active)		**2**	Operation enabled
3	Enable operation		**3**	Fault
4	Rfg enable		**4**	Voltage enabled
5	Rfg unlock		**5**	Quick stop (Low active)
6	Rfg use ref		**6**	Switch on disabled
7	Fault reset		**7**	Warning
8	Halt		**8**	-
9	-		**9**	Remote
10	-		**10**	Target reached (not used)
11	-		**11**	Internal limit active
12	-		**12**	-
13	-		**13**	-
14	-		**14**	-
15	-		**15**	Warning 2

Block diagram



Bit 4/rfg enable

Rfg enable = 0 the reference speed comes from a manufacturer-specific special function.

Rfg enable = 1 The reference speed corresponds to the ramp output.

The special function is evaluated only if **1299** *S. Special Function Generator* ≠ "9-Zero". If **1299** *S. Special Function Generator* = "9-Zero", the value of the ramp output will always be used.

Bit 5/rfg unlock

Rfg unlock = 0 The last speed will be maintained and used.

Rfg unlock = 1 The ramp function is active and changes according to the reference value and the ramp.

Bit 6/rfg use ref

Rfg use ref = 0 Reference value "0" is used.

Rfg use ref = 1 The setpoint of 0x6042 *vl target Velocity* is used.

Bit 8 Halt

HALT = 0 → **Execute positioning.**

HALT = 1 → **Stop axis.** (The frequency inverter remains enabled in "Operation enabled" state.)

If special function generator **1299** *Q. Special Function Generator* ≠ "9-Zero", the reference value from the ramp output will also be used if bit 4 "rfg enable" = 1, and if bit 4 "rfg enable" = 0, the reference value from the source specified in **1299** *Q. Special Function Generator* will be used.

Reference value source		
	1299 *Q. Special Function Generator* ≠ "9-Zero"	**1299** *Q. Special Function Generator* = "9-Zero"
Bit 4 rfg enable = 0	Reference value from special function	Reference value from ramp output
Bit 4 rfg enable = 1	Reference value from ramp output	

Sequence example

In order to start "velocity mode", the correct sequence must be sent by the PLC.

1 1	Control word = Status word =	0x0000 0x0050	Disable voltage Switch On Disabled
2	Modes of Operation =	2	(Velocity mode)
3	Control word = Status word =	0x0006 0x0031	Shutdown Ready to switch on
4	Control word = Status word =	0x0007 0x0033	Switch On Switched On
5	Control word = Status word =	0x000F 0xnn37	Enable Operation, no change to previous state if already enabled. Operation enabled
6a	Control word = Status word =	0x007F 0xnn37	Start Velocity mode with Reference speed from object 0x6042 *vl* target velocity. Operation enabled
6b	Control word = Status word =	0x006F 0xnn37	**1299** *Q. Special Function Generator*: = "9-Zero" Starts "Velocity mode" with reference value from Object 0x6042 *vl* target velocity. **1299** *Q. Special Function Generator*: ≠ "9-Zero" Starts "with reference value from **1299** *Q. Special Function Generator* Operation enabled
6c	Control word = Status word =	0x003F 0xnn37	Starts "Velocity mode" with reference value "0". Operation enabled
6d	Control word = Status word =	0x002F 0xnn37	**1299** *Q. Special Function Generator*: = "9-Zero" Starts "Velocity mode" with reference value "0". **1299** *Q. Special Function Generator*: ≠ "9-Zero" Starts with reference value from source from **1299** *Q. Special Function Generator* Operation enabled
6e	Control word = Status word =	0x005F 0xnn37	Starts "Velocity mode" at current speed – current ramps will be canceled. Operation enabled
6f	Control word = Status word =	0x004F 0xnn37	**1299** *Q. Special Function Generator*: = "9-Zero" Starts "Velocity mode" with actual speed – a running ramp is interrupted. **1299** *Q. Special Function Generator*: unequal "9-Zero" Starts with reference value from source from **1299** *Q. Special Function Generator* Operation enabled
7	Control word =	0x01xx	HALT: The drive is stopped with ramp 0x6049 *vl velocity deceleration*.

Status word =	0xnn37	Operation enabled

WARNING

Dangerous state due to new mode!

When 0x6060 *Modes of Operation* is changed during operation (Control word = 0xnnnF), a dangerous state can occur in the new mode.

- Check the status word before changing 0x6060 *Modes of Operation* (i.e. check state 0xnn33).

Once the sequence of the first four status words has been processed correctly, the ANG is ready for operation (dark table area).

In state "operation enabled" (0xnnnF), the state of the Motion Control can be changed (white table area).

With control word transition from 0xnnnF to 0x0007, "Velocity mode" will be stopped. Then, the mode can be restarted via 0xnnnF.

As long as 0x0007 is active, the mode of operation can be edited safely. Once 0x6060 *modes of operation* has been set to another value, operation can be started with a corresponding sequence.

12.4.2 Profile Velocity mode [u/s]

WARNING

Dangerous state due to new mode!

When 0x6060 *Modes of Operation* is changed during operation (Control word = 0xnnnF), a dangerous state can occur in the new mode.

- Check the status word before changing 0x6060 *Modes of Operation* (i.e. check state 0xnn33).

The profile *velocity mode* is selected via object 0x6060/0 *Modes of operation* =**3**.

In profile *position mode* the inverter receives a reference speed in [u/s].

Relevant objects:

0x6040	Control word
0x6041	Status word
0x6046	*vl* velocity min max amount
0x6060	Modes of operation
0x6061	Modes of operation display
0x606C	Velocity Actual value
0x606D	Velocity Window
0x606E	Velocity Window Time

0x606F	Velocity Threshold
0x6070	Velocity Threshold Time
0x6083	Profile acceleration
0x6084	Profile deceleration
0x6085	Quick stop deceleration
0x6086	Motion Profile Type
0x60F8	Max Slippage
0x60FF	Target Velocity

The Ramp Rise/Fall times are set up via parameters **1176** and **1178** and object 0x6086.

In profile *velocity mode*, the operation mode specific bits of the control word and the status word are used as follows:

Control word *Modes of operation* =3			Status word	
Bit			**Bit**	
0	Switch on		**0**	Ready to switch on
1	Enable voltage		**1**	Switched on
2	Quick stop (Low active)		**2**	Operation enabled
3	Enable operation		**3**	Fault
4	-		**4**	Voltage enabled
5	-		**5**	Quick stop (Low active)
6	-		**6**	Switch on disabled
7	Fault reset		**7**	Warning
8	Halt		**8**	-
9	-		**9**	Remote
10	-		**10**	Target reached
11	-		**11**	Internal limit active
12	-		**12**	Velocity
13	-		**13**	Max Slippage
14	-		**14**	-
15	-		**15**	Warning 2

The profile *velocity mode* is used to set the reference speed in user units [u/s]. The reference speed 0x60FF *Target Velocity* is taken over in mode "*operation enabled*" immediately (0xnn37). The acceleration and deceleration ramp are specified by objects 0x6083 *Profile acceleration* and 0x6084 *Profile deceleration*.

Setting Bit 8 "*Halt*" of the control word delays the drive with ramp 0x6084 *Profile deceleration* and holds the drive at standstill. Resetting Bit 8 results in an acceleration with ramp 0x6083 *Profile acceleration* to the actual reference velocity.

Control word Bit 8: Halt

HALT = 0 → **Execute Profile Velocity Mode.**

HALT = 1 → **Halt Axis.** (The Frequency inverter remains in state "Operation enabled".)

The actual velocity in [u/s] can be displayed in a PLC via map able Object 0x606D.

Via Objects 0x606D *Velocity Window* and 0x606E *Velocity Window time* Bit 10 "*Target reached*" of the status word is set.

Via Objects 0x606F *Velocity Threshold* and 0x6070 *Velocity Threshold time* Bit 12 "*Velocity*" of the status word is set.

Via Object 0x60F8 *Max Slippage* a slip monitoring via Bit 13 "*Max Slippage*" of the status word can be set up.

Status word Bit 10: Target reached

Target reached = 0 → The actual velocity does not match the reference velocity.

Target reached = 1 → **The actual velocity matches the reference velocity.**

The actual velocity differs at least from the defined time period in object 0x606E *Velocity Window time* up to the defined amount [us] in Object 0x606D *Velocity Window*.

Status word Bit 12: Velocity

Velocity = 0 → **The Actual Velocity matches the comparison speed.**

The Actual Velocity has exceeded for a defined time (Object 0x6070 *Velocity Threshold time*) a defined Velocity in user units per seconds [u/s] (Object 0x606F *Velocity Threshold*).

Velocity = 1 → The Actual Velocity does not match the Comparison Velocity.

Status word Bit 13: Maximum Slippage

Maximum Slippage = 0 → **The actual Slippage speed is smaller than defined.** The comparison value of the slippage speed is defined Object 0x60F8 *Max Slippage*.

Maximum Slippage = `1` → **The actual Slippage speed is bigger than defined.** The comparison value of the slippage speed is defined Object 0x60F8 *Max Slippage*.

Sequence Example

To start the Profile Velocity mode, the correct sequence has to be sent from the PLC.

1 1	Control word = Status word =	0x0000 0x0050	Disable voltage Switch On Disabled
2	Modes of Operation =	3	(Profile Velocity mode)
3	Control word = Status word =	0x0006 0x0031	Shutdown Ready to switch on
4	Control word = Status word =	0x0007 0x0033	Switch On Switched On
5	Control word = Status word =	0x0007 ↓ 0x000F 0xnn37	Enable Operation, no change to previous state if already enabled. The Profile Velocity mode is started with reference velocity 0x60FF *Target Velocity* and Ramp profile 0x6084 *Profile acceleration* and 0x6084 *Profile deceleration*. Changes to Target Velocity and Ramps are taken over immediately. Operation enabled

A profile consists of the following entries. If a value is not changed, the old value will still be active.

- 0x6081 *Profile velocity*
- 0x6083 *Profile acceleration*
- 0x6084 *Profile deceleration*
- 0x60FF *Target Velocity*

Once the sequence of the first four status words has been processed correctly, the ANG is ready for operation (dark table area). In state "operation enabled" (0xnnnF), the state of the Motion Control can be changed (white table area).

With control word transition from 0xnnnF to 0x0007, "Profile position mode" will be stopped. Then, the mode can be restarted via 0xnnnF.

While 0x0007 is active, it is also possible to change the modes of operation without any danger. After changing 0x6060 *modes of operation* to another value you can start the new operation mode with the according sequence.

12.4.3 Profile position mode

Profile position mode can be selected via object 0x6060/0 *Modes of operation* = `1`.

In profile position mode, the frequency inverter receives a vl target position, followed by the command to travel to this target.

Relevant objects:

0x6040	Control word	0x607A	Target position
0x6041	Status word	0x6081	Profile velocity
0x6046	*vl* velocity min max amount	0x6083	Profile acceleration
0x6060	Modes of operation	0x6084	Profile deceleration
0x6061	Modes of operation display	0x6085	Quick stop deceleration

The ramp times are specified via parameters **1176** and **1178** and object 0x6086. In profile position mode, the mode-specific bits of the control word and the status word are used as follows:

Control word *Modes of operation* =1			Status word	
Bit			**Bit**	
0	Switch on		**0**	Ready to switch on
1	Enable voltage		**1**	Switched on
2	Quick stop (Low active)		**2**	Operation enabled
3	Enable operation		**3**	Fault
4	New set-point		**4**	Voltage enabled
5	Change set immediatelys		**5**	Quick stop (Low active)
6	Abs/rel		**6**	Switch on disabled
7	Fault reset		**7**	Warning
8	Halt		**8**	-
9	Change set-point		**9**	Remote
10	-		**10**	Target reached
11	-		**11**	Internal limit active
12	-		**12**	Set-point acknowledge
13	-		**13**	Following error
14	-		**14**	-
15	-		**15**	Warning 2

Control word

Change on set-point **Bit 9**	**Change set-point immediately** **Bit 5**	**New set-point** **Bit 4**	**Description**
0	0	0 → 1	Once the sequence of the first four status words has been processed correctly, the ANG is ready for operation (dark table area).
X	1	0 → 1	In state "operation enabled" (`0xnnnF`), the state of the Motion Control can be changed.
1	0	0 → 1	With control word transition from `0xnnnF` to `0x0007`, "Profile position mode" will be stopped. Then, the mode can be restarted via 0xnnnF.

Identification	**Value**	**Description**
Abs/rel **Bit 6**	0	The target position is an absolute value
	1	The target position is a relative value
Halt **Bit 8**	0	Execute positioning operation.
	1	Stop axle with *profile deceleration* (if not supported by *profile acceleration*), the frequency inverter will remain in status "Operation enabled".

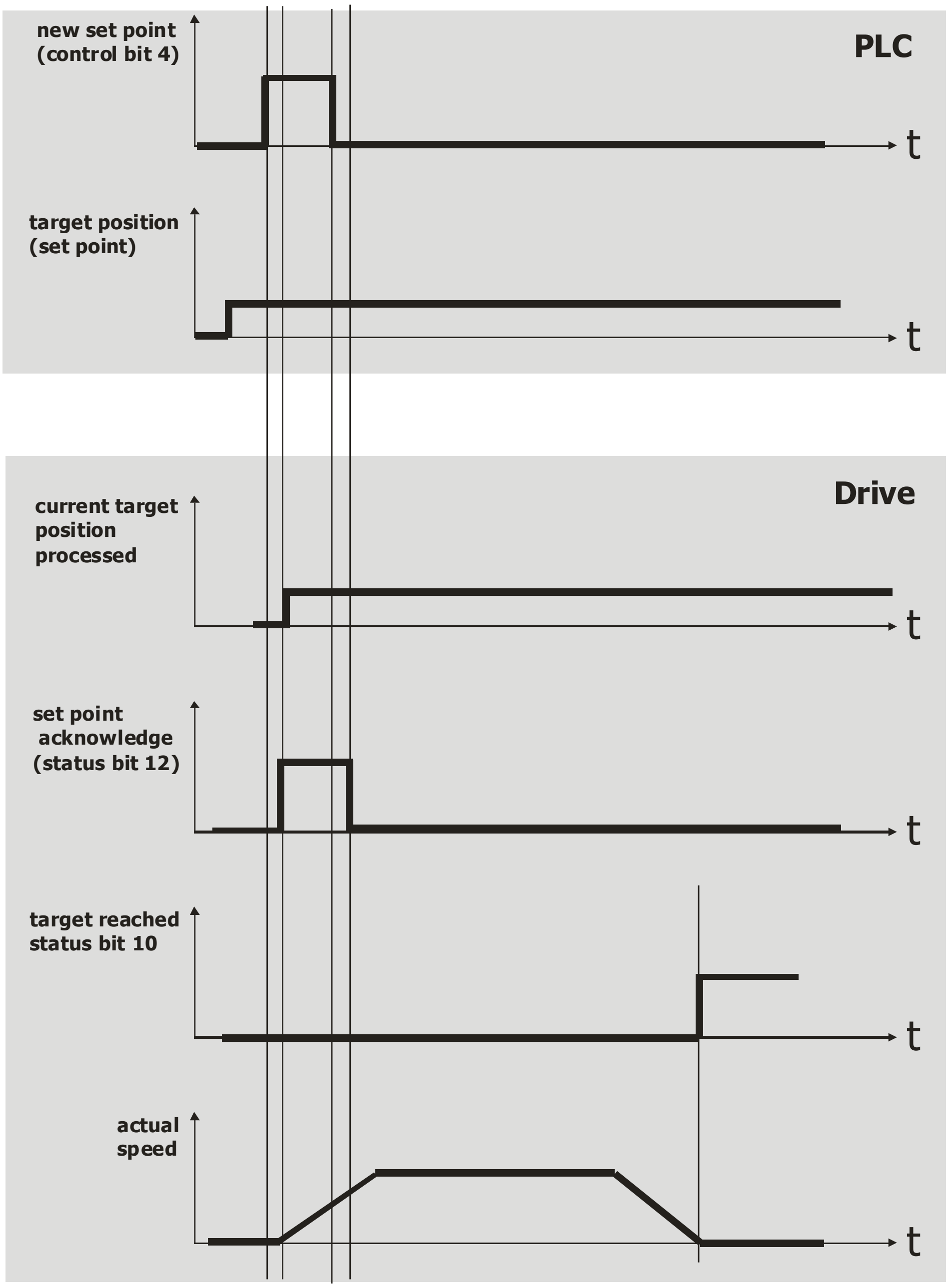
Status word

Identification	**Value**	**Description**
Vl target reached **Bit 10**	0	Stop = 0:*target position* (still) not reached
		Stop = 1: Axis decelerated
	1	Stop = 0: *target position* reached
		Stop = 1: Speed of axis is 0
Set-point acknowledge **Bit 12**	0	The travel profile calculation has not applied the position value (yet).
	1	The travel profile calculation has applied the position value.
Following error **Bit 13**	0	No following error
	1	Following error

Example: single set-point

- control bit change on set-point = `0`
- control bit change set immediately = `0`

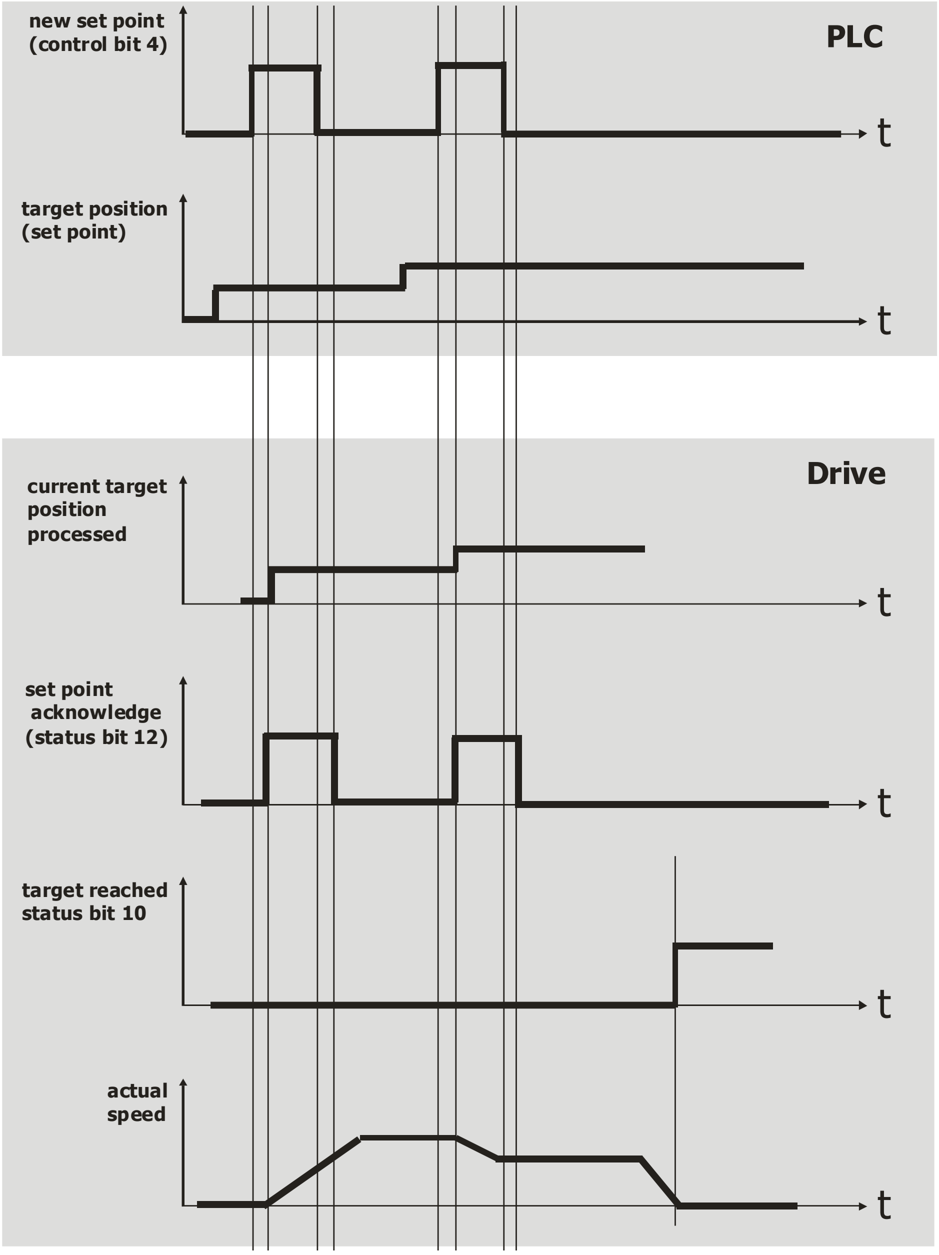
Once a reference value has been transmitted to the drive, the controller signals a permissible value in the control word by a rising signal edge for the bit "New reference value". The drive responds by setting the bit "Reference value confirmed" and starts moving to the new vl target position. After that, the controller resets the bit "New reference value", and the drive resets the bit "Reference value confirmed". Once the bit "Reference value confirmed" has been reset, the drive is ready for receiving a new vl target position.



Example: single set-point

- control bit change on set-point = 0
- control bit change set immediately = 1

A new reference value is confirmed by the control bit "New reference value" (rising edge) while a reference value is being processed. The new reference value is processed immediately.



Example: set of set-points

- control bit change on set-point = `0/1`
- control bit change set immediately = `0`

The travel profile is changed during an active positioning operation.

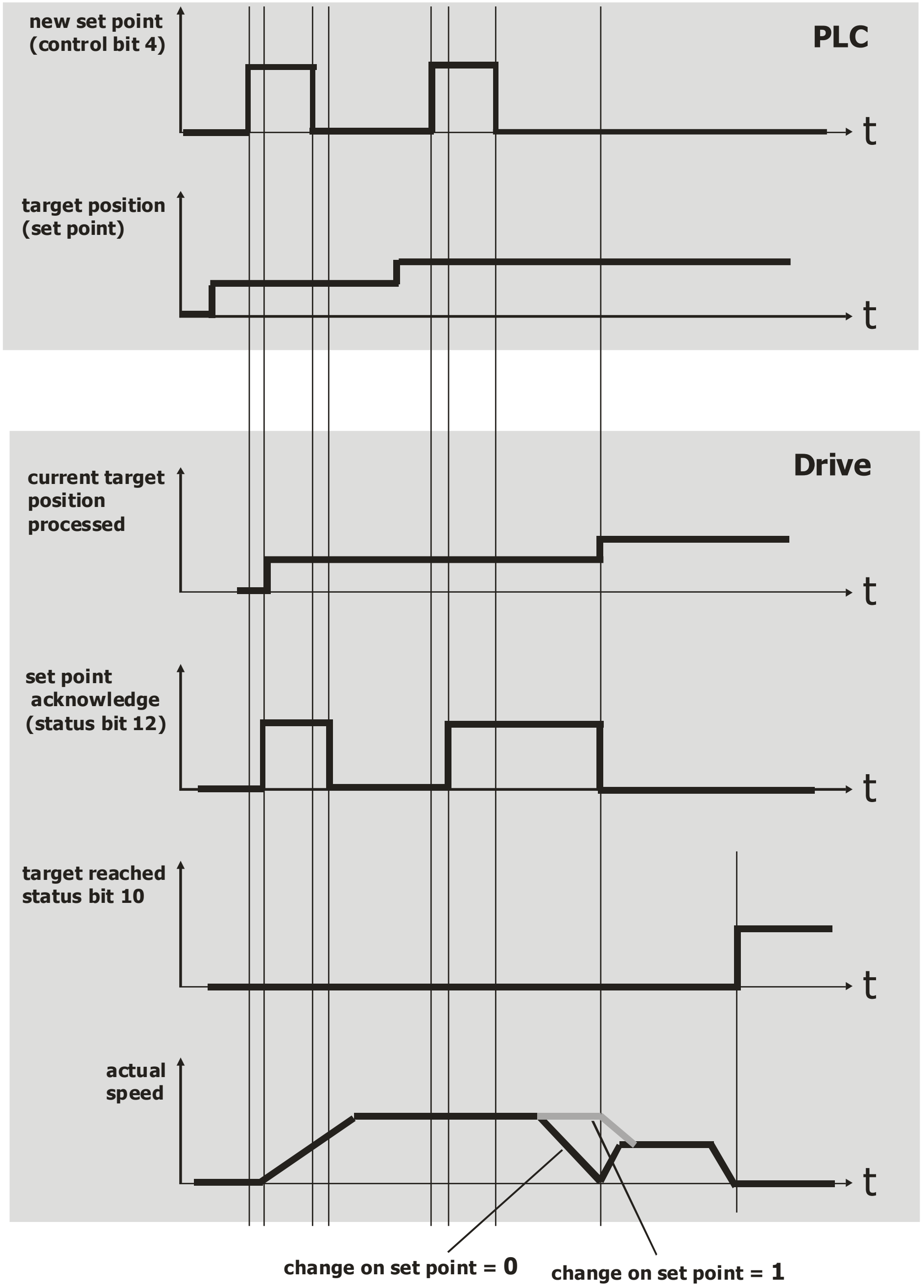
Change on set point = 0

The current target position is approached with a **Stop**. Once the position has been reached, the new reference value is set.

Change on set point = 1

The current target position is approached at the active speed. Once the current vl target position has been reached, the new reference value is applied without reducing the speed to zero.

The gray line in the segment "Actual speed" shows the speed behaviors when the control bit "Switch at reference value" is set to 1.


PLC
new set point (control bit 4)
target position (set point)
Drive
current target position processed
set point acknowledge (status bit 12)
target reached status bit 10
actual speed
t
change on set point = 0
change on set point = 1

Sequence example

In order to start "Profile position mode", the correct sequence must be sent by the PLC.

1 1	Control word = Status word =	0x0000 0x0050	Disable voltage Activation disabled
2	Modes of Operation =	1	Profile position mode
3	Control word = Status word =	0x0006 0x0031	Stop Ready for activation
4	Control word = Status word =	0x0007 0x0033	Start Active
5	Control word = Status word =	0x0007 ↓ 0x000F 0xnn37	Enable operation. Positioning operation is not started. Operation enabled

6a	Control word =	0x0007 or 0x000F ↘ ↙ 0x001F	Operation enabled, start **absolute** positioning with profile[1]. If a positioning operation is already in process, this operation will be completed. Then, the new profile will be used.
	Status word =	0xnn37	Operation enabled
6b	Control word =	0x0007 or 0x000F ↘ ↙ 0x005F	Operation enabled, start **relative** positioning with profile[1]. If a positioning operation is already in process, this operation will be completed. Then, the new profile will be used.
	Status word =	0xnn37	Operation enabled
6c	Control word =	0x0007 or 0x000F ↘ ↙ 0x003F	Operation enabled, start **absolute** positioning with profile[1]. Running positioning operations will change and apply the new profile
	Status word =	0xnn37	Operation enabled
6d	Control word =	0x0007 or 0x000F ↘ ↙ 0x007F	Operation enabled, start **relative** positioning with profile[1]. Running positioning operations will change and apply the new profile
	Status word =	0xnn37	Operation enabled
7	Control word =	0x01nF	HALT: The drive is stopped with ramp 0x6049 *vl velocity deceleration*.
	Status word =	0xnn37	Operation enabled

[1] A profile consists of the following entries. If a value is not changed, the old value will still be active:

- 0x607A *Target Position*
- 0x6081 *Profile velocity*
- 0x6083 *Profile acceleration*
- 0x6084 *Profile deceleration*

WARNING

Dangerous state due to new mode!

When 0x6060 *Modes of Operation* is changed during operation (Control word = 0xnnnF), a dangerous state can occur in the new mode.

- Checking the status word before changing 0x6060 *Modes of Operation* (i.e. check state 0xnn33).

After the sequence of the first four Control word s was processed correctly, the ANG is enabled (table area with a darker background).

In "Operation enabled" state (0xnnnF) the Motion Control states can be changed (white marked area in table).

With the control word transition from 0xnnnF to 0x0007 the velocity mode is stopped. After that it is possible to start again with 0xnnnF.

While 0x0007 is active, it is also possible to change the modes of operation without any danger. After changing 0x6060 *modes of operation* to another value you can start the new operation mode with the according sequence.

In order to start a profile, you don’t have to set the control word to 0x0007 first before switching to 0xnnnF.

Once a profile has been processed, a new profile can be started with the bit "New Setpoint" (bit 4) in control word 0xnnnF.

While a profile is being processed, you can start a new profile without stopping by using the bits "Change Setpoint immediately" (bit 5) and "New Setpoint" (bit 4).

12.4.4 Homing mode

Homing mode can be selected via object 0x6060/0 *Modes of operation* = **6**. In homing mode, the frequency inverter moves the drive to a reference position. The method used for this movement is defined by parameter 0x6098 *homing method*.

Relevant objects:

0x6040	Control word
0x6041	Status word
0x6046	*vl* velocity min max amount
0x6060	Modes of operation
0x6061	Modes of operation display
0x6098	Homing method
0x6099	Homing speeds
0x609A	Homing acceleration

The Ramp Rise/Fall times are set up via parameters **1135**.

In homing, the mode-specific bits of the control word and the status word are used as follows:

Control word *Modes of operation* =6			Status word	
Bit			**Bit**	
0	Switch on		**0**	Ready to switch on
1	Enable voltage		**1**	Switched on
2	Quick stop (Low active)		**2**	Operation enabled
3	Enable operation		**3**	Fault
4	Homing operation start		**4**	Voltage enabled
5	-		**5**	Quick stop (Low active)
6	-		**6**	Switch on disabled
7	Fault reset		**7**	Warning
8	Halt		**8**	-
9	-		**9**	Remote
10	-		**10**	Target reached
11	-		**11**	Internal limit active
12	-		**12**	Homing attained
13	-		**13**	Homing error
14	-		**14**	-
15	-		**15**	Warning 2

Control word

Identification	**Value**	**Description**
Homing operation start **Bit 4**	0	Homing not active
	0 → 1	Start homing
	1	Homing active
	1 → 0	Stop homing
Halt **Bit 8**	0	Execute command from bit 4 "Start homing"
	1	Stop axis with acceleration value (as deceleration) for homing. (The frequency inverter remains enabled in "Operation enabled" status.)

Status word

Identification	**Value**	**Description**
Vl target reached **Bit 10**	0	Stop = 0: Home position (still) not reached
		Stop = 1: Axle decelerated
	1	Stop = 0: Home position reached
		Stop = 1: Axle has speed 0
Homing attained **Bit 12**	0	Homing not completed yet
	1	Homing completed successfully
Homing error **Bit 13**	0	No homing error
	1	Homing error occurred, homing not completed successfully

For an exact description of the various homing modes refer to the application manual "Positioning".

Sequence example

In order to start "homing mode", the correct sequence must be sent by the PLC.

1 1	Control word = Status word =	0x0000 0x0050	Disable voltage Switch On Disabled
2	Modes of Operation =	6	(Homing)
3	Control word = Status word =	0x0006 0x0031	Shutdown Ready to switch on
4	Control word = Status word =	0x0007 0x0033	Switch On Switched On
5	Control word = Status word =	0x000F 0xnn37	Enable Operation. Operation enabled
6a	Control word = Status word =	0x001F 0x1n37	Enable Operation and start Homing. Operation enabled and homing attained.

WARNING

Dangerous state due to new mode!

When 0x6060 *Modes of Operation* is changed during operation (Control word = 0xnnnF), a dangerous state can occur in the new mode.

- Checking the status word before changing 0x6060 *Modes of Operation* (i.e. check state 0xnn33).

Once the sequence of the first four status words has been processed correctly, the ANG is ready for operation (table area with darker background).

In state "operation enabled" (0xnnnF), the state of the Motion Control can be changed (white table area).

With control word transition from 0x0007 (or 0x000F) to 0x001F the homing operation is started. Bit "Homing attained" (Bit 12) returns the status in the status word.

As long as 0x0007 is active, the mode of operation can also be changed safely. Once *modes of operation* has been set to another value, operation can be started with a corresponding sequence.

12.4.5 Table travel record

Table travel record mode can be selected via object 0x6060/0 *Modes of operation* = **0xFF** =**-1**. Table travel record mode used pre-defined positions. Each vl target position is defined by a motion block. Several motion blocks can be defined.

For a description of motion blocks, refer to the "Positioning" application manual.

Relevant objects:

0x6040	Control word	0x6064	Position actual value
0x6041	Status word	0x6065	Following error window
0x6046	*vl* velocity min max amount	0x6066	Following error time
0x6060	Modes of operation	0x6067	Position window
0x6061	Modes of operation display	0x6068	Position window time
0x5FF0	Active motion block	0x6085	Quick stop deceleration
0x5FF1	Motion block to resume		

In table travel record mode, the mode-specific bits of the control word and the status word are used as follows:

Control word *Modes of operation* =-1	
Bit	
0	Switch on
1	Enable voltage
2	Quick stop (Low active)
3	Enable operation
4	Sequence mode
5	-
6	Resume
7	Fault reset
8	Halt
9	Start motion block
10	-
11	Motion block select 0
12	Motion block select 1
13	Motion block select 2
14	Motion block select 3
15	Motion block select 4

Status word	
Bit	
0	Ready to switch on
1	Switched on
2	Operation enabled
3	Fault
4	Voltage enabled
5	Quick stop (Low active)
6	Switch on disabled
7	Warning
8	Motion block in progress
9	Remote
10	Target reached
11	Internal limit active
12	In gear
13	Following error
14	-
15	Warning 2

Control word

Identification	Value	Description
Sequence mode **Bit 4**	0	Single motion block
	1	Automatic sequence
Resume **Bit 6**	0	Start motion block = motion block switching
	1	Start motion block = last active motion block The motion block to be resumed can be read out via object 0x5FF0.
Halt **Bit 8**	0	Execute command from bit 4 "Sequence mode"
	1	Stop axis with ramp of current motion block The frequency inverter remains in "Operation – enabled" status.
Start motion block **Bit 9**	0	Stop axis with ramp of current motion block
	0 → 1	Execute motion block(s)
Motion block select 0…4 **Bit 11…15**	n	Start motion block = n + 1

Motion block select:

Control word															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Motion block select						Sta	Halt		Res		Seq				
4	3	2	1	0											

Start motion block = motion block select + 1:

Motion block select					resulting start motion block
4	3	2	1	0	
0	0	0	0	0	1
0	0	0	1	1	4
1	0	0	0	0	17
1	1	1	1	1	32

Status word

Identification	Value	Description	
Motion block in progress **Bit 8**	0	Single motion Automatic Sequence	Motion block complete Sequence completed
	1	Single motion/automatic sequence active	
Target reached **Bit 10**	0	Halt = 0:	Vl target position not reached yet (only motion blocks with positioning)
		Halt = 1:	Axle decelerated
	1	Halt = 0:	Vl target position reached (only motion blocks with positioning)

		Halt = 1: Axle has velocity 0
In gear **Bit 12**	0	Electronic gear not coupled
	1	Electronic gear coupled
Following error **Bit 13**	0	No following error
	1	Following error

Basic functions

The control bit "Automatic sequence" defines if a single motion (*Automatic sequence* = 0) or and automatic motion block sequence (*Automatic sequence* = 1) is to be executed.

In both cases, the selection of the required motion block (motion block number of single motion or start motion block number of automatic sequence) is calculated by the motion block switching feature with the rising edge of "*Start motion block"*.

"Motion block is being processed" is set to "1" while a selected motion block or an automatic sequence is being executed. "Motion block is being processed" will remain set to "1" until the motion block sequence is complete. When a single motion block is executed, "Motion block is being processed" will be set to "0" once the single motion block is complete. When an automatic sequence is executed, "Motion block is being processed" will be set to "0" once a motion block with setting 0 for Next motion block (end of motion block), -1 (error stop), -2 (Stop, error) or -3 (emergency stop, error) is reached.

During the automatic processing of motion blocks, the currently processed motion block is indicated by object 0x5FF0 *active motion block*.

If the execution of motion blocks is interrupted by setting "Start motion block" to "0", the drive will stop with the ramp set in the current motion block. The interrupted motion block or automatic motion block sequence can be continued by setting "Resume" and a rising signal edge for "Start motion block". If "Resume" is set to "1" and no valid motion block is available, the motion block selected by the motion block switching function will be used. A valid motion block is indicated by object 0x5FF1 *motion block to resume*. *Motion block to resume* displays reads -1, if no valid motion block is present or if the last motion block or motion block sequence was not interrupted.

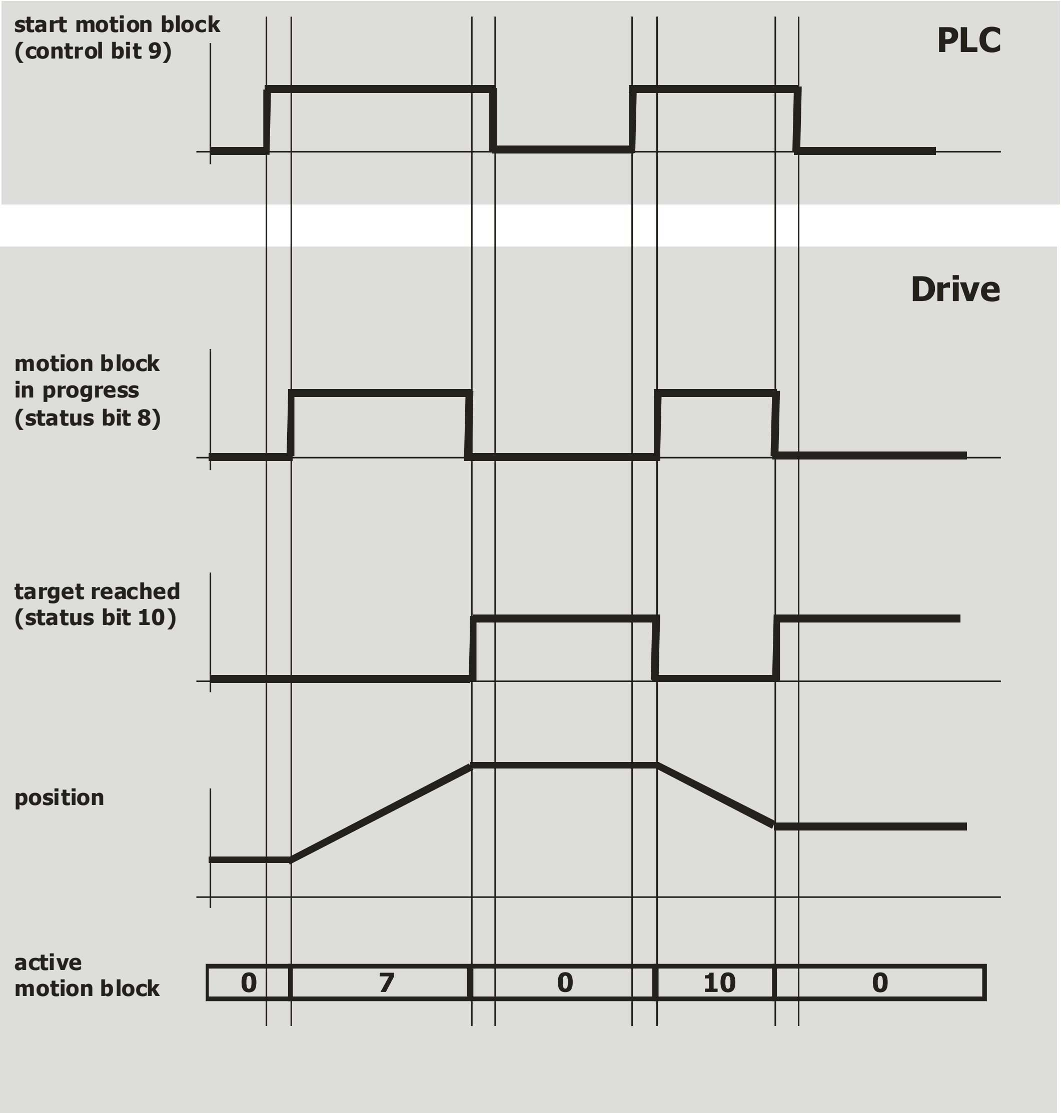
"Vl target reached" is set if the actual position of motion blocks with absolute or relative positioning reaches the *position window*.

"In Gear" is set when the electronic gear function is used and the electronic gear is coupled (synchronous running).

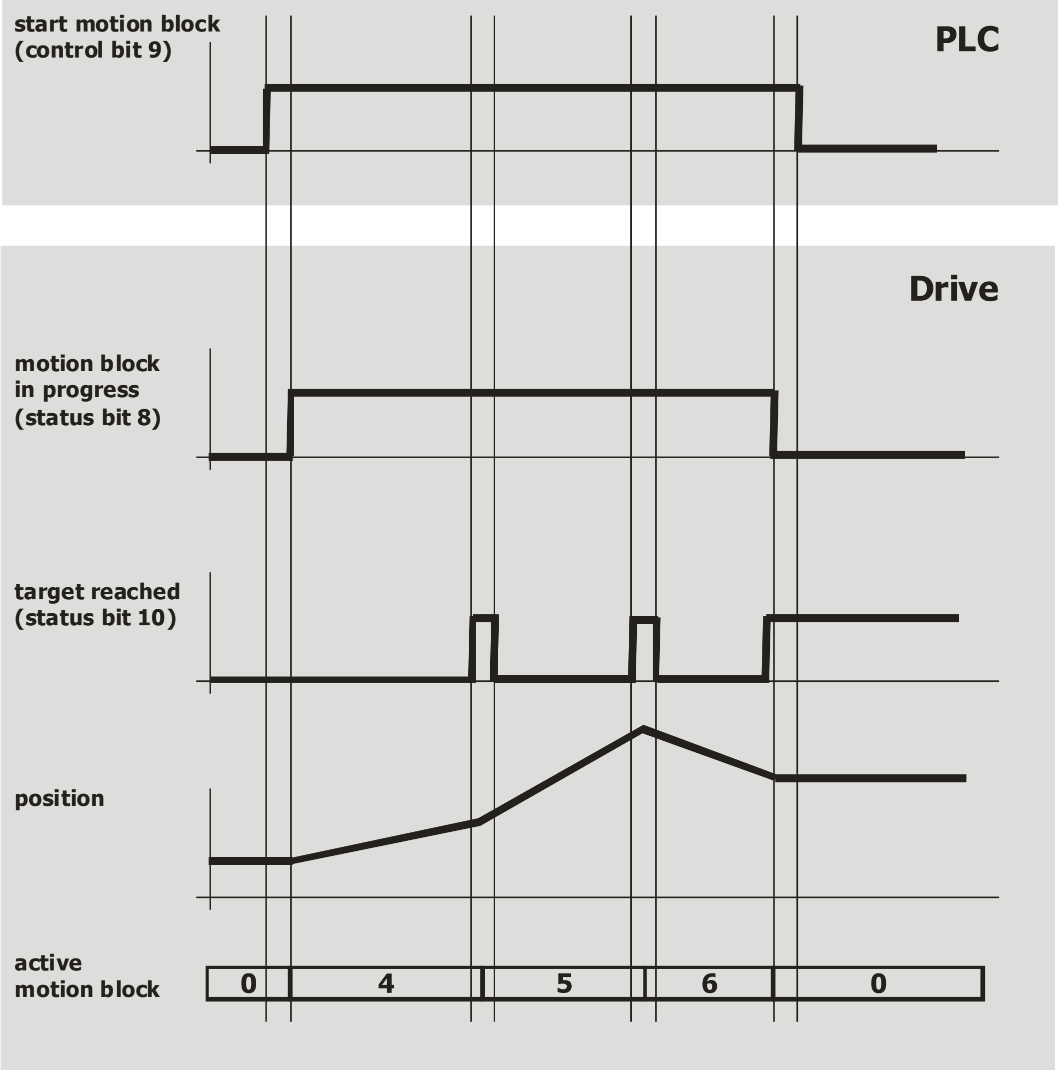
Setting *Halt* to *"*1" will stop a currently executed motion block. The axle is stopped th the ramp set in the current motion block. "Vl target reached" is set to "1" when the speed reaches value 0. The drive remains in "Operation – enabled" status. To continue the interrupted motion block, reset *Halt* to "0".

Examples:

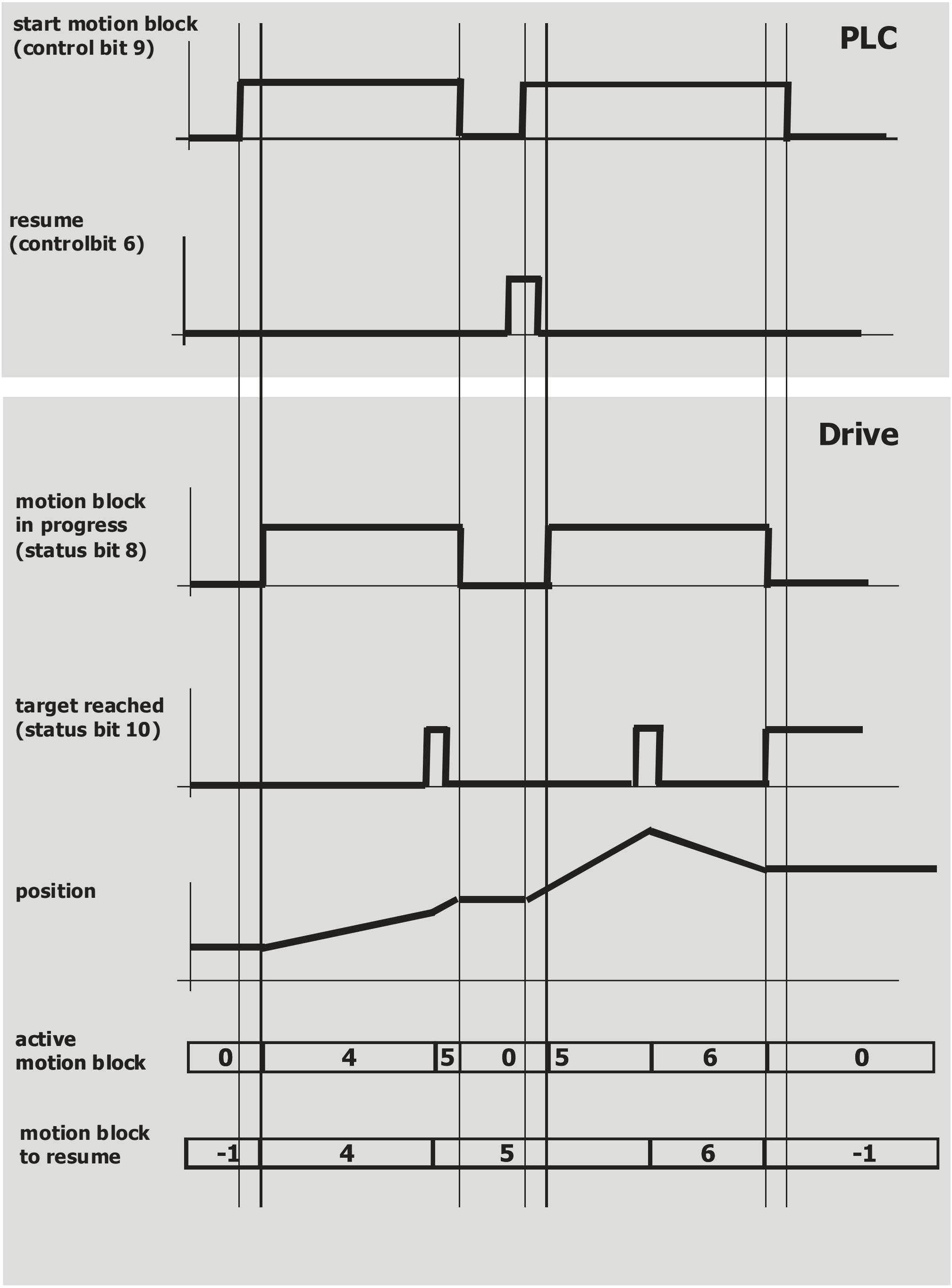
"single motion block" **sequence mode (control bit 4) = 0**
2 motion blocks 7 + 10



" motion block sequence" sequence mode (control bit 4) = 1
sequence = motion block 4, 5, 6



“interrupted motion block sequence” **sequence mode (control bit 4) = 1**
sequence = motion block 4, 5, 6
motion block 5 interrupted



Sequence Example

In order to start "Table travel record mode", the correct sequence must be sent by the PLC.

1 1	Control word = Status word =	0x0000 0x0050	Disable voltage Switch On Disabled
2	Modes of Operation =	-1	(Table travel record mode)
3	Control word = Status word =	0x0006 0x0031	Shutdown Ready to switch on
4	Control word = Status word =	0x0007 0x0033	Switch On Switched On
5a	Control word =	0x000F	Enable Operation.

	Status word =	0xnn37	Operation enabled
5b	Control word = Status word = Status word =	0x020F 0xn337 0xn637	Start Motion Block 1 as Single Motion Block. Operation enabled and Positioning active. Operation enabled and Target reached.
5c	Control word = Status word = Status word =	0x0A0F 0xn337 0xn637	Start Motion Block 2 as Single Motion Block. Operation enabled and Positioning active. Operation enabled and Target reached.
5d	Control word = Status word = Status word =	0x120F 0xn337 0xn637	Start Motion Block 3 as Single Motion Block. Operation enabled and Positioning active. Operation enabled and Target reached.
5e	Control word = Status word = Status word =	0x021F 0xn337 0xn637	Start Motion Block 1 in Sequence Mode Operation enabled and Positioning active. Operation enabled and Target reached.
5f	Control word = Status word = Status word =	0x004F 0xn337 0xn637	Resume last Motion Block as Single Motion Block Operation enabled and Positioning active. Operation enabled and Target reached.
5g	Control word = Status word = Status word =	0x005F 0xn337 0xn637	Resume last Motion Block in Sequence Mode Operation enabled and Positioning active. Operation enabled and Target reached.

WARNING

Dangerous state due to new mode!

When 0x6060 *Modes of Operation* is changed during operation (Control word = 0xnnnF), a dangerous state can occur in the new mode.

- Check the status word before changing 0x6060 *Modes of Operation* (i.e. check state 0xnn33).

Once the sequence of the first four status words has been processed correctly, the ANG is ready for operation (dark table area).

In state "operation enabled" (0xnnnF), the state of the Motion Control can be changed (white table area).

Bit 9 "Start motion block" must be active during positioning. If bit 9 is reset to "0", the positioning operation is interrupted.

As long as 0x0007 is active, the mode of operation can also be changed safely. Once 0x6060 *modes of operation* has been set to another value, operation can be started with a corresponding sequence.

12.4.6 Move away from limit switch mode

Move away from limit switch mode can be selected via object 0x6060/0 *Modes of operation* = **0xFE** = -2.

In *Move away from limit switch* mode, the drive moves back from a triggered limit switch to the permissible travel range.

Relevant objects:

0x6040	Control word	0x6085	Quick stop deceleration
0x6041	Status word	0x6099/2	Homing: Creep Speed
0x6046	*vl* velocity min max amount amount	0x609A	Homing: Acceleration
0x6060	Modes of operation	0x6085	Quick stop deceleration
0x6061	Modes of operation display		

In table travel mode the "operation mode specific" and "manufacturer specific" bits of *Control word* and *Status word* are used as shown:

Control word *Modes of operation* =-2			Status word	
Bit			**Bit**	
0	Switch on		**0**	Ready to switch on
1	Enable voltage		**1**	Switched on
2	Quick stop (Low active)		**2**	Operation enabled
3	Enable operation		**3**	Fault
4	Move away from limit switch		**4**	Voltage enabled
5	-		**5**	Quick stop (Low active)
6	-		**6**	Switch on disabled
7	Fault reset		**7**	Warning
8	Halt		**8**	
9	-		**9**	Remote
10	-		**10**	Target reached
11	-		**11**	Internal limit active
12	-		**12**	-
13	-		**13**	-
14	-		**14**	-
15	-		**15**	Warning 2

NOTICE

Equipment damage!

"Move away from limit switch mode" must not be used, if one of the following error messages occurs:

– F1444 Pos. limit switch < Neg. limit switch

– F1445 Both limit switches at the same time

– F1446 Wrong limit switch wiring

If one of these errors has occurred, the wiring and parameter settings must be checked first before resuming operation.

Control word

Identification	Value	Description
Move away from limit switch **Bit 4**	0	Do not start or stop movement
	1	Start (or resume) movement from limit switch to travel range
Halt **Bit 8**	0	Execute command from bit 4 "Move away from limit switch"
	1	Stop axis with ramp of current motion block (The frequency inverter remains enabled in "Operation enabled" status.)

Status word

Name	Value	Description	
Target reached **Bit 10**	0	Halt = 0:	Limit switch still active
		Halt = 1:	Axle decelerated
	1	Halt = 0:	Limit switch cleared
		Halt = 1:	Axle has speed 0

Basic functions

In mode -2 "Move away from limit switch", the drive is cleared from a triggered hardware limit switch. The direction of rotation depends on the active limit switch: If the positive limit switch is active, the drive moves to negative direction and vice versa.

"`Move away from limit switch`" mode is started in status "`Operation enabled`" by control word bit **4** "*Move away from limit switch*". The drive is accelerated with the ramp from Object 0x609A *Homing acceleration* to the speed set in Object 0x6099 *Homing speeds / Subindex 2* "Homing Speed 2 - search for zero" Once the active limit switch has been cleared, the drive is stopped. Once speed 0 has been reached, status word bit **10** "*Vl target reached*" will be set.

When both directions of rotation are blocked, e.g. due to simultaneous triggering of positive and negative limit switch, error message "`F1449 Both directions locked`" is triggered. In this case, the function "Move away from limit switch" cannot be used.

NOTICE

In the clearing phase of a hardware limit switch, the hysteresis defined in parameter *Hysteresis* **1149** will be active. After detection of the limit switch edge, the axis will be moved on, at least by the defined hysteresis distance.

Setting *Halt* to "1" will stop the started clearing operation. The axis will be stopped. Status bit "Vl target reached" is set to "1" when the speed reaches value 0. The drive remains in "Operation enabled" status. By resetting *Halt* to "0", the interrupted clearing operation will be continued, and "Vl target reached" will be reset to "0".

Sequence example

In order to clear the limit switches, the correct sequence must be sent by the PLC.

1 1	Control word = Status word =	0x0000 0x0050	Disable voltage Switch On Disabled
2	Modes of Operation =	-2	(Move away from limit switch)
3	Control word = Status word =	0x0006 0x0031	Shutdown Ready to switch on
4	Control word = Status word =	0x0007 0x0033	Switch On Switched On
5	Control word = Status word =	0x000F 0xnn37	Enable Operation. Operation enabled
6	Control word = Status word = Status word =	0x001F 0xn2B7 0xn637	Move away from limit switch. Operation enabled and Positioning active. Operation enabled and Limit switch no longer active (Target reached).

WARNING

Dangerous state due to new mode!

When 0x6060 *Modes of Operation* is changed during operation (Control word = 0xnnnF), a dangerous state can occur in the new mode.

- Checking the status word before changing 0x6060 *Modes of Operation* (i.e. check state 0xnn33).

Once the sequence of the first four status words has been processed correctly, the ANG is ready for operation (dark table area).

In state "operation enabled" (0xnnnF), the state of the Motion Control can be changed (white table area).

Bit 4 "Move away from limit switch" must be active in the clearing phase. If bit 4 is reset to "0", the clearing operation is interrupted.

As long as 0x0007 is active, the mode of operation can also be changed safelyOnce 0x6060 *modes of operation* has been set to another value, operation can be started with a corresponding sequence.

12.4.7 Electronic Gear: Slave

The mode *Electronic gear: Slave* can be selected via *modes of operation* = **0xFD** = `-3`.

In *Electronic gear slave* mode the drive follows a master drive as a slave drive.

Relevant objects:

0x6040	Control word	0x6064	Position actual value
0x6041	Status word	0x6065	Following error window
0x6060	Modes of operation	0x6066	Following error time
0x6046	*v* velocity min max amount	0x6067	Position window
0x6061	Modes of operation display	0x6068	Position window time
0x5F10	Electronic Gear: Gear factor	0x6085	Quick stop deceleration
0x5F11	Electronic Gear: Phasing 1	0x5F18	M/S Synchronization offset
…	…		
0x5F14	Electronic Gear: Phasing 4		

In *Electronic gear: Slave* mode, the operation mode specific bits of the control word and the status word are used as follows:

Control word *Modes of operation* =-3		Status word	
Bit		**Bit**	
0	Switch on	**0**	Ready to switch on
1	Enable voltage	**1**	Switched on
2	Quick stop (Low active)	**2**	Operation enabled
3	Enable operation	**3**	Fault
4	Start Gearing	**4**	Voltage enabled
5	Start M/S Correction	**5**	Quick stop (Low active)
6	Direct sync	**6**	Switch on disabled
7	Fault reset	**7**	Warning
8	Halt	**8**	Phasing Done or M/S Correction Done
9	Start Phasing	**9**	Remote
10	-	**10**	Target reached / in gear
11	-	**11**	Internal limit active
12	Phasing switch over 0	**12**	M/S Position Correction successful
13	Phasing switch over 1	**13**	Following error
14	-	**14**	-
15	-	**15**	Warning 2

WARNING

Dangerous state due to faulty parameterization

The function Mater/Slave Position Correction is only allowed to be used after complete setup of this function.

- Check for parameter setup chapter 12.4.7.2 "Master/Slave Position Correction".

Control word

Name	Value	Description
Start El. Gear **Bit 4**	0	Stop drive with ramp 0x6084
	1	Start electronic gear with master speed reference value with ramp 0x6083
Start M/S Correction **Bit 5**	0	M/S Correction not started.
	1	Start Master/Slave Position correction. See chapter 12.4.7.2.
Direct Sync **Bit 6**	0	Direct Synchronisation enabled.
	1	Direct Synchronisation disabled.
Halt **Bit 8**	0	Execute command from bit 4 "Start el. Gear"
	1	Stop axis with ramp of current motion block The frequency inverter remains in "Operation enabled" status.
Start Phasing **Bit 9**	0	Phasing disabled / aborted
	1	Start Phasing with profile defined by Bits 12 & 13
Phasing select 0…1 **Bit 12…13**	n	Phasing Profile= n + 1

Phasing select

Control word															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
		Ph.-Sel.				Pha	Halt		DS		SG				
		1	0												

Phasing Profile= Phasing switch over +1

Phasing select		Phasing Profile
Bit 13	Bit 12	
0	0	1
0	1	2
1	0	3
1	1	4

Status word

Identification	Value	Description
Phasing Done **Bit 8**	0	Phasing in process or not started yet. Or M/S correction in process or not started yet.
	1	Phasing done. Or M/S correction done.
Target reached/InGear **Bit 10**	0	Halt (Control bit 8) = 0: Electronic gear (still) not in gear
		Halt (Control bit 8) = 1: Axis decelerated
	1	Halt (Control bit 8) = 0: Electronic gear in gear
		Halt (Control bit 8) = 1 Axis has speed 0
M/S Position Correction successful **Bit 12**	0	M/S Correction is running or wasn't started yet.
	1	M/S Correction finished. See chapter 12.4.7.2.
Following error **Bit 13**	0	No following error
	1	Following error

Basic functions

Mode "-3 Electronic gear: Slave" implements a mode for a slave drive in the electronic gear to a master drive. The master of the electronic gear must be connected to the slave via signal cables or Systembus (recommended). The master input is selected in the Slave via parameter *Master position source* **1122** .

Master Position Source 1122	Function
0 - Off	No source selected.
1 - Encoder 1	The current speed and position of the master drive is taken over from encoder input 1.
2 - Encoder 2 / Resolver	The current speed and position of the master drive is taken over from encoder input 2 or resolver.

11 - RxPDO1.Long1 extrapolated	The current position of the master drive is taken over by the process data channel RxPDO1.Long1 of the systembus. Additionally, the data received are extrapolated, even for slow settings of TxPDO Time of the master. Depending on the application, select a setting of the corresponding TxPDO.Long of the master: "606 – Internal act. Position (16/16)", mechanical position of master drive. Value **will not change abruptly** when a homing operation of the master drive is completed. "607 – Act. Position (16/16)", mechanical position of master drive. Value will jump when the master drive carries out a homing operation. "620 – motion profile gen.: internal reference position", reference position of master drive; advantage: Improved controller properties. Value **will not change abruptly** when a homing operation of the master drive is completed. "627 - Motion profile gen.: reference position", reference position of master drive; advantage: Improved controller properties. Value will jump when the master drive carries out a homing operation. Settings 607 and 627 are only to be used in exceptional situations. In most applications, source 606 or 620 is the better setting.

In setting "11 - RxPDO1.Long1 extrapolated" of parameter *Master position source* **1122**, the *Operation mode* **1180** of the systembus synchronization must be set to 1 or 10 to ensure reliable functional operation.

Operation mode 1180	
0 -	Off [1]
1 -	RxPDO1 [2]
2 -	RxPDO2 [3]
3 -	RxPDO3 [3]
10 -	SYNC

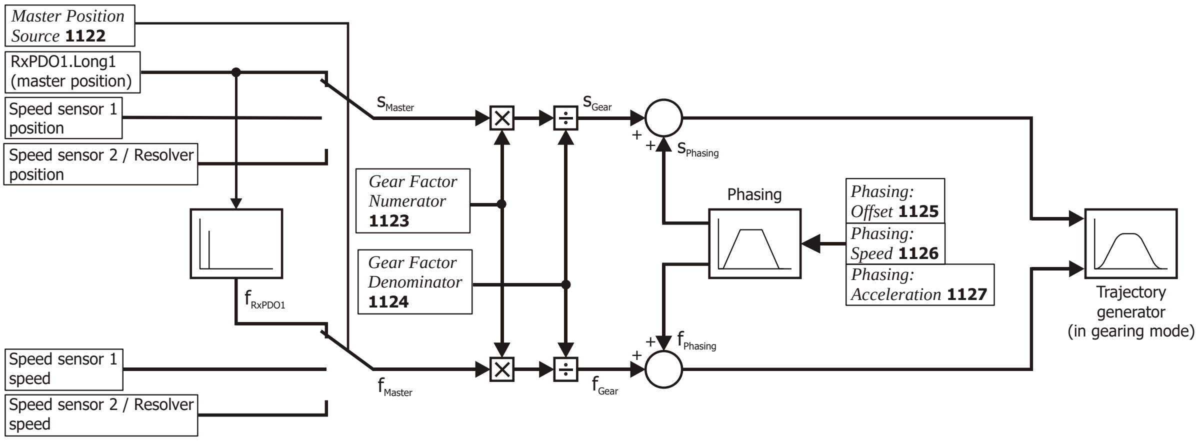
[1] If the error message "F1453 Systembus-Synchronization not activated" is displayed when the slave drive is started, one of the operation modes 1, 2, 3 or 10 must be selected.
[2] Synchronization of processing with data message or cyclic sending of SYNC message.
[3] Not recommended for el. Gear because no extrapolation carried out.

Synchronization between several drives must be performed at high updating rates in order to guarantee optimum results. In the transmitter of the TxPDO object, set a low value for the time (e.g. *TxPDO1 Time* **931**). If you use the SYNC function of Systembus, set parameter *SYNC time* **919** to a lower value.

Note that, due to these settings, the bus load of the systembus must provide for sufficient reserves for proper operation.

Systembus is described in the manuals of the expansion modules with Systembus interface.

Block diagram: electronic gear and phasing function


Master Position Source 1122
RxPDO1.Long1 (master position)
Speed sensor 1 position
Speed sensor 2 / Resolver position
f_{RxPDO1}
Speed sensor 1 speed
Speed sensor 2 / Resolver speed
s_{Master}
f_{Master}
Gear Factor Numerator 1123
Gear Factor Denominator 1124
s_{Gear}
f_{Gear}
+
+
$s_{Phasing}$
$f_{Phasing}$
Phasing
Phasing: Offset 1125
Phasing: Speed 1126
Phasing: Acceleration 1127
Trajectory generator (in gearing mode)

The master position and speed are multiplied by the *gear factor*. When phasing is started, the phasing profile is added to the master speed until the phasing offset is reached.

The *gear factor* is defined by the following Objects or parameters:

Object		Parameter	
0x5F10/1	Gear factor Numerator	**1123**	*Gear Factor Numerator*
0x5F10/2	Gear factor Denominator	**1124**	*Gear Factor Denominator*
0x5F10/3	Gear factor Resync on change	**1142**	*Resync. On Change of Gear-Factor*

The *Phasing* is defined by the following Objects or parameters:

Object		Parameter	
0x5F11/1	Phasing 1: Offset	**1125.1**	*Phasing: Offset*
0x5F12/1	Phasing 2: Offset	**1125.2**	
0x5F13/1	Phasing 3: Offset	**1125.3**	
0x5F14/1	Phasing 4: Offset	**1125.4**	
0x5F11/2	Phasing 1: Speed	**1126.1**	*Phasing: Speed*
0x5F12/2	Phasing 2: Speed	**1126.2**	
0x5F13/2	Phasing 3: Speed	**1126.3**	
0x5F14/2	Phasing 4: Speed	**1126.4**	
0x5F11/3	Phasing 1: Acceleration	**1127.1**	*Phasing: Acceleration*
0x5F12/3	Phasing 2: Acceleration	**1127.2**	
0x5F13/3	Phasing 3: Acceleration	**1127.3**	
0x5F14/3	Phasing 4: Acceleration	**1127.4**	

Start Electronic Gear and Status bits

The electronic gear is started with Bit 4 "Start Electronic Gear". The drive accelerates as specified in Object 0x6083 *Profile acceleration*. Once the slave speed is coupled into the master, status word bit 10 "target reached/Gear in" is set. The conditions for "In Gear" status are set via Objects 0x5F15 *In gear threshold* and 0x5F16 *In gear time*.

"Target reached/In Gear" is set when the electronic gear function is used and electronic gear synchronous running is reached.

Setting *Halt "*1" will stop a currently executed movement. The axle is stopped with 0x6084. "Target reached" is set to "0" to start the deceleration and to "1" when the speed reaches value 0. The drive remains in "Operation – enabled" status. To continue the interrupted movement, reset *Halt* to "0". Bit "target reached" is set to "0" to start the acceleration and to "1" when the conditions for "In Gear" of Vl Objects 0x5F15 *In gear threshold* und 0x5F16 *In gear time* are reached.

Phasing

With the phasing function, the slave position is offset from the master position received by the value of one phasing position 1125.

The Phasing is described in chapter 9.9.4.18 "0x5F11/n…0x5F14/n Phasing 1…4".

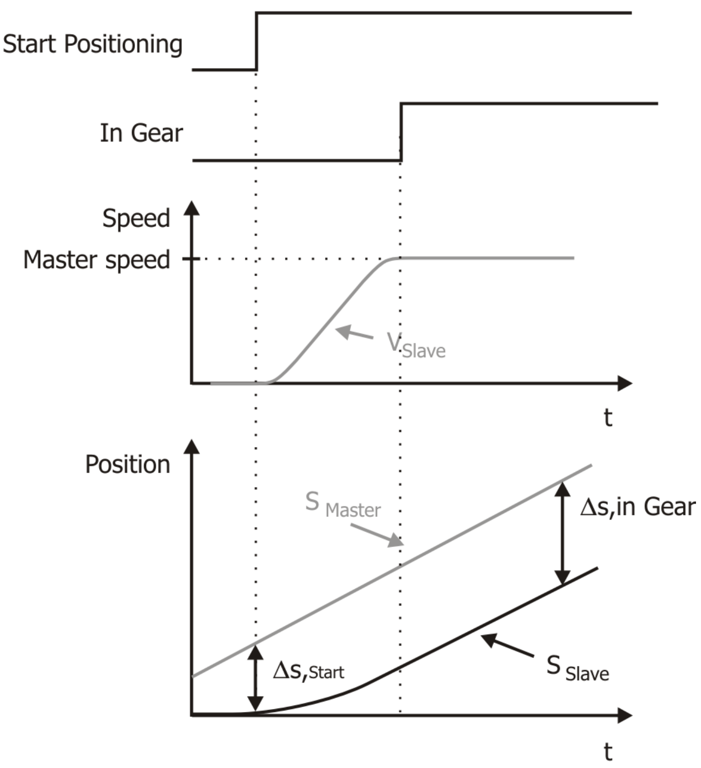
12.4.7.1 Direct Synchronization

Function without Direct Synchronization ("Standard Synchronization")

The drive accelerates the master speed with the ramps parameterized in the motion block. As soon as the master speed is reached for the first time, the drive is synchronized with the master drive. The slave is engaged at the current position and operates at a synchronous angle to the master. In the case of a relative positioning operation, this engaging position is used as the start position.

The acceleration and deceleration for synchronizations follow an S-curve.

The relative position change due to acceleration is not compensated.

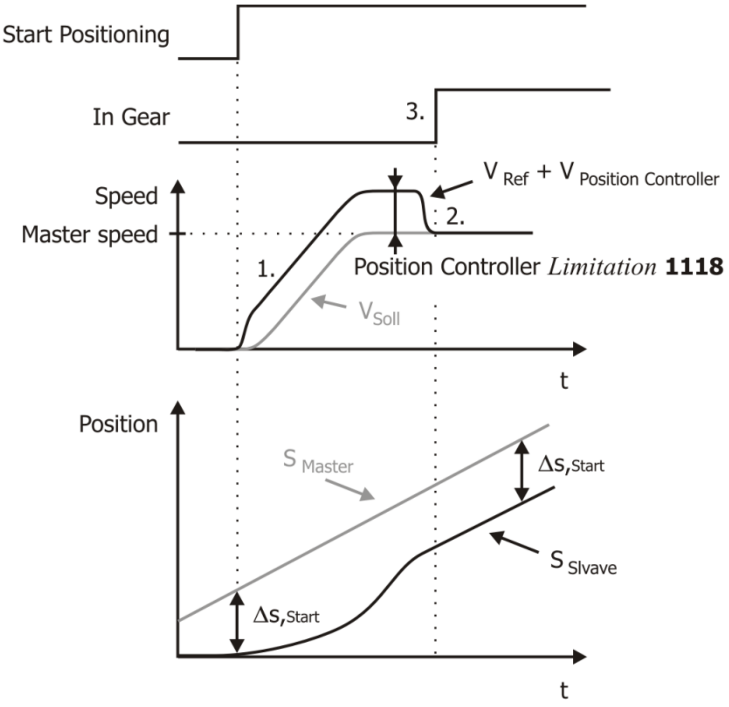


Function with Direct Synchronization

The drive accelerates the master speed with the ramps parameterized in the motion block. When the motion block is started, the drive is synchronized with the master drive directly. The master position is processed directly by the position controller.

The acceleration and deceleration for synchronizations follow an S-curve.

The relative position change due to acceleration is compensated by the position controller.



12.4.7.2 Master/Slave Position Correction

NOTICE

When using this functionality, master drive and slave drive have to use the same mechanical characteristics (i.e. gear transmission ratios) and use the same reference system.

The Master/Slave Position Correction offers as part of the Electronic Gear the possibility to synchronize the absolute Position of the Slave to the absolute Position of the master.

This function is helpful in example in applications, in which drives often work independently from each other and have to work together for certain activities. For example this could be the case in crane applications, where normal loads are operated independently from each other and which are switched together for heavy loads. To speed up the switching together process, the Master/Slave Position correction can be used to synchronize the absolute position of the Slave with the absolute position of the Master.

Additionally by using an Offset a relative reference can be set up in the target position.

Preparations Master drive

The Master drive must be set up as follows:

TxPDO2 Identifier **927 =** 640 (or a different not used Identifier)

TxPDO2 Function **932** = 1 – controlled by time or 2 – controlled by SYNC

TxPDO2.Long1 **964** = 743 – Act. Position [User Units]

Additionally the following parameters must be set according to the electronic gear: *TxPDO1.Long1* **954** corresponding to the description of *Master Position Source* **1122**

TxPDO1 Identifier **925 =** 384 (or a different not used Identifier)

TxPDO1 Function **930** = 1 – controlled by time or 2 – controlled by SYNC

Preparations Slave drive

The Slave drive must be set up as follows:

RxPDO2 Function **926 =** 640 (or the Identifier defined in the Master drive)

Additionally the following parameters must be set according to the electronic gear:

RxPDO1 Function **924 =** 384 (or the Identifier defined in the Master drive)

Source Master position **1122** = 11 – RxPDO1.Long

The function Master/Slave Position Correction expects the Target Position [u] always in RxPD2.Long. When using this function RxPDO2.Long1 and also RxPDO2.Word1, RxPDO2.Word2, RxPDO2.Boolean1 and RxPDO2.Boolean2 are not allowed to be used for any other purpose.

Starting of Master/Slave Position Correction in Slave drive

To start the Master/Slave Position correction at first Bit 4 and then Bit 5 have to be set in the Control word. Bit 5 is only allowed to be set when Bit 10 In Gear is shown in the Status word.

By setting Bit 5 in the Control word the Slave drive is started to position to the Master position + Offset.

The acceleration is done with the object 0x609A/0 0Homing acceleration (or Parameter *Acceleration* **1134**). The used velocity can be set up via 0x6099/1 Homing speed (or Parameter *Fast Speed* **1132**).

As long as the Master/Slave Position correction is executed, Bit 12 is deactivated in the Status word. When the Master/Slave Position correction was finished successfully Bit 12 is set.

During the Correction sequence the Status word bit 8 "Master/Slave Position correction" is set to "Low". As soon as the Master/Slave Position correction is finished or cancelled, the Bit is set to "High". After first switch-on (or after a device reset) the "Phasing Done" bit is also "Low".

Since Bit 8 is also used for Phasing, always the last started function is signaled by this bit.

Offset Reference

The Offset for the M/S Synchronization can be set via Object 0x5F18/0.

Object		Parameter	
0x5F18/0	M/S Synchronization offset	1284	*M/S Synchronization offset*

Application limitations

The function can be used in most of all applications without any limitations. In applications with very long travelling distances the following must be checked:

- The position difference to be compensated must not be greater than 215-1 motor revolutions.
- The position difference to be compensated must not be greater than 231-1 user units.

Depending on the used reference system it can vary, which limit is decisive. Always the smaller limit must be complied with.

A motor with a reference speed of 6000 rpm would have to travel for around 5.5 minutes into one direction to exceed this limit.

12.4.7.3 Sequence example

To start the Electronic Gear: Slave mode, the correct sequence has to be sent from the PLC.

1 1	Control word = Status word =	0x0000 0x0050	Disable voltage Switch On Disabled
2	Modes of Operation =	-3	(Electronic Gear: Slave mode)
3	Control word = Status word =	0x0006 0x0031	Shutdown Ready to switch on
4	Control word = Status word =	0x0007 0x0033	Switch On Switched On
5	Control word = Status word =	0x000F 0xnn37	Enable Operation, Reference speed "0". Operation enabled
6a	Control word = Status word = Status word = Status word = Status word =	0x001F 0xn327 0xn337 0xn727 0xn737	Start Electronic Gear without Direct Synchronisation Operation enabled, Slave not coupled (yet), Phasing not finished. Operation enabled, Slave not coupled (yet), Phasing finished. Operation enabled, Slave coupled, Phasing not (yet) finished. Operation enabled, Slave coupled, Phasing finished.
6b	Control word = Status word =	0x005F See 6a	Start Electronic Gear with Direct Synchronisation See 6a
7a	Control word = Status word =	0x021F See 6a	Start Electronic Gear **without** Direct Synchronisation and Phasing Profile 1 See 6a
7b	Control word = Status word =	0x121F See 6a	Start Electronic Gear without Direct Synchronisation and Phasing Profile 2 See 6a
7c	Control word = Status word =	0x221F See 6a	Start Electronic Gear without Direct Synchronisation and Phasing Profile 3 See 6a
7d	Control word = Status word =	0x321F See 6a	Start Electronic Gear without Direct Synchronisation and Phasing Profile 4 See 6a
8a	Control word = Status word =	0x025F See 6a	Start Electronic Gear **with** Direct Synchronisation and Phasing Profile 1 See 6a
8b	Control word =	0x125F	Start Electronic Gear **with** Direct Synchronisation and Phasing Profile 2

	Status word =	See 6a	See 6a
8c	Control word =	0x225F	Start Electronic Gear **with** Direct Synchronisation and Phasing Profile 3
	Status word =	See 6a	See 6a
8d	Control word =	0x325F	Start Electronic Gear **with** Direct Synchronisation and Phasing Profile 4
	Status word =	See 6a	Disable voltage
9	Control word =	0x001F 0x003F	Enable Operation, the Slave drive synchronizes to the Master position.
	Status word =	0xnn37 0x1n37	Operation enabled M/S Position Correction finished.

WARNING

Dangerous state due to new mode!

When 0x6060 *Modes of Operation* is changed during operation (Control word = 0xnnnF), a dangerous state can occur in the new mode.

- Checking the status word before changing 0x6060 *Modes of Operation* (i.e. check state 0xnn33).

Once the sequence of the first four status words has been processed correctly, the ANG is ready for operation (dark table area).

In state "operation enabled" (0xnnnF), the state of the Motion Control can be changed (white table area).

Bit 4 "Start electronic gear" must be active during the movement. If bit 4 is reset to "0", the movement is interrupted.

As long as 0x0007 is active, the mode of operation can also be changed safely. Once 0x6060 *modes of operation* has been set to another value, operation can be started with a corresponding sequence.

Bit 5 "Start Position Correction" is only allowed to be used when the Slave is in gear (Status word Bit 10).

Bit 5 "Start Position Correction" should be used for optimum results when the master drive does not move.

When Bit 5 of the Control word is reset to "0" the movement is interrupted.

13 Analog input/Analog output MF4

The function of terminal X410B.4 is configured by *Operation mode X410B.4* **502**.

Operation mode X410B.4		Function
1 -	Voltage input 0…10 V	Unipolar Voltage input DC 0…10 V.
2 -	Current input 0…20 mA	Unipolar Current input DC 0…20 mA.
3 -	Digital input	Usage as digital input
4 -	Voltage input -10…10 V	Bipolar Voltage input DC -10…10 V.
5 -	Voltage output 0…10 V	Unipolar Voltage output DC 0…10 V.
11 -	Motor PTC	Evaluation of a Motor PTC. Also consider *Therm.Contact* **204** and *Motortemp. Operation Mode* **570**.
12 -	Motor KTY	Evaluation of a Motor KTY. Also consider *Therm.Contact* **204** and *Motortemp. Operation Mode* **570**.
13 -	Motor PT1000	Evaluation of a Motor PT1000. Also consider *Therm.Contact* **204** und *Motortemp. Betriebsart* **570**.

13.1Analog input MF4IA

The terminal X410B.4 can be used as voltage input or current input.

13.1.1 General

The analog input of the EM-AUT-21 expansion module can optionally be configured as a voltage or a current input. Parameterization of the input signal is done by the definition of a linear characteristic and assignment as a

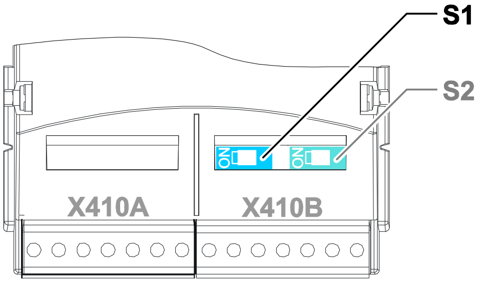
- reference value source
  (can be selected via the parameter *Reference frequency source* **475**),
- reference percentage source
  (can be selected via the parameter *Reference percentage source* **476**),
- actual percentage source
  (can be selected via the parameter *Actual percentage source* **478**, in configuration **x11**) or
- limit value source
  (can be selected via the parameter *Limit Source* **734…737**).

The terminal X410B.4 is assigned as function in the selection tables with MF4 (MF4ID for digital input, MF4IA for analog input, MF4OA for analog output).

13.1.2 Configuration voltage/current input

The terminal X410B.4 can be used optionally as analog input, analog output, PTC, KTY, PT1000 or digital input.

EM-AUT-21: **Analog input/ Analog output MF4:** Switch S1 enables a switchover of the operation mode for an analog current signal of 0… 20 mA between "`Input`" and "`Output`". The selection of current input and voltage input is done via the selection of the operation mode in the software.



Operation mode – switch S1	Function
OFF - Operation mode "input"	OFF (to the right) – MF4IA is configured as an input. The operation modes "analog", "digital", "voltage" or "current" are configured within the software.
ON - Operation mode "output"	ON (to the left) – MF4IA is configured as a voltage output.

13.1.3 Characteristic

The mapping of the analog input signals onto a frequency or percentage reference value is possible for various demands. The parameterization is to be done via two points of the linear characteristic of the reference channel.

The characteristic point 1, with the coordinates X1 and Y1, and the characteristic point 2, with the coordinates X2 and Y2, are to be set in the four data sets.

The characteristic points X1 and X2 are stated as percentages, as the analog input can be switched as a current or voltage input via switch S1.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Default**
564	Characteristic point X1	-100.00 %	100.00 %	-98.00 %
565	Characteristic point Y1	-100.00 %	100.00 %	-100.00 %
566	Characteristic point X2	-100.00 %	100.00 %	98.00 %
567	Characteristic point Y2	-100.00 %	100.00 %	100.00 %

The coordinates of the characteristic points are percentage-related to the analog signal, with 10 V or 20 mA, and the parameter *Maximum Frequency* **419** or parameter *Maximum reference percentage* **519**. The change of direction of rotation can be done via the digital inputs and/or by selecting the characteristic points.

The definition of the analog input characteristic can be calculated via the two-point form of the straight-line equation. The speed Y of the drive mechanism is controlled according to the analog control signal X.

$$Y = \frac{Y2 - Y1}{X2 - X1} \cdot (X - X1) + Y1$$

Monitoring of the analog input signal via the parameter *Error/Warning Behavior* **563** demands a check of the characteristic parameters. Sensible use is only possible if the *Characteristic point X1* **564** is within the positive range.

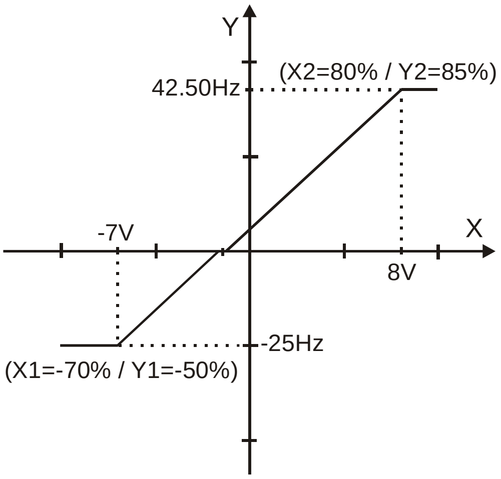
13.1.3.1 Examples

The analog input signal is mapped onto a reference value as a function of the characteristic. The following examples show the operation modes for an analog voltage signal. The parameter *Minimum Frequency* **418** is set to the value 0.00 Hz. The characteristic point 100% for the Y-axis corresponds to the parameter *Maximum Frequency* **419** of 50.00 Hz in the examples.

The various operation modes change the input characteristic as a function of the parameterized characteristic points. In the following examples, the areas of the coordinate system from which a characteristic point is displaced are marked.

Operation mode "4 – Voltage input -10…10 V"

In operation mode "4 – voltage input", the characteristic of the analog input can be freely set by stating two characteristic points.



Characteristic point 1:
X1 = -70.00% · 10 V = -7.00 V
Y1 = -50.00% · 50.00 Hz = -25.00 Hz

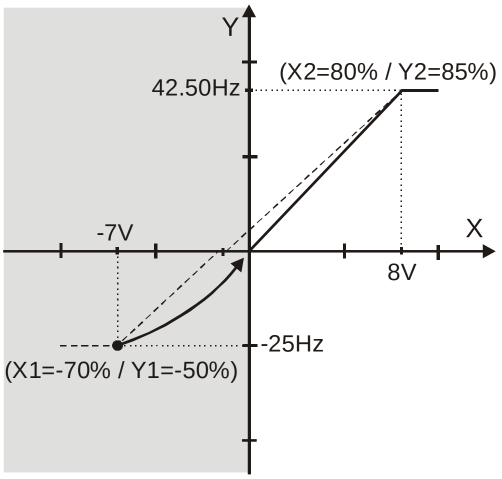
Characteristic point 2:
X2 = 80.00% · 10 V = 8.00 V
Y2 = 85.00% · 50.00 Hz = 42.50 Hz

Tolerance band:
∆X = 2.00% · 10 V = 0.20 V

The change of direction of rotation is done in the example at an analog input signal of -1.44 V, with a tolerance band of ±0.20 V.

Operation mode "1 – Voltage input 0…10 V"

In operation mode "1 – Voltage input", the characteristic points are displaced to the origin of the characteristics with a negative value for the X-axis.



Characteristic point 1:
X1 = -70.00% · 10 V = -7.00 V
Y1 = -50,00 % · 50.00 Hz = -25.00 Hz

Characteristic point 2:
X2 = 80.00% · 10 V = 8.00 V
Y2 = 85.00% · 50.00 Hz = 42.50 Hz

Tolerance band:
∆X = 2.00% · 10 V = 0.20 V

The characteristic point 1 has been displaced to the origin. The parameter *Tolerance band* **560** is not taken into account in this example, as no change of sign of the reference frequency value takes place.

13.1.4 Scaling

The analog input signal is mapped to the freely configurable characteristic. The maximum admissible setting range of the drive mechanism is to be set according to the configuration selected via the frequency limits or the percentage value limits. In the parameterization of a bipolar characteristic, the minimum and maximum limit for both directions of rotation are taken over. The percentage values of the characteristic points are relative to the maximum limits selected.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Factory setting**
418	Minimum frequency	0.00 Hz	599.00 Hz	3.50 Hz [1)] 0.00 Hz [2)]
419	Maximum frequency	0.00 Hz	599.00 Hz	50.00 Hz

The factory settings depend on the setup of parameter *Configuration* **30**:

[1)] 3.50 Hz in Configurations 1xx, 4xx; 6xx
[2)] 0.00 Hz in Configurations 2xx, 5xx

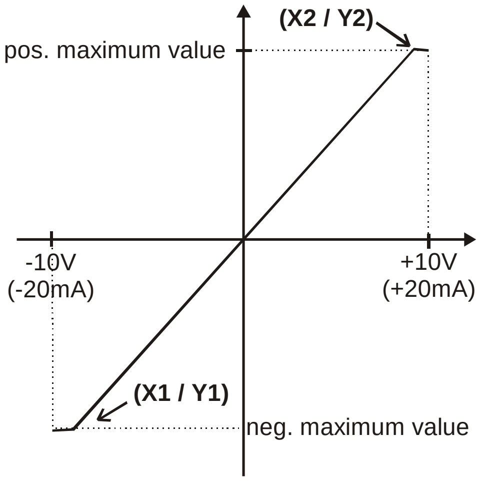
The controls use the maximum value of the output frequency, which is calculated from the parameter *Maximum Frequency* **419** and the compensated slip of the drive mechanism. The frequency limits define the speed range of the drive mechanism and the reference percentage values supplement the scaling of the input characteristic according to the configured functions.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Default**
518	Minimum reference percentage	0.00 %	300.00 %	0.00 %
519	Maximum reference percentage	0.00 %	300.00 %	100.00 %

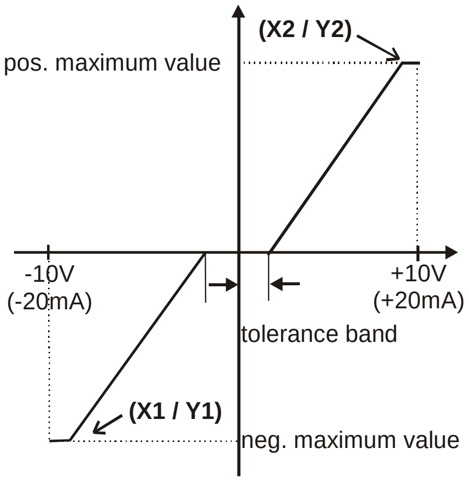
13.1.5 Tolerance band and hysteresis

The analog input characteristic with change of sign of the reference value can be adapted by the parameter *Tolerance band* **560** of the application. The tolerance band to be defined extends the zero crossing of the speed relative to the analog control signal. The percentage parameter value is relative to the maximum current or voltage signal.

Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Default**
560	Tolerance band	0.00 %	25.00 %	2.00 %

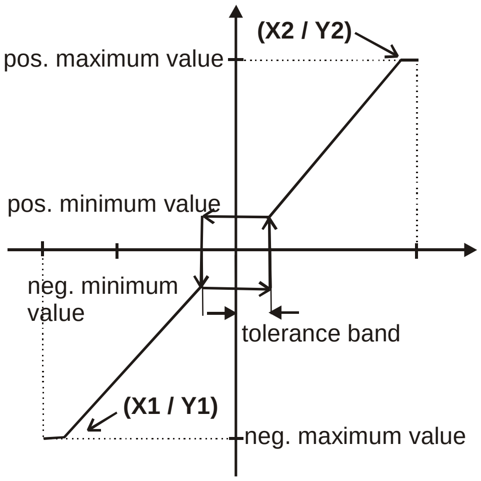


Without tolerance band



With tolerance band

The *Minimum Frequency* **418** or the *Minimum reference percentage* **518** set in the factory extends the parameterized tolerance band to the hysteresis.



With tolerance band and minimum value

For example, the output variable resulting from the positive input signals is kept at the positive minimum value until the input signal is below the value for the tolerance band in a negative direction. After that proceed on the set characteristic.

13.1.6 Error and warning behavior

The monitoring of the analog input signal necessary according to the application is to be configured via the parameter *Error/Warning behavior* **563**.

Operation mode		Function
0 -	Off	The input signal is not monitored.
1 -	Warning < 1 V / 2 mA	If the input signal is less than 1 V or 2 mA, there is a warning message.
2 -	Shutdown < 1V / 2 mA	If the input signal is less than 1 V or 2 mA, there is a warning message; the drive mechanism is decelerated according to stopping behavior 2.
3 -	Fault switch-off < 1 V / 2 mA	If the input signal is less than 1 V or 2 mA, there is a warning and fault message and the drive mechanism stops freely.

The monitoring of the analog input signal is active independent of the release of the frequency inverter according to the operation mode selected.

In **operation mode 2**, the drive mechanism is decelerated independent of the stopping behavior set (Parameter *Operation mode* **630**) according to stopping behavior 2 (shutdown and stop). If the set holding time has expired, there is a fault message. A repeat start of the drive mechanism is possible by switching the start signal on and off if the fault has been cleared.

Operation mode 3 defines the free stoppage of the drive mechanism, independent of the stopping behavior selected, which is stipulated with the parameter *Operation Mode* **630**.

NOTICE

Monitoring of the analog input signal via the parameter *Error/Warning Behavior* **563** demands a check of the characteristic parameters.

13.1.7 Filter time constant

The time constant of the filter for the reference analog value can be set via the parameter *Filter time constant* **561**.

The time constant states the time for which the input signal is averaged by means of a low pass filter, e.g. in order to eliminate fault effects.

The setting range is a range of values between 0 ms and 5000 ms in 15 steps.

Operation mode		Function
0 -	Time constant 0 ms	Filter deactivated – analog reference value is forwarded unfiltered
2 -	Time constant 2 ms	Filter activated – averaging of the input signal via the set value of the filter time constants
4 -	Time constant 4 ms	
8 -	Time constant 8 ms	
16 -	Time constant 16 ms	
32 -	Time constant 32 ms	
64 -	Time constant 64 ms	
128 -	Time constant 128 ms	
256 -	Time constant 256 ms	
512 -	Time constant 512 ms	
1000 -	Time constant 1000 ms	
2000 -	Time constant 2000 ms	
3000 -	Time constant 3000 ms	
4000 -	Time constant 4000 ms	
5000 -	Time constant 5000 ms	

13.2Analog output MF4OA

The terminal X410B.4 can be used as voltage output if the switch S1 is ON (see chapter 14.1.2).

NOTICE

The MF4OA output used as analog output (terminal X410B.4) cannot permanently withstand short circuits and external voltages. External voltages can damage the module.

- Avoid short circuits and external voltages.

13.2.1 General

The analog output is a voltage output with the range of 0..10 V. The parameterization is done by selecting the operation mode and a linear characteristic with the consideration of an offset and amplification.

To use terminal X410B.4 as Analog output *Operation mode X410B.4* **502** has to be set up.

Operation mode 502	Function
5 – Voltage Output 0 – 10 V	Voltage signal (MFO2A), 0 V ... 10 V

13.2.2 Function for analog output MFO2A

The selected configuration determines which actual values can be selected for parameter *Analog operation* **584** of analog output 2.

Operation mode 584		Function
0 -	Off	Analog operation MFO1 is switched off.
1 -	Abs. Fs	Abs. value of the Stator Frequency 0.00 Hz ... *Maximum frequency* **419**.
2 -	Abs. Fs betw. fmin/fmax	Abs. value of the Stator Frequency *Minimum frequency* **418**...*Maximum frequency* **419**.
3 -	Abs. Speed Sensor 1	Abs. value of speed sensor signal 1, 0.00 Hz ... *Maximum frequency* **419**.
4 -	Abs. Speed Sensor 2	Abs. value of speed sensor signal 2, 0.00 Hz ... *Maximum frequency* **419**.
5 -	Abs. Speed Sensor 3	Abs. value of speed sensor signal 3, 0.00 Hz ... *Maximum frequency* **419**.
7 -	Abs. Actual Frequency	Abs. value of act. frequency, 0.00 Hz ... *Maximum frequency* **419**.
10 -	Abs. Ref. Percentage	Abs. value of ref. percentage, 0.00 % ... *Maximum Reference Percentage* **519**.
11 -	Abs. Ref. Percent. betw. %min / %max	Abs. value of ref. percentage betw. %min / %max *Minimu Reference Percentage* **518**... *Maximum Reference Percentage* **519**.
20 -	Abs. Iactive	Abs. value of current effective current I_{Active}, 0.0 A ... FU rated current.
21 -	Abs. Isd	Abs. value of flux-forming current component, 0.0 A ... FU rated current.
22 -	Abs. Isq	Abs. value of torque-forming current component, 0.0 A ... FU rated current.
30 -	Abs. Pactive	Abs. value of current active power P_{Active}, 0.0 kW ... *Rated mech. power* **376**.
31 -	Abs. M	Abs. value of calculated torque M, 0.0 Nm ... Rated torque.
32 -	Abs. Inside Temperature	Abs. value of measured inside temperature, 0 °C ... 100 °C
33 -	Abs. Heat Sink Temperature	Abs. value of measured heat sink temperature, 0 °C ... 100 °C
40 -	Abs. Analog Input MF1IA	Abs. signal value on analog input 1, 0.0 V ... 10.0 V or -10 V .. 10 V.

Operation mode 584		Function
41 -	Abs. Analog Input EM-MF1IA	Abs. signal value on analog input 1 of EM, 0.0 V ... 10.0 V or -10 V .. 10 V.
50 -	Abs. I	Abs. current value of measured output currents, 0.0 A ... FU rated current.
51 -	DC –Link Voltage	DC link voltage du, 0.0 V ... 1000.0 V.
52 -	V	Output voltage U, 0.0 V ... 1000.0 V.
53 -	Volume Flow	Abs. value of calculated volume flow 0.0 m^3/h ... *Nominal volumetric flow* **397**.
54 -	Pressure	Abs. value of calculated pressure 0.0 kPa ... *Reference pressure* **398**.
61 -	Abs. Val. FT-Output percentage 1	Abs. value of FT-Output percentage 1, 0.00 % ... 327.67%.
62 -	Abs. Val. FT-Output percentage 2	Abs. value of FT-Output percentage 2, 0.00 % ... 327.67%.
101 to 162		Operation modes in analog operation with signs.

13.2.3 Zero adjustment and amplification

With the parameters *Offset* **585** (zero adjustment) and *Amplification* **586** the voltage of the output signal at 0% and 100% of the reference signals can be set.

The zero adjustment with the parameter *Offset* **585** is done specific to the application as a percentage of the final value of the analog output (10 V).

Via the parameter *Amplification* **586** the amplification can be set as a percentage of the final value of the analog output (10 V).

In the factory setting, the zero point has been set at 0% Offset, i.e. minimum value of the reference signal equal to 0 V output signal. The factory setting amplification equal to 100% means that the output signal is 10 V when the reference value is reached.

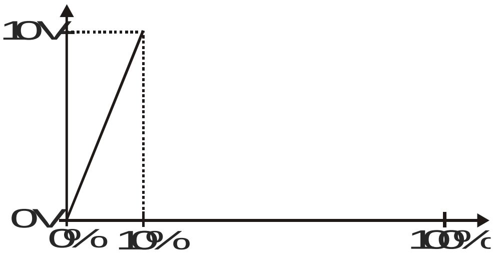
Parameter		Setting		
No.	**Description**	**Min.**	**Max.**	**Default**
585	Offset	-100.00%	100.00%	0.00%
586	Amplification	5.0%	1000.0%	100.0%

13.2.3.1 Examples

The actual value parameter is mapped to the analog output signal as a function of the selected parameters *Offset* **585** and *Amplification* **586**. The following examples show the application-specific adaptation for an analog voltage signal.

Example 1:

Parameter		Setting
No.	**Description**	**Example**
585	Offset	0.00 %
586	Amplification	1000.0 %

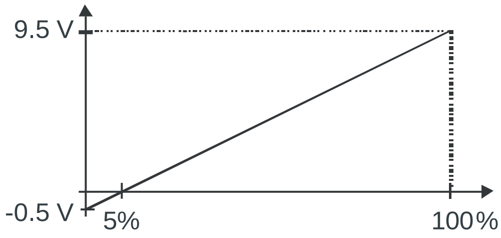


The setting of the parameter *Offset* **585** to 0.00% and the parameter *Amplification* **586** to 1000.0% means that the output signal:

- is 0 V at 0% of the reference signal,
- is 10 V at 10% of the reference signal

Example 2:

Parameter		Setting
Nº.	**Description**	**Example**
585	Offset	0.00 %
586	Amplification	1000.0 %



The setting of the parameter *Offset* **585** to -5,00% and the parameter *Amplification* **586** to 100.0% means that the output signal:

- is -0.5 V at 0% of the reference signal,
- is 9.5 V at 100% of the reference signal.

13.3Operation mode MF4ID Digital input

The terminal X410B.4 can be used according to parameter *Operation mode X410B.4* **502** as digital input.

13.4Operation mode Motor temperature monitoring

The terminal X410B.4 can be used for motor temperature monitoring.

The characteristics according to parameter *Operation mode X410B.4* **502** are available. Also consider *Therm. Contact* **204** and *Motor Temp. Operation Mode* **570**.

14 Reference frequency and percentage value channel

The various functions for the statement of the reference figures are connected in the various configurations by the reference frequency or percentage value channel. The *Reference Frequency Source* **475** and the *Reference Percentage Source* **476** determine the additive connection of the available reference sources as a function of the installed hardware.

Operation mode		Function
2 -	Abs. analog value MF4IA	Reference source is the analog input MF4IA (X410B.4).
4 -	Abs. value MF1IA + MF4IA	Reference sources are the multifunctional input MF1IA and the analog input MF4IA (X410B.4).
14 -	Abs. value MF1IA + MF4IA + FP (or FF)	Reference sources are the multifunctional input MF1IA, analog input MF4IA (X410B.4) and fixed percentage FP *(or the fixed frequency FF).*
24 -	Abs. value MF1IA + MF4IA + MP	Reference sources are the multifunctional input MF1IA, analog input MF4IA (X410B.4) and the motor potentiometer function MP.
102 to 124		Operation modes with signs (+/-).

Alongside the operation modes listed, those stated in the operating instructions of the frequency inverter in the chapter "Frequency reference channel", and in the chapter "Percentage reference channel" also apply.

15 Motor temperature

The temperature monitoring is a part of the error and warning behavior which can be freely configured. The connected load can be monitored by the connection of a measurement resistor (motor PTC resistor / PTC) with a temperature characteristic to DIN 44081 or with a bimetallic temperature sensor (NC contact).

The operation mode of the motor PTC port can be selected via parameter *Motor Temp. Operation mode* **570** The operation modes described in chapter "Motor Temperature" of the frequency inverter operating instructions are complemented by the following operation modes with the expansion module:

Operation mode 570		Function
0 -	Off	Motor temperature monitoring switched off.
1 -	Therm.-Cont.: Warning only	The critical point of operation is displayed by the control unit and parameter *Warnings* **269**. Selection Therm. Contact via *Therm. Contact* **204**.
2 -	Therm.-Cont.: Error Switch-Off	The fault switch-off is displayed by message F0400. The fault switch-off can be acknowledged via the control unit or the digital input. Selection Therm. Contact via *Therm. Contact* **204**.
3 -	Therm.-Cont.: Error Switch-Off 1 min del.	The fault switch-off according to operation mode 2 is delayed by one minute.
4 -	Therm.-Cont.: Error Switch-Off 5 min del.	The fault switch-off according to operation mode 2 is delayed by five minutes.
5 -	Therm.-Cont.: Error Switch-Off 10 min del.	The fault switch-off according to operation mode 2 is delayed by ten minutes.
31 -	MF1IA: Warning only	The critical point of operation is displayed by the control unit and parameter *Warnings* **269**. Selection Temperature monitoring via *Operation Mode* **452** / X210B.6.
32 -	MF1IA: Fault shutdown	The fault switch-off is displayed by message F0400. The fault switch-off can be acknowledged via the control unit or the digital input. Selection Temperature monitoring via *Operation Mode* **452** / X210B.6.
33 -	MF1IA: Fault shutdown 1 min del.	The fault switch-off according to operation mode 32 is delayed by one minute.
34 -	MF1IA: Fault shutdown 5 min del.	The fault switch-off according to operation mode 32 is delayed by five minutes.
35 -	MF1IA: Fault shutdown 10 min del.	The fault switch-off according to operation mode 32 is delayed by ten minutes.
41 -	MF4IA: Warning only	The critical point of operation is displayed by the control unit and parameter *Warnings* **269**. Evaluation Temperature monitoring via *Operation Mode* **502** / X410B.4.
42 -	MF4IA: Fault shutdown	The fault switch-off is displayed by message F0400. The fault switch-off can be acknowledged via the control unit or the digital input. Evaluation Temperature monitoring via *Operation Mode* **502** / X410B.4.
43 -	MF4IA: Fault shutdown 1 min del.	The fault switch-off according to operation mode 42 is delayed by one minute.
44 -	MF4IA: Fault shutdown 5 min del.	The fault switch-off according to operation mode 42 is delayed by five minutes.
45 -	MF4IA: Fault shutdown 10 min del.	The fault switch-off according to operation mode 42 is delayed by ten minutes.
111 -	X412-PTC: Warning only	The critical point of operation is displayed by the control unit and parameter *Warnings* **269**. Evaluation Temperature monitoring via X412.

Operation mode 570		Function
112 -	X412-PTC: Fault shutdown	The fault switch-off is displayed by message F0400. The fault switch-off can be acknowledged via the control unit or the digital input. Evaluation Temperature monitoring via X412.
113 -	X412-PTC: Fault shutdown 1 min del.	The fault switch-off according to operation mode 112 is delayed by one minute.
114 -	X412-PTC: Fault shutdown 5 min del.	The fault switch-off according to operation mode 112 is delayed by five minutes.
115 -	X412-PTC: Fault shutdown 10 min del.	The fault switch-off according to operation mode 112 is delayed by ten minutes.
121 -	X412-KTY: Warning only	The critical point of operation is displayed by the control unit and parameter *Warnings* **269**. Evaluation Temperature monitoring via X412.
122 -	X412-KTY: Fault shutdown	The fault switch-off is displayed by message F0400. The fault switch-off can be acknowledged via the control unit or the digital input. Evaluation Temperature monitoring via X412.
123 -	X412-KTY: Fault shutdown 1 min del.	The fault switch-off according to operation mode 122 is delayed by one minute.
124 -	X412-KTY: Fault shutdown 5 min del.	The fault switch-off according to operation mode 122 is delayed by five minutes.
125 -	X412-KTY: Fault shutdown 10 min del.	The fault switch-off according to operation mode 122 is delayed by ten minutes.
131 -	X412-PT1000: Warning only	The critical point of operation is displayed by the control unit and parameter *Warnings* **269**. Evaluation Temperature monitoring via X412.
132 -	X412-PT1000: Fault shutdown	The fault switch-off is displayed by message F0400. The fault switch-off can be acknowledged via the control unit or the digital input. Evaluation Temperature monitoring via X412.
133 -	X412-PT1000: Fault shutdown 1 min del.	The fault switch-off according to operation mode 132 is delayed by one minute.
134 -	X412-PT1000: Fault shutdown 5 min del.	The fault switch-off according to operation mode 132 is delayed by five minutes.
135 -	X412-PT1000: Fault shutdown 10 min del.	The fault switch-off according to operation mode 32 is delayed by ten minutes.

The function to be adjusted by parameter *Motor Temp. Operation mode* **570** results in signaling the overtemperature by the red LED of the frequency inverter, irrespective of the selected operation modes of the control inputs and outputs.

The operation modes with error-switch-off result in the fault message "FAULT" with fault number "F0400" being displayed on the control unit KP500.

The fault message can be acknowledged via parameter *Program* **34** or the logic signal linked with parameter *Error acknowledgement* **103**.

16 Actual value display

The actual value of rotary encoder 1 can be read out via the parameters *Encoder 1 Frequency* **217** and *Encoder 1 Speed* **218**.

The actual value of rotary encoder 2 can be read out via the parameters *Encoder 2 Frequency* **219** and *Encoder 2 Speed* **220**.

The analog input signal on analog input MF4IA is displayed via actual value parameter *Analog Input MF4IA* **253**.

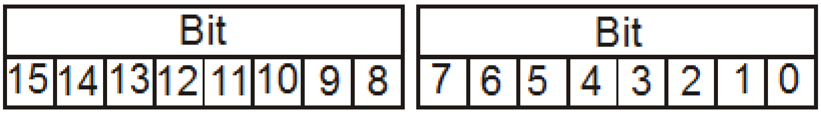
16.1Actual position

Act. position **1108** shows the current actual value (position) in user units [u] in positioning configurations x40.

16.2Status of Digital Signals

The status of the digital signals can be read in decimal coding via the parameters *Digital inputs* **250** and *Digital outputs* **254**. The display of the digital input signals enables checking the various control signals and their connections with the software functions in question, in particular in commissioning.

Coding of the status of the digital signals



Bit								Bit							
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0

Assignment:

Bit	Decimal value	Terminal	Designation
Bit 00	1	X210A.3 & X210B.2	STO
Bit 01	2	X210A.4	S2IND
Bit 02	4	X210A.5	S3IND
Bit 03	8	X210A.6	S4IND
Bit 04	16	X210A.7	S5IND
Bit 05	32	X210B.1	S6IND
Bit 06	64	X210B.6	MF1ID
Bit 07	128	X410B.4	MF4ID
Bit 08	256	VIRTUAL	REMOTE 1
Bit 09	512	VIRTUAL	REMOTE 2
Bit 10	1024	X210B.3	MF3ID
Bit 11	2048	X210B.4	MF2ID
Bit 12	4096		
Bit 13	8192		
Bit 14	16384		
Bit 15	32768		

17 Parameter list

The parameter list is structured according to the menu branches of the control unit. For better clarity, the parameters have been marked with pictograms:

- ▤ The parameter is available in the four data sets.
- ☑ The parameter value is adjusted by the SETUP routine if a control method for a synchronous machine is selected for parameter *Configuration* **30**.
- ⊗ This parameter cannot be written when the frequency inverter is in operation.

Parametrization

As an alternative to the control unit, you can also use the optional PC user software **VPlus** for parameterization, monitoring and maintenance of the inverter.

In order to use the Vplus software, the PC with Vplus installed must be connected to the inverter. The communication between VPlus and the inverters can be carried out via serial interfaces (COM Ports, internal or USB-COM Adapter), CAN-Systembus with a special USB-CAN Adapter or Modem (COM Port, internal or external).

If no interface is configured, as is the case after installation of VPlus, the Inverter Manager automatically starts with the "COM Properties" window. Per default, the first COM-Port found is displayed here and can be immediately configured. The same behaviour occurs if all interfaces are deleted and VPlus restarted.

After the conection to the inverter is established, the *.vcb-File containing the inverter configuration can be downloaded from the connected inverter. Subsequently, the file can be edited using the Vplus Software to configure the parameters.

The parameters accessible via the software VPlus are divided in 3 control levels.

Parameter *Control Level* **28** defines the effective control level.

- **Control level 1** provides inverter information, actual values, an error list and contains parameters with which the operating behavior can be influenced.
- **Control level 2** contains additional parameters, actual values and functions.
- **Control level 3** contains additional error information, functions and actual values. Additionally, it enables fundamental changes of the operating behavior.

NOTICE

With the communication interface, it is possible to access **ALL** frequency inverter parameters from a controller. There is no access control via the control level as with the KP500 manual control unit or the VPlus PC software. Changing parameters, the functions of which are not known to the user, can result in unintended movements of system components and material losses, and/or personal injury as well as inoperativeness of the frequency inverter.

NOTICE

Not all parameters of the inverter are listed in the parameter list given below.

This parameter list focuses on the parameters specific to the EM-AUT Interface. For further information see the applicable operating instruction document.

For more information on using the VPlus software, see the online help in the VPlus software.

17.1Actual values

No.	Description	Unit	Display range	Chapter
Inverter data				
16	EM Software Version			6.2.4
Actual values of the machine				
219	Encoder 2 Frequency	Hz	0.0 ... 599,99	16
220	Encoder 2 Speed	rpm	0 ... 60000	16
Actual values of the machine				
228	Internal reference frequency	Hz	-1000.00 ... 1000.00	12.3.3
249	Active data set	-	1 ... 4	12
253	Analog input MF4IA	V	-10 ... +10	16
260	Current error	-	0 ... 0xFFFF	9.9.5.2 18.4
270	Warnings	-	0 ... 0xFFFF	18.2
274	Warnings Application	-	0 ... 0xFFFF	18.3
282	Reference bus frequency	Hz	-1000.00 ... 1000.00	12.3.3
283	Reference ramp frequency	Hz	-1000.00 ... 1000.00	12.3.3
Actual values of the Systembus				
978	Node-State	-	1 ... 3	10.9
979	CAN-State	-	1 ... 3	10.9
Actual values of the Motion Control Interface				
1108	Actual Position	u	--2147483647 ... 2147483647	16.1
1109	Act. Contouring Error	u	--2147483647 ... 2147483647	16.1
Actual values of CANopen				
1290	Node-State (NMT)	-	0 … 127	8.9
Actual values of Ethernet				
1431	Module Info	-	String	6.2.4
1453	OS SyncSource Act	-	Selection	6.3

Parameter *Warning Dig. Encoder* **1273** is intended for read-out via a PLC, parameter *Warning Dig. Encoder* **1274** provides a brief description of the information in VPlus and the keypad KP500.

17.2Parameters

	No.	Description	Unit	Setting range	Default value	Chapter
		CAN Bus				
	385	CAN Baudrate	-	Selection		9.4
	387	CAN Node Number	-	-1 ... 127	-1	9.5
	388	Bus Error Behavior	-	Selection	1 - Error	9.6. 9.9.5.1
		Bus control				
	392	State-transition 5	-	Selection	2 - Ramp	12.3.2
▤	412	Local/Remote	-	Selection	44 - Ctrl. Cont.+KP. Dir. Cont.+KP	12
		Data set change-over				
	414	Data set selection	-	0 ... 4	0	12
		Frequency ramps				
▤	420	Acceleration (Clockwise)	Hz/s	0.00 ... 9999.99	5.00	9.9.5.9
▤	421	Deceleration (Clockwise)	Hz/s	0.01 ... 9999.99	5.00	9.9.5.10
▤	422	Acceleration Anticlockwise	Hz/s	-0.01 ... 9999.99	-0.01	9.9.5.9
▤	423	Deceleration Anticlockwise	Hz/s	-0.01 ... 9999.99	-0.01	9.9.5.10
▤	424	Emergency Stop Clockwise	Hz/s	0.01 ... 9999.99	5.00	9.9.5.11. 12.3.1
▤	425	Emergency Stop Anticlockwise	Hz/s	0.01 ... 9999.99	5.00	9.9.5.11. 12.3.1
	434	Ramp set point	-	Selection	3 - Internal + Line Setpoint	12.3.3
		Encoder 2 Gear factors				
	513	EC2 Gear Factor Numerator	-	-300.00 ... 300.00		7.2.5
	514	EC2 Gear Factor Denominator	-	0.01 ...300.00		7.2.5
		Max. Control deviation				
	549	Max. Control deviation	%	0.01 ... 20.00	5.00	12.1.12.2
		Stopping behavior				
▤	560	Tolerance band	%	0.00 ... 25.00	2.00 %	13.1.5
	561	Filter time constant	-	Selection		13.1.7
	562	Operation mode	-	Selection		13.1.2
	563	Error / Warning behavior	-	Selection		13.1.6
▤	564	Characteristic point X1	%	-100.00 ... 100.00	-98.00 %	13.1.3
▤	565	Characteristic point Y1	%	-100.00 ... 100.00	-100.00 %	13.1.3
▤	566	Characteristic point X2	%	-100.00 ... 100.00	98.00 %	13.1.3
▤	567	Characteristic point Y2	%	-100.00 ... 100.00	100.00 %	13.1.3
	585	Offset	%	-100.00 ... 100.00	0.00%	13.2.3
	586	Amplification	%	5.00 ... 1000	100.00 %	13.2.3
		Stopping behavior				
▤	637	Switch-off threshold	%	0.0 ... 100.0	1.0	12.3.1. 12.3.2
▤	638	Holding time	s	0.0 ... 200.0	1.0	12.3.1. 12.3.2
		Speed controller				
	766	Actual Speed Source	-	Selection		7.7
		Systembus				
	900	Node-ID	-	-1 ... 63		10.2
	903	Baud-Rate	-	Selection		10.1
	904	Boot-Up Delay	ms	3500 ... 50000		10.5.4
	918	SYNC-Identifier	-	0 ... 2047		10.5.2
	919	SYNC-Time	ms	0 ... 50000		10.6.2
	921	RxSDO1-Identifier	-	0 ... 2047		10.6.4
	922	TxSDO1-Identifier	-	0 ... 2047		10.6.4
	923	SDO2 Set Active	-	Selection		10.6.4
	924	RxPDO1-Identifier	-	0 ... 2047		10.8.1
	925	TxPDO1-Identifier	-	0 ... 2047		10.8.1
	926	RxPDO2-Identifier	-	0 ... 2047		10.8.1
	927	TxPDO2-Identifier	-	0 ... 2047		10.8.1
	928	RxPDO3-Identifier	-	0 ... 2047		10.8.1

No.	Description	Unit	Setting range	Default value	Chapter
929	TxPDO3-Identifier	-	0 ... 2047		10.8.1
930	TxPDO1 Function	-	Selection		10.8.2
931	TxPDO1 Time	ms	0 ... 50000		10.8.2
932	TxPDO2 Function	-	Selection		10.8.2
933	TxPDO2 Time	ms	0 ... 50000		10.8.2
934	TxPDO3 Function	-	Selection		10.8.2
935	TxPDO3 Time	ms	0 ... 50000		10.8.2
936	RxPDO1 Function	-	Selection		10.4.2
937	RxPDO2 Function	-	Selection		10.4.2
938	RxPDO3 Function	-	Selection		10.4.2
939	SYNC Timeout	ms	0 ... 60000		10.8.3
941	RxPDO1 Timeout	ms	0 ... 60000		10.8.3
942	RxPDO2 Timeout	ms	0 ... 60000		10.8.3
945	RxPDO3 Timeout	ms	0 ... 60000		10.8.3
946	TxPDO1 Boolean1	-	Selection		10.8.5.1
947	TxPDO1 Boolean2	-	Selection		10.8.5.1
948	TxPDO1 Boolean3	-	Selection		10.8.5.1
949	TxPDO1 Boolean4	-	Selection		10.8.5.1
950	TxPDO1 Word1	-	Selection		10.8.5.1
951	TxPDO1 Word2	-	Selection		10.8.5.1
952	TxPDO1 Word3	-	Selection		10.8.5.1
953	TxPDO1 Word4	-	Selection		10.8.5.1
954	TxPDO1 Long1	-	Selection		10.8.5.1
955	TxPDO1 Long2	-	Selection		10.8.5.1
956	TxPDO2 Boolean1	-	Selection		10.8.5.1
957	TxPDO2 Boolean2	-	Selection		10.8.5.1
958	TxPDO2 Boolean3	-	Selection		10.8.5.1
959	TxPDO2 Boolean4	-	Selection		10.8.5.1
960	TxPDO2 Word1	-	Selection		10.8.5.1
961	TxPDO2 Word2	-	Selection		10.8.5.1
962	TxPDO2 Word3	-	Selection		10.8.5.1
963	TxPDO2 Word4	-	Selection		10.8.5.1
964	TxPDO2 Long1	-	Selection		10.8.5.1
965	TxPDO2 Long2	-	Selection		10.8.5.1
966	TxPDO3 Boolean1	-	Selection		10.8.5.1
967	TxPDO3 Boolean2	-	Selection		10.8.5.1
968	TxPDO3 Boolean3	-	Selection		10.8.5.1
969	TxPDO3 Boolean4	-	Selection		10.8.5.1
972	TxPDO3 Word1	-	Selection		10.8.5.1
973	TxPDO3 Word2	-	Selection		10.8.5.1
974	TxPDO3 Word3	-	Selection		10.8.5.1
975	TxPDO3 Word4	-	Selection		10.8.5.1
976	TxPDO3 Long1	-	Selection		10.8.5.1
977	TxPDO3 Long2	-	Selection		10.8.5.1
989	Emergency Reaction	-	Selection		10.5.3
Electronic Gear					
⊗ 1115	Feed constant	u/U	1 ... 231-1		7.3.5 [1)
⊗ 1116	Gear: Shaft revolutions		1 ... 65535		
⊗ 1117	Gear: Motor revolutions		1 ... 65535		
🗐 1141	Position Actual source		Selection		
Electronic Gear with position					
1122	Source Master Position	-	Selection	0-Off	12.4.7

[1) For Positioning and using the Motion Control Interface comply with the application manual "Positioning".

Systembus Synchronization					
1180	Operation mode	-	Selection	0-Off	12.4.7
Master/Slave Position Correction					
1284	Master/Slave Synchronization Offset	-	Selection	0 u	12.4.7.2
Encoder Interface X412					
1183	Division marks	-	0 … 8192		7.4.1
⊗ 1184	Tracks/Protocol	-	Selection		7.4.2
1186	Power supply	-	Selection		7.4.3
1187	Supply voltage	V	5.00 … 12.0		7.4.4
1188	Offset	°	-360.0 … 360.0		7.4.6
1189	Abs. Encoder: Filter time constant	µs	125 … 8000	125	7.4.5
1268	SSI: Sampling interval	-	Factor (x 125 us)	0	7.4.10
1269	SSI: Error-/Extra-Bits (Low)	-	Special		7.4.9
1270	SSI: Error-/Extra-Bits (High)	-	Special		7.4.9
1271	Bits/Turn	Bit/U	0 … 32		7.4.7
1272	Bits Multiturn	Bit	0 … 32		7.4.8
1273	Warning Dig. Encoder	Bit	0 … 32		7.4.12
1274	Warning Dig. Encoder	Bit	0 … 32		7.4.12
Master Slave Position Correction					
1284	Master/Slave Synchonization Offset	-	Selection		12.4.7.2
Motion Control Interface					
1285	S. Target velocity pv [u/s]	-	Selection	816 – 0x60FF	11.3
1292	S. Modes of Operation	-	Selection	801 – 0x6060	11.3
1293	S. Target Position	-	Selection	802 – 0x607A	11.3
1294	S. Profile Velocity	-	Selection	803 – 0x6081	11.3
1295	S. Acceleration	-	Selection	804 – 0x6083	11.3
1296	S. Deceleration	-	Selection	805 – 0x6084	11.3
1297	S. Target Velocity vl [rpm]	-	Selection	806 – 0x6042	11.3
1299	S. Special Function Generator	-	Selection	9-Zero	11.3
CANopen Mux/Demux					
1420	CANopen Mux Input Index (Write) [1]	-	EEPROM: 0 … 16 RAM: 17 … 33	1	9.9.4.5
1421	CANopen Mux Input Index (Read) [1]	-	EEPROM: 0 … 16 RAM: 17 … 33	1	9.9.4.5
1422	CANopen Mux Inputs	-	Selection	7 - Off	9.9.4.5
1423	CANopen Percentage Actual Value Source	-	Selection	52 - Analog Input MF1IA	9.9.4.7
1451	OS Synctime	-	700…900 us	800 us	6.3
1452	OS_SyncSource	-	Selection	52 - Analog Input MF1IA	6.3
Motion Control Override					
1454	Override Modes Of Operation	-	Selection	0	11.4
1455	Override Target Position	-		-1 u	11.4
1456	Override Profile Velocity	-		-1 u/s	11.4
1457	Override Profile Acceleration	-		-1 u/s²	11.4
1458	Override Profile Deceleration	-		-1 u/s²	11.4
1459	Override Target velocity vl [rpm]	-		-1 rpm	11.4
1460	Override Target velocity pv [u/s]	-		-1 u/s	11.4

[1]

Non volatile (fixed Parameterization)		Volatile	
0:	All indexes in EEPROM	17:	All indexes in RAM
1…16:	One Index in EEPROM	18…33:	One Index 1…16 in RAM

Setting "0" for *CANopen Mux input index (write)* **1420** changes all data in EEPROM and/or RAM.

Parameter *Data set selection* **414** is only accessible via Manufacturer objects 0x2nnn. It cannot be addressed via the VPlus control software or the control panel.

18 Annex

18.1 Control Word/Status word Overview

18.1.1 Control Word overview (without Sync Modes)

The tables on this page list in an overview the functionality of the **Control Word** bits.

Bit	*Standard (No Positioning)*	*Positioning without MCI*	*MCI: Velocity Mode*	*MCI: Profile Velocity Mode*	*MCI: Profile Position Mode*
0	Switch On	Switch On	Switch On	Switch On	Switch On
1	Enable Voltage	Enable Voltage	Enable Voltage	Enable Voltage	Enable Voltage
2	Quick Stop (low active)	Quick Stop (low active)	Quick Stop (low active)	Quick Stop (low active)	Quick Stop (low active)
3	Enable Operation	Enable Operation	Enable Operation	Enable Operation	Enable Operation
4			Rfg enable		New setpoint
5			Rfg unlock		Change set immediately
6			Rfg use ref		Abs/rel
7	Fault reset	Fault reset	Fault reset	Fault reset	Fault reset
8	Halt	Halt	Halt	Halt	Halt
9					Change on setpoint
10					
11					
12					
13					
14					
15					

Bit	*MCI: Interpol. Position Mode*	*MCI: Homing Mode*	*MCI: Table travel record Mode*	*MCI: Move away from Limit Sw.*	*MCI: Electronic Gear: Slave*
0	Switch On	Switch On	Switch On	Switch On	Switch On
1	Enable Voltage	Enable Voltage	Enable Voltage	Enable Voltage	Enable Voltage
2	Quick Stop (low active)	Quick Stop (low active)	Quick Stop (low active)	Quick Stop (low active)	Quick Stop (low active)
3	Enable Operation	Enable Operation	Enable Operation	Enable Operation	Enable Operation
4	Enable ip-mode	Homing operat.start	Sequence mode	Move away from LS	Start Gearing
5					
6			Resume		Direct Sync
7	Fault reset	Fault reset	Fault reset	Fault reset	Fault reset
8	Halt	Halt	Halt	Halt	Halt
9			Start motion block		Start Phasing
10					
11			Motion Block Select 0		
12			Motion Block Select 1		Phasing Profile Sel. 1
13			Motion Block Select 2		Phasing Profile Sel. 2
14			Motion Block Select 3		
15			Motion Block Select 4		

18.1.2 Status Word overview (without Sync modes)

The tables on this page list in an overview the functionality of the **Status Word** bits.

Bit	*Standard (No Positioning)*	*Positioning without MCI*	*MCI: Velocity Mode*	*MCI: Profile Velocity Mode*	*MCI: Profile Position Mode*
0	Ready to Switch On	Ready to Switch On	Ready to Switch On	Ready to Switch On	Ready to Switch On
1	Switched On	Switched On	Switched On	Switched On	Switched On
2	Operation enabled	Operation enabled	Operation enabled	Operation enabled	Operation enabled

3	Fault	Fault	Fault	Fault	Fault
4	Voltage enabled	Voltage enabled	Voltage enabled	Voltage enabled	Voltage enabled
5	Quick Stop (low active)	Quick Stop (low active)	Quick Stop (low active)	Quick Stop (low active)	Quick Stop (low active)
6	Switch On Disabled	Switch On Disabled	Switch On Disabled	Switch On Disabled	Switch On Disabled
7	Warning	Warning	Warning	Warning	Warning
8		Homing done			
9	Remote	Remote	Remote	Remote	Remote
10	Target reached	Target reached	Target reached	Target reached	Target reached
11	Internal limit active	Internal limit active	Internal limit active	Internal limit active	Internal limit active
12				Speed	Set-point acknowl.
13				Max slippage error	Following error
14		Target Pos. reached			
15	Warning 2	Warning 2	Warning 2	Warning 2	Warning 2

Bit	*MCI: Interpol. Position Mode*	*MCI: Homing Mode*	*MCI: Table travel record Mode*	*MCI: Move away from Limit Sw.*	*MCI: Electronic Gear: Slave*
0	Ready to Switch On	Ready to Switch On	Ready to Switch On	Ready to Switch On	Ready to Switch On
1	Switched On	Switched On	Switched On	Switched On	Switched On
2	Operation enabled	Operation enabled	Operation enabled	Operation enabled	Operation enabled
3	Fault	Fault	Fault	Fault	Fault
4	Voltage enabled	Voltage enabled	Voltage enabled	Voltage enabled	Voltage enabled
5	Quick Stop (low active)	Quick Stop (low active)	Quick Stop (low active)	Quick Stop (low active)	Quick Stop (low active)
6	Switch On Disabled	Switch On Disabled	Switch On Disabled	Switch On Disabled	Switch On Disabled
7	Warning	Warning	Warning	Warning	Warning
8			Motion Block in Progress		Phasing Done
9	Remote	Remote	Remote	Remote	Remote
10	Target reached	Target reached	Target reached	Target reached	Target reached
11	Internal limit active	Internal limit active	Internal limit active	Internal limit active	Internal limit active
12	IP-mode active	Homing attained	In gear		
13		Homing error	Following error		Following error
14					
15	Warning 2	Warning 2	Warning 2	Warning 2	Warning 2

18.1.3 Control Word overview for Sync Modes

The table on this page list in an overview the functionality of the **Control Word** bits.

Bit	*MCI: Sync Position Mode*	*MCI: Sync Velocity Mode*
0	Switch On	Switch On
1	Enable Voltage	Enable Voltage
2	Quick Stop (Low active)	Quick Stop (Low active)
3	Enable Operation	Enable Operation
4		
5		
6		
7	Fault reset	Fault reset

8	Halt	Halt
9		
10		
11		
12		
13		
14		
15		

18.1.4 Status Word overview for Sync modes

The table on this page list in an overview the functionality of the **Status Word** bits.

Bit	*MCI: Sync Position Mode*	*MCI: Sync Velocity Mode*
0	Ready to Switch On	Ready to Switch On
1	Switched On	Switched On
2	Operation enabled	Operation enabled
3	Fault	Fault
4	Voltage enabled	Voltage enabled
5	Quick Stop	Quick Stop
6	Switch On Disabled	Switch On Disabled
7	Warning	Warning
8		
9	Remote	Remote
10		
11		
12	Target Position ignored	Target velocity ignored
13	Following error	
14		
15	Warning 2	Warning 2

18.2Warning messages

The different control methods and the hardware of the frequency inverter include functions for continuous monitoring of the application. In addition to the messages documented in the frequency inverter Operating Instructions, further warning messages are activated by the Field Bus communication´.

The bit-coded warning reports are issued via parameter *Warnings* **270** according to the following pattern:

Parameter *Warnings* **269** indicates the warnings as plain text in the VPlus PC software and Keypad KP500.

For more information on VPlus, see chapter 17.

Warning messages		
Bit no.	**Warning code**	**Meaning**
0	0x0001	Warning Ixt
1	0x0002	Warning Short Term - Ixt
2	0x0004	Warning Long Term - Ixt
3	0x0008	Warning Heat sink Temperature Tc
4	0x0010	Warning Inside Temperature Ti
5	0x0020	Warning Limit
6	0x0040	Warning Init
7	0x0080	Warning Motor Temperature
8	0x0100	Warning Mains Failure
9	0x0200	Warning Motor Protective Switch
10	0x0400	Warning Fmax

11	0x0800	Warning Analog Input MF1IA
12	0x1000	Warning Analog Input MF14A
13	0x2000	Warning Systembus
14	0x4000	Warning Udc
15	0x8000	Warning *Application warning status* **273**

The meaning of the individual warnings are described in detail in the Operating Instructions.

18.3Application warning Messages

The "Warning Message Application" is an additional information to the "Warning" bit. The Application warning messages are given via parameter *Application Warnings* **274**, bit-coded according to the following scheme.

Parameter *Application Warnings* **273** shows the warnings in clear text on the operator panel and the PC software tool VPlus.

Use Parameter *Application Warnings* **274** to access the Application warning codes via Field bus.

Warning Messages Application

Bit-No.	**Warning Code**	**Description**	
0	0x0001	BELT	- Belt
1	0x0002	SW-LIM CW	– SW Limit Switch Positive
2	0x0004	SW-LIM CCW	– SW Limit Switch Negative
3	0x0008	HW-LIM CW	– HW Limit Switch Positive
4	0x0010	HW-LIM CCW	– HW Limit Switch Negative
5	0x0020	CONT	– Contouring Error
6	0x0040	ENC	– Warning Absolute encoder
7	0x0080	User 1	– User Warning 1
8	0x0100	User 2	– User Warning 2
9	0x0200	(reserved)	
10	0x0400	(reserved)	
11	0x0800	(reserved)	
12	0x1000	(reserved)	
13	0x2000	(reserved)	
14	0x4000	(reserved)	
15	0x8000	(reserved)	

For details on the warnings, refer to the frequency inverter Operating Instructions and the "Positioning" application manual.

The Warning Bit 6 "Absolute encoder" can be read out via Parameter **1274** in VPlus or **1273** via field bus.

18.4Error messages

The fault code that is stored after a fault occurs is made up of the fault group FXX (high Byte, hexadecimal) followed by the code number XX (low Byte, hexadecimal).

	Communication fault		
	Code		**Meaning**
Motion Control Interface	F04	00	Motor temperature too high or temperature evaluation connection defective. Check cables and connections.
		04	Control Deviation Position Controller
	F14	02	Reference value signal on analog input MF4IA faulty, check signal.
		20	Inconsistent direction of rotation of the resolver in comparison to the motor • Swap two motor phases, for example U and V. Note the direction of rotation of the motor. • Invert the direction of rotation of the resolver by exchanging Sin+ and Sin-. Ensure that the analog part and digital part of the encoder use the same sense of direction. • Invert the direction of rotation of the resolver via parameter *Operation mode* **493**.
		21	Resolver synchronization not successful. Check resolver signal for interferences.
		22	Resolver counting error: Check resolver signal for interferences.
		23	Resolver pole pair number is invalid. The ratio of the no. of pole pairs to the no. of resolver pole pairs must be an integer number. Check parameters *No. of Pole Pairs* **373** for the motor and und *No. of Pole Pairs* **381** for the resolver, correct if necessary.
		24	Open circuit: check resolver connections and lines.
		30	Speed sensor signal is faulty, check connections
		31	One track of the speed sensor signal is missing, check connections.
		32	Direction of rotation of speed sensor wrong, check connections.
		33	Speed sensor 2, divisions marks wrong, check speed sensor
		34	Division marks of speed sensor signal 2 too low, check speed sensor.
		35	Division marks of speed sensor signal 2 too high, check speed sensor.
		42	Pos. SW-Limit Switch
		43	Neg. SW-Limit Switch
		44	Pos. SW-Lim. Switch < Neg. SW-Lim. Switch
		45	Pos. and Neg. HW-Lim Switch Simultaneously
		46	Limit Switch Incorrect Wired!
		47	Pos. HW Limit Switch
		48	Neg. HW Limit Switch
		49	Both directions locked
		51	Clockwise Operation Locked
		52	Anti-Clockwise Operation Locked
		53	Systembus-Synchronization not activated
		55	Flying Homing: No edge selected for P. 1280
		60	Pos. HW-Lim. Switch: Illegal Signal Source
		61	Pos. HW-Lim. Switch: Input disabled by PWM-/FF-Input
		62	Pos. HW-Lim. Switch: Input disabled by Index-Contr.
		63	Pos. HW-Lim. Switch: Wrong Op.-Mode for MF1 (terminal X210B.6)
		64	Pos. HW-Lim. Switch: Input disabled by Encoder 1
		65	Pos. HW-Lim. Switch: Input disabled by Encoder 2
		67	Pos. HW-Lim. Switch: Input disabled by MF3 (terminal X210B.3)
		68	Pos. HW-Lim. Switch: Input disabled by MF4 (terminal X410B.4)
		70	Neg. HW-Lim. Switch: Illegal Signal Source
		71	Neg. HW-Lim. Switch: Input disabled by PWM-/FF-Input
		72	Neg. HW-Lim. Switch: Input disabled by Index-Contr.
		73	Neg. HW-Lim. Switch: Wrong Op.-Mode for MF1I
		74	Neg. HW-Lim. Switch: Input disabled by Encoder 1
		75	Neg. HW-Lim. Switch: Input disabled by Encoder 2
		77	Neg. HW-Lim. Switch: Input disabled by MF3 (terminal X210B.3)
		78	Neg. HW-Lim. Switch: Input disabled by MF4 (terminal X410B.4)
		80	Error during EM module initialization. Initialization of expansion module failed. Check if expansion module is plugged in correctly.
		81	EM-module communication failure. Communication between expansion module and frequency inverter is faulty. Check EMC.
		82	General EM module error. Error on expansion module. One of the following errors F1483 … F1493 has occurred.

Communication fault

	Code		Meaning	
		83	EM-AUT: No ext. 24V. Via parameter *Power supply* **1186**, an operation mode using an external power supply was selected, but there is no external voltage. Connect external power source or change operation mode.	
		84	EM-AUT: Ext. 24V voltage level too low. The voltage level of the external power supply is too low or the external power supply is overloaded. Check the voltage level of the external power supply.	
		85	EM-AUT: Int. 24V voltage level too low. The internal power supply to the encoder provided by the frequency inverter is overloaded. Check the connections at the control terminals.	
		86	EM-AUT: No sense line. • Via parameter Power supply 1186, an operation mode using a sense measuring line was selected, but no sense line is connected. Connect sense measuring line or select another operation mode. • Sense measuring line defective or broken. Check cables and connections.	
		87	EM-AUT: A/B track missing. A/B track not found. Connect A/B track. A/B track cable broken. Check cables and connections. Check set *Division marks* **1183**. In some cases, you may have to reset the device if the error occurs directly after connection of mains supply.	
		88	EM-AUT: Fault correction A/B track. Error during evaluation of A/B track. Required measuring accuracy not reached. The offset and amplification error correction for the A/B track has reached its maximum.	
Motion Control Interface	F15	xx	User-Defined Error in Motion Block xx (1 ≤ xx ≤ 32)	
		70	No Homing Done	
		71	Homing : Encoder-Mode w.o. Z-Impulse	
		72	Both Directions Locked	
		73	No Touch Probe Signal Detected	
		74	M/S Position Correction: Master Position source not set	
		75	P. 1118 is set too high	
CANopen	F20	21	Bus OFF	
		22	Guarding failure	
		23	Error state	
		24	SYNC error (SYNC timing)	
		25	NMT state change (operational → xxx)	
		26	RxPDO1 length error	(number of received bytes different to mapping)
		27	RxPDO2 length error	
		28	RxPDO3 length error	
		2A	RxPDO1 Timeout	(The RxPDO was not received in the set time. Check object 0x140n/5 Event time.)
		2B	RxPDO2 Timeout	
		2C	RxPDO3 Timeout	
Systembus	F21	nn	Fault report to systembus master in fault in systembus slave nn = Node ID of slave (hex)	
	F22	00	Communication fault, systembus, timeout SYNC telegram	
		01	Communication fault, systembus, timeout RxPDO1	
		02	Communication fault, systembus, timeout RxPDO2	
		03	Communication fault, systembus, timeout RxPDO3	
		10	Communication fault, systembus, bus OFF	

Communication fault

	Code		Meaning
CANopen	F23	nn	Heartbeat failure – nn = node address of the failed subscriber (hex)

The Actual error message can be read out by parameter access via parameter *Actual Error* **260** and via the Emergency Message or Object 0x1014.

Parameter *Actual Error* **259** shows the actual error in clear text on the operator panel and the PC software tool VPlus.

In addition to the fault messages stated, there are further fault messages described in the Operating Instructions. The faults of the Motion Control Interface (F14xx, F15xx) are described detailed in the application manual "Positioning".

For more information on VPlus, see chapter 17.

18.5Conversions

The speeds can be converted into other speed formats using the formulas in this chapter:

Frequency [Hz] to	Speed [rpm]
	Speed in user units [u/s]
Speed [rpm] to	Frequency [Hz]
	Speed in user units [u/s]
Speed in user units [u/s]	Speed [rpm]
	Frequency [Hz]

18.5.1 Speed [rpm] to Frequency [Hz]

$$f\ [\mathrm{Hz}] = \frac{n\left[\mathrm{min}^{-1}\right] \times No.of\ pole\ pairs\ (P.373)}{60}$$

18.5.2 Frequency [Hz] to Speed [rpm]

$$n[rpm] = \frac{f\ [\mathrm{Hz}] \times 60}{No.of\ pole\ pairs\ (P.373)}$$

18.5.3 Speed in user units [u/s] to Frequency [Hz]

$$f\ [\mathrm{Hz}] = v\ [\frac{\mathrm{u}}{\mathrm{s}}] \times \frac{No.of\ pole\ pairs\ (P.373)}{Feed\ Constant\ (P.1115)} \times \frac{Gear\ Box: Motor\ Shaft\ Revolutions\ (P.1117)}{Gear\ Box: Driving\ Shaft\ Revolutions(P.1116)}$$

18.5.4 Frequency [Hz] to Speed in user units [u/s]

$$v\ [\frac{\mathrm{u}}{\mathrm{s}}] = f\ [\mathrm{Hz}] \times \frac{Feed\ Constant\ (P.1115)}{No.of\ pole\ pairs\ (P.373)} \times \frac{Gear\ Box: Driving\ Shaft\ Revolutions(P.1116)}{Gear\ Box: Motor\ Shaft\ Revolutions\ (P.1117)}$$

18.5.5 Speed in user units [u/s] to Speed [rpm]

$$n\ [rpm] = v\ [\frac{\mathrm{u}}{\mathrm{s}}] \times \frac{60}{Feed\ Constant\ (P.1115)} \times \frac{Gear\ Box: Motor\ Shaft\ Revolutions\ (P.1117)}{Gear\ Box: Driving\ Shaft\ Revolutions\ (P.1116)}$$

18.5.6 Speed [rpm] to Speed in user units [u/s]

$$v\,[\frac{\mathrm{u}}{\mathrm{s}}] = n\,[rpm] \times \frac{Feed\ constant\ (P.1115)}{60} \times \frac{Gear\ Box{:}\ Driving\ Shaft\ Revolutions\ (P.1116)}{Gear\ Box{:}\ Motor\ Shaft\ Revolutions\ (P.1117)}$$

18.6Object support in the Software versions and XML files

The support of CANopen was extended in various steps in the firmware. The following table lists which objects are supported with the different software version. Objects, that were added or where changes were made are marked in light blue color.

Long Object names are shortened sensibly in the table to maintain the overview.

Firmware GSDXML EDS	8.0.6+ BV_ANG.eds	Firmware GSDXML EDS	8.0.6+ BV_ANG.eds
0x1000 Device Type	x	0x2nnn ANG parameter access	x
0x1001 Error register	x	0x3000 Sync Jitter [1)]	x
0x1005 COB-ID SYNC Object [1)]	x	0x3001 Digital In actual values	x
0x1006 Comm. Cycle Period [1)]	x	0x3002 Digital Out act. values	x
0x1007 Syn. Window length [1)]	x	0x3003 Digital Out set values	x
0x1008 Manuf. Device name	x	0x3004 Boolean Mux	x
0x1009 Manuf. Hardw. Vers.	x	0x3005 Boolean DeMux	x
0x100A Manuf. Softw. Vers.	x	0x3006 Percentage Set value	x
0x100C Guard Time [1)]	x	0x3007 Percentage Act. value 1	x
0x100D Life time factor [1)]	x	0x3008 Percentage Act. value 2	x
0x1010 Store parameters	x	0x3011 Act. Value Word 1	x
0x1011 Restore parameters	x	0x3012 Act. Value Word 2	x
0x1014 COB-ID emerg. object [1)]	x	0x3021 Act. Value Long 1	x
0x1016 Consumer heartb. time [1)]	x	0x3022 Act. Value Long 2	x
0x1017 Producer heartb. time [1)]	x	0x3111 Ref. Value Word 1	x
0x1018 Identity object	x	0x3112 Ref. Value Word 2	x
0x1029 Error behavior	x	0x3121 Ref. Value Long 1	x
0x1200 Server SDO param. [1)]	x	0x3122 Ref. Value Long 2	x
0x1400 RxPDO1 comm. param. [1)]	x	0x5F10 Gear Factor	x
0x1401 RxPDO2 comm. param. [1)]	x	0x5F11 Phasing 1	x
0x1402 RxPDO3 comm. param. [1)]	x	0x5F12 Phasing 2	x
0x1600 RxPDO1 map. param.	x	0x5F13 Phasing 3	x
0x1601 RxPDO2 map. param.	x	0x5F14 Phasing 4	x
0x1602 RxPDO3 map. param.	x	0x5F15 In Gear Threshold	x
0x1800 TxPDO1 comm. param. [1)]	x	0x5F16 In Gear Time	x
0x1801 TxPDO2 comm. param. [1)]	x	0x5F17 Position Controller	x
0x1802 TxPDO3 comm. param. [1)]	x	0x5F18 M/S Synchronization Offset	x
0x1A00 TxPDO1 map. param.	x	0x5FF0 Active motion block	x
0x1A01 TxPDO2 map. param.	x	0x5FF1 Motion block to resume	x
0x1A02 TxPDO3 map. param.	x		

1) CANopen only.

Firmware GSDXML EDS	8.0.6+ BV_ANG.eds	Firmware GSDXML EDS	8.0.6+ BV_ANG.eds
0x2nnn ANG parameter access	x	0x6007 Abort connect. option c.	x
0x3000 Sync Jitter	x	0x603F Error code	x
0x3001 Digital In actual values	x	0x6040 Controlword	x
0x3002 Digital Out act. values	x	0x6041 Statusword	x
0x3003 Digital Out set values	x	0x6042 *v*/target velocity	x
0x3004 Boolean Mux	x	0x6043 *v*/Velocity demand	x
0x3005 Boolean DeMux	x	0x6044 *v*/velocity actual value	x
0x3006 Percentage Set value	x	0x6046 *v*/velocity min max amount	x
0x3007 Percentage Act. value 1	x	0x6048 *v*/velocity acceleration	x
0x3008 Percentage Act. value 2	x	0x6049 *v*/velocity deceleration	x
0x3011 Act. Value Word 1	x	0x604A *v*/velocity quick stop	x
0x3012 Act. Value Word 2	x	0x6060 Modes of Operation	x
0x3021 Act. Value Long 1	x	0x6061 Modes of Op. display	x
0x3022 Act. Value Long 2	x	0x6064 Position actual value	x
0x3111 Ref. Value Word 1	x	0x6065 Following error window	x
0x3112 Ref. Value Word 2	x	0x6066 Following error timeout	x
0x3121 Ref. Value Long 1	x	0x6067 Position Window	x
0x3122 Ref. Value Long 2	x	0x6068 Position Window time	x
0x5F10 Gear Factor	x	0x606C Velocity act. value	x
0x5F11 Phasing 1	x	0x606D Velocity window	x
0x5F12 Phasing 2	x	0x606E Velocity window time	x
0x5F13 Phasing 3	x	0x606F Velocity Threshold	x
0x5F14 Phasing 4	x	0x6070 Velocity Threshold time	x
0x5F15 In Gear Threshold	x	0x6071 Target Torque	x
0x5F16 In Gear Time	x	0x6077 Torque Actual value	x
0x5F17 Position Controller	x	0x6078 Current Actual value	x
0x5F18 M/S Synchronization Offset	x	0x6079 DC link circuit voltage	x
0x5FF0 Active motion block	x	0x607A Target Position	x
0x5FF1 Motion block to resume	x	0x607C Home Offset	x
		0x6081 Profile Velocity	x
		0x6083 Profile Acceleration	x
		0x6084 Profile Deceleration	x
		0x6085 Quick Stop deceleration	x
		0x6086 Motion Profile type	x
		0x6091 Gear ratio	x
		0x6092 Feed constant	x
		0x6098 Homing method	x
		0x6099 Homing speeds	x
		0x609A Homing acceleration	x
		0x60F4 Following err. Act. Val.	x
		0x60F8 Max. Slippage	x
		0x60FF Target Velocity	x
		0x6502 Supported Drive modes	x

18.7Recommended encoder settings

Please note that the settings described in the following are only recommendations for standard variant of the relevant encoders. When it comes to setup, always refer to the encoder manufacturer's data sheet.

In the case of synchronous servomotors, the *Offset* **1188** must be set up according to chapter **7.4.6** "**Offset Absolute encoder**".

18.7.1 SinCos encoders:

Encoder	B.C.	1183	1184	1186	1187	1271	1272	1270
Heidenhain ERN 1387 Variante: 2048 Ampl.	S1	2048	700	1)	5,0 V	2)	2)	2)

Heidenhain ERN 1185 Variante: 512 Ampl.	S2	512	700	[1]	5,0 V	[2]	[2]	[2]
Heidenhain ERN 1185 Variante: 2048 Ampl.	S3	2048	700	[1]	5,0 V	[2]	[2]	[2]

B.C. = Bonfiglioli Code used at motors of series BCR & BTD.

[1] Please refer to chapter **7.4.3** for setup of parameter *Power supply* **1186**.

[2] Not evaluated due to the *Tracks/Protocol* **1184** settings chosen.

Due to the great number of encoder types and special solutions not documented publicly, Bonfiglioli Vectron will not accept any responsibility for the settings specified.

18.7.2 Hiperface encoders:

Encoder	B.C.	1183	1184	1186	1187	1271	1272	1270
Sick SRS50	H1	1024	3109	[1]	8,0 V	15	0	[2]
Sick SRM50	H2	1024	3109	[1]	8,0 V	15	12	[2]
Sick SKS36	H3	128	3109	[1]	8,0 V	12	0	[2]
Sick SKM36	H4	128	3109	[1]	8,0 V	12	12	[2]
Sick SEL37	H5	16	3109	[1]	8,0 V	9	12	[2]
Sick SEK37	H6	16	3109	[1]	8,0 V	9	0	[2]
Sick SEL52	H7	16	3109	[1]	8,0 V	9	12	[2]
Sick SEK52	H8	16	3109	[1]	8,0 V	9	0	[2]

B.C. = Bonfiglioli Code used at motors of series BCR & BTD.

[1] Please refer to chapter **7.4.3** for setup of parameter *Power supply* **1186**.

[2] Not evaluated due to the *Tracks/Protocol* **1184** settings chosen.

Due to the great number of encoder types and special solutions not documented publicly, Bonfiglioli Vectron will not accept any responsibility for the settings specified.

18.7.3 EnDat2.1 encoders:

Encoder	B.C.	1183	1184	1186	1187	1271	1272	1270
Heidenhain ECI 1319	D1	32	1101	[1]	5,0 V	[3]	[3]	[2]
Heidenhain EQI 1331	D2	32	1101	[1]	5,0 V	[3]	[3]	[2]
Heidenhain ECN 1113	D3	512	1101	[1]	5,0 V	[3]	[3]	[2]
Heidenhain EQN 1125	D4	512	1101	[1]	5,0 V	[3]	[3]	[2]
Heidenhain ECN 1313 Variante: 512 Ampl.		512	1101	[1]	5,0 V	[3]	[3]	[2]
Heidenhain ECN 1313 Variante: 2048 Ampl.		2048	1101	[1]	5,0 V	[3]	[3]	[2]
Heidenhain EQN 1325 Variante: 512 Ampl.		512	1101	[1]	5,0 V	[3]	[3]	[2]
Heidenhain EQN 1325 Variante: 2048 Ampl.		2048	1101	[1]	5,0 V	[3]	[3]	[2]

B.C. = Bonfiglioli Code used at motors of series BCR & BTD.

[1] Please refer to chapter **7.4.3** for setup of parameter *Power supply* **1186**.

[2] Not evaluated due to the *Tracks/Protocol* **1184** settings chosen.

[3] Parameters *Bits/Turn* **1270** and *Bits Multiturn* **1271** are not evaluated due to the setting of *Track signal* **1184**=1101. The values are applied directly from the EnDat 2.1 encoder.

Due to the great number of encoder types and special solutions not documented publicly, Bonfiglioli Vectron will not accept any responsibility for the settings specified.

18.7.4 SSI encoders, rotary:

Due to the great number of SSI encoder variants, only an extract of specifications can be shown here. Please refer to the encoder manufacturer's data sheets for the parameter settings.

Encoder / Parameter	1183	1184	1186	1187	1271	1272	1268	1269	1270
Sick AFM60B-BxPC032768 (without incremental track)	32768	50xx	1-internal	5.0 V	15	12	125 us	HHH	-
Kübler Sendix 5863 (with SinCos track)	2048	61xx	1-internal	5.0 V	17	12	125 us	-	-
Sick AFM60B-TxKx001024	1024	61xx	1-internal	5.0 V	10	12	125 us	HHH	-

(SinCos)									

[1] Please refer to chapter **7.4.3** for setup of parameter *Power supply* **1186**.

Due to the great number of encoder types and special solutions not documented publicly, Bonfiglioli Vectron will not accept any responsibility for the settings specified.

18.7.5 SSI encoders, linear encoders:

Encoder	1183	1184	1186	1187	1271	1272	1268	1269	1270
Leuze AMS304i 1120	---	50xx	1 [2]	5.0 V	24 Bit distributed [3]		2	H [4]	-
Sick DME4000-111	---	50xx	1 [2]	5.0 V	24 Bit distributed [3]		8	H	-
Vahle LIMAX2S-03-050-1000-SSG0-U	---	50xx	1 [2]	5.0 V	24 Bit distributed [3]		16	H	-

[1] Please refer to chapter **7.4.3** for setup of parameter *Power supply* **1186**.

[2] Due to the high power consumption, this encoder requires an external power supply. In this case, set *Power supply* **1186** = "1-Internal" and connect the encoder to an external power supply.

[3] For setup of this parameter, refer to chapter **7.3.5**.

[4] Note behavior changeable via encoder parameters.

Due to the great number of encoder types and special solutions not documented publicly, Bonfiglioli Vectron will not accept any responsibility for the settings specified.

Index

Bonfiglioli Worldwide Locations

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
⚙ | 🔧 | 🛒 | 🛠

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 Vectron GmbH
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 - Mechatronic & Motion
Survey No. 528/1
Perambakkam High Road Mannur Village,
Sriperumbudur Taluk Chennai - 602 105
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 Bazzane, 33/A
40012 Calderara di Reno
Tel. +39 051 6473111
⚙ | 🔧 | 🛒 | 🛠

Mobility & Wind Industries
Via Enrico Mattei, 12 Z.I. Villa Selva
47100 Forlì
Tel. +39 0543 789111
⚙ | 🔧 | 🛒 | 🛠

Discrete Manufacturing & Process Industries
Via Sandro Pertini lotto 7b
20080 Carpiano
Tel. +39 02985081
🔧 | 🛒 | 🛠

Bonfiglioli Mechatronic Research S.p.A
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
⚙ | 🔧 | 🛒 | 🛠

Bonfiglioli

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 121 R1

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