

CANopen Communication module CM-CAN

Frequency Inverter 230V / 400V

ACTIVE Cube



General points on the documentation

The present supplement of the documentation is valid for frequency inverters of the device series ACU 201 and ACU 401. The information necessary for the assembly and application of the CANopen communication module CM-CAN is documented in this guidance.

For better clarity, the user documentation is structured according to the customer-specific demands made on the frequency inverter.

Brief instructions "Quick Start Guide"

The brief instruction "Quick Start Guide" describes the fundamental steps for mechanical and electrical installation of the frequency inverter. The guided commissioning supports you in the selection of necessary parameters and the software configuration of the frequency inverter.

Operating instructions

The operating instructions document the complete functionality of the frequency inverter. The parameters necessary for specific applications for adaptation to the application and the extensive 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 connected with the use of the frequency inverter is described specific to the application.

Installation instructions

As a complement to the brief instructions and the operating instructions, the installation instructions describe the installation and use of devices.

The documentation and additional information can be requested via your local representation of the BONFIGLIOLI company. The following pictograms and signal words are used for the purposes of the current documentation:



Danger!

means a direct threatening danger. Death, serious damage to persons and considerable damage to property will occur if the precautionary measure is not taken.



Warning!

marks a possible threat. Death, serious damage to persons and considerable damage to property can be the consequence if attention is not paid to the text.



Caution!

refers to an indirect threat. Damage to people or property can be the result.

Attention!

refers to a possible operational behavior or an undesired condition that can occur in accordance with the reference text.

Note

marks information that facilitates handling for you and supplements the corresponding part of the documentation.



Warning! In installation and commissioning, comply with the information in the documentation. You as a qualified person must read the documentation carefully before the start of the activity and obey the safety instructions. For the purposes of the instructions, "qualified person" designates a person acquainted with the installation, assembly, commissioning and operation of the frequency inverters and possessing the qualification corresponding to the activity.

Contents

1	General safety and application information	5
1.1	General information.....	5
1.2	Proper use.....	6
1.3	Transport and storage	6
1.4	Handling and installation.....	6
1.5	Electrical connection.....	7
1.6	Operating information	7
1.7	Maintenance and servicing	7
2	Introduction	8
3	Installation/Disassembly of the communication module	10
3.1	Installation	10
3.2	Disassembly	11
4	Connector pin assignment/bus termination/line	12
5	Baud rate setting/line lengths	13
6	Setting the node number	13
7	Assigning the CANopen interface	14
8	Operational behavior on bus failure	15
9	CANopen Overview	16
9.1	Communication Objects.....	16
9.2	Application Objects.....	16
9.3	SDO Function	17
9.4	PDO Function	17
9.5	Emergency Function	18
9.6	SYNC	18
9.7	NMT Functions	18
9.8	Guarding	20
9.9	Heartbeat.....	21
10	Objects	22
10.1	Objects tabular overview	22
10.1.1	Communication objects	22
10.1.2	Manufacturer objects	25
10.1.3	Device profile objects.....	26
10.2	Communication Objects (0x1nnn)	30
10.2.1	0x1000/0 Device Type.....	30
10.2.2	0x1001/0 Error Register	30
10.2.3	0x1005/0 COB-ID SYNC Message	31
10.2.4	0x1006/0 Communication Cycle Period	32

10.2.5	0x1007/0 Synchronous window length.....	33
10.2.6	0x1008/0 Manufacturer Device Name	33
10.2.7	0x1009/0 Manufacturer Hardware Version	33
10.2.8	0x100A/0 Manufacturer Software Version	33
10.2.9	0x100C/0 Guard Time	33
10.2.10	0x100D/0 Lifetime Factor	33
10.2.11	0x1010/n Store Parameters.....	34
10.2.12	0x1011/n Restore default Parameters.....	34
10.2.13	0x1014/0 COB-ID Emergency Message.....	35
10.2.14	0x1016/n Consumer Heartbeat Time	36
10.2.15	0x1017/0 Producer Heartbeat Time.....	36
10.2.16	0x1018/n Identity Object	36
10.2.17	0x1029/n Error Behavior	37
10.2.18	0x1200/n SDO Server Parameter	37
10.2.19	0x1400/n, 0x1401/n, 0x1402/n RxPDO Communication Parameters.....	38
10.2.20	0x1600/n, 0x1601/n, 0x1602 RxPDO Mapping Parameters	39
10.2.21	0x1800/n, 0x1801/n, 0x1802/n TxPDO Communication Parameters	40
10.2.22	0x1A00/n, 0x1a01/n, 0x1A02/n TxPDO Mapping Parameters.....	43
10.3	Manufacturer objects (0x2nnn)	44
10.3.1	Handling of data sets/cyclic writing	44
10.3.2	SDO examples (expedited transfer only)	45
10.4	Manufacturer objects (0x3000 ... 0x5FFF).....	50
10.4.1	0x3000/0 SYNC Jitter	50
10.4.2	0x3001/0 Digital In actual value.....	51
10.4.3	0x3002/0 Digital Out actual value.....	51
10.4.4	0x3003/0 Digital Out set values	51
10.4.5	0x3004/0 Boolean Mux.....	52
10.4.6	0x3005/0 Boolean DeMux.....	53
10.4.7	0x3006/0 Percentage set value	54
10.4.8	0x3007/0 Percentage actual value.....	55
10.4.9	0x5FF0/0 Active motion block	55
10.5	Device Profile Objects (0x6nnn)	56
10.5.1	0x6007/0 Abort Connection option code	56
10.5.2	0x603F/0 Error code	59
10.5.3	0x6040/0 Controlword.....	60
10.5.4	0x6041/0 Statusword	61
10.5.5	0x6042/0 Target velocity.....	62
10.5.6	0x6043/0 Target velocity demand	62
10.5.7	0x6044/0 Control effort.....	62
10.5.8	0x6046/n Velocity min max amount.....	63
10.5.9	0x6048/n Velocity acceleration.....	64
10.5.10	0x6049/n Velocity deceleration	65
10.5.11	0x604A/n Velocity quick stop.....	66
10.5.12	0x6060/0 Modes of operation	67
10.5.13	0x6061/0 Modes of operation display	67
10.5.14	0x6064/0 Position actual value	67
10.5.15	0x6065/0 following error window	68
10.5.16	0x6066/0 following error time out.....	68
10.5.17	0x6067/0 position window.....	69
10.5.18	0x6068/0 position window time	69
10.5.19	0x6071/0 Target Torque	69
10.5.20	0x6077/0 Torque actual value	69
10.5.21	0x6078/0 Current actual value.....	70
10.5.22	0x6079/0 DClk circuit voltage.....	70
10.5.23	0x607A/0 Target position	70
10.5.24	0x607C/0 Home offset	70
10.5.25	0x6081/0 Profile velocity	70

10.5.26	0x6083/0 Profile acceleration.....	70
10.5.27	0x6084/0 Profile deceleration	70
10.5.28	0x6085/0 Quick stop deceleration	71
10.5.29	0x6086/0 Motion profile type.....	71
10.5.30	0x6091/n Gear ratio.....	72
10.5.31	0x6092/n Feed constant.....	72
10.5.32	0x6098/0 Homing method.....	73
10.5.33	0x6099/n Homing speeds.....	75
10.5.34	0x609A/0 Homing acceleration	75
10.5.35	0x60C1/1 Interpolation data record.....	76
10.6	Special object and parameter dependencies.....	76
11	Inverter Control	78
11.1	Control via digital inputs/remote digital inputs.....	79
11.2	Control via state machine.....	82
11.3	Non motion control configurations	86
11.3.1	Behavior in quick stop.....	86
11.3.2	Behavior in transition 5	87
11.3.3	Reference value / actual value	88
11.4	Motion control configurations	89
11.4.1	Velocity mode	90
11.4.2	Profile position mode	92
11.4.3	Interpolated position mode.....	98
11.4.4	Homing mode.....	101
11.4.5	Table travel record.....	103
12	Parameter list.....	109
12.1	Actual values	109
12.2	Parameters	109
13	Annex	111
13.1	Warning messages	111
13.2	Fault messages	111
14	Motion-control-interface for Profibus connection	112

1 General safety and application information

This documentation has been produced with the greatest of care and extensively and repeatedly checked. For reasons of clarity, not all the detailed information on all types of the product and also not every imaginable case of installation, operation or maintenance has been taken into account. If you require further information or if specific problems which are not dealt with extensively enough in the documentation exist, you can request the necessary information via the local representation of the BONFIGLIOLI company.

We would also point out that the contents of this documentation are not part of a previous or existing agreement, assurance or legal relationship and are not intended to amend the same. All obligations of the manufacturer result from the underlying purchase contract, which also contains the complete and solely valid warranty regulation. These contractual warranty provisions are neither extended nor limited by the production of this documentation.

The manufacturer reserves the right to correct or amend the contents and the product information as well as omissions without prior notification and assumes no kind of liability for damage, injuries or expenditure to be put down to the aforementioned reasons.

1.1 General information



Warning! BONFIGLIOLI VECTRON frequency inverters have high voltage levels during operation, depending on their protection class, drive moving parts and have hot surfaces.

In the event of inadmissible removal of the necessary covers, improper use, wrong installation or operation, there is the risk of serious damage to persons or property.

To avoid any damage, only qualified staff may carry out the transport, installation, setup or maintenance work required. Comply with the standards EN 50178, IEC 60364 (Cenelec HD 384 or DIN VDE 0100), IEC 60664-1 (Cenelec HD 625 or VDE 0110-1), BGV A2 (VBG 4) and national provisions. Qualified persons within the meaning of this principal safety information are people acquainted with the installation, fitting, commissioning and operating of frequency inverters and the possible hazards and in possession of qualifications matching their activities.

1.2 Proper use



Warning! The frequency inverters are electrical drive components intended for installation in industrial plants or machines. Commissioning and start of intended operation are not allowed until it has been established that the machine corresponds to the provisions of the EC machine directive 98/37/EEC and EN 60204. According to the CE sign, the frequency inverters additionally fulfill the requirements of the low-voltage directive 2006/95/EC and standards EN 50178/DIN VDE 0160 and EN 61800-2. Responsibility for compliance with the EMC directive 89/336/EEC is with the user. Frequency inverters are available in a limited way and as components exclusively intended for professional use within the meaning of the EN 61000-3-2.

With the issue of the UL according to UL508c, the requirements of the CSA Standard C22.2-No. 14-95 have also been fulfilled.

The technical data and the information on connection and ambient conditions the rating plate and the documentation be complied with. The instructions must be read and fully understood before starting work on the device.

Do not connect any capacitive loads.

1.3 Transport and storage

Transport and storage are to be carried out in an adequate way in the original packaging. Storage shall be in dry rooms protected against dust and moisture with slight temperature fluctuations. Please observe the climatic conditions according to EN 50178 and the marking on the packaging.

The duration of storage without connection to the admissible reference voltage may not exceed one year.

1.4 Handling and installation



Warning! Damaged or destroyed components may not be put into operation because they may be a health hazard.

The frequency inverters are to be used according to the documentation, the directives and the standards. Handle carefully and avoid mechanical overload. Do not bend the components or change the insulation distances. Do not touch electronic components or contacts. The devices contain electrostatic sensitive components which can easily be damaged by improper handling. Any use of damaged or destroyed components shall be considered as a non-compliance with the applicable standards. Do not remove any warning signs from the device.

1.5 Electrical connection



Warning! Before any assembly or connection work, de-energize the frequency inverter. Make sure that the frequency inverter is de-energized. Do not touch the sockets, because the capacitors may still be charged. Comply with the information given in the operating instructions and on the frequency inverter label.

While working on the frequency inverters, obey the applicable standards BGV A2 (VBG 4), VDE 0100 and other national directives. Comply with the information in the documentation on electrical installation and the relevant directives. Responsibility for compliance with and examination of the limit values of the EMC product standard EN 61800-3 for variable-speed electrical drive mechanisms is with the manufacturer of the industrial plant or machine.

The documentation contains information on installation correct for EMC. The cables connected to the frequency inverters may not be subjected to an isolation test with a high test voltage without previous circuit measures.

1.6 Operating information



Warning! The frequency inverter may be connected to the power supply every 60s. Consider this for a jog operation of a mains contactor. 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. Install protective equipment if personal injury or material damage is possible. Before commissioning and the start of the intended operation, attach all the covers and check the sockets. Check additional monitoring and protective devices pursuant to EN 60204 and the safety directives applicable in each case (e.g. Working Machines Act, Accident Prevention Directives etc.). No connection work may be performed, while the system is in operation.

1.7 Maintenance and servicing



Warning! Unauthorized opening and improper interventions can lead to physical injury or damage to property. Repairs on the frequency inverters may only be carried out by the manufacturer or persons authorized by the latter. Check protective equipment regularly.

2 Introduction

This document describes the features of the CANopen communication for frequency inverters of the ACU series.

CANopen communication is available either with module CM-CAN or an EM-SYS module or another expansion module EM with CAN terminals on board like EM-IO-01. The expansion modules can be used with either System bus or CANopen, depending of the selection of the CANopen interface.

Possible combinations

CM-CAN	EM-SYS / EM with CAN driver
---	System bus
CANopen	System bus
CANopen	---
---	CANopen

The frequency inverter must be extended by either the CANopen communication module CM-CAN or a fitting EM module for the CAN connection.

The CM-CAN CANopen module is enclosed with the frequency inverter as a separate component and must be fitted by the user. This is described in detail in the "Assembly" chapter. For the assembly of the EM modules refer to the corresponding manual.

Note:

CM-CAN offers decoupled drivers, while EM modules have coupled drivers. BONFIGLIOLI VECTRON recommends using the CM-CAN module, especially in environments with critical EMC behavior.

For reasons of better readability, in the following chapters CM-CAN module is used representative for all modules able to establish CANopen communication.

Note:

These instructions are not to be understood as fundamental information on CANopen. They presuppose underlying knowledge of the methods and mode of effect of CAN open on the part of the user.

In some chapters, as an alternative to the KP500 control unit, the setting and displaying of values is described with the help of the VPlus control software. Operation of a PC with the VPlus control software requires an optional KP232 interface adapter.

In this document, connecting the hardware, relevant parameters and the available objects are shown.

The available objects are sub-divided according to:

Communication objects	(0x1nnn)	to DS301 V4.01
Manufacturer objects	(0x2nnn)	
Standardized objects	(0x6nnn)	to DS402 V1.1

The functions and objects are described as far as necessary in these instructions. For further information, reference is made here to the Draft Standards of the CiA. The standards to which reference is made are DS102, DS301 and DS402, which are available from:

CiA, CAN in AUTOMATION
Am Weichselgarten 26
D-91058 Erlangen

Tel.: +49 9131 69086-0
Fax: +49 9131 69086-79

Attention! With the help of the CM-CAN CANopen communication module, it is possible to access **ALL** parameters of the frequency inverter from the external control unit. Control of the access via the operation level, as with the KP500 control unit or the VPlus PC software, does not exist. A change of parameters with an unknown meaning to the user can lead to the inoperability of the frequency inverter.

Attention! ACTIVE CUBE inverters support two different types of configuration:

- Non motion control configurations
- Motion control configurations

Motion control configurations are set when parameter *configuration 30* = x40.

The inverter's behavior with respect to *controlword/statusword* and *modes of operation/modes of operation display* is different in the two different types of configuration.

Refer to chapter 11.3 for inverter control with non motion control and chapter 11.4 for inverter control with motion control.

3 Installation/Disassembly of the communication module

3.1 Installation

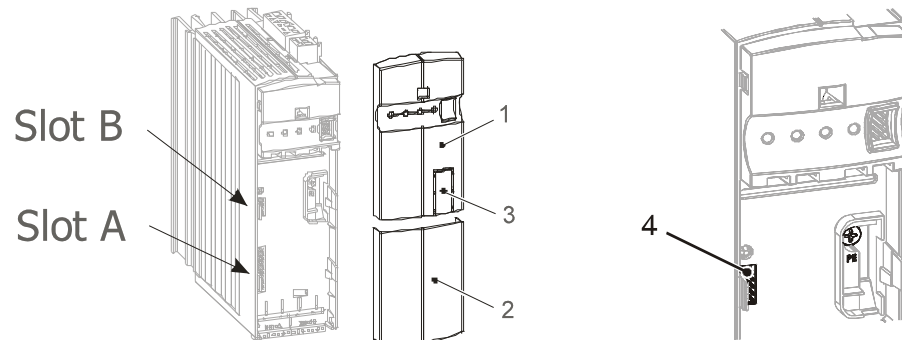
The communication module CM-CAN is pre-assembled in a case. Additionally, a PE spring is enclosed for PE connection (shield).



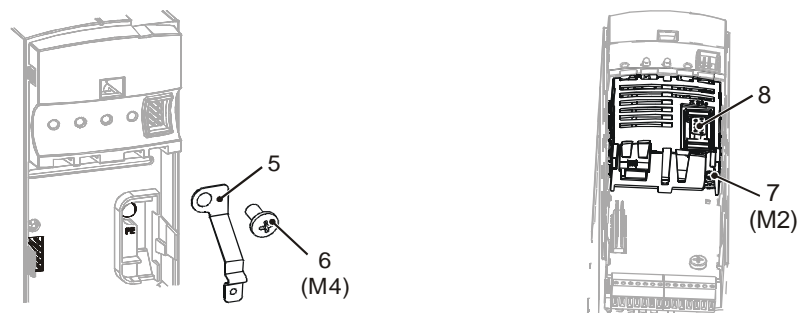
Caution! The frequency inverter must be disconnected from the power supply before installation of the communication module. Assembly under voltage is not permissible and will destroy the frequency inverter and/or the communication module. Do not touch the PCB visible on the back of the module, otherwise components may be damaged.

Work steps:

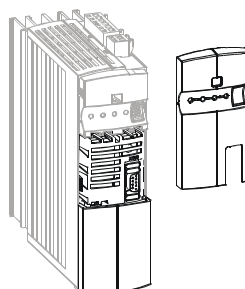
- Disconnect the frequency inverter from the mains voltage and protect it against being energized unintentionally.
- Remove covers (1) and (2) of the frequency inverter. Slot B (4) for the communication module is now accessible.



- Mount the supplied PE spring (5) using the M4 screw (6) in the unit. The spring must be aligned centrally.
- Insert the communication module in slot B (4) until it engages audibly.
- Fix the communication module by screwing the M2 screw (7) of the module to the PE spring (5).

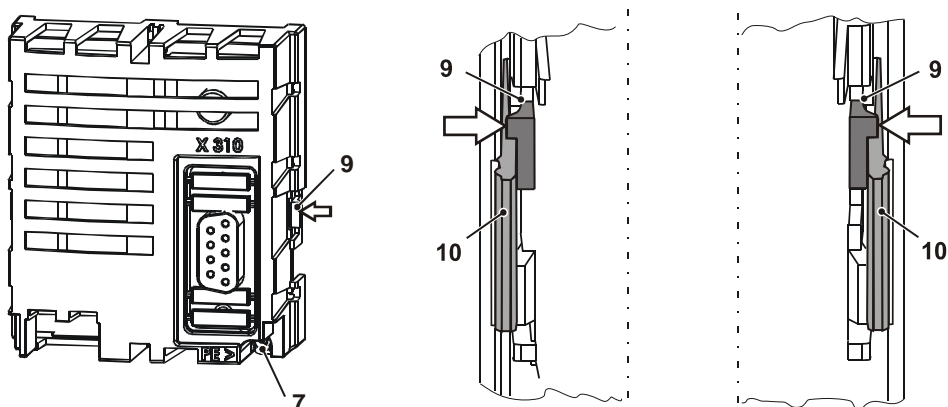


- In the upper cover (1), break out the pre-punched cutout (3) for the plug X310 (8).
- Mount the two covers (1) and (2).



3.2 Disassembly

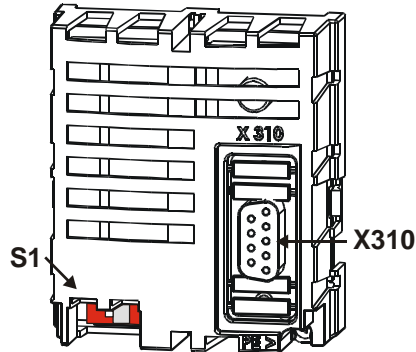
- Disconnect the frequency inverter from mains voltage and protect it against being energized unintentionally.
- Remove covers (1) and (2) of the frequency inverter.



- Loosen the M2 screw (7) on the communication module
- Unplug the communication module from Slot B (4) by unlocking the locking hooks (9) on the right and left hand side of the module from the case of the frequency inverter using a small screwdriver.
 The locking hooks (9) are located at the place where the locking hooks (10) for the upper cover (1) project from the case of the frequency inverter.
 - To do this, carefully insert the screwdriver in the gap between the case of the module and the frequency inverter and push the locking hook inwards in the direction of the arrow (←). As soon as the right hand side is unlocked, pull the module out a bit on the right hand side and hold it.
 - Hold the module on the right hand side while unlocking the locking hook on the left hand side in the same way (⇒).
 - Pull the module out of the slot by gently pulling on the right and left hand side alternately.
- Disassemble the PE spring (5).
- Mount the two covers (1) and (2).

4 Connector pin assignment/bus termination/line

The CAN connection is physically designed according to the ISO 11898 standards (CAN High-Speed).



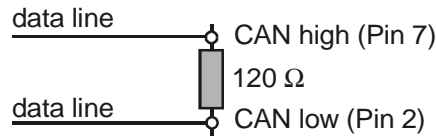
The **X310** (9-pole Sub-D) bus plug has been designed according to DS102 Version 2.0 (Bus node, option A).

Details can be seen from the following table on the occupancy of the bus plug.

The bus termination necessary on a phase in the physically first and last subscriber can be activated via **DIP switch S1** on the communication module.

The factory setting for the bus termination is **OFF**.

As an alternative, this is also possible via corresponding switching in the bus connection plugs.



Attention! Make absolutely sure that only one of the two possibilities for the bus termination is used and the bus termination is only switched on with the first and last subscriber. Otherwise, operation of the CANopen communication is not possible. The CAN Controller State is displayed via actual value parameter *CAN-State* **1291**.

Bus plug X310		
Pin	Name	Function
Housing	Shield	connected with PE
1	CAN_L	CAN Low bus interface, short-circuit resistant and function-insulated, max. current 60 mA
2	CAN_L	CAN Low bus interface, short-circuit resistant and function-insulated, max. current 60 mA
3	CAN_GND	Earth / GND
4	n.c.	not used
5	n.c.	not used
6	CAN_GND	Earth / GND
7	CAN_H	CAN High bus interface, short-circuit resistant and function-insulated, max. current 60 mA
8	CAN_H	CAN-High Bus-interface, short-circuit resistant and function-insulated, max. current 60 mA
9	-	Do NOT connect.

The drilled and shielded line is to be used for the bus line. The shield is to be implemented as a harness shield (**not a film shield**).

Attention! Connect the line screen with PE at both ends.

5 Baud rate setting/line lengths

The transmission speed of the CANopen communication module CM-CAN can be set via the parameter *CAN Baud rate* **385**.

Parameter		Setting		
No.	Description	Min.	Max.	Fact. sett.
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

Note:

Changing the baud rate causes a restart of the CANopen system (NOT a reset of the inverter).

6 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.	Fact. sett.
387	CAN Node Number	-1	127	-1

Note:

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.

Note:

Changing the node number causes a restart of the CANopen system (NOT a reset of the inverter).

7 Assigning the CANopen interface

Normally a CANopen connection is set up using the CM-CAN module. As an alternative for special applications, the CANopen connection can be switched to an EM-module with a CAN driver connection via parameter *CAN Interface 276*. This is only possible when parameter *Node-Id 900* of the system bus is set to **-1** !

<i>CAN Interface 276</i>	
Operation mode	Function
1 - CM-CAN	CM-CAN is used for the CANopen connection. Factory setting.
2 - EM-xxx	EM-xxx is used for the CANopen connection

Note:

The setting of *CAN interface 276* = 2 is only possible when an EM-module with CAN system bus is installed.

Even if only an EM-module with CAN system bus is installed, **276** first displays the value "1 – CM-CAN" which must then be changed to "2 – EM-xxx" in order to activate the EM-module for the CANopen connection.

If *CAN interface 276* = 2 is set, the transmission speed is set with *CAN Baud rate 385*. *Baud Rate 903* (System bus) is deactivated by setting **900** = -1. The same applies to all other parameters which have a function when using the system bus.

8 Operational behavior on bus failure

The operational behavior if the CANopen system fails due to BusOff, guarding, heart-beat, SYNC error, RxPDO length error or NMT state change (leaving NMT state operational) can be parameterized. The required behavior is set with parameter *CAN Error Behavior 388*.

<i>CAN Error Behavior 388</i>	Function
0 - No Reaction	Operating point is maintained
1 - Error	Device state machine changes immediately to state "fault" (factory setting)
2 - Stop	Device state machine processes command <i>'disable voltage'</i> and changes to state "switch on disabled"
3 - Quick-Stop	Device state machine processes command <i>'quick stop'</i> and changes to state "switch on disabled"
4 - Ramp-Stop + Error	Device state machine processes command <i>'disable operation'</i> and changes to state "fault" after the drive is stopped
5 - Quick-Stop + Error	Device state machine processes command <i>'quick stop'</i> and changes to state "fault" after the drive is stopped

Attention! The parameter settings *CAN Error Behavior 388* = 2 ... 5 are only relevant if parameter *Local/Remote 412* = "1 - Control via state machine" has been set as a supplement.

Parameter *CAN Error Behavior 388* corresponds to the device profile object 0x6007 *abort connection option code*.

For an exact description of the inverter's functional behavior, see chapter 10.5.1 object 0x6007 *abort connection option code*.

The error and warning behavior of the frequency inverter can be parameterized in various ways. If a failure of the bus system occurs in the setting *CAN Error Behavior 388* = 1, 4 or 5, the frequency inverter reports one of the following errors:

Communication error		
Code		Meaning
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 (number of received bytes different to mapping)
	28	RxPDO3 length error (number of received bytes different to mapping)
F23	nn	Heartbeat failure – nn = node address of the failed subscriber (hex)

9 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 (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 0xnxxx (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**).

9.1 Communication Objects

The communication objects are located in the index range 0x1xxx. They describe the communication behavior 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.

9.2 Application Objects

The application objects are divided into two groups again. The index range 0x2000 – 0x5FFF is reserved for manufacturer specific objects and the index range 0x6xxx is reserved for device profile specific objects. Device profile specific objects 0x6xxx are defined by DS402 drives and motion control. They are used for controlling the device application (start/stop, speed, motion control functions).

9.3 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 which 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 10.3 "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 8 data bytes.

SDO-message:

Byte	0	1	2	3	4	5	6	7
	command specifier	index		sub-index	data	data	data	data
	nn	LSB	MSB					

Default Identifiers:

TxSDO 384 + Node-ID
RxSDO 512 + Node-ID

9.4 PDO Function

The PDO (Process Data Objects) messages are messages with up to eight bytes of process data. The process data objects are mapped to the Rx/Tx-PDO's with the help of communication objects (communication/mapping parameter). Active Cube inverters support 3 RxPDO's (PLC → inverter) and 3 TxPDO's (inverter → PLC).

Process data objects are directly linked to application functions of the inverter.

PDO-message:

Byte	0	1	2	3	4	5	6	7
	data	data	data	data	data	data	data	data

The number of data bytes is 1 ... 8 and depends on the mapped objects. The byte alignment is in 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	180 + Node-ID
RxPDO1	512 + Node-ID	200 + Node-ID
TxPDO2	640 + Node-ID	280 + Node-ID
RxPDO2	798 + Node-ID	300 + Node-ID
TxPDO3	896 + Node-ID	380 + Node-ID
RxPDO3	1024 + Node-ID	400 + Node-ID

9.5 Emergency Function

In the event of a communication error or an error inside the inverter, the inverter sends an emergency message. This emergency message includes the relevant error information. After error acknowledgement (fault reset), an emergency message is sent with all data bytes set to zero.

9.6 SYNC

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.

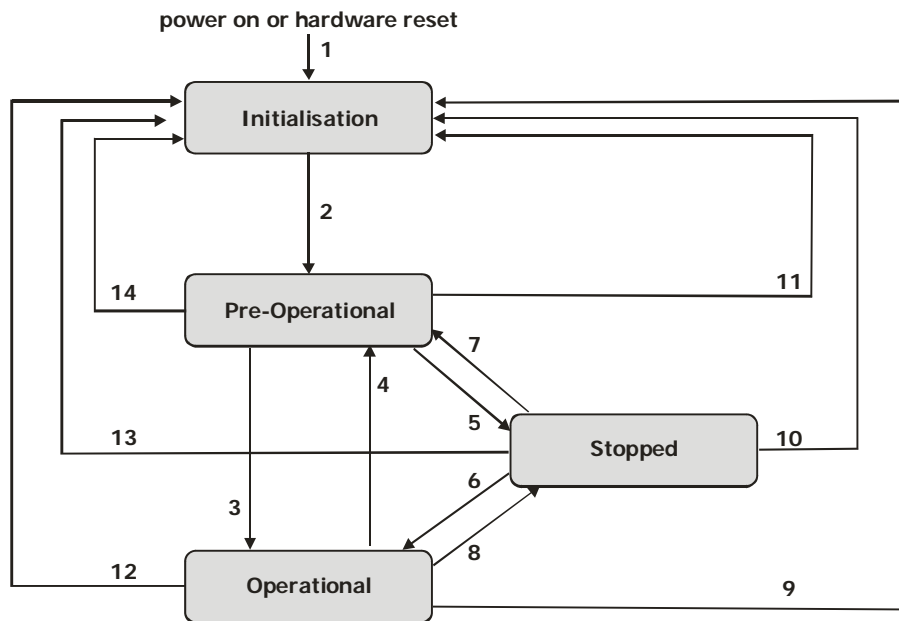
The SYNC message is a message with no data. The default Identifier = 128.

9.7 NMT Functions

The NMT (Network management) functions describe the NMT state machine and NMT error control 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:



Note:

A change of NMT-State may also be triggered by a communication (Bus-off, Guarding, etc.). The behavior of the NMT state machine in such a case is described in Chapter 10.2.17 "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
12, 13, 14	Reset Communication

In state transition "2" Initialisation → Pre-Operational the device sends the Boot-Up message.

Boot-Up message:

Identifier	Byte 0
1792 + Node-ID	0

NMT commands:

Identifier	Byte 0	Byte 1
0	Command Specifier	Node-ID
	cs	id

id = 0 command accepted by **all** devices
id = 1...127 command accepted by device with Node-ID = id

cs: 1 Start Remote Node
 2 Stop Remote Node
 127 Enter Pre-Operational
 129 Reset Node
 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

9.8 Guarding

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 on reception of the first guarding request.

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 behavior*.

Guarding sequence:

The PLC sends via a RTR (Remote Transmission Request) a guarding request with Identifier = 1792 + Node-ID (no data bytes). This remote frame is answered by the inverter with Identifier = 1792 and one data byte. The data byte contains the NMT state of the inverter.

PLC:

Identifier	
1792 + Node-ID	RTR

Inverter:

Identifier	Byte 0							
	NMT state + toggle bit							
1792 + 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 Zustand: 0 Boot-Up
 4 Stopped
 5 Operational
 127 Pre-Operational

9.9 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*. 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 behavior*.

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.

heartbeat message:

Identifier	Byte 0							
	NMT state							
1792 + Node-ID	7	6	5	4	3	2	1	0
	r	NMT state						

t: reserved (always 0)

NMT state:

0	Boot-Up
4	Stopped
5	Operational
127	Pre-Operational

10 Objects

The available objects are marked via Index/Subindex and are to be addressed via this identification. This chapter describes all available objects.

10.1 Objects tabular overview

The objects are displayed in the next tables. The following definitions apply:

Access type	
Read only	The PLC is only allowed to read the data from the ACU.
Read/write	The PLC is granted full access (read and write) to the ACU data.
Data type	
Unsigned32	32 Bit value: $0 \dots 2^{32}-1$
Unsigned16	16 Bit value: $0 \dots 2^{16}-1$ (0...65535)
Unsigned8	8 Bit value: $0 \dots 2^8-1$ (0...255)
Integer32	Signed 32 Bit value: $2^{31} \dots 2^{31}-1$
Integer16	Signed 16 Bit value: $2^{15} \dots 2^{15}-1$ (-32768...32767)
Integer8	Signed 8 Bit value: $2^7 \dots 2^7-1$ (-128...127)
PDO mapping	
No	This object cannot be used for PDO exchange, only SDO is applicable.
Tx	This object can be transmitted as PDO from ACU.
Rx	This object can be transmitted as PDO to ACU.

Note:

“Highest Sub-index supported” displays the highest Sub-index that is supported by this object.

10.1.1 Communication objects

Index	SubIndex	Name	Access type	Data type	PDO-mapping
0x1000	0	Device type	Read only	Unsigned32	No
0x1001	0	Error register	Read only	Unsigned8	No
0x1005	0	COB-ID SYNC object	Read/write	Unsigned32	No
0x1006	0	Communication cycle period	Read/write	Unsigned32	No
0x1007	0	Synchronous window length	Read/write	Unsigned32	No
0x1008	0	Manufacturer device name	Read only	Visible string	No
0x1009	0	Manufacturer hardware version	Read only	Visible string	No
0x100A	0	Manufacturer software version	Read only	Visible string	No
0x100C	0	Guard time	Read/write	Unsigned16	No
0x100D	0	Life time factor	Read/write	Unsigned8	No
0x1010	0	Store parameters Highest sub-index supported	Read only	Unsigned8	No
	1	Save all parameters	Read/write	Unsigned32	No
	2	Save communication parameters	Read/write	Unsigned32	No
	3	Save application parameters	Read/write	Unsigned32	No

Index	SubIndex	Name	Access type	Data type	PDO-mapping
0x1011		Restore default parameters	Read only	Unsigned8	No
	0	Highest sub-index supported			
	1	Restore all default parameters	Read/write	Unsigned32	No
	2	Restore communication default parameters	Read/write	Unsigned32	No
	3	Restore application default parameters	Read/write	Unsigned32	No
0x1014	0	COB-ID emergency object	Read/write	Unsigned32	No
0x1016		Consumer heartbeat time	Read only	Unsigned8	No
	0	Highest sub-index supported			
	1	Consumer heartbeat time 1	Read/write	Unsigned32	No
	2	Consumer heartbeat time 2	Read/write	Unsigned32	No
	3	Consumer heartbeat time 3	Read/write	Unsigned32	No
0x1017	0	Producer heartbeat time	Read/write	Unsigned16	No
0x1018		Identity object	Read only	Unsigned8	No
	0	Highest sub-index supported			
	1	Vendor ID	Read only	Unsigned32	No
	2	Product code	Read only	Unsigned32	No
	3	Revision number	Read only	Unsigned32	No
	4	Serial number	Read only	Unsigned32	No
0x1029	0	Error behavior	Read only	Unsigned8	No
	1	Communication error	Read/write	Unsigned8	No
0x1200	0	Server SDO parameter	Read only	Unsigned8	
	1	COB-ID Rx	Read only	Unsigned32	No
	2	COB-ID Tx	Read only	Unsigned32	No
0x1400		RxPDO1 communication parameter	Read only	Unsigned8	No
	0	Highest sub-index supported			
	1	COB-ID	Read/write	Unsigned32	No
	2	Transmission type	Read/write	Unsigned8	No
0x1401		RxPDO2 communication parameter	Read only	Unsigned8	No
	0	Highest sub-index supported			
	1	COB-ID	Read/write	Unsigned32	No
	2	Transmission type	Read/write	Unsigned8	No
0x1402		RxPDO3 communication parameter	Read only	Unsigned8	No
	0	Highest sub-index supported			
	1	COB-ID	Read/write	Unsigned32	No
	2	Transmission type	Read/write	Unsigned8	No
0x1600		RxPDO1 mapping parameter	Read/write	Unsigned8	No
	0	No. of mapped objects			
	1	1. mapped obj.	Read/write	Unsigned32	No
	2	2. mapped obj.	Read/write	Unsigned32	No
	3	3. mapped obj.	Read/write	Unsigned32	No
	4	4. mapped obj.	Read/write	Unsigned32	No
	5	5. mapped obj.	Read/write	Unsigned32	No
	6	6. mapped obj.	Read/write	Unsigned32	No
	7	7. mapped obj.	Read/write	Unsigned32	No
8	8. mapped obj.	Read/write	Unsigned32	No	

Index	SubIndex	Name	Access type	Data type	PDO-mapping
0x1601		RxPDO2 mapping parameter	Read/write	Unsigned8	No
	0	No. of mapped objects			
	1	1. mapped obj.	Read/write	Unsigned32	No
	2	2. mapped obj.	Read/write	Unsigned32	No
	3	3. mapped obj.	Read/write	Unsigned32	No
	4	4. mapped obj.	Read/write	Unsigned32	No
	5	5. mapped obj.	Read/write	Unsigned32	No
	6	6. mapped obj.	Read/write	Unsigned32	No
	7	7. mapped obj.	Read/write	Unsigned32	No
8	8. mapped obj.	Read/write	Unsigned32	No	
0x1602		RxPDO3 mapping parameter	Read/write	Unsigned8	No
	0	No. of mapped objects			
	1	1. mapped obj.	Read/write	Unsigned32	No
	2	2. mapped obj.	Read/write	Unsigned32	No
	3	3. mapped obj.	Read/write	Unsigned32	No
	4	4. mapped obj.	Read/write	Unsigned32	No
	5	5. mapped obj.	Read/write	Unsigned32	No
	6	6. mapped obj.	Read/write	Unsigned32	No
	7	7. mapped obj.	Read/write	Unsigned32	No
8	8. mapped obj.	Read/write	Unsigned32	No	
0x1800		TxPDO1 communication parameter	Read only	Unsigned8	No
	0	Highest sub-index supported			
	1	COB-ID	Read/write	Unsigned32	No
	2	Transmission type	Read/write	Unsigned8	No
	3	Inhibit time		Unsigned16	No
	4	-	-	-	No
5	Event time	Read/write	Unsigned16	No	
0x1801		TxPDO2 communication parameter	Read only	Unsigned8	No
	0	Highest sub-index supported			
	1	COB-ID	Read/write	Unsigned32	No
	2	Transmission type	Read/write	Unsigned8	No
	3	Inhibit time	Read/write	Unsigned16	No
	4	-	-	-	No
5	Event time	Read/write	Unsigned16	No	
0x1802		TxPDO3 communication parameter	Read only	Unsigned8	No
	0	Highest sub-index supported			
	1	COB-ID	Read/write	Unsigned32	No
	2	Transmission type	Read/write	Unsigned8	No
	3	-inhibit time	Read/write	Unsigned16	No
	4	-	-	-	No
5	Event time	Read/write	Unsigned16	No	
0x1A00		TxPDO1 mapping parameter	Read/write	Unsigned8	No
	0	No. of mapped objects			
	1	1. mapped obj.	Read/write	Unsigned32	No
	2	2. mapped obj.	Read/write	Unsigned32	No
	3	3. mapped obj.	Read/write	Unsigned32	No
	4	4. mapped obj.	Read/write	Unsigned32	No
	5	5. mapped obj.	Read/write	Unsigned32	No
	6	6. mapped obj.	Read/write	Unsigned32	No
	7	7. mapped obj.	Read/write	Unsigned32	No
8	8. mapped obj.	Read/write	Unsigned32	No	

Index	SubIndex	Name	Access type	Data type	PDO-mapping
0x1A01		TxPDO2 mapping parameter	Read/write	Unsigned8	No
	0	No. of mapped objects			
	1	1. mapped obj.	Read/write	Unsigned32	No
	2	2. mapped obj.	Read/write	Unsigned32	No
	3	3. mapped obj.	Read/write	Unsigned32	No
	4	4. mapped obj.	Read/write	Unsigned32	No
	5	5. mapped obj.	Read/write	Unsigned32	No
	6	6. mapped obj.	Read/write	Unsigned32	No
	7	7. mapped obj.	Read/write	Unsigned32	No
8	8. mapped obj.	Read/write	Unsigned32	No	
0x1A02		TxPDO3 mapping parameter	Read/write	Unsigned8	No
	0	No. of mapped objects			
	1	1. mapped obj.	Read/write	Unsigned32	No
	2	2. mapped obj.	Read/write	Unsigned32	No
	3	3. mapped obj.	Read/write	Unsigned32	No
	4	4. mapped obj.	Read/write	Unsigned32	No
	5	5. mapped obj.	Read/write	Unsigned32	No
	6	6. mapped obj.	Read/write	Unsigned32	No
	7	7. mapped obj.	Read/write	Unsigned32	No
8	8. mapped obj.	Read/write	Unsigned32	No	

10.1.2 Manufacturer objects

Index	SubIndex	Name	Access type	Data type	PDO-mapping
0x2nnn	0, 1, ... 9	Manufacturer specific Direct access to inverter parameters Read/write access by SDO transfer only			
0x3000	0	Sync Jitter	Read/write	Unsigned16	No
0x5FF0	0	Active motion block	Read only	Integer8	Tx
0x5FF1	0	Motion block to resume	Read only	Integer8	Tx

10.1.3 Device profile objects

Index	SubIndex	Name	Access type	Data type	PDO-mapping	Factory setting	min	max	Corresponding parameter
0x6007	0	Abort connection option code	Read/write	Integer16	No	1	-2	3	p.388
0x603F	0	Error code	Read only	Unsigned16	No	-	-	-	-
0x6040	0	controlword	Read/write	Unsigned16	Rx	-	-	-	p.410
0x6041	0	statusword	Read/only	Unsigned16	Tx	-	-	-	p.411
0x6042	0	Target velocity	Read/write	Integer16	Rx	0	-32768	32767	-
0x6043	0	Velocity demand	Read only	Integer16	Tx	-	-	-	-
0x6044	0	Control effort	Read only	Integer16	Tx	-	-	-	-
0x6046	0	Velocity min max	Read only	Unsigned8	No	-	-	-	-
	1	Highest sub-index supported	Read/write	Unsigned32	No	0	0	32767	p.418
	2	Velocity min amount	Read/write	Unsigned32	No	32767	0	32767	p.419
0x6048	0	Velocity acceleration	Read only	Unsigned8	No	-	-	-	-
	1	Highest sub-index supported	Read/write	Unsigned32	No	150	1	32767	p.420
	2	Delta time	Read/write	Unsigned16	No	1	1	65535	p.422
0x6049	0	Velocity deceleration	Read only	Unsigned8	No	-	-	-	-
	1	Highest sub-index supported	Read/write	Unsigned32	No	150	1	32767	p.421
	2	Delta time	Read/write	Unsigned16	No	1	1	65535	p.423
0x604A	0	Velocity quick stop	Read only	Unsigned8	No	-	-	-	-
	1	Highest sub-index supported	Read/write	Unsigned32	No	150	1	32767	p.421
	2	Delta time	Read/write	Unsigned16	No	1	1	65535	p.423
0x6060	0	Modes of operation	Write only	Integer8	Rx	2	-1	7	-
0x6061	0	Modes of operation display	Read only	Integer8	Tx	2	-	-	-
0x6064	0	Position actual value	Read only	Integer32	Tx	-	0x8000.0000	0x7FFF.FFFF	-
0x6065	0	Following error window	Read/write	Unsigned32	No	0xFFFF.FFFF	0	0xFFFF.FFFF	p.1105

Index	Sub Index	Name	Access type	Data type	PDO-mapping	Factory setting	min	max	Corresponding parameter
0x6066	0	Following error time out	Read/write	Unsigned16	No	10	0	65535	p.1119
0x6067	0	Position window	Read/write	Unsigned32	No	0xFFFF.FFFF	0	0xFFFF.FFFF	p.1165
0x6068	0	Position window time	Read/write	Unsigned16	No	10	0	65535	p.1166
0x607A	0	Target position	Read/write	Integer32	Rx	0	0x8000.0000	0x7FFF.FFFF	-
0x607C	0	Home offset	Read/write	Integer32	No	0	0x8000.0000	0x7FFF.FFFF	p.1131
0x6081	0	Profile velocity	Read/write	Unsigned32	Rx	0x0005.0000	1	0x7FFF.FFFF	-
0x6083	0	Profile acceleration	Read/write	Unsigned32	Rx	0x0005.0000	1	0x7FFF.FFFF	-
0x6084	0	Profile deceleration	Read/write	Unsigned32	Rx	0x0005.0000	1	0x7FFF.FFFF	-
0x6085	0	Quick stop deceleration	Read/write	Unsigned32	No	0x000A.0000	1	0x7FFF.FFFF	-
0x6086	0	Motion profile type	Read/write	Integer16	No	3	0	3	-
0x6091	0	Gear ratio	Read only	Unsigned8	No	-	-	-	-
	0	Highest sub-index supported							
	1	Motor revolutions	Read/write	Unsigned32	No	1	1	65535	p.1116
	2	(Driving) Shaft revolutions	Read/write	Unsigned32	No	1	1	65535	p.1117
0x6092	0	Feed constant	Read only	Unsigned8	No	-	-	-	-
	0	Highest sub-index supported							
	1	Feed	Read/write	Unsigned32	No	0x0001.0000	1	0x7FFF.FFFF	p.1115
	2	(Driving) Shaft revolutions	Read/write	Unsigned32	No	1	1	1	
0x6098	0	Homing method	Read/write	Integer8	No	0	0	35	p.1130
0x6099	0	Homing speeds	Read only	Unsigned8	No	-	-	-	-
	0	Highest sub-index supported							
	1	Speed during search for switch	Read/write	Unsigned32	No	0x0005.0000	1	0x7FFF.FFFF	p.1132
	2	Speed during search for zero	Read/write	Unsigned32	No	00x0002.0000	1	0x7FFF.FFFF	p.1133
0x609A	0	Homing acceleration	Read/write	Unsigned32	No	0x0005.0000	1	0x7FFF.FFFF	p.1134
0x60C1	0	Interpolation data record	Read only	Unsigned8	No	-	-	-	-
	0	Highest sub-index supported							
	1	Interpolation data record 1	Read/write	Integer32	Rx	0	0x8000.0000	0x7FFF.FFFF	

Attention!

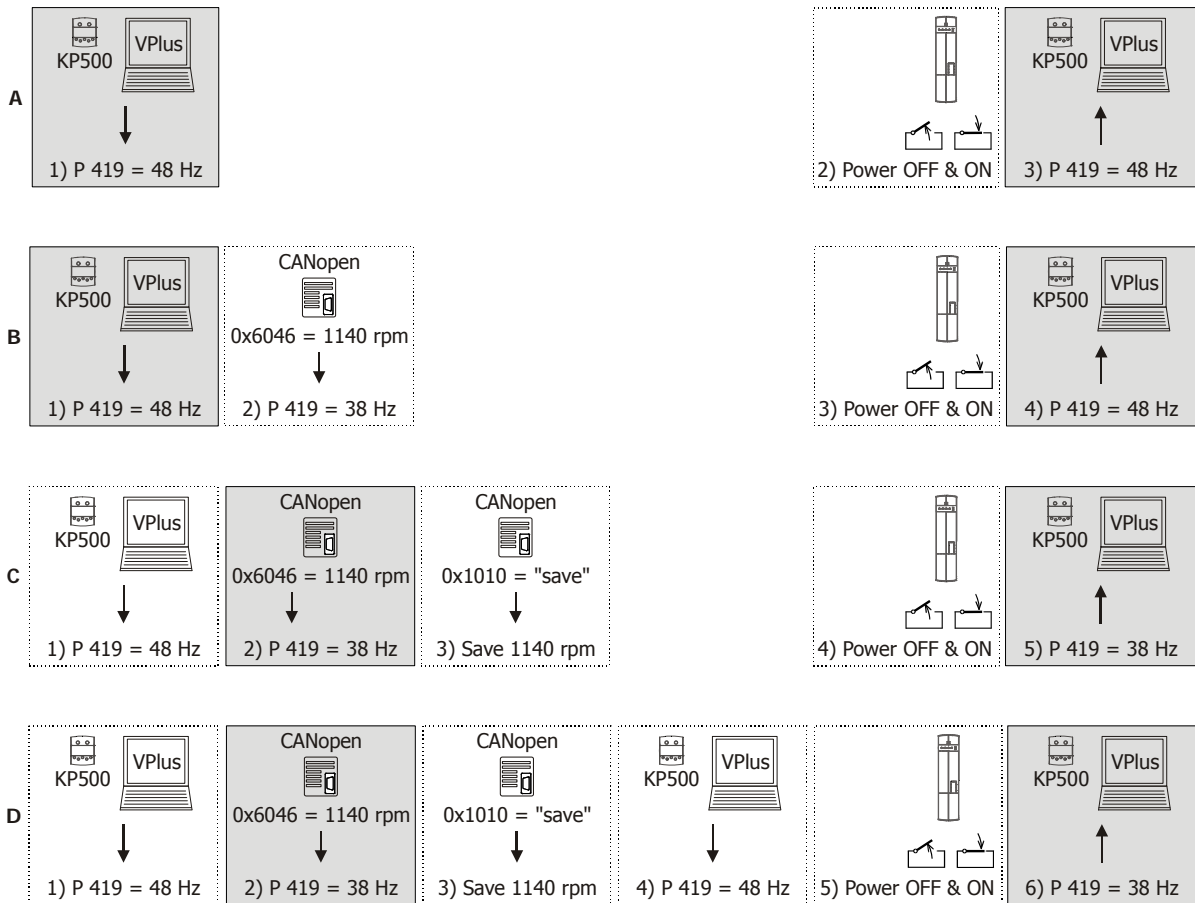
Some of the above listed CANopen objects have corresponding inverter parameters.

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, the next power on cycle overwrites the value set by VPlus with the value stored by the "save" command.

Effect of the "save" command

(sequences of writing parameters and objects, examples)



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).

Attention!

There are inverter parameters calculated from CANopen objects which require the no. of pole pairs for calculating the corresponding value for inverter parameters (e. g. deceleration or acceleration parameters). These calculations always use the no. of pole pairs from data set 1. If the no. of pole pairs is different in the data sets, the result of this operation will not be clear for the user. To avoid confusion it is recommended to write the inverter parameters via the SDO channel using the 0x2nnn (manufacturer) objects and not to use the CANopen objects.

All CANopen objects with corresponding inverter parameters described in this manual have a special information "**Note**".

10.2 Communication Objects (0x1nnn)

10.2.1 0x1000/0 Device Type

The device identification is carried out during the start of the network. The information on the device type and the functionality (type) are prescribed by the CANopen standards.

Object 0x1000/0				
Additional Information			Device Profile Number	
Mode Bits	Type			
31	24	23	16	15
				0

The "Drives and Motion Control" standard device profile used by the frequency inverter is portrayed as device profile number 402. The additional information specifies the device functionality of the frequency inverter.

Device Profile Number	= 402	drives and motion control
Type	= 1	frequency converter
Type	= 2	servo drive
Mode bits	= 0	unused

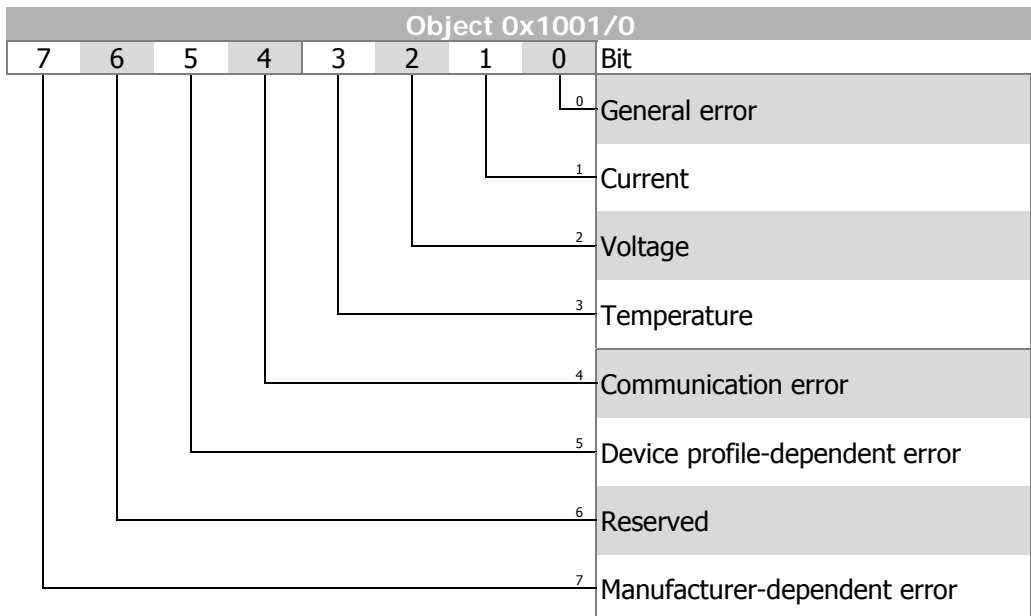
Note:

"Type" depends on the setting of parameter *Configuration 30*.

A motion control configuration (*Configuration 30* = x40) sets type = 2 "servo drive". Other configurations set type = 1 "frequency converter".

10.2.2 0x1001/0 Error Register

Object 0x1001/0 is the error register for internal errors of the frequency inverter. The status error-free (0x1001/0 = 0) or error exists (0x1001/0 ≠ 0) is displayed.



10.2.3 0x1005/0 COB-ID SYNC Message

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

10.2.4 0x1006/0 Communication Cycle Period

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.

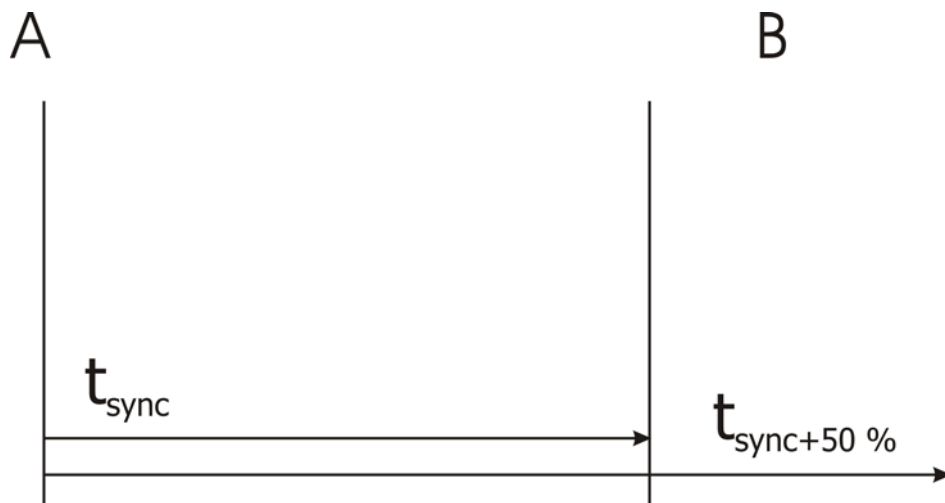
The synchronization of the inverter to an external clock has to be met under the condition, that at least one RxPDO or TxPDO is defined as synchronous object and is activated. The definitions of the TxPDO / RxPDO objects can be changed via objects 0x1400 / 0x1800.

Note:

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.



Note:

The time distance between two consecutive SYNC messages is monitored. If object 0x1006/0 *communication cycle period* is set to a value other than zero, then a communication error event is triggered whenever the time defined by 0x1006/0 is exceeded by more than 50%.

After SYNC telegram "A", SYNC telegram "B" has to be received latest after the set SYNC time + 50 %.

If object 0x1006/0 *communication cycle period* is not set (= zero), then this monitoring function is **not** active.

10.2.5 0x1007/0 Synchronous window length

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 *communication cycle period* is given in multiples of micro seconds.

Note:

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*.

10.2.6 0x1008/0 Manufacturer Device Name

The device name is displayed as a sequence of ASCII characters.

Example : "ACTIVE CUBE"

10.2.7 0x1009/0 Manufacturer Hardware Version

The device version is displayed as a sequence of ASCII characters.

Example : "ACU 400 512 344"

10.2.8 0x100A/0 Manufacturer Software Version

The software version is displayed as a sequence of ASCII characters.

Example : "5.0.4"

10.2.9 0x100C/0 Guard Time

The response monitoring time is calculated by the multiplication of the objects *guard 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*.

10.2.10 0x100D/0 Lifetime Factor

The object "Lifetime Factor" is the multiplier for *guard time*. *Lifetime factor* = 0 deactivates the guarding function.

10.2.11 0x1010/n Store Parameters

With object 0x1010/n parameter/object settings can be stored to non-volatile memory. This object supports 3 sub-indexes with different functions.

Object 0x1010/n				
Index	Sub-index	Meaning	Data type	Value
0x1010	0	Highest sub-index supported	Unsigned8	3
	1	Store all parameters	Unsigned32	See text
	2	Store communication parameters	Unsigned32	See text
	3	Store application parameters	Unsigned32	See text

Writing "save" to 0x1010/3 stores all application parameters (0x6nnn) to non-volatile memory.

Specification of write "save" command

LSB		MSB	
"s"	"a"	"v"	"e"
0x73	0x61	0x76	0x65

Note:

Writing a value other than "save" results in an SDO abort. The store command is **not** processed.

10.2.12 0x1011/n Restore default Parameters

With object 0x1011/n parameters/objects can be set to their default values. This object supports 3 sub-indexes with different functions.

Object 0x1011/n				
Index	Sub-index	Meaning	Data type	Value
0x1011	0	Highest sub-index supported	Unsigned8	3
	1	Restore all parameters	Unsigned32	See text
	2	Restore communication parameters	Unsigned32	See text
	3	Restore application parameters	Unsigned32	See text

Writing "load" to 0x1011/3 restores all application parameters (0x6nnn).

Specification of write "load" command

LSB		MSB	
"l"	"o"	"a"	"d"
0x6C	0x6F	0x61	0x64

Note:

Writing a value other than "load" results in an SDO abort. The restore defaults command is **not** processed.

10.2.13 0x1014/0 COB-ID Emergency Message

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.

10.2.14 0x1016/n Consumer Heartbeat Time

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*.

Object 0x1016/n				
Index	Subindex	Meaning	Data type	Value
0x1016	0	Highest sub-index supported	Unsigned8	3
	1	Consumer Heartbeat Time	Unsigned32	See text
	2	Consumer Heartbeat Time	Unsigned32	See text
	3	Consumer Heartbeat Time	Unsigned32	See text

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

10.2.15 0x1017/0 Producer Heartbeat Time

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.

Object 0x1017/0				
Index	Subindex	Meaning	Data type	Value
0x1017	0	Producer heartbeat time	Unsigned16	ms

10.2.16 0x1018/n Identity Object

The identity object provides information on the device manufacturer as well as the device itself.

Object 0x1018/0				
Index	Subindex	Meaning	Data type	Value
0x1018	0	Highest sub-index supported	Unsigned8	4
	1	Vendor ID	Unsigned32	0x000000D5
	2	Product code	Unsigned32	xxxx xxxx
	3	Revision number	Unsigned32	xxxx xxxx
	4	Serial number	Unsigned32	xxxx xxxx

The vendor ID "0xD5" identifies the manufacturer **Bonfiglioli Vectron GmbH**. This vendor ID has been assigned by the CANopen users' organization "CAN in Automation" (CiA) in Erlangen/Germany (www.can-cia.org).

Product code displays the inverter's type code.

Revision number displays the inverter's CANopen system revision.

Serial number displays the inverter's serial number.

10.2.17 0x1029/n Error Behavior

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).

Object 0x1029				
Index	Subindex	Meaning	Data type	Value
0x1029	0	Highest sub-index supported	Unsigned8	1
	1	Communication error	Unsigned8	xx

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

10.2.18 0x1200/n SDO Server Parameter

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				
Index	Subindex	Meaning	Data type	Value
0x1200	0	Highest sub-index supported	Unsigned8	2
	1	COB-ID client → server (Rx)	Unsigned32	xxxx xxxx
	2	COB-ID server → client (Tx)	Unsigned32	xxxx xxxx

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

10.2.19 0x1400/n, 0x1401/n, 0x1402/n RxPDO Communication Parameters

RxPDO Communication parameters:

0x1400/n RxPDO1

0x1401/n RxPDO2

0x1402/n RxPDO3

These communication parameters define the COB-ID and transmission type used by the RxPDOs. Only sub-index 1 and 2 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).

Receive-PDO No. 1/2/3				
Index	Subindex	Meaning	Data type	Value
0x1400 0x1401 0x1402	0	Highest sub-index supported	Unsigned8	2
	1	COB ID	Unsigned32	RxPDO1: 512 + Node ID RxPDO2: 768 + Node ID RxPDO3: 1024 + Node ID
	2	Transmission type	Unsigned8	0 ... 255

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.

10.2.20 0x1600/n, 0x1601/n, 0x1602 RxPDO Mapping Parameters

RxPDO Mapping parameters:

0x1600/n RxPDO1

0x1601/n RxPDO2

0x1602/n RxPDO3

Mapping Parameter				
Index	Subindex	Meaning	Data type	Value
0x1600 0x1601 0x1602	0	Number of mapped objects	Unsigned8	0 ... 8
	1	1 st mapped obj.	Unsigned32	xxxx xxxx
	2	2 nd mapped obj.	Unsigned32	xxxx xxxx
	3	3 rd mapped obj.	Unsigned32	xxxx xxxx
	4	4 th mapped obj.	Unsigned32	xxxx xxxx
	5	5 th mapped obj.	Unsigned32	xxxx xxxx
	6	6 th mapped obj.	Unsigned32	xxxx xxxx
	7	7 th mapped obj.	Unsigned32	xxxx xxxx
	8	8 th mapped obj.	Unsigned32	xxxx xxxx

0x1600/0 = 0 = no objects mapped

0x1600/0 = 1 ... 8 = 1 ... 8 objects mapped

Mapping entry:

MSB		LSB	
Object index		Sub index	Length (no. of bits)
High byte	Low byte	si	ll

Examples:

Mapping of 0x6040/0 *controlword* (unsigned16) to 1st mapped object in RxPDO1:

0x1600/1 = 0x60400010

Mapping of 0x60C1/1 *interpolation data record 1* (integer32) to 2nd mapped object in RxPDO1:

0x1600/2 = 0x60C10120

Default mapping

RxPDO1	0x1600/0	0x1600/1	0x1600/2	0x1600/3...8
	2	0x6040 controlword	0x6042 target velocity	0x00000000
RxPDO2	0x1601/0			
	0	No mapping		
RxPDO3	0x1602/0			
	0	No mapping		

Mapping Sequence

The mapping sequence requires five steps:

- Step 1: Set PDO to "not valid"
- Step 2: Set sub index 0 to 0 (deactivate current mapping)
- Step 3: Set sub index 1 ... n to the new objects
- Step 4: Set sub index 0 to the number of mapped objects (activate new mapping)
- Step 5: Set PDO valid

Note:

The number of mappable objects depends on the object's length. The maximum number of bytes that can be mapped is 8.

10.2.21 0x1800/n, 0x1801/n, 0x1802/n TxPDO Communication Parameters

TxPDO Communication parameters:

0x1800/n TxPDO1

0x1801/n TxPDO2

0x1802/n TxPDO3

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).

Transmit-PDO No. 1/2/3				
Index	Subindex	Meaning	Data type	Value
0x1800 0x1801 0x1802	0	Highest sub-index supported	Unsigned8	5
	1	COB ID	Unsigned32	TxPDO1: 384 + Node ID TxPDO2: 640 + Node ID TxPDO3: 896 + Node ID
	2	Transmission type	Unsigned8	0 ... 255
	3	Inhibit time	Unsigned16	0 ... 65535
	4	-	-	-
	5	Event time	Unsigned16	0 ... 65535

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 0x1400/0x1401/0x1402 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 inhibit time is set in hundreds of microseconds, e. g. a value of 300 is $300 * 100 \text{ us} = 30 \text{ ms}$.

Note:

The internal time resolution for the inhibit time is in milliseconds. An inhibit time value = 37 is truncated to 30 [3.7 ms → 3 ms].
Values less then 10 are interpreted as 0.

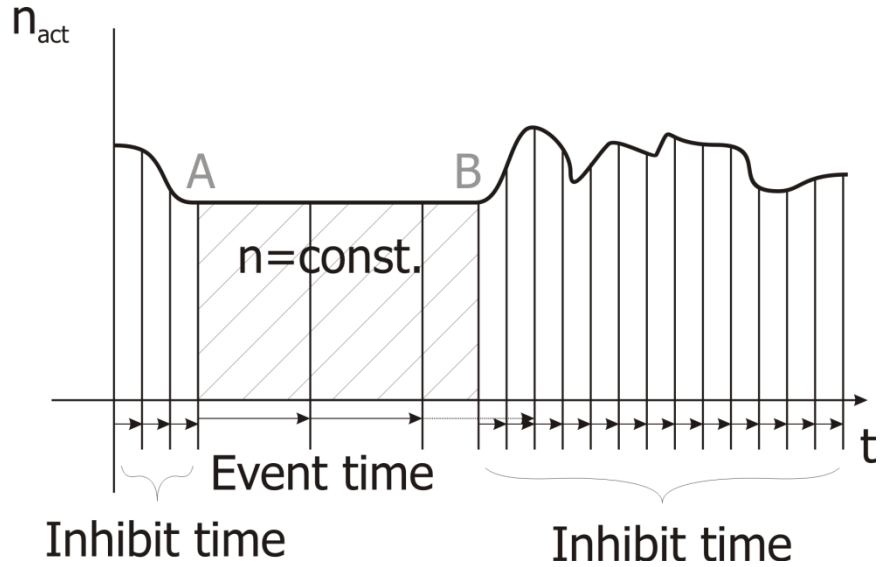
Event time:

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 event time is set in milliseconds, e.g. a value of 2000 = 2000 ms.

Example:

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.

10.2.22 0x1A00/n, 0x1a01/n, 0x1A02/n TxPDO Mapping Parameters

TxPDO Mapping parameters:

0x1A00/n TxPDO1

0x1a01/n TxPDO2

0x1A02/n TxPDO3

Mapping Parameter				
Index	Subindex	Meaning	Data type	Value
0x1A00 0x1A01 0x1A02	0	Number of mapped objects	Unsigned8	0 ... 8
	1	1 st mapped obj.	Unsigned32	xxxx xxxx
	2	2 nd mapped obj.	Unsigned32	xxxx xxxx
	3	3 rd mapped obj.	Unsigned32	xxxx xxxx
	4	4 th mapped obj.	Unsigned32	xxxx xxxx
	5	5 th mapped obj.	Unsigned32	xxxx xxxx
	6	6 th mapped obj.	Unsigned32	xxxx xxxx
	7	7 th mapped obj.	Unsigned32	xxxx xxxx
	8	8 th mapped obj.	Unsigned32	xxxx xxxx

0x1A00/0 = 0 = no object mapped

0x1A00/0 = 1 ... 8 = 1 ... 8 objects mapped

Mapping entry:

MSB		LSB	
Object index		Sub index	Length (no. of bits)
High byte	Low byte	si	ll

Examples:

Mapping of 0x6041/0 *statusword* (unsigned16) to 1st mapping object inTxPDO1:

0x1A00/1 = 0x60410010

Mapping of 0x6064/0 *position actual value* (integer32) to 2nd mapping object in TxPDO1:

0x1A00/2 = 0x60640020

Default mapping

TxPDO1	0x1A00/0	0x1A00/1	0x1A00/2	0x1A00/3...8
	2	0x6041 statusword	0x6044 control effort	0x00000000
TxPDO2	0x1A01/0			
	0	No mapping		
TxPDO3	0x1A02/0			
	0	No mapping		

Mapping Sequence

The mapping sequence requires five steps:

- Step 1: Set PDO to "not valid"
- Step 2: Set sub index 0 to 0 (deactivate current mapping)
- Step 3: Set sub index 1 ... n to the new objects
- Step 4: Set sub index 0 to the number of mapped objects (activate new mapping)
- Step 5: Set PDO valid

Note:

The number of mappable objects depends on the object's length. The maximum number of bytes that can be mapped is 8.

10.3 Manufacturer objects (0x2nnn)

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:

$$\begin{aligned}\text{Index} &= \text{Parameter number} + 0x2000 \\ \text{Sub-index} &= \text{required data set (0, 1 ... 4, 5, 6 ... 9)}\end{aligned}$$

Note:

The mapping of numeric data is always in integer or long data type. Values which contain decimal places are extended accordingly: (e.g. value 17.35 is transmitted as 1735)

10.3.1 Handling of data sets/cyclic writing

Access to the parameter values is carried out on the basis of the parameter number and the required data set. 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 is reported.



Caution! The values are entered automatically into the EEPROM on the controller. If values are to be written cyclically with a high repetition rate, there must be no entry into the EEPROM, as it only has a limited number of admissible writing cycles (about 1 million cycles). If the number of admissible writing cycles is exceeded, the EEPROM is destroyed.

To avoid this, cyclically written data can be entered exclusively into the RAM without a writing cycle taking place on the EEPROM. The data are volatile, i.e., they are lost on power-off and have to be written again after power-on.

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

Writing to a virtual data set in the 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

10.3.2 SDO examples (expedited transfer only)

Writing parameters:

Client → Server SDO Download (expedited)

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
	LSB	MSB	0xnn				
0x2B			uint/int	LSB	MSB	--	--
0x23			long	LSB	MSB

Server → Client Download Response → writing process free of errors

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x60	LSB	MSB	0xnn	-			

Server → Client Abort SDO Transfer → writing process with error

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x80	LSB	MSB	0xnn	Error code			

If an error occurs during the writing process, the corresponding error code is given in Bytes 4 ... 7.

Reading parameters:

Client → Server SDO Upload (expedited)

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x40	LSB	MSB	0xnn	-			

Server → Client Upload Response → reading process without errors

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
	LSB	MSB	0xnn	LSB			MSB
0x4B			uint/int	LSB	MSB	--	--
0x43			long	LSB	MSB

Server → Client Abort SDO Transfer → reading process faulty

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x80	LSB	MSB	0xnn	Error code			

If an error occurs during the reading process, the corresponding error code is given in Bytes 4 ... 7.

Error code table:

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

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	abort code low		abort code high	
0x80	LSB	MSB	0xnn	LSB	MSB	LSB	MSB

Error codes			
Abort code high	Abort code low	Description to 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 (Lesen/ schreiben / checksum)
0x0607	0x0010	Data type does not match	- Parameter has a different data type
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
0x0800	0x0021	Data cannot be transferred because of local control	- parameter cannot be written in operation

Examples of writing parameters:

Writing parameter *Rated Speed* **372** (type: uint) in data set 2 with the parameter value 2980.

Index = 372 + 0x2000 = 0x2174, value = 2980 = 0x0BA4

Client → Server SDO Download (expedited)

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x2B	0x74	0x21	0x02	0xA4	0x0B	--	--

Writing parameter *warning limit Tk* **407** (type: int) in data set 0 with the parameter value -15.

Index = 407 + 0x2000 = 0x2197, value = -15 = 0xFFFF1

Client → Server SDO Download (expedited)

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x2B	0x97	0x21	0x00	0xF1	0xFF	--	--

Writing parameter *Fixed frequency I* **480** (type: long) in data set 1 with the parameter value 100.00 Hz.

Index = 480 + 0x2000 = 0x2174, value = 10000 = 0x00002710

Client → Server SDO Download (expedited)

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x23	0x74	0x21	0x01	0x10	0x27	0x00	0x00

Writing parameter *Fixed Frequency I* **480** (type: long) in data set 3 with the parameter value -50.00 Hz.

Index = 480 + 0x2000 = 0x2174, value = -5000 = 0xFFFFEC78

Client → Server SDO Download (expedited)

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x23	0x74	0x21	0x03	0x78	0xEC	0xFF	0xFF

Examples of reading parameters:

Reading parameter *Rated speed* **372** (type: uint) in data set 2 with the current parameter value 1460.

Index = 372 + 0x2000 = 0x2174, value = 1460 = 0x05B4

Client → Server SDO Upload (expedited)

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x40	0x74	0x21	0x02	--	--	--	--

Server → Client Upload Response

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x4B	0x74	0x21	0x02	0xB4	0x05	--	--

Reading parameter *Warning Limit Tk* **407** (type: int) in data set 0 with the current parameter value -5.

Index = 407 + 0x2000 = 0x2197, value = -5 = 0xFFFB

Client → Server SDO Upload (expedited)

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x40	0x97	0x21	0x00	--	--	--	--

Server → Client Upload Response

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x4B	0x97	0x21	0x00	0xFB	0xFF	--	--

Reading parameter *Fixed Frequency 1* **480** (type: long) in data set 1 with the current parameter value 75.00 Hz.

Index = 480 + 0x2000 = 0x2174, value = 7500 = 0x00001D4C

Client → Server SDO Upload (expedited)

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x40	0x74	0x21	0x01	--	--	--	--

Server → Client Upload Response

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x43	0x74	0x21	0x01	0x4C	0x1D	0x00	0x00

Reading parameter *Fixed Frequency 1* **480** (type: long) in data set 3 with the current parameter value -10.00 Hz.

Index = 480 + 0x2000 = 0x2174, value = -1000 = 0xFFFFFC18

Client → Server SDO Upload (expedited)

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x40	0x74	0x21	0x03	--	--	--	--

Server → Client Upload Response

0	1	2	3	4	5	6	7
Control byte	Index		Subindex	Data			
0x43	0x74	0x21	0x03	0x18	0xFC	0xFF	0xFF

10.4 Manufacturer objects (0x3000 ... 0x5FFF)

In addition to the device profile objects the following manufacturer specific objects are implemented.

10.4.1 0x3000/0 SYNC Jitter

DS301 does not include an object for monitoring the jittering of the SYNC message. ACTIVE CUBE 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:

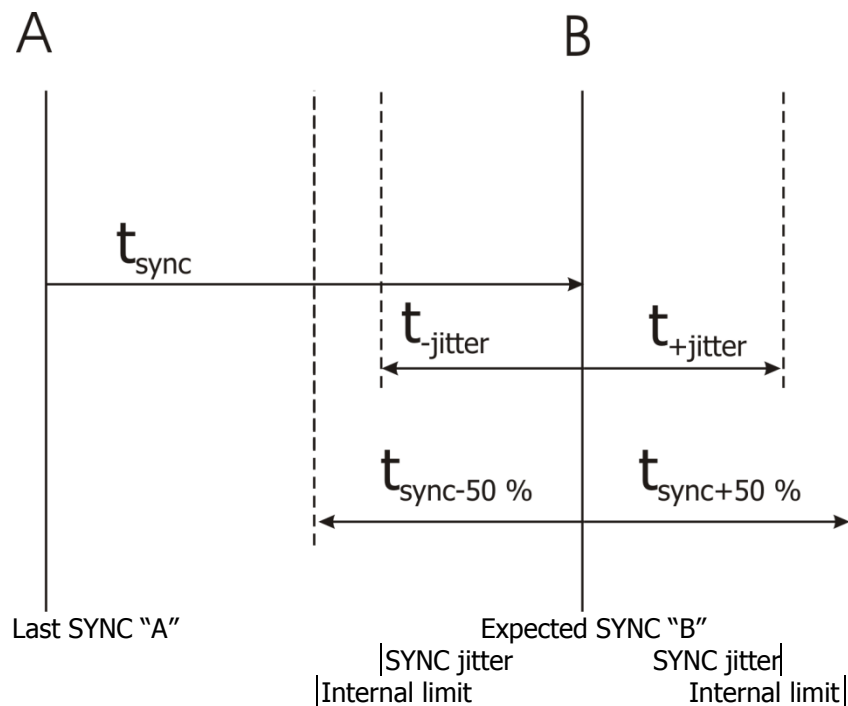
$$0x1006/0 \text{ communication cycle period } +/- 0x3000/0 \text{ SYNC Jitter}$$

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 (μ s) 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 $\neq 0$ 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).



Object 0x3000/0				
Index	Subindex	Meaning	Data type	Value
0x3000	0	SYNC Jitter	Unsigned16	xxxx xxxx

Note:

Object 0x3000 *SYNC Jitter* is located in the application object area and is saved by objects 0x1010/3 *save application objects* and 0x1010/1 *save all objects*.

10.4.2 0x3001/0 Digital In actual value

Object 0x3001 *Digital In actual value* displays the current state of the digital inputs and of the multifunctional input 1 (if in *Operation mode 452*-digital input) as described in parameter *Digital Inputs 250*.

10.4.3 0x3002/0 Digital Out actual value

Object 0x3002 *Digital Out actual value* displays the current state of the up to 4 - depending on optional hardware - digital outputs and of the multifunctional output 1 (if in *Operation mode 550* – digital) as described in parameter *Digital Outputs 254*.

10.4.4 0x3003/0 Digital Out set values

Via object 0x3003 there are 5 digital sources available for use with parameters, which require digital values.

Object 0x3003			
Bit	Source no.	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
5	-	-	-
6	-	-	-

Digital outputs use these sources as operation mode 90 ... 94 *Obj 0x3003 DigOut 1 ... 5* respectively inverted as 190 ... 194 *inv. Obj 0x3003 DigOut 1 ... 5* (see e.g. parameter *Op. Mode Digital Output 1 530*). The mapping of this object bits to the outputs is arbitrary.

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 ...

The sources can be chosen directly by the selection of 810...814 *Obj 0x3003 DigOut 1 ... 5* from the parameters choice list. This can be used e.g. for direct setting of Boolean inputs.

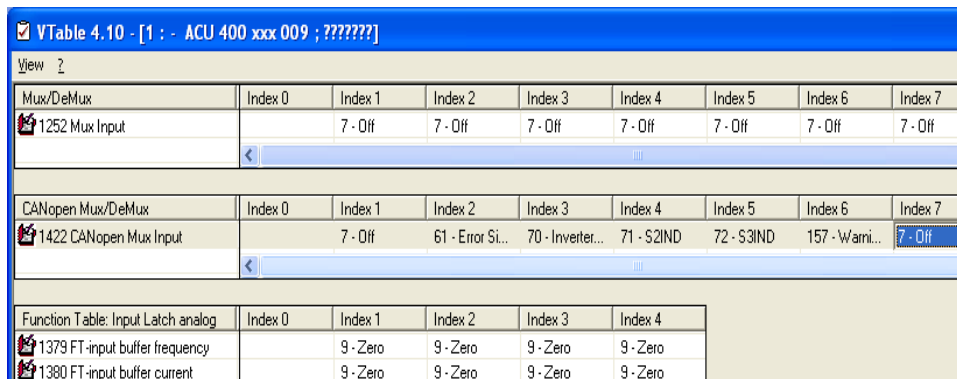
10.4.5 0x3004/0 Boolean Mux

Via object 0x3004 up to 16 boolean values can be read in a compressed manner. Each bit in 16 bit object 0x3004 displays the actual value of the connected boolean source.

Note:

Bit number 0 ... 15 corresponds with index number 1 ... 16!

The sources for the 16 bits can be chosen from a choice list via the index parameter *CANopen Mux Input. 1422*. Parameters **1420** and **1421** are the associated write and read parameters which you have to set prior to writing/reading parameter **1422**. By using VTable this process is easier and more clearly laid out.



VTable 4.10 - [1 : - ACU 400 xxx 009 ; ???????]

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
CANopen Mux/DeMux								
CANopen Mux/DeMux	Index 0	Index 1	Index 2	Index 3	Index 4	Index 5	Index 6	Index 7
1422 CANopen Mux Input		7 - Off	61 - Error Si...	70 - Inverter...	71 - S2IND	72 - S3IND	157 - Wami...	7 - Off
Function Table: Input Latch analog								
Function Table: Input Latch analog	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

10.4.6 0x3005/0 Boolean DeMux

Via object 0x3005 up to 16 boolean values can be written in a compressed manner. These values are available as sources which can be chosen by the selection of 832 ... 847 *Obj 0x3005 Demux Out 1...16* from a parameters choice list.

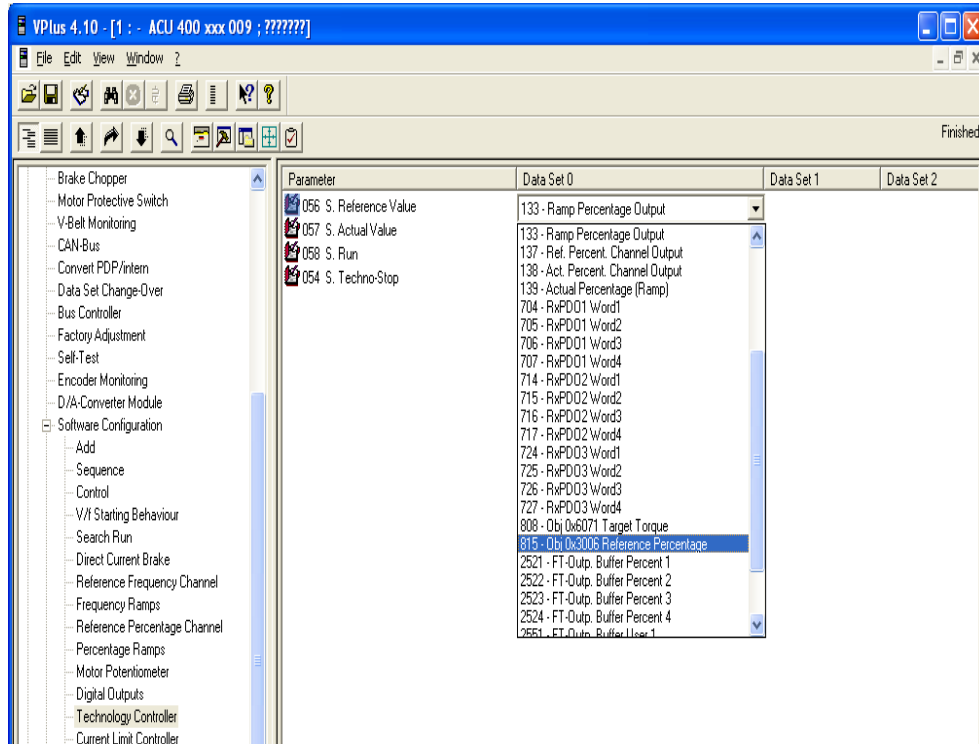
Obj. 0x3005		
Bit no.	Source No.	Source name
0	832	Obj. 0x3005 Demux Out 1
1	833	Obj. 0x3005 Demux Out 2
2	834	Obj. 0x3005 Demux Out 3
3	835	Obj. 0x3005 Demux Out 4
4	836	Obj. 0x3005 Demux Out 5
5	837	Obj. 0x3005 Demux Out 6
6	838	Obj. 0x3005 Demux Out 7
7	839	Obj. 0x3005 Demux Out 8
8	840	Obj. 0x3005 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

10.4.7 0x3006/0 Percentage set value

Via object 0x3006 it is possible to write to a percentage source like parameter *S. Reference Value 056*.

The value of object 0x3006 is available as source which can be chosen by the selection of "815 - Obj 0x3006 Reference Percentage" from a parameters choice list.

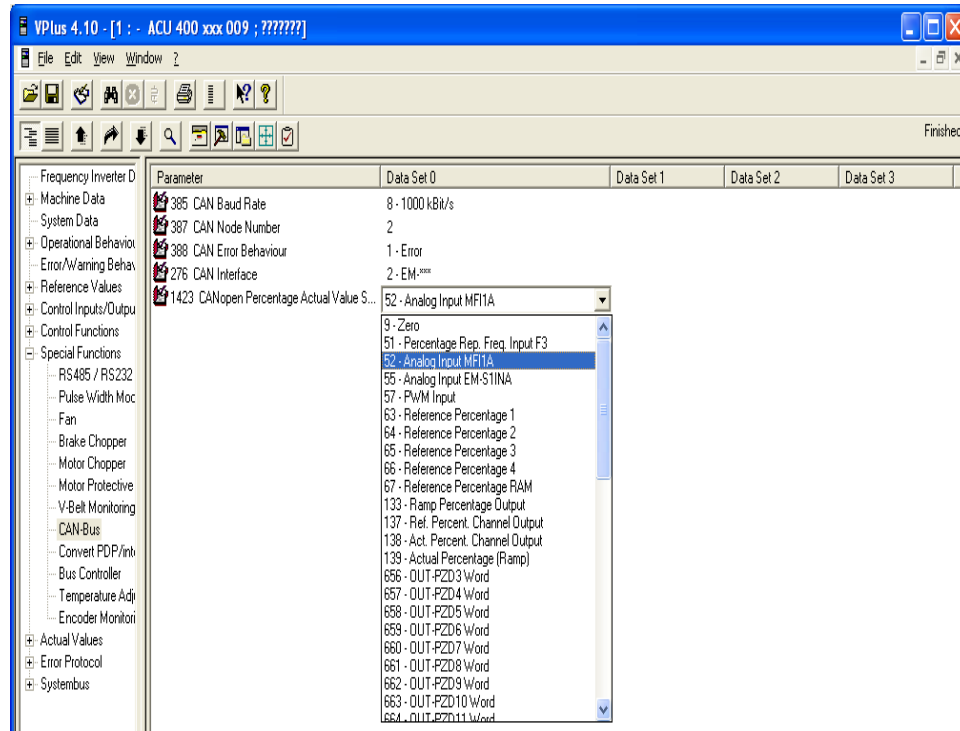
Example: Technology controller parameter *S. Reference Value 056*.



The percentage value is scaled as percent * 100 (e.g. 5678 represents 56.78%).

10.4.8 0x3007/0 Percentage actual value

Object 0x3007 displays the value of a percentage source which is selectable via parameter *CANopen Percentage Actual Value Source* 1423. Default source is 52 – Analog Input MF11A.



The percentage value is scaled as percent * 100 (e.g. 5678 represents 56.78%).

10.4.9 0x5FF0/0 Active motion block

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.

10.5 Device Profile Objects (0x6nnn)

10.5.1 0x6007/0 Abort Connection option code

Object *abort connection option code* specifies the operational behavior of the frequency inverter if the bus connection fails due to BusOff, guarding, heartbeat, SYNC error, RxPDO length error or NMT state change (leaving NMT state operational).

Object 0x6007/0	
Operation mode	Function
0 - No reaction	Operating point is maintained
1 - Error	Device state machine changes to state "fault" immediately (factory setting)
2 - Switch-off	Device state machine processes command ' <i>disable voltage</i> ' and changes to state "switch on disabled"
3 - Quick stop	Device state machine processes command ' <i>quick stop</i> ' and changes to state "switch on disabled"
-1 - Ramp stop + Error (minus 1)	Device state machine processes command ' <i>disable operation</i> ' and changes to state "fault" after the drive is stopped
-2 - Quick stop + Error (minus 2)	Device state machine processes command ' <i>quick stop</i> ' and changes to state "fault" after the drive is stopped

The object *abort connection option code* corresponds to the inverter parameter *CAN Error behavior 388*.

p.388	0x6007
0	0
1	1
2	2
3	3
4	-1
5	-2

Note:

Writing P.388 or writing object 0x6007 has the same effect.

If object 0x6007 was written and then a save parameters command (object 0x1010) processed, the value of 0x6007 is stored in non volatile memory. After the next power on of the inverter the previously set value for 0x6007 is reactivated and overwrites the setting of P.388.

Note:

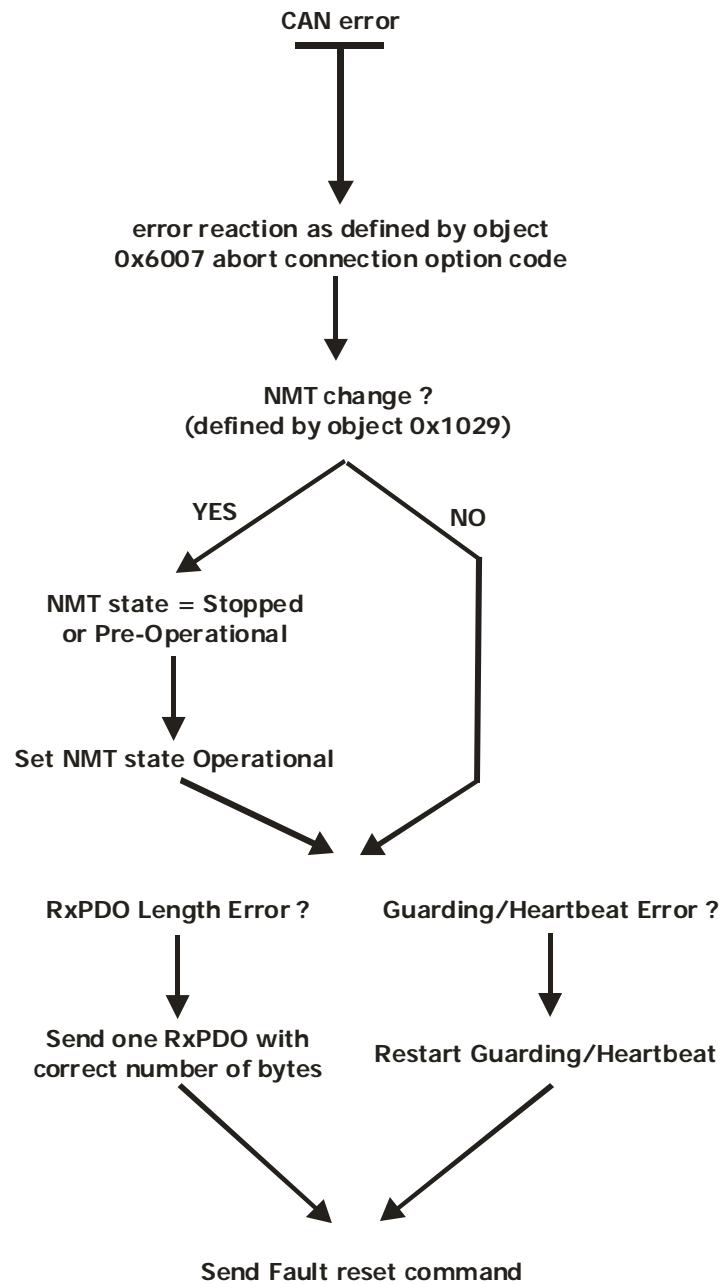
The behavior described above is ONLY relevant if parameter *Local/Remote* **412** is set to **1** "Control via Statemachine".

Otherwise the functional behavior is different. If *abort connection option code* is set to 0 "Operating point is maintained", there is no reaction. For EVERY other setting the inverter reacts with an immediate change to state "fault" if a CAN error occurs.

Attention! The behavior for fault reset corresponds to object 0x1029 *Error Behavior*. Depending on the setting of object *Error behavior*, the NMT state may change (leaving NMT state operational). In this case the NMT state must be set to operational again before a fault reset command is accepted by the inverter.

In addition, the error reason must also be reset. E.g. for a guarding error, guarding must be started again before a fault reset command is accepted by the inverter. In the case of a RxPDO length error, a RxPDO frame with the correct number of bytes must be received before new RxPDO data is accepted.

Typical recovery sequence after CAN error:



Warning! With the setting of object 0x6007 *abort connection option code* = 0 (no reaction), the inverter does not react to any CAN error and remains in its current state (e. g. drive running).

It is strongly recommended to use a setting for object 0x6007 *abort connection option code* that forces the inverter into the Fault state (setting = 1, -1 or -2).

10.5.2 0x603F/0 Error code

The object *error code* is used to store the last error that occurred.

In DS402, a large number of possible error codes are specified. The following list shows the relationship between the error code displayed internally by the frequency inverter and on the KP500 control unit, and the error secured in object *error code*.

Error reports				
Error	xx	Error code		Meaning
F00	xx	00	00	No error has occurred
Overload				
F01	xx	23	10	Frequency inverter has been overloaded
Case				
F02	xx	42	10	Case temperature outside the temperature limits
Inside				
F03	xx	41	10	Inside temperature outside the 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 the 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 reports

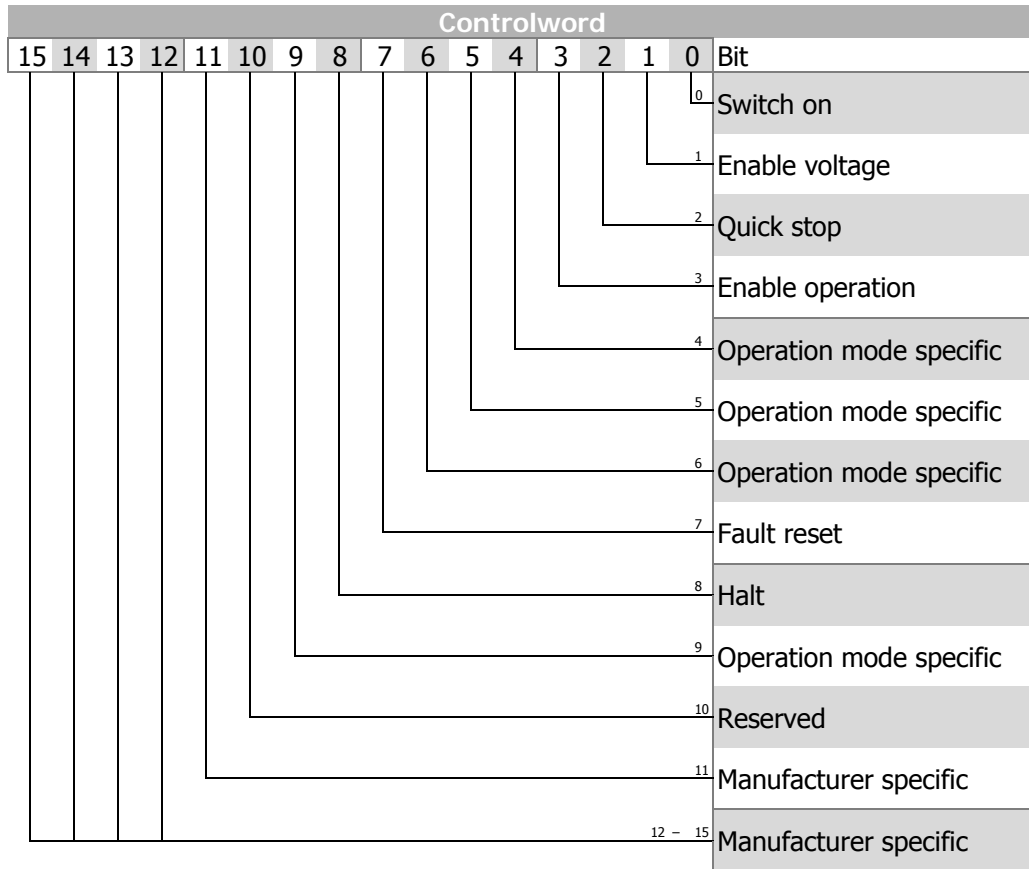
If the value 1000 (= generic error) appears as the error code, then the inverter error code can be read via parameter *Current error* **260** (unsigned16). This parameter contains the error code in the product-internal format.

The assignment table of the error code can be taken from the operating instructions.

In the emergency message, the error code of the frequency inverter is transmitted in bytes 4 ... 7.

10.5.3 0x6040/0 Controlword

Object 0x6040/0 *controlword* is relevant to the inverter remote state machine whenever parameter *LocalRemote* 412 is set to 1 (remote state machine).

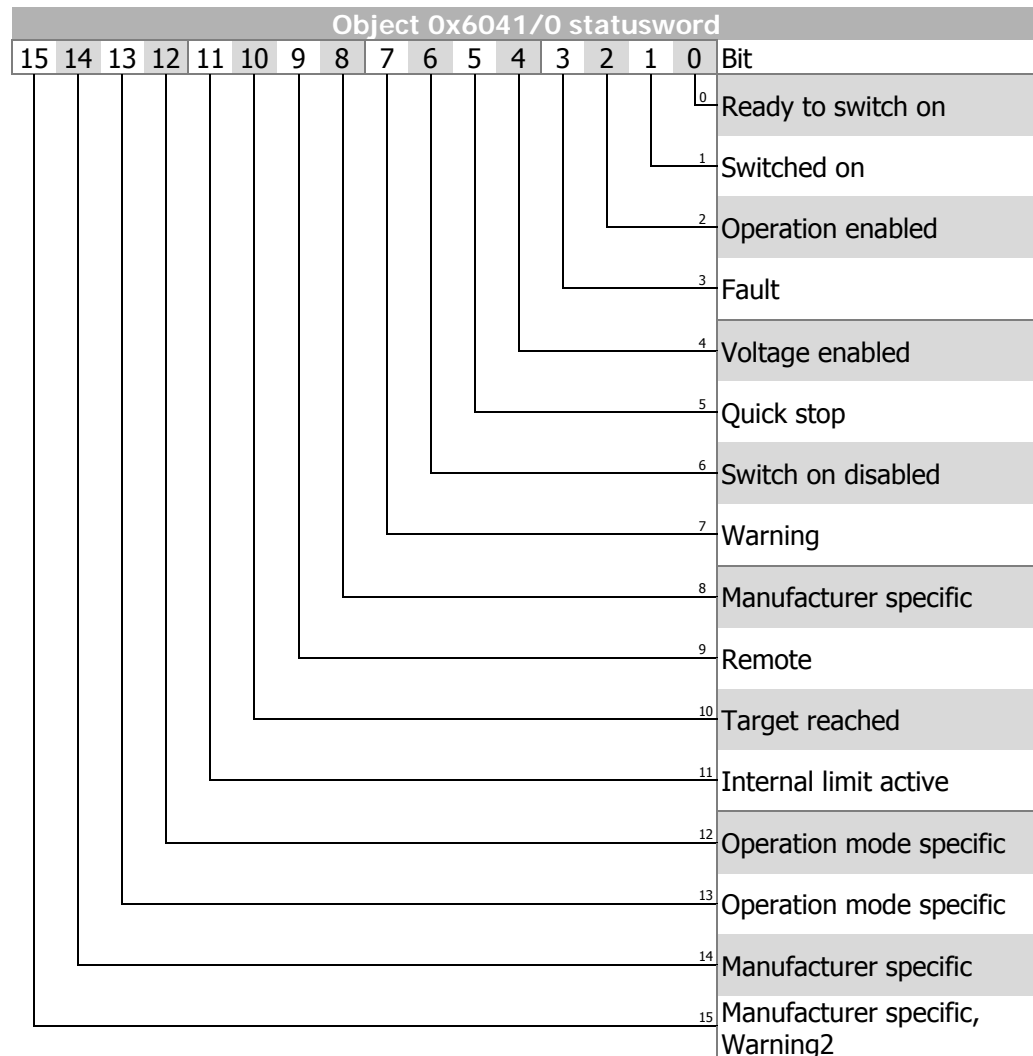


Bits 9 ... 15 unused

Bits 4, 5, 6 *operation mode specific* and bit 8 *halt* are used in motion control configurations (p.30 = x40) only.
See chapter 11 "Inverter Control".

10.5.4 0x6041/0 Statusword

Object 0x6041/0 *statusword* displays the actual state of the inverter.



Bits 8 and 14 unused

Bits 12 and 13 *operation mode specific* are used in motion control configurations (p.30 = x40) only.

See chapter 11 "Inverter Control".

10.5.5 0x6042/0 Target velocity

Object *target velocity* is the speed reference value for the frequency inverter in velocity mode. *Target velocity* is interpreted as a speed with the unit RPM. The inverter's internal reference frequency is calculated from the target velocity in RPM taking into account parameter *No. of Pole Pairs 373* (value of P.373 in the active data set).

Parameter		Setting	
No.	Object	Min.	Max.
0x6042	Target velocity	-32768	32767

The 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.

Note:

The parameter *No. of Pole Pairs 373* has four different data sets. In motion control applications only the data set 1 is used.

Non motion control applications sometimes have more than one motor connected to the inverter (only one at a time, switched over by contactor). These motors may have a different no. of pole pairs. The entry in *No. of Pole Pairs 373* is then different in the four data sets. After change-over of the motor, the object *target velocity* must be written at least once in order to recalculate the internal reference frequency of the inverter using the correct no. of pole pairs.

10.5.6 0x6043/0 Target velocity demand

Object *target velocity demand* is the output value of the ramp function in RPM. The object has the same notation as the object *target velocity* and can be read as an actual value. For calculating *target velocity demand* the parameter *No. of Pole Pairs 373* (value of P.373 in active data set) is taken into account in the same way as described for object target velocity.

10.5.7 0x6044/0 Control effort

Object *control effort* is the actual speed of the drive in RPM. The object has the same notation as the object *target velocity* and can be read as an actual value. For calculating *control effort* the parameter *No. of Pole Pairs 373* (value of P.373 in the active data set) is taken into account in the same way as described for object target velocity.

10.5.8 0x6046/n Velocity min max amount

Object velocity min max amount comprises the sub-index 1 = velocity min amount and sub-index 2 = velocity max amount.

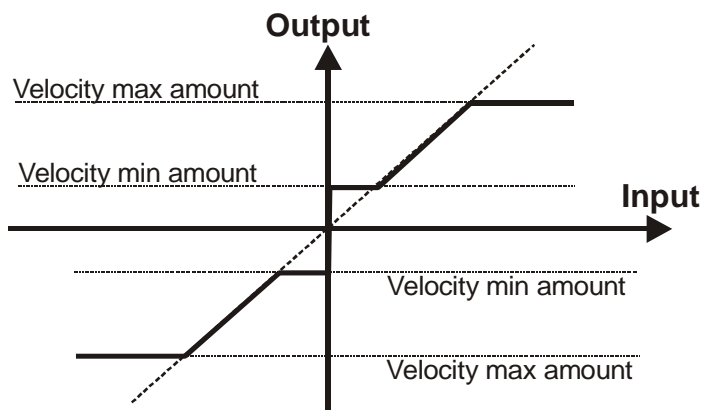
Velocity min max amount			
Index	Subindex	Meaning	Value
0x6046	0	Highest sub-index supported (unsigned 8)	2
	1	Velocity min amount (RPM)	(unsigned 32)
	2	Velocity max amount (RPM)	(unsigned 32)

The unit of *velocity min amount* is in RPM (positive values only). Writing to object *velocity min amount* automatically generates a write command to parameter *Minimum Frequency 418* (data set 5, all data sets in RAM only !). The value of *velocity min amount* is converted internally to a frequency value, taking into account the parameter *No. of Pole Pairs 373* (in data set 1 !).

The unit of *velocity max amount* is in RPM (positive values only). Writing to object *velocity max amount* automatically generates a write command to parameter *Maximum Frequency 419* (data set 5, all data sets in RAM only !). The value of *velocity max amount* is converted internally to a frequency value, taking into account parameter *No. of Pole Pairs 373* (in data set 1 !).

If the input reference value of object *target velocity* is less than the object value *velocity min amount* or greater than *velocity max amount*, then *target velocity* is limited accordingly.

No.	Object	Min.	Max.
0x6046/1	Velocity min amount (RPM)	0	32767
0x6046/2	Velocity max amount (RPM)	0	32767



Note:

If objects 0x6046/1 or 0x6046/2 were written and then a save parameters command (object 0x1010) processed, the object values are stored in non volatile memory. After the next power on of the inverter, the previously set values are reactivated and overwrite the settings of parameters 418/419.

10.5.9 0x6048/n Velocity acceleration

The acceleration in velocity mode is set with object *velocity acceleration*. The object *velocity acceleration* consists of *delta speed* in RPM and *delta time* in seconds.

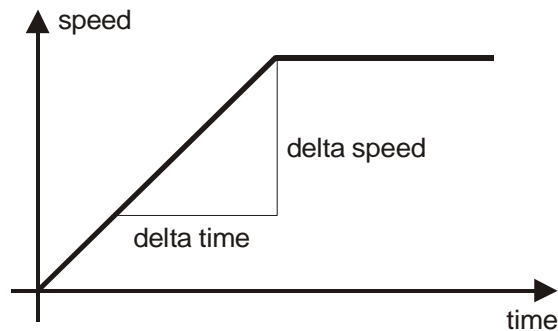
Velocity acceleration			
Index	Subindex	Meaning	Value
0x6048	0	Highest sub-index supported (unsigned 8)	2
	1	Delta speed (RPM)	(unsigned 32)
	2	Delta time (sec)	(unsigned 16)

The gradient of the frequency in the acceleration period is written to parameters *Acceleration (clockwise)* **420** and *Acceleration (anti-clockwise)* **422** (data set 5, all data sets in RAM only !). Both parameters are set to the same value.

The values of p.420 and p.422 are converted internally to a frequency/sec value, taking into account parameter *No. of Pole Pairs* **373** (in data set 1 !).

The gradient is changed internally by altering the objects delta time or delta speed.

No.	Object	Min.	Max.
0x6048/1	Delta speed (RPM)	1	32767
0x6048/2	Delta time (sec)	1	65535



Note:

If objects 0x6048/1 or 0x6048/2 were written and then a save parameters command (object 0x1010) processed, the object values are stored in non volatile memory. After the next power on of the inverter, the previously set values are reactivated and overwrite the settings of p.420/422.

10.5.10 0x6049/n Velocity deceleration

The deceleration in velocity mode is set with object *velocity deceleration*. The object *velocity deceleration* consists of *delta speed* in rpm and *delta time* in seconds.

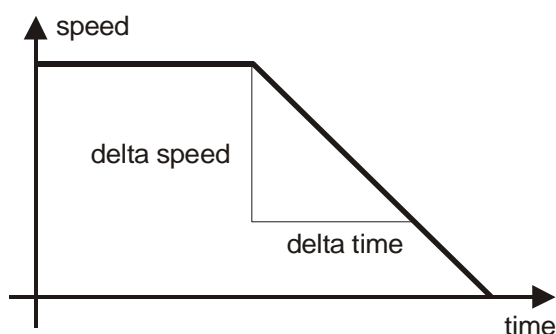
Velocity acceleration			
Index	Subindex	Meaning	Value
0x6049	0	Highest sub-index supported (unsigned 8)	2
	1	Delta speed (RPM)	(unsigned 32)
	2	Delta time (sec)	(unsigned 16)

The gradient of the frequency in the deceleration period is written to parameters *Deceleration (clockwise)* **421** and *Deceleration (anti-clockwise)* **423** (data set 5, all data sets in RAM only !). Both parameters are set to the same value.

The values of p.421 and p.423 are converted internally to a frequency/sec value, taking into account the parameter *No. of Pole Pairs* **373** (in data set 1 !).

The gradient is changed internally by altering the objects delta time or delta speed.

Parameter		Setting	
No.	Object	Min.	Max.
0x6049/1	Delta speed (RPM)	1	32767
0x6049/2	Delta time (sec)	1	65535



Note:

If objects 0x6049/1 or 0x6049/2 were written and then a save parameters command (object 0x1010) processed, the object values are stored in non volatile memory. After the next power on of the inverter, the previously set values are reactivated and overwrite the settings of p.421/423.

10.5.11 0x604A/n Velocity quick stop

The quick stop deceleration in velocity mode is set with object *velocity quick stop*. Object *velocity quick stop* consists of *delta speed* in RPM and *delta time* in seconds.

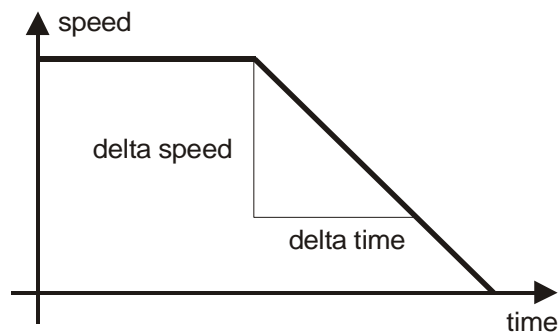
Velocity quick stop			
Index	Subindex	Meaning	Value
0x604A	0	Highest sub-index supported (unsigned 8)	2
	1	Delta speed (RPM)	(unsigned 32)
	2	Delta time (sec)	(unsigned 16)

The gradient of the frequency in the deceleration period is written to parameters *Emergency Stop (clockwise) 424* and *Emergency Stop (anti-clockwise) 425* (data set 5, all data sets in RAM only !). Both parameters are set to the same value.

The values of p.424 and p.425 are converted internally to a frequency/sec value, taking into account the parameter *No. of Pole Pairs 373* (in data set 1 !).

The gradient is changed internally by altering the objects delta time or delta speed.

Parameter		Setting	
No.	Object	Min.	Max.
0x604A/1	Delta speed (RPM)	1	32767
0x604A/2	Delta time (sec)	1	65535



Note:

If objects 0x604A/1 or 0x604A/2 were written and then a save parameters command (object 0x1010) processed, the object values are stored in non volatile memory. After the next power on of the inverter, the previously set values are reactivated and overwrite the settings of p.424/425.

10.5.12 0x6060/0 Modes of operation

With object *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):

<i>Modes of operation</i>
1 – position profile mode
2 – velocity mode
6 – homing mode
7 – interpolated position mode
-1 – table travel record (manufacturer specific mode)

Available value for *modes of operation* with inverter in non motion control configuration (p.30 ≠ x40):

<i>Modes of operation</i>
2 – velocity mode

The inverter in non motion control configuration ignores all settings other than "2".

For further information see chapter 11 "Inverter Control".

10.5.13 0x6061/0 Modes of operation display

Object *modes of operation display* acknowledges the previously set value of *modes of operation* by displaying the same value as *modes of operation*.

Note:

After setting *modes of operation*, the PLC must wait for this acknowledgement before sending any other command to the inverter.

For further information see chapter 11 "Inverter Control".

10.5.14 0x6064/0 Position actual value

Object 0x6064 *position actual value* represents the actual value of the position measurement device in user units.

10.5.15 0x6065/0 following error window

Note:

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 value range of object 0x6065/0 *following error window* is 0 ... $(2^{31}-1)$. Writing a value of 2^{31} ... $(2^{32}-2)$ results in an SDO abort (value range).

If the value of the *following error window* is set to $2^{32}-1$ OR 0, the *following error window* is switched off.

Note:

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 !).

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 p.1105.

10.5.16 0x6066/0 following error time out

When a following error (contouring error) occurs longer than the defined value of object 0x6066 *following error time out* given in milliseconds, the corresponding bit in the statusword (bit 13 *following error*) is set to one.

Note:

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 !).

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 p.1119.

10.5.17 0x6067/0 position window

Object 0x6066 *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.

The value range of object 0x6067/0 *position window* is 0 ... $(2^{31}-1)$. Writing a value of 2^{31} ... $(2^{32}-2)$ results in an SDO abort (value range).

If the value of *position window* is set to $2^{32}-1$ OR 0, the position window control is switched off.

Note:

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 p.1165.

10.5.18 0x6068/0 position window time

When the actual position is within the *position window* during the defined *position window time* (given in milliseconds), then the corresponding bit in the statusword (bit 10 *target reached*) is set to one.

Note:

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 p.1166.

10.5.19 0x6071/0 Target Torque

The value transmitted via Object 0x6071 is selectable as source Q.808 for various parameters (e.g. *FT Input buffer percentage* **1381**).

It is also available as operation mode 95 or inverted as 195 (e.g. for parameter *Reference Percentage Source* **476** in configurations with torque control p.**30 = x30**).

A value of 1000 corresponds to rated motor torque.

10.5.20 0x6077/0 Torque actual value

Object 0x6077 *Torque actual value* displays the torque actual value.

A value of 1000 corresponds to rated motor torque.

10.5.21 0x6078/0 Current actual value

Object 0x6078 *Current actual value* displays the current actual value.

A value of 1000 corresponds to the rated motor current.

10.5.22 0x6079/0 DClink circuit voltage

Object 0x6079 *DC link circuit voltage* displays the voltage actual value of the voltage intermediate circuit in mV (see parameter *DC-Link Voltage 222*).

A value of 100.000 corresponds to 100.0 V.

10.5.23 0x607A/0 Target position

Object 0x607A *target position* defines the position (in user units) that the drive should move to in profile position mode.

10.5.24 0x607C/0 Home offset

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.

Note:

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 1131.

10.5.25 0x6081/0 Profile velocity

Object 0x6081 *profile velocity* is the velocity (in user units per second) at the end of the acceleration ramp in profile position mode.

10.5.26 0x6083/0 Profile acceleration

Object 0x6083 *profile acceleration* is the acceleration (in user units per second²) in profile position mode.

10.5.27 0x6084/0 Profile deceleration

Object 0x6084 *profile deceleration* is the deceleration (in user units per second²) in profile position mode.

10.5.28 0x6085/0 Quick stop deceleration

Object 0x6085 *quick stop deceleration* is the deceleration (in user units per second²) in profile position mode for quick stop mode (controlword bit 2 = 0).

Note:

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 1179.

10.5.29 0x6086/0 Motion profile type

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

10.5.30 0x6091/n Gear ratio

Object 0x6091 *gear ratio* defines the ratio of motor shaft revolutions to driving shaft revolutions.

$$\text{Gear ratio} = \frac{0x6091/1 \text{ motor shaft revolutions}}{0x6091/2 \text{ driving shaft revolutions}}$$

Note:

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 p.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 1116.

10.5.31 0x6092/n Feed constant

Object 0x6092 *feed constant* defines the feed (in user units) per driving shaft revolutions.

$$\text{Feed constant} = \frac{0x6092/1 \text{ feed}}{0x6092/2 \text{ driving shaft revolutions}}$$

Note:

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 1115.

10.5.32 0x6098/0 Homing method

Object 0x6098/0 *homing method* determines the method that will be used during homing.

For an exact description of the different homing modes see the application manual "Positioning".

Note:

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 1130.

<i>Homing Method 0x6098/0</i>		<i>Function</i>
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).
8 -	Pos. Lim.-Sw., Ref.-Sig. right of left Edge of Home-Sw.	Reversal of direction of rotation when positive HW limit switch is reached.
9 -	Pos. Lim.-Sw., Ref.-Sig. left of right Edge of Home-Sw.	Home position is the first encoder ref. signal to the left or right of the left or right edge of the home switch signal.
10 -	Pos. Lim.-Sw., Ref.-Sig. right of right Edge of Home-Sw.	
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).
12 -	Neg. Lim.-Sw., Ref.-Sig. left of right Edge of Home-Sw.	Reversal of direction of rotation when negative HW limit switch is reached.
13 -	Neg. Lim.-Sw., Ref.-Sig. right of left Edge of Home-Sw.	Home position is the first encoder ref. signal to the left or right of the left or right edge of the home switch signal.
14 -	Neg. Lim.-Sw., Ref.-Sig. left of left Edge of Home-Sw.	

<i>Homing Method 0x6098/0</i>		<i>Function</i>
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 <i>Home-Offset 1131</i>) is taken over as actual position value.

10.5.33 0x6099/n Homing speeds

Object 0x6099/1 *speed during search for switch* defines the speed (in user units per second) during search for switch.

Object 0x6091/2 *speed during search for zero* defines the speed (in user units per second) during search for zero.

Note:

Writing to object *speed during search for switch* automatically generates a write command to parameter *Fast Speed 1132* (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 1132.

Writing to object *speed during search for zero* automatically generates a write command to parameter *Creep speed 1133*(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 1133.

10.5.34 0x609A/0 Homing acceleration

Object 0x609A/0 *homing acceleration* defines acceleration and deceleration (in user units per second²) during homing.

Note:

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 1134.

10.5.35 0x60C1/1 Interpolation data record

Object 0x60C1/1 *interpolation data record 1* is the target position (in user units) used in interpolation position mode.

Interpolation position mode uses synchronous RxPDOs. The last received value for object 0x60C1/1 is activated with the next SYNC.

10.6 Special object and parameter dependencies

Some objects have special dependencies on parameters and functional behavior. This depends on the actual mode of operation, control commands and error reaction to communication errors (see object 0x6007/0 *abort connection option code*).

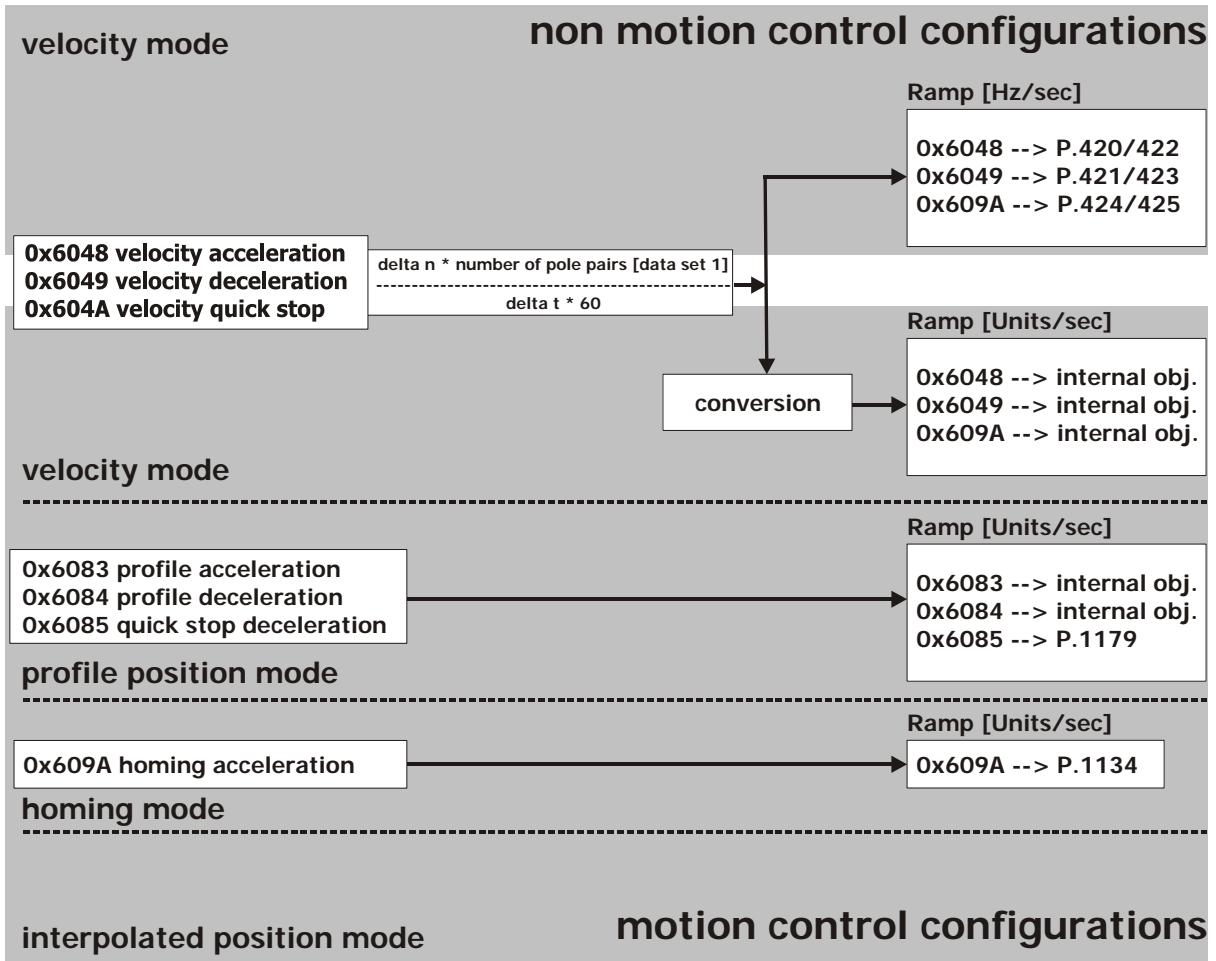
Related objects to the different modes of operation:

	acceleration	deceleration	quick stop deceleration
Velocity mode	0x6048 <i>velocity acceleration</i>	0x6049 <i>velocity deceleration</i>	0x604A <i>velocity quick stop</i>
Profile position mode	0x6083 <i>profile acceleration</i>	0x6084 <i>profile deceleration</i>	0x6085 <i>quick stop deceleration</i>
Homing mode	0x609A <i>homing acceleration</i>	0x609A <i>homing acceleration</i>	0x6085 <i>quick stop deceleration</i>
Interpolated position mode	0x6083 <i>profile acceleration</i>	0x6084 <i>profile deceleration</i> on stop command OR HALT	0x6085 <i>quick stop deceleration</i> on quick stop command

Related objects to mode of operation and used deceleration objects in the event of a communication error:

	deceleration	quick stop deceleration
Velocity mode	0x6049 <i>velocity deceleration</i>	0x604A <i>velocity quick stop</i>
Profile position mode	0x6084 <i>profile deceleration</i>	0x6085 <i>quick stop deceleration</i>
Homing mode	0x609A <i>homing acceleration</i>	0x6085 <i>quick stop deceleration</i>
Interpolated position mode	0x6084 <i>profile deceleration</i>	0x6085 <i>quick stop deceleration</i>

Correlation of objects, parameters and value conversion:



11 Inverter Control

The control of the frequency inverter can, in principle, be carried out using three operation modes. These are set via the data set change-over capable parameter *Local/Remote* **412**.

Parameter		Setting		
No.	Description	Min.	Max.	Fact. sett.
412	Local/Remote	0	44	44

For operation with CANopen, only the settings 0, 1 and 2 are relevant. The remaining settings relate to the possibilities of control via the KP500 control unit.

Control mode	Function
0 - Control via digital inputs	The Start and Stop command as well as the statement of the direction of rotation are via digital input signals.
1 - Control via state machine	The inverter is controlled by the <i>controlword</i> . Only in this control mode are the motion control functions supported by <i>controlword</i> and <i>modes of operation</i> as defined with CANopen DS402.
2 - Control via remote digital inputs	The Start and Stop command as well as the statement of the direction of rotation are via digital input signals emulated by the bits of the <i>controlword</i> .

Note:

Parameter *Local/Remote* **412** is data set change-over capable. Thus, it is possible to switch over between the various control modes via the data set selection.

The data set change-over can be carried out locally on the frequency inverter via digital inputs or via the bus. For data set change-over via the bus, parameter *Data set selection* **414** is used.

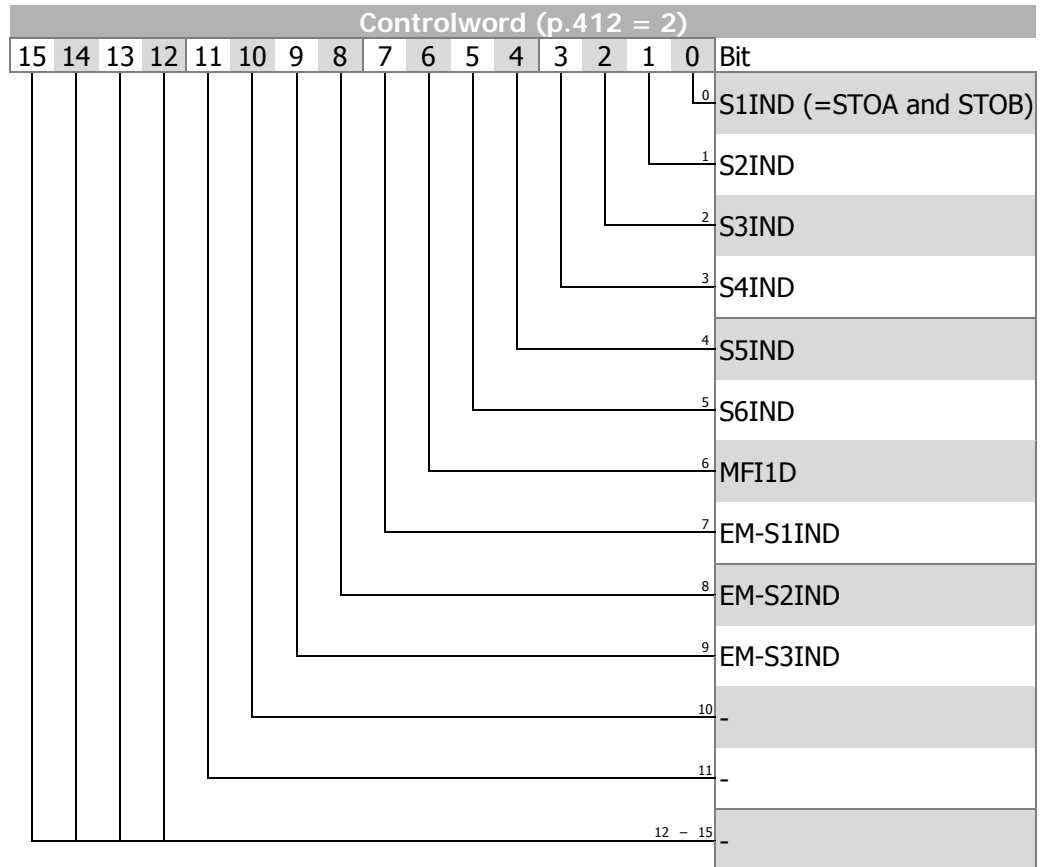
Parameter		Setting		
No.	Description	Min.	Max.	Fact. sett.
414	Data set selection	0	4	0

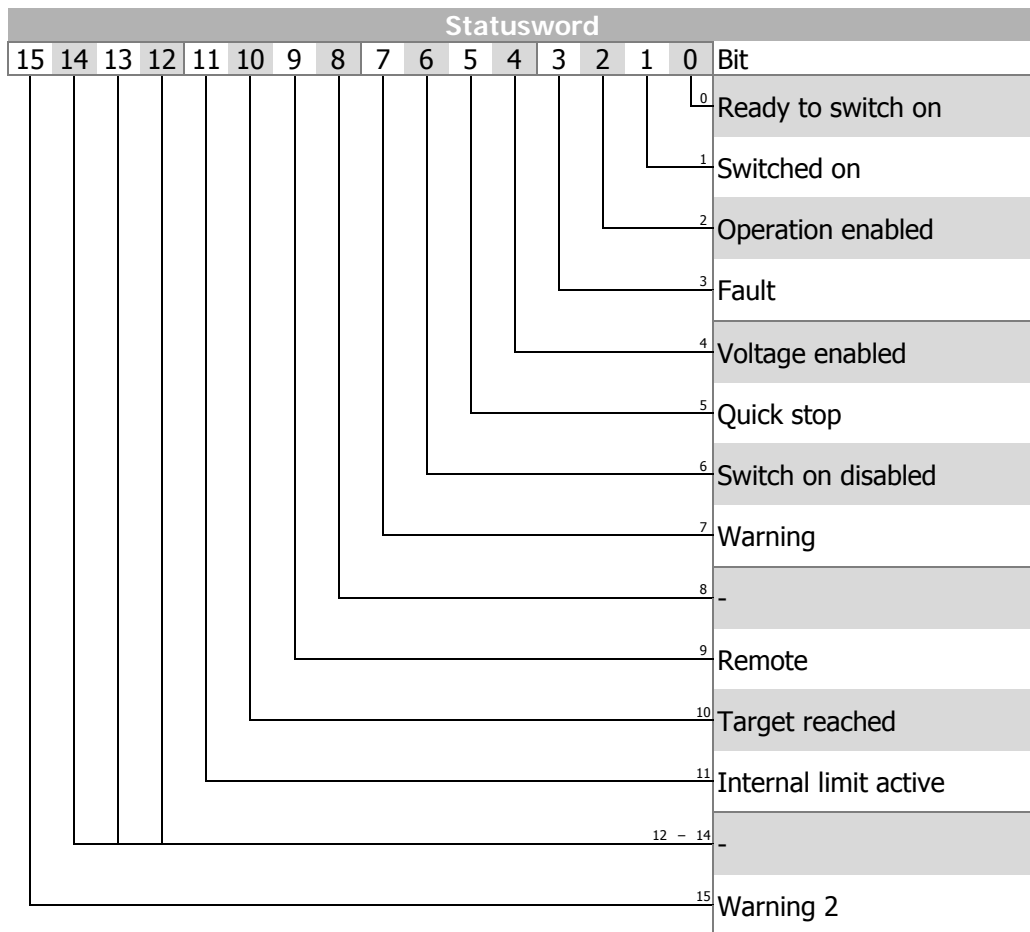
With *Data set selection* **414** = 0, the data set change-over via digital inputs is active. If *Data set selection* **414** has been set to 1, 2, 3, or 4, the corresponding data set is activated. Data set change-over via the digital inputs is then disabled.

Via parameter *Active data set* **249**, the currently selected data set can be read out. *Active data set* **249** states the activated data set with the value 1, 2, 3 or 4. This is independent of whether the data set change-over was carried out via digital inputs or via *Data set selection* **414**.

11.1 Control via digital inputs/remote digital inputs

In the operation mode Control via Contacts or Control via Remote-Contacts (parameter *Local/Remote* **412** = 0 or 2), the frequency inverter is controlled via the digital inputs S1IND (STOA AND STOB), S2IND to EM-S3IND directly, or by digital input emulation with help of the individual bits in the *controlword*. The meaning of these inputs can be taken from the operating instructions.





Note:

When using the control mode control via remote digital inputs, the digital inputs STOA AND STOB at X210A.3/X210B.2 must be set AND bit 0 of *controlword* must be set in order to get the drive started.

The two control modes, digital inputs and remote digital inputs, are only relevant to the *mode of operation* "velocity mode".

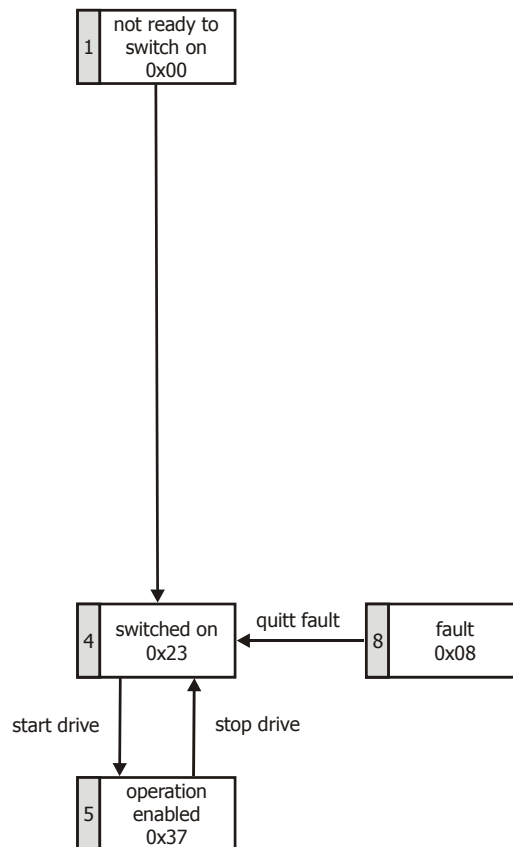
Note:

ACTIVE CUBE inverters support an external 24V supply for control logic. Even if the mains are not switched on, communication between the PLC and the inverter can still be established.

Bit 4 "Voltage enabled" of the *statusword* 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.

State machine:


Stateword	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

Note:

Bits marked "x" are don't care.

Bit no. 7, Warning, can be set at any time. It indicates a device-internal warning message. The evaluation of the present warning is carried out by reading out the warning status with parameter *Warnings 270*.

Bit no. 10, Target reached, is set when the specified reference value has been reached. In the special case of power failure regulation, the bit is also set if the power failure regulation has reached the frequency 0 Hz (see operating instructions). For "Reference value reached" there is a hysteresis (tolerance range), which can be set via parameter *max. Control deviation 549* (see operating instructions).

Bit no. 11, Internal limit active, indicates that an internal limit is active. This can, for example, be the present current limit, the torque limit or the over-voltage limit. All of these limit functions lead to the reference value being quit or not reached.

Bit no. 15, Warning 2, indicates a warning which leads to a fault switch-off of the frequency inverter within a short period of time. This bit is set if there is a warning for motor temperature, heat sink/inside temperature, Ixt monitoring or mains phase failure.

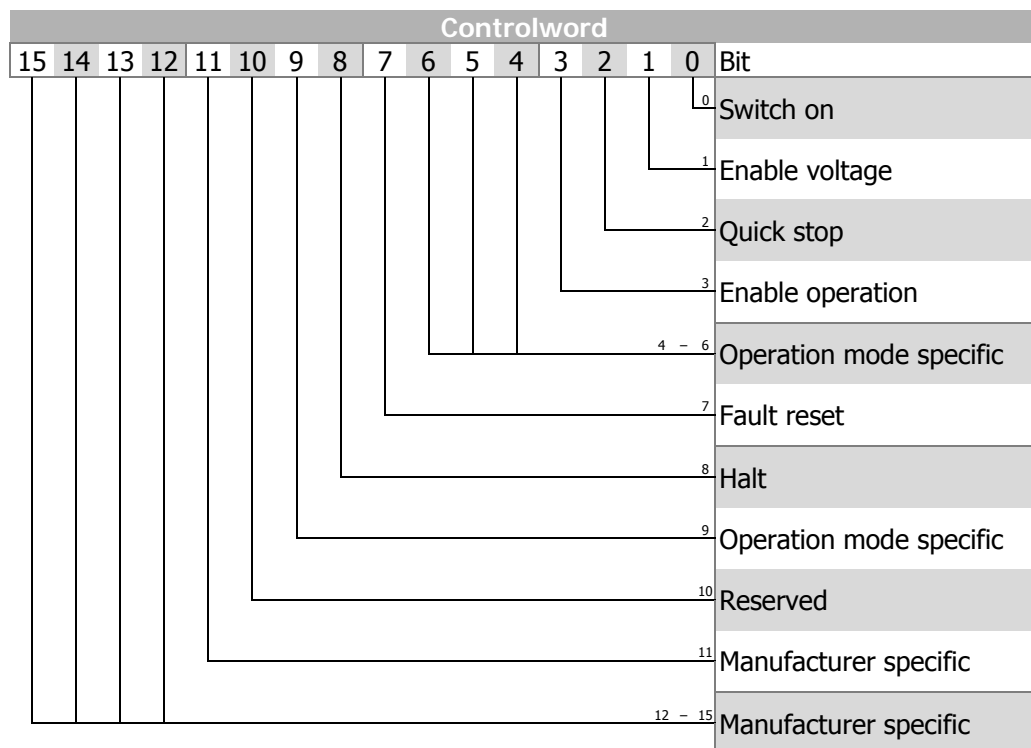
11.2 Control via state machine

In the operation mode "control via state machine" (parameter *Local/Remote* **412** = 1), the frequency inverter is controlled via the *controlword*.

State transition 4 to state "Operation enabled" is only possible when:

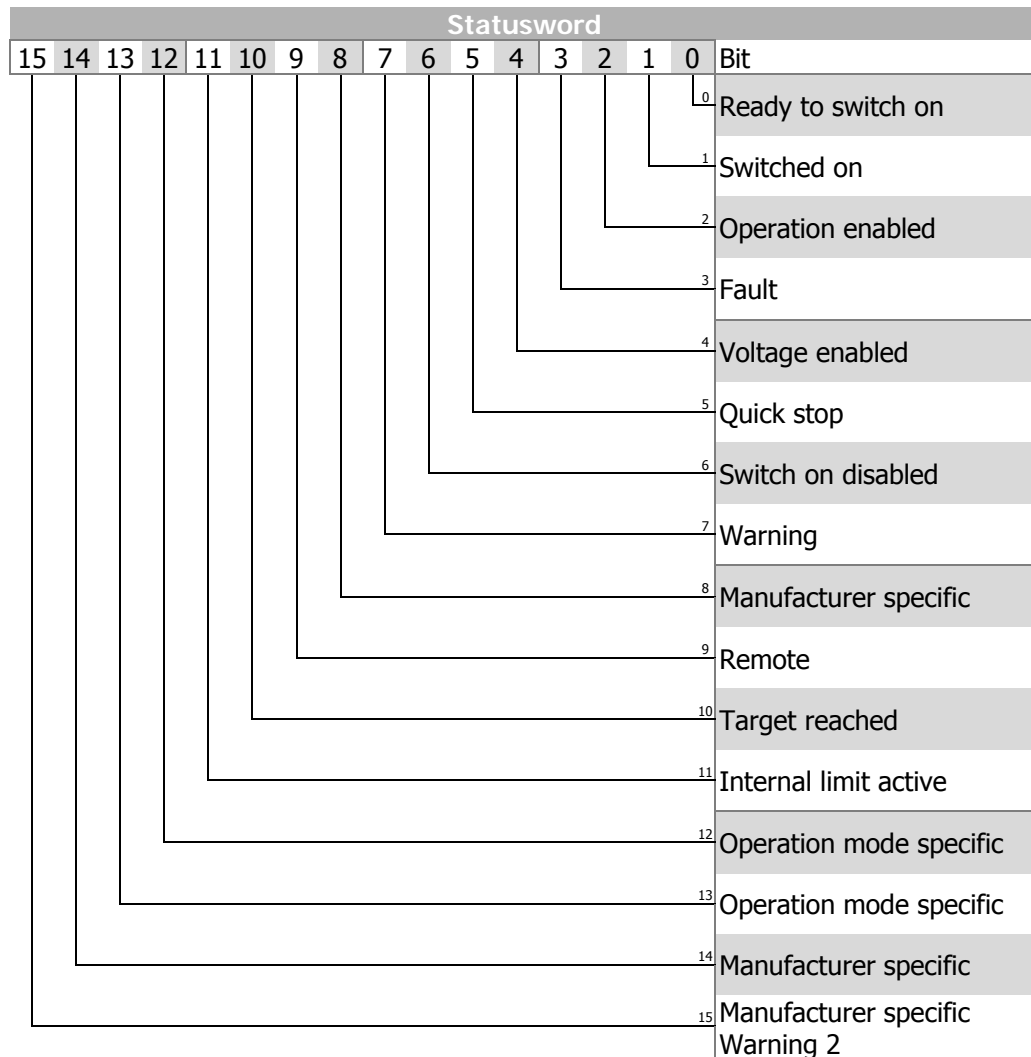
- In Motion control configuration (parameter *Configuration* **30** = x40) digital input S1IND (= STOA AND STOB) is set.
- In other control configurations (parameter *Configuration* **30** ≠ x40) digital input S1IND (= STOA AND STOB) AND (S2IND OR S3IND) is set; S2IND = start clockwise, S3IND = start anticlockwise

Object 0x6040/0 *controlword* is relevant to the inverter whenever parameter *Local/Remote* **412** is set to 1 (remote state machine).



Bits 9 ... 15 unused

Controlword bits 4, 5, 6 *operation mode specific* and bit 8 *halt* are used in motion control configurations (p.30 = x40) only.



Bit 14 unused

Statusword bits 12 and 13 *operation mode specific* are used in motion control configurations (p.30 = x40) only.

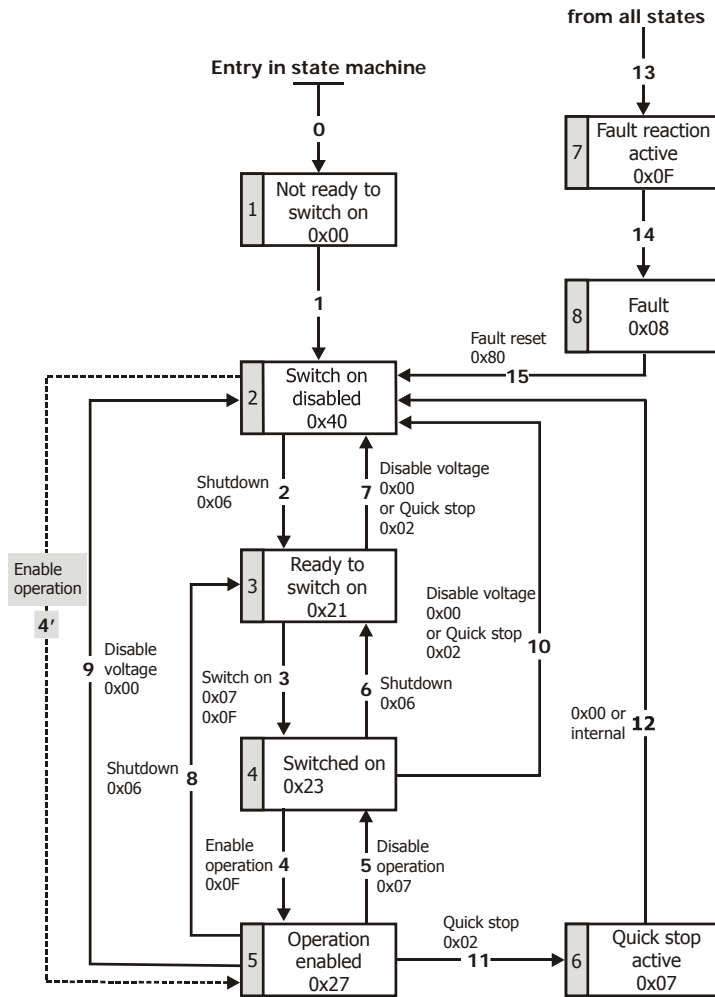
Note:

ACTIVE CUBE inverters support an external 24V supply for control logic. Even if the mains are not switched on, communication between the PLC and the inverter can still be established.

Bit 4 "**Voltage enabled**" of the *statusword* indicates the current state of the mains power supply.

Bit 4 "**Voltage enabled**" = 0 signals "no mains voltage" and the state transition "Ready to switch on" → "Switched on" is **not possible**.

Bit 4 "**Voltage enabled**" = 1 signals "mains voltage switched on" and the state transition "Ready to switch on" → "Switched on" is **possible**.

State machine:


The device control commands are triggered by the following bit pattern in the *controlword*:

Controlword						
Command	Bit 7	Bit 3	Bit 2	Bit 1	Bit 0	Transitions
	Fault reset	Enable operation	Quick stop	Enable voltage	Switch on	
Shutdown	X	X	1	1	0	2, 6, 8
Switch on	X	0	1	1	1	3
Switch on	X	1	1	1	1	3
Disable voltage	X	X	X	0	X	7, 9, 10, 12
Quick stop	X	X	0	1	X	7, 10, 11
Disable operation	X	0	1	1	1	5
Enable operation	X	1	1	1	1	4
Fault reset	0 ⇒ 1	x	x	x	x	15

Bits marked **X** are irrelevant

Note:

State transition 3 (command "Switch on") is only processed if bit no. 4 "Voltage enabled" of the statusword is set.

State transition 4 is only available for non motion control configurations (p.30 ≠ x40) and is only processed if bit no. 4 "Voltage enabled" of the statusword is set. This function is for downward compatibility to older software versions.

The *statusword* displays the current operation state.

Statusword						
state	Bit 6	Bit 5	Bit 3	Bit 2	Bit 1	Bit 0
	Switch on disabled	Quick stop	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	0	0	0	1	1	1
Fault reaction active	0	X	1	1	1	1
Fault	0	X	1	0	0	0

Bits marked **X** are irrelevant

Bit 7, Warning, can be set at any time. It indicates a device-internal warning message. The evaluation of the warning reason is carried out by reading out the warning status with the parameter *Warnings 270*.

Bit 9, Remote, is set if the operation mode "control via state machine" (*Local/Remote 412* = 1) has been set **and** the hardware release is available.

Logic linking of the digital control signals:

(S1IND (= STOA AND STOB) **AND** (Start clockwise **OR** Start anticlockwise))

The inverter can only be controlled via the state machine if the logic linking is true. The logic inputs for Start clockwise / Start anticlockwise can be connected directly to ON/OFF (p.68, p.69).

Note:

For motion control configurations (p.30 = **x40**), only the digital input S1IND (STOA AND STOB) must be set. Start clockwise and start anticlockwise have no function in

Bit 10, Target reached, is set whenever the specified reference value has been reached.

In non motion control configurations (p.30 \neq **x40**), target reached is related to the reference speed object 0x6042 *target velocity*. In the special case of power failure regulation, the bit is also set if the power failure regulation has reached the frequency 0 Hz (see operating instructions). For "Target reached" there is a hysteresis (tolerance range), which can be set via parameter *max. Control deviation 549* (see operating instructions).

Bit 11, Internal limit active, indicates that an internal limit is active. This can, for example, be the present current limit, the torque limit or the over-voltage limit. All of these limit functions lead to the reference value being quit or not reached.

Bit 15, Warning 2, indicates a warning which leads to a fault switch-off of the frequency inverter within a short period of time. This bit is set if there is a warning for motor temperature, heat sink/inside temperature, Ixt monitoring or mains phase failure.

11.3 Non motion control configurations

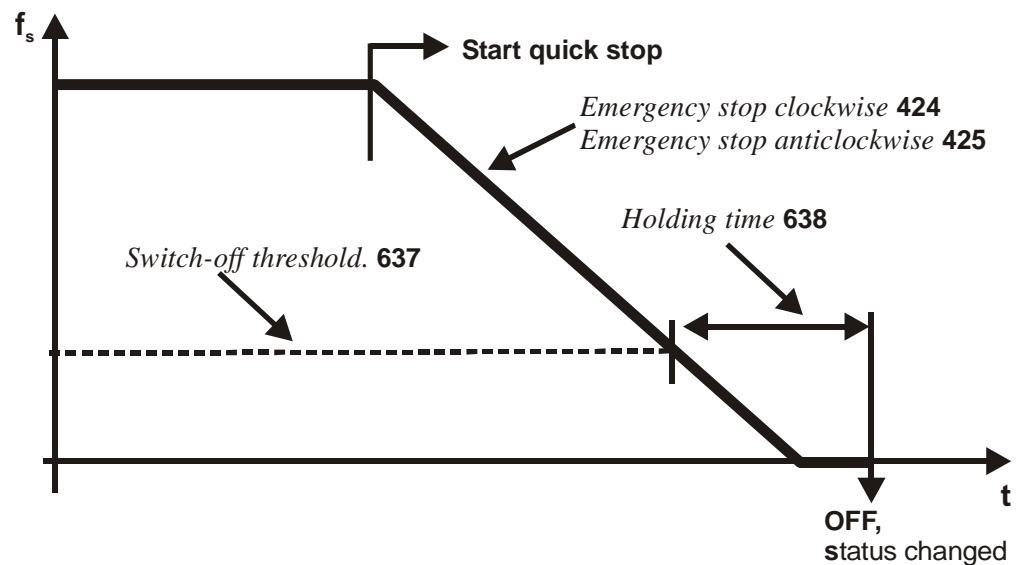
In non motion control configurations (p.30 \neq x40), object 0x6060 *modes of operation* is fixed to "2" *velocity mode*. Object 0x6061 *modes of operation display* is always "2" *velocity mode*. This cannot be changed.

Related objects:

0x6040	Controlword
0x6041	Statusword
0x6042	Target velocity
0x6043	Velocity demand
0x6044	Control effort
0x6046	Velocity min max amount
0x6048	Velocity acceleration
0x6049	Velocity deceleration

11.3.1 Behavior in quick stop

In quick stop, the parameters *Switch-off threshold* 637 (percent of f_{max}) and *Holding time* 638 (holding time after falling short of the switch-off threshold) are relevant. In a quick stop the drive is shutdown via the emergency stop ramps (*Emergency stop clockwise* 424 or *Emergency stop anti-clockwise* 425).



If frequency/speed zero has been reached during the holding time, the drive continues to be supplied with direct current until the switch-off time has expired. With this measure, there is an assurance that the drive is stationary in a change of state.

Note:

"Behavior in quick stop" is only relevant for non motion control configurations (p.30 \neq x40).

11.3.2 Behavior in transition 5

The *behavior in transition 5* from "Operation enabled" to "Switched on" can be parameterized. The behavior is set via parameter *State transition 5* **392**.

Parameter		Setting		
No.	Description	Min.	Max.	Fact. sett.
392	State transition 5	0	2	2

Operation mode	Function
0 - Coast to stop	immediate transition from "Operation enabled" to "Switched on", free stoppage of the drive
1 - DC brake	activation of DC brake; at the end of DC braking there is a change of state from "Operation enabled" to "Switched on"
2 - Ramp	transmission with normal stop ramp; after reaching standstill, there is a change of state from "Operation enabled" to "Switched on"

Note:

Setting operation mode "1 - DC brake" is only possible in applications with v/f characteristic (e.g. configuration 110), as other applications do not know such an operation mode.

If the frequency inverter is operated with a configuration which does not know the DC braking operation mode (e.g. configuration 210, field-orientation speed controlled), value "1" cannot be set. It is also not offered in the selection menus of the KP500 control unit or the VPlus program.

Note:

The default value for parameter *State transition 5* **392** is operation mode "2 - Ramp". For configurations with torque control, the default value is operation mode "0 - Coast to stop". If the configuration is changed, the value set for *State transition 5* **392** is also altered, if necessary.

Note:

"Behavior in transition 5" is only relevant for non motion control configurations (p.30 ≠ x40).

If transition 5 has been triggered with *State transition 5* **392** = "1 - DC brake", a new control word is only accepted after the completion of the transition process. The change of state from "Operation enabled" to "Ready" is carried out after the time parameterized for the DC brake *Braking time* **632** has expired.

If the parameter *State transition 5* **392** = "2 - Ramp" has been set, the *controlword* can be set back to "Enable operation" during the stoppage of the drive. In this way, the drive runs back up to its set reference value and remains in the state "Operation enabled".

The change of state from "Operation enabled" to "Switched on" is carried out after the set switch-off threshold has been reached **and** the set holding time (equivalent to the behavior in a quick stop) has expired. In this, the parameters *Switch-off threshold* **637** (percent of fmax) and *Holding time* **638** (holding time after switch-off threshold reached) are relevant.

11.3.3 Reference value / actual value

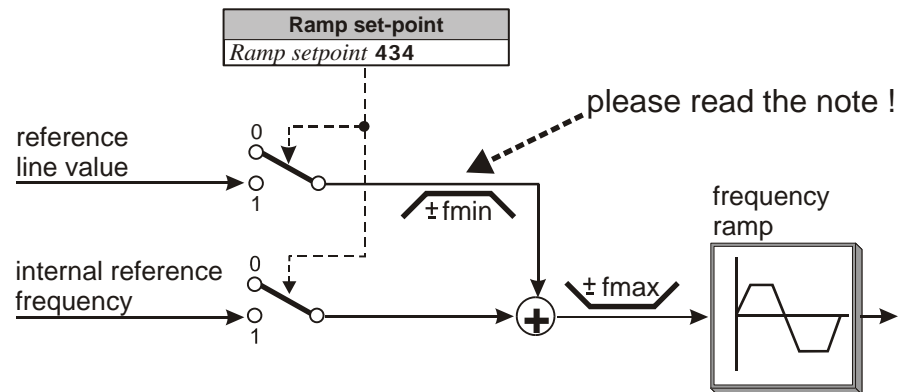
The PLC gives its reference value to the frequency inverter via object 0x6042/0 *target velocity* in the RxPDO used and receives the information on its actual value back via object 0x6044/0 *control effort* in the TxPDO used.

The use of the reference/actual value channel depends on the set configuration (control system). The actual value is generated from the appropriate source depending on the control system used.

Note:

The reference value in object 0x6042/0 *target velocity* and the actual value in object 0x6044/0 *control effort* 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 *target velocity* is connected to 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.	Fact. sett.
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

Note:

This function is only relevant for non motion control configurations (p.30 ≠ x40).

Note:

If *Ramp set-point 434* = 2 (only reference line value), then this reference line value is limited to *fmin*. Please remember that the sign in front of *fmin* at reference value = 0 is derived from the sign in front of the last reference line value $\neq 0$.

After Power 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.

Actual values		
Parameter	Contents	Format
<i>Internal reference frequency 228</i>	Internal reference value from the frequency reference value channel	xxx.xx Hz
<i>Reference bus frequency 282</i>	Reference line value from the CANopen bus	xxx.xx Hz
<i>Reference ramp frequency 283</i>	Sum of internal + reference line value	xxx.xx Hz

11.4 Motion control configurations

The function of the state machine describes the basic behavior of the inverter. In motion control configurations (p.30 = 0x40). The previously described objects *controlword* and *statusword* support the bits marked as "operation mode specific".

The meaning of these bits and the bit "Target reached" are different for the different motion control modes set by object 0x6060 *modes of operation*. The following chapters describe the use of the operation mode specific bits in *controlword* and *statusword* depending on the different motion control modes.

Principle functions:

Before a motion control command can be set by the "operation mode specific" bits of the *controlword*, the state machine must be set to "Operation enabled".

After the PLC has set a mode to object *modes of operation*, commands for this mode are not accepted until this mode is displayed in object *modes of operation display*.

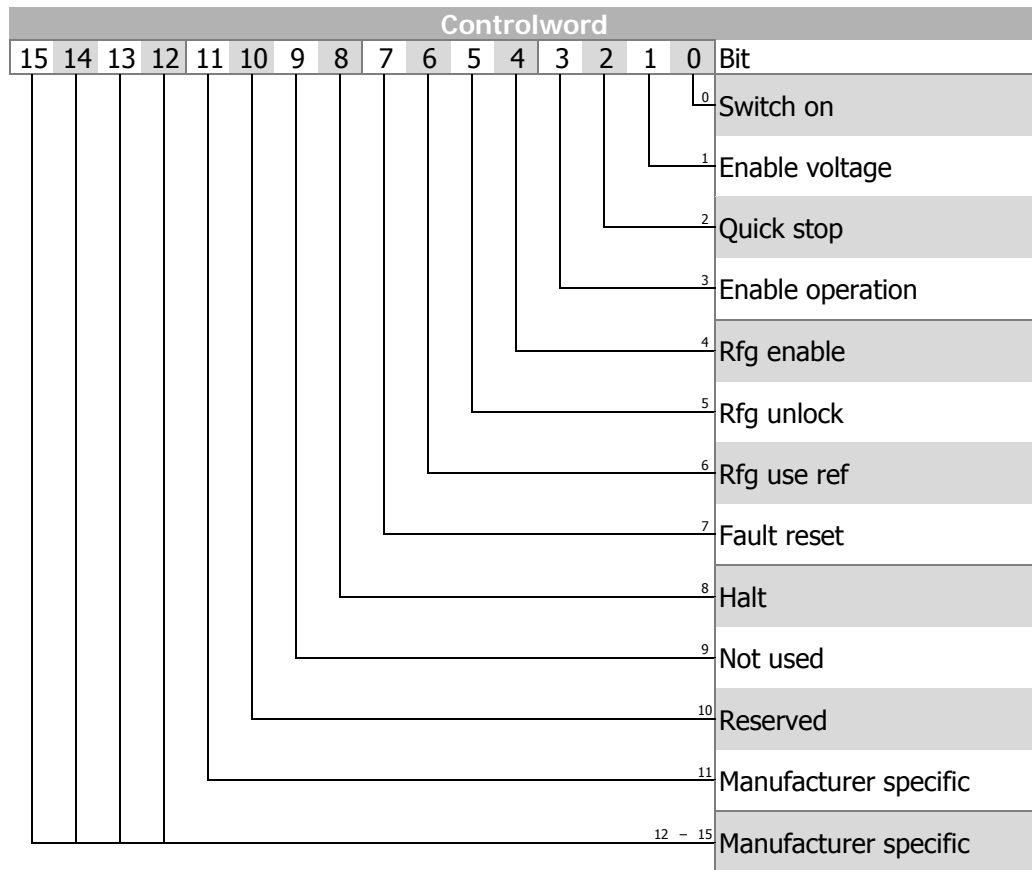
The bits in *controlword* and *statusword* marked "operation mode specific" are only supported in motion control configurations (p.30 = x40).

11.4.1 Velocity mode

In velocity mode the "operation mode specific" bits of the *controlword* control the ramp function generator "rfg". The function is explained in the block diagram.

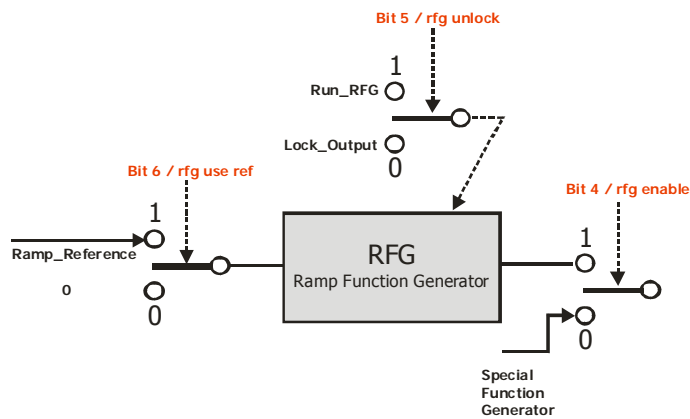
Related objects:

0x6040	Controlword
0x6041	Statusword
0x6042	Target velocity
0x6060	Modes of operation
0x6061	Modes of operation display
0x6043	Velocity demand
0x6044	Control effort
0x6046	Velocity min max amount
0x6048	Velocity acceleration
0x6049	Velocity deceleration
0x604A	Velocity quick stop



Statusword																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit
															0	Ready to switch on
															1	Switched on
															2	Operation enabled
															3	Fault
															4	Voltage enabled
															5	Quick stop
															6	Switch on disabled
															7	Warning
															8	Manufacturer specific
															9	Remote
															10	Target reached (not used)
															11	Internal limit active
															12	Not used
															13	Not used
															14	Manufacturer specific
															15	Manufacturer specific Warning2

Block diagram



HALT = 0 → execute motion
HALT = 1 → stop axle (inverter remains in state "operation enabled")

Note:

The "rfg enable" bit (bit 4) is not supported by the inverter and has no function.

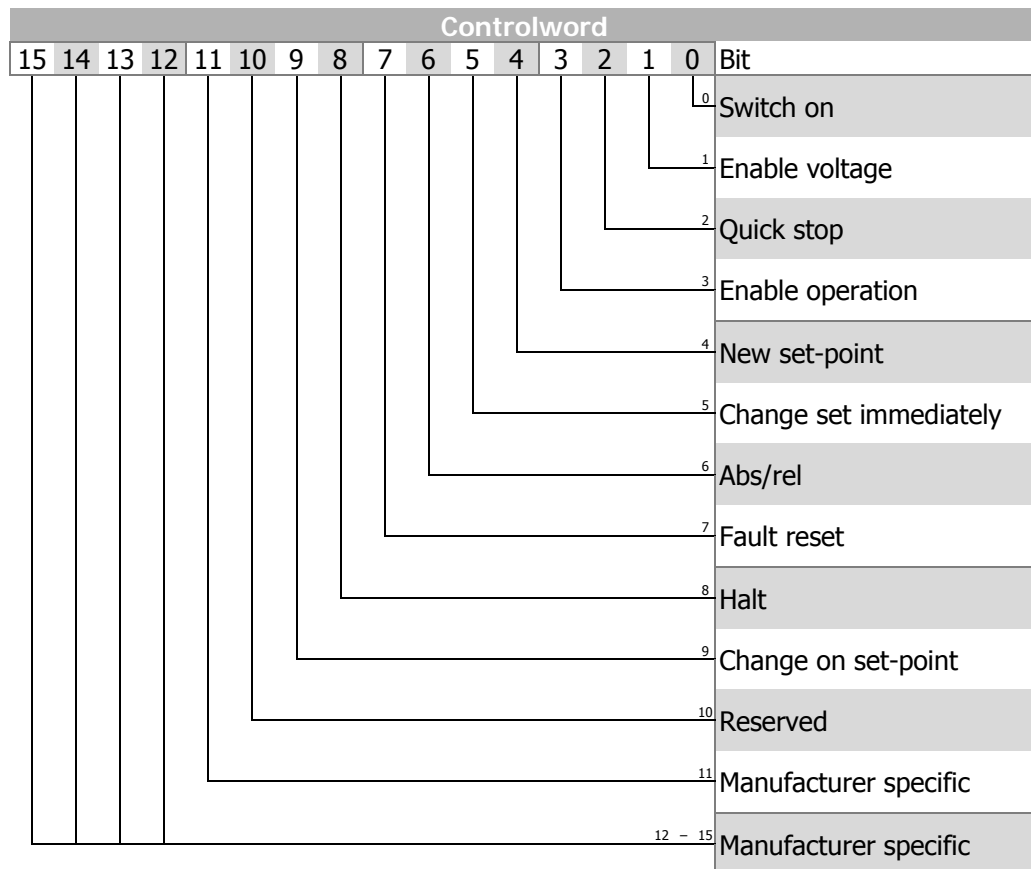
11.4.2 Profile position mode

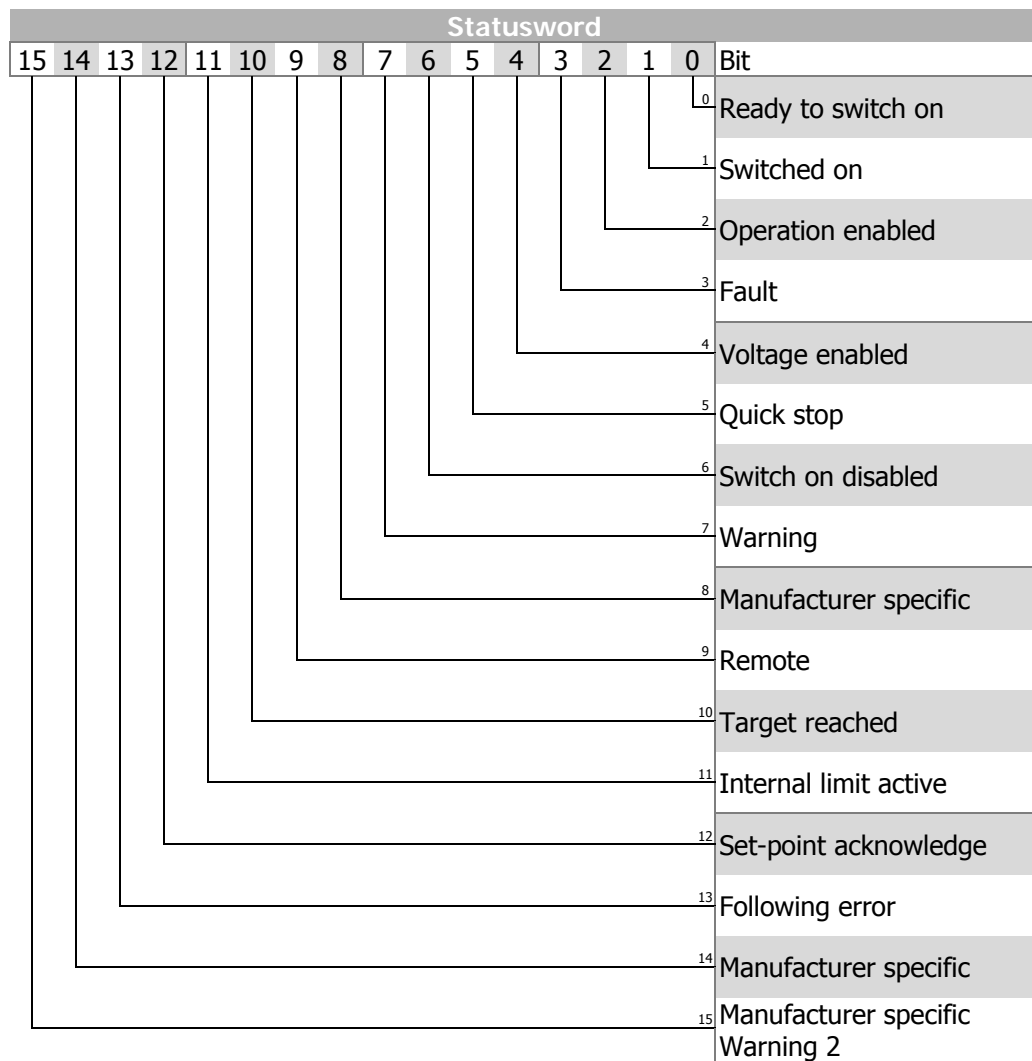
In profile position mode the inverter receives a target position followed by a command to move to this position.

Related objects:

0x6040	Controlword
0x6041	Statusword
0x6060	Modes of operation
0x6061	Modes of operation display
0x607A	Target position
0x6081	Profile velocity
0x6083	Profile acceleration
0x6084	Profile deceleration
0x6085	Quick stop deceleration

In profile position mode the "operation mode specific" bits of controlword and statusword are used as shown:





Controlword

Change on set-point	Change set-point immediately	New set-point	Description
0	0	0 → 1	Positioning shall be completed (target reached) before the next one is started
X	1	0 → 1	Next position shall be started immediately
1	0	0 → 1	Positioning with the current profile velocity up to the current set-point shall be proceeded and then the next positioning shall be applied

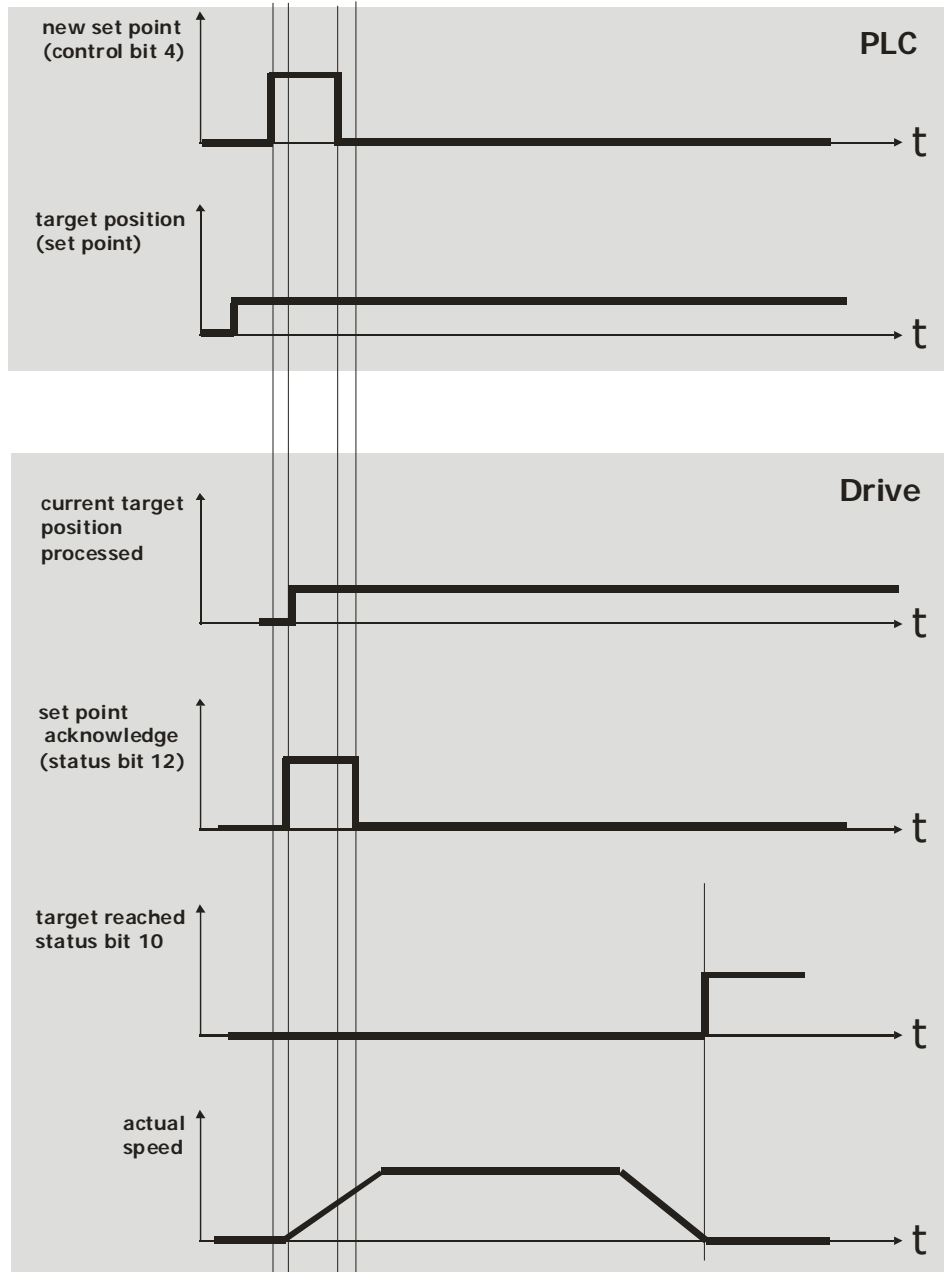
Name	Value	Description
Abs/rel	0	<i>Target position</i> is an absolute value
	1	<i>Target position</i> is a relative value
Halt	0	Execute positioning
	1	Stop axle with <i>profile deceleration</i> (if not supported with <i>profile acceleration</i>), the inverter remains in state "operation enabled"

Statusword

Name	Value	Description
Target reached	0	Halt = 0: <i>target position</i> not reached
		Halt = 1: axle decelerates
	1	Halt = 0: <i>target position</i> reached
		Halt = 1: velocity of axle is 0
Set-point acknowledge	0	Trajectory generator has not assumed the positioning value (yet)
	1	Trajectory generator has assumed the positioning value
Following error	0	No following error
	1	Following error

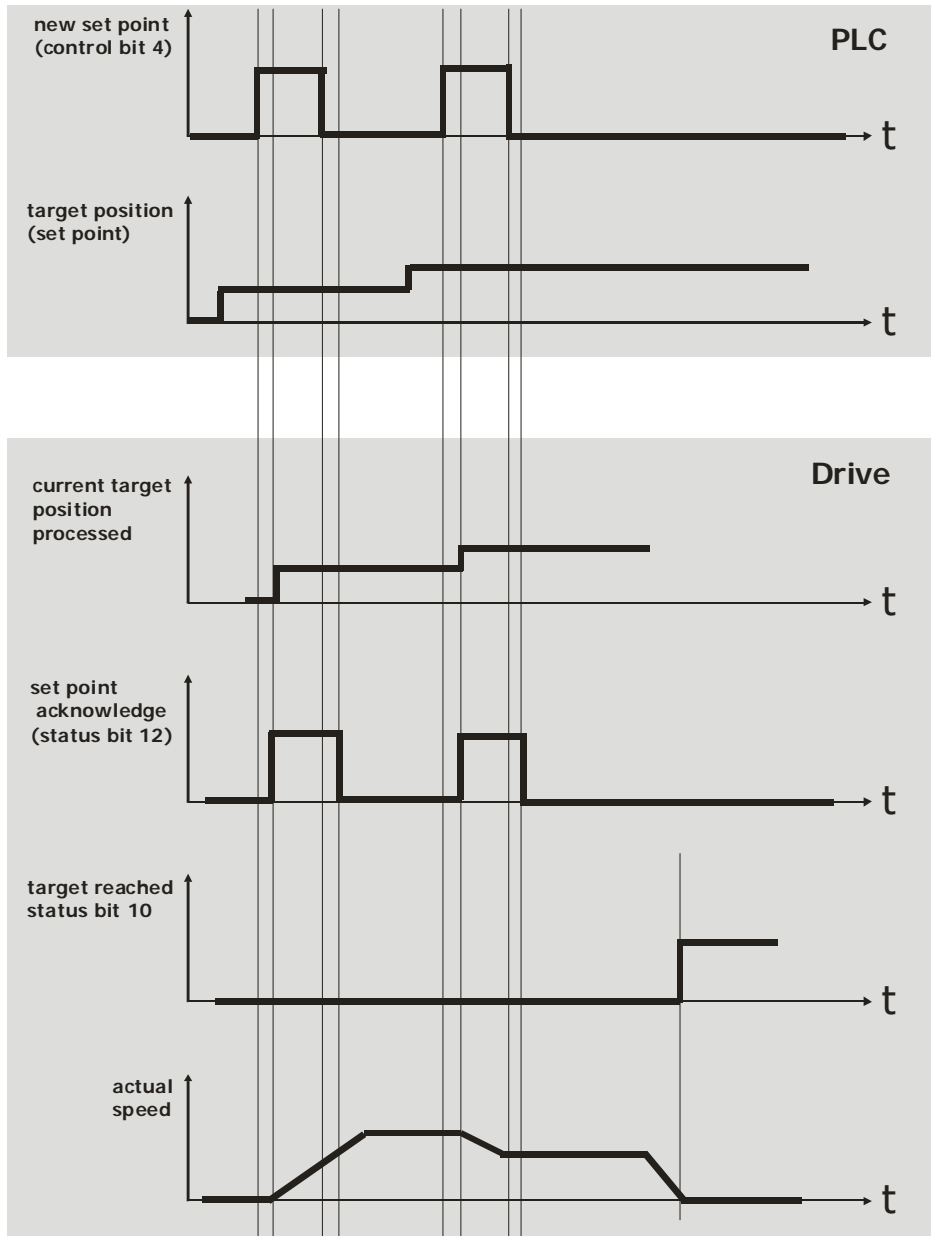
Example: single set-point
 control bit *change on set-point* = 0
 control bit *change set immediately* = 0

After a set-point is sent to the drive, the control device signals set-point valid by a rising edge on bit *new set-point* in the controlword. The drive answers by setting bit *set-point acknowledge* and starts moving to the new target position. After that, the control device clears the bit *new set-point* and the drive also clears the bit *set-point acknowledge*. After clearing the bit *set-point acknowledge* the drive is able to accept a new target position.



Example: **single set-point**
 control bit *change on set-point* = 0
 control bit *change set immediately* = 1

When a set-point is in progress and a new set-point is validated by control bit *new set-point* (rising edge), the new set-point is processed immediately.



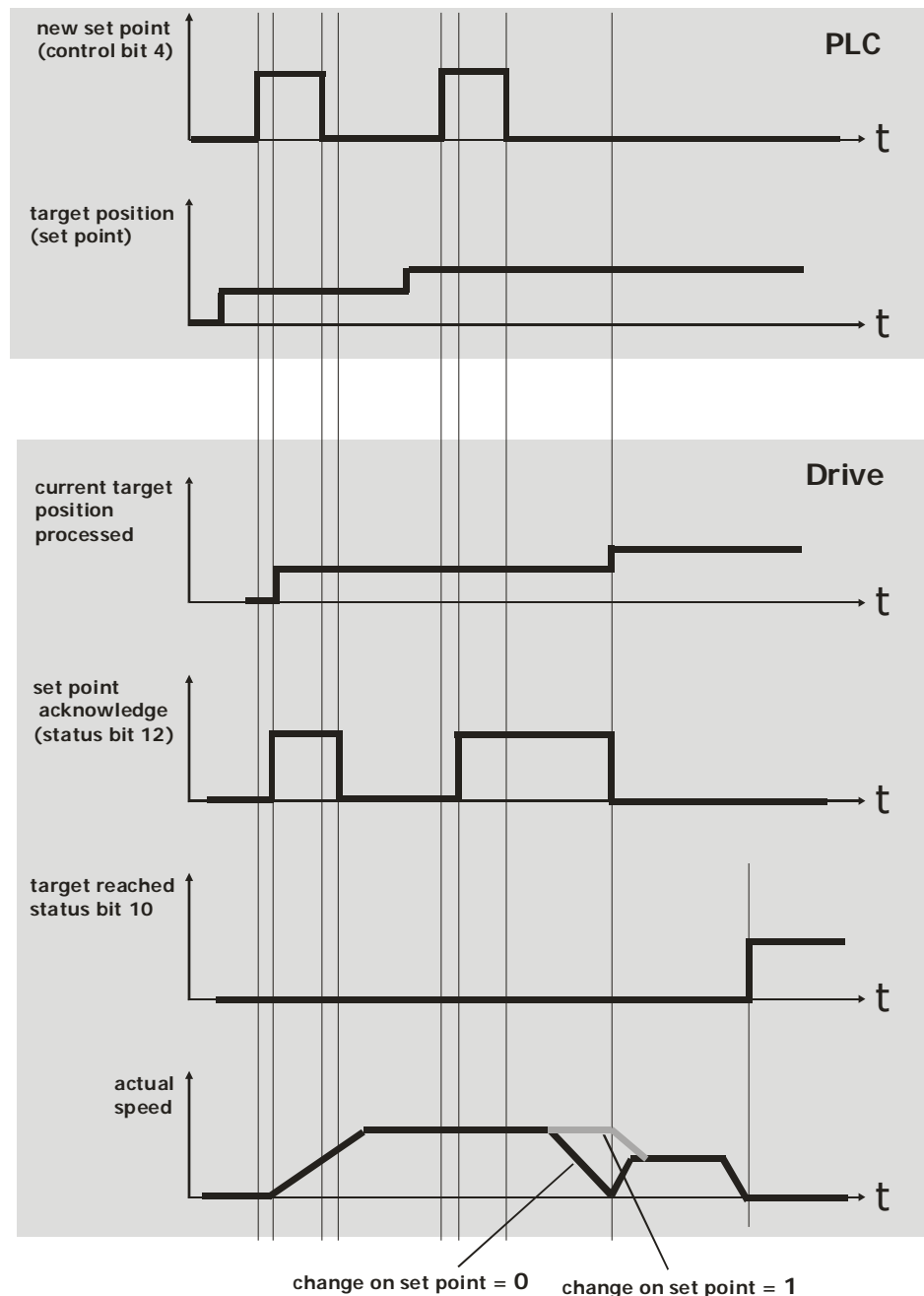
Example: set of set-points
 control bit *change on set-point* = 0/1
 control bit *change set immediately* = 0

While a Positioning is in progress, the set point is changed.

Change on set point = 0 Current target position will be **stopped** at. After the position is reached, the new set point is taken over.

Change on set point = 1 The current target position will be **driven to in current speed**. As soon as the position is reached, the new set point is taken over.

The grey line in segment "actual speed" shows the actual speed behavior if control bit *change of set-point* is set (= 1).



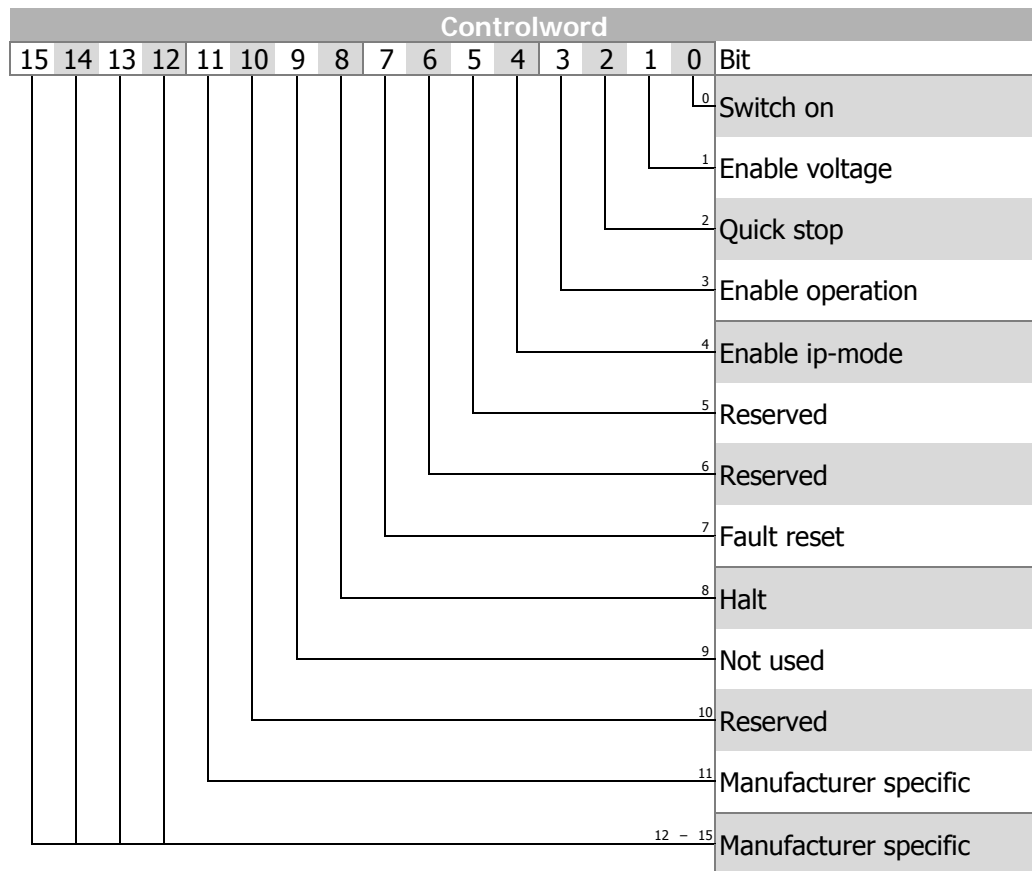
11.4.3 Interpolated position mode

In interpolated position mode the inverter receives a target position at equidistant time intervals.

Related objects:

0x6040	Controlword
0x6041	Statusword
0x6060	Modes of operation
0x6061	Modes of operation display
0x60C1	Interpolation data record

In interpolated position mode the "operation mode specific" bits of *controlword* and *statusword* are used as shown:



Statusword																
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0	Bit
															0	Ready to switch on
															1	Switched on
															2	Operation enabled
															3	Fault
															4	Voltage enabled
															5	Quick stop
															6	Switch on disabled
															7	Warning
															8	Manufacturer specific
															9	Remote
															10	Target reached
															11	Internal limit active
															12	Ip-mode active
															13	Reserved
															14	Manufacturer specific
															15	Manufacturer specific Warning 2

Only linear interpolation is available (for this, object 0x60C0/0 *interpolation submode select* is NOT implemented). For proper operation, object 0x60C1/1 *interpolation data record* must be sent by a synchronous RxPDO. The time interval of the SYNC messages MUST be constant. For the evaluation of the SYNC interval, object 0x1006/0 *communication cycle period* must be set or at least eleven SYNC messages are necessary before the interpolated position mode is activated by setting control bit 4 *enable_ip_mode*. For the SYNC interval only multiples of milliseconds are allowed (1, 2, 3, 4, ...; refer to chapter 10.2.4 "0x1006/0 Communication Cycle Period").

The activation of interpolated position mode is displayed by status bit 12 *ip_mode_active*.

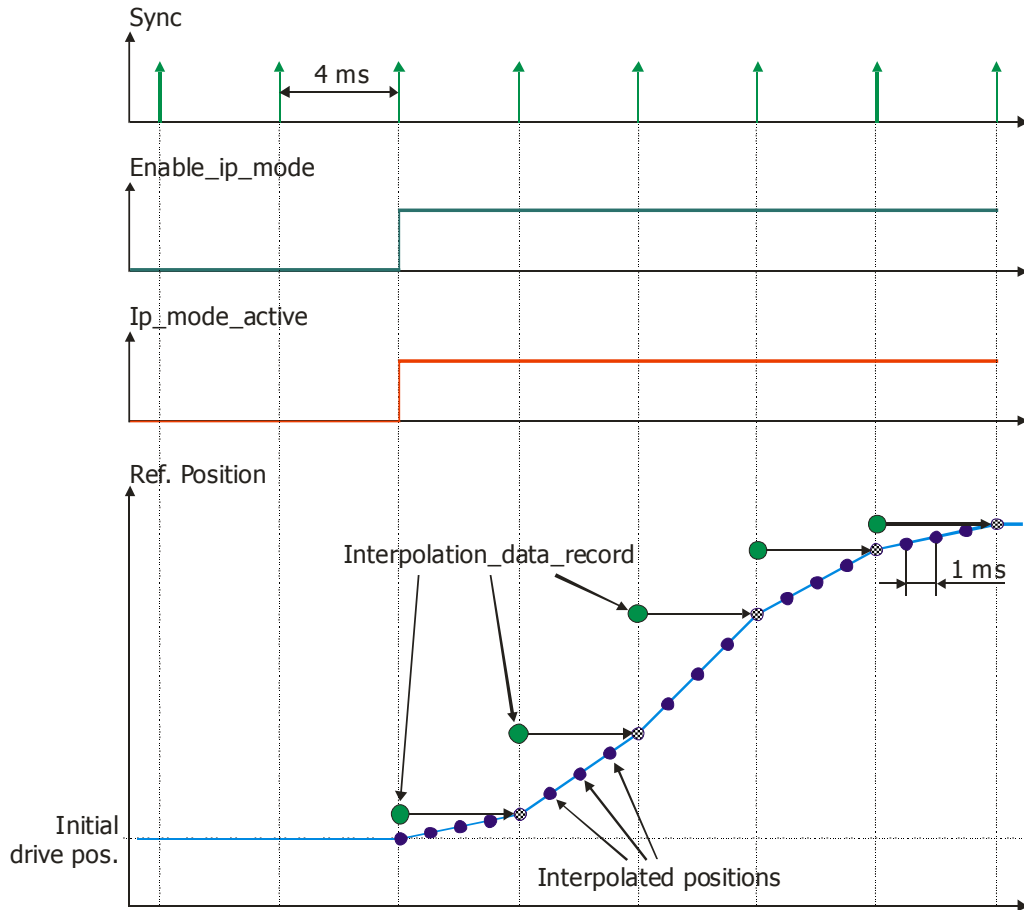
With each SYNC message a new target position is transmitted to the drive by object 0x60C1/1 *interpolation data record*. The new interpolated reference positions and an additional speed reference are calculated from the last reference position, the *interpolation data record* and the time interval of the SYNC messages. As shown, the target position actually received will be reached at the time of the next SYNC message.

Controlword

Name	Value	Description
Enable ip-mode	0	Interpolated position mode inactive
	1	Interpolated position mode active
Halt	0	Execute the instruction of bit 4 "enable ip-mode"
	1	Stop axle , inverter remains in state "operation enabled"

Statusword

Name	Value	Description
Target reached	0	Halt = 0: position not (yet) reached
		Halt = 1: axle decelerates
	1	Halt = 0: position reached
		Halt = 1: axle has velocity 0
Ip-mode active	0	Interpolated position mode inactive
	1	Interpolated position mode active



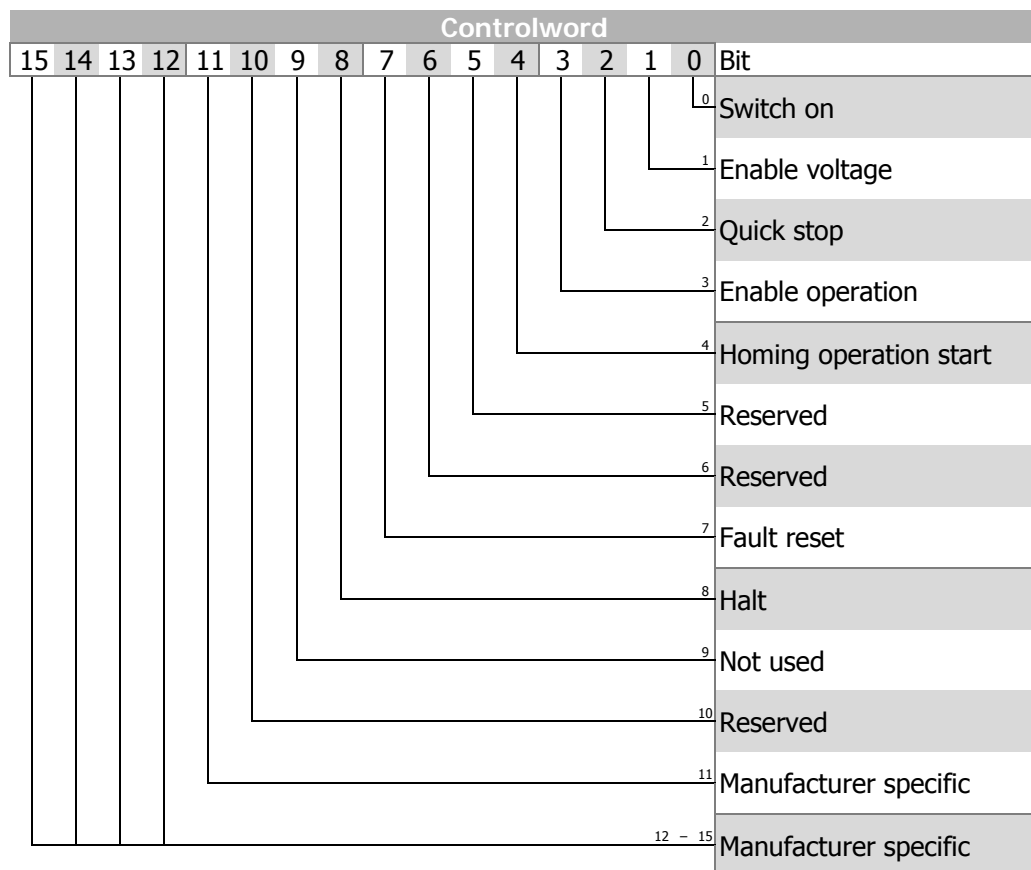
11.4.4 Homing mode

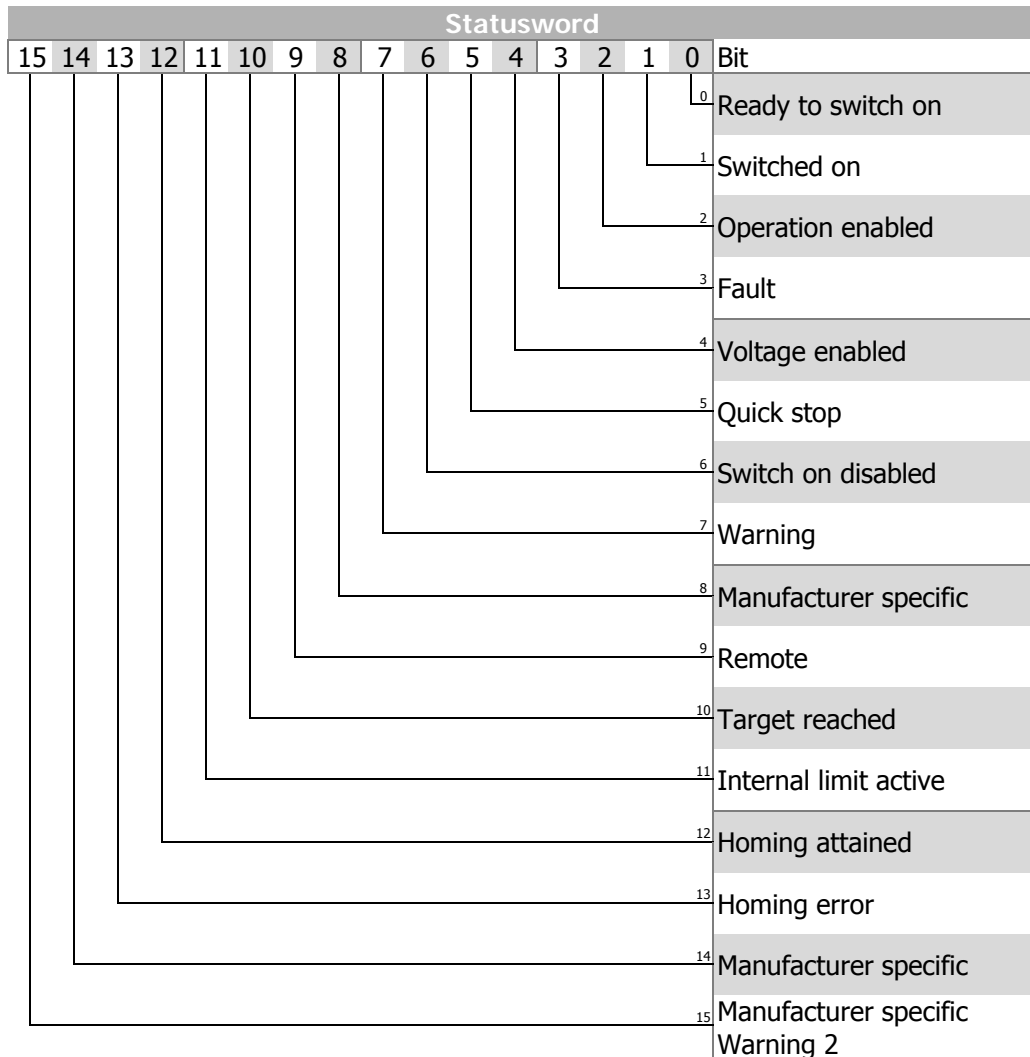
In homing mode the inverter moves the drive to a reference position. The method used for this movement is defined by object 0x6098 *homing method*.

Related objects:

0x6040	Controlword
0x6041	Statusword
0x6060	Modes of operation
0x6061	Modes of operation display
0x6098	Homing method
0x6099	Homing speeds
0x609A	Homing acceleration

In homing mode the "operation mode specific" bits of *controlword* and *statusword* are used as shown:





Controlword

Name	Value	Description
Homing operation start	0	Homing mode inactive
	0 → 1	Start homing mode
	1	Homing mode active
	1 → 0	Interrupt homing mode
Halt	0	Execute instruction of bit 4 "homing operation start"
	1	Stop axle with homing acceleration, , inverter remains in state "operation enabled"

Statusword

Name	Value	Description
Target reached	0	Halt = 0: home position not reached Halt = 1: axle decelerates
	1	Halt = 0: home position reached Halt = 1: axle has velocity 0
Homing attained	0	Homing not yet completed
	1	Homing mode carried out successfully
Homing error	0	No homing error
	1	Homing error occurred Homing mode carried out unsuccessfully

For an exact description of the various homing modes refer to the application manual "Positioning".

11.4.5 Table travel record

In table travel record mode the inverter moves the drive autonomous to consecutive positions.

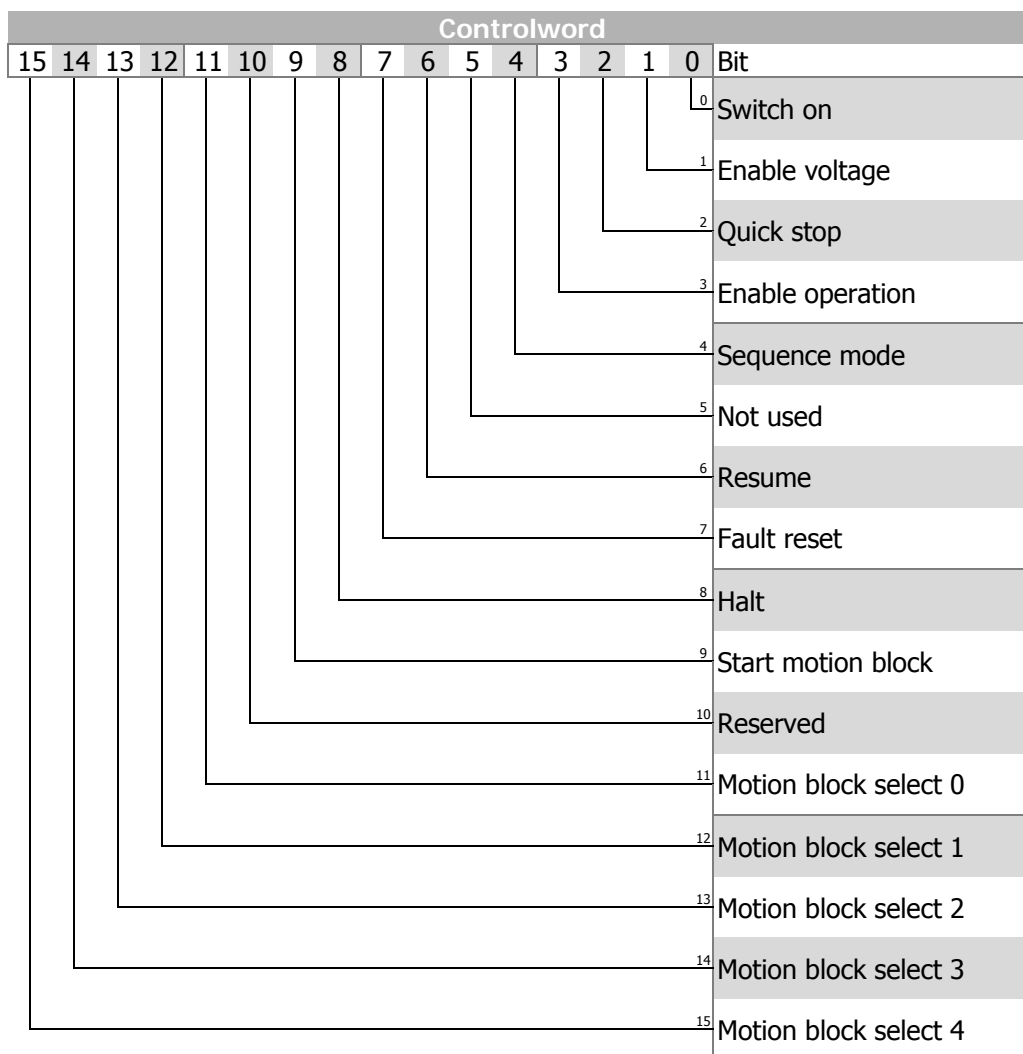
The table travel record mode uses predefined sets of positions. Every target position is defined by one motion block. Several sets of motion blocks may be defined.

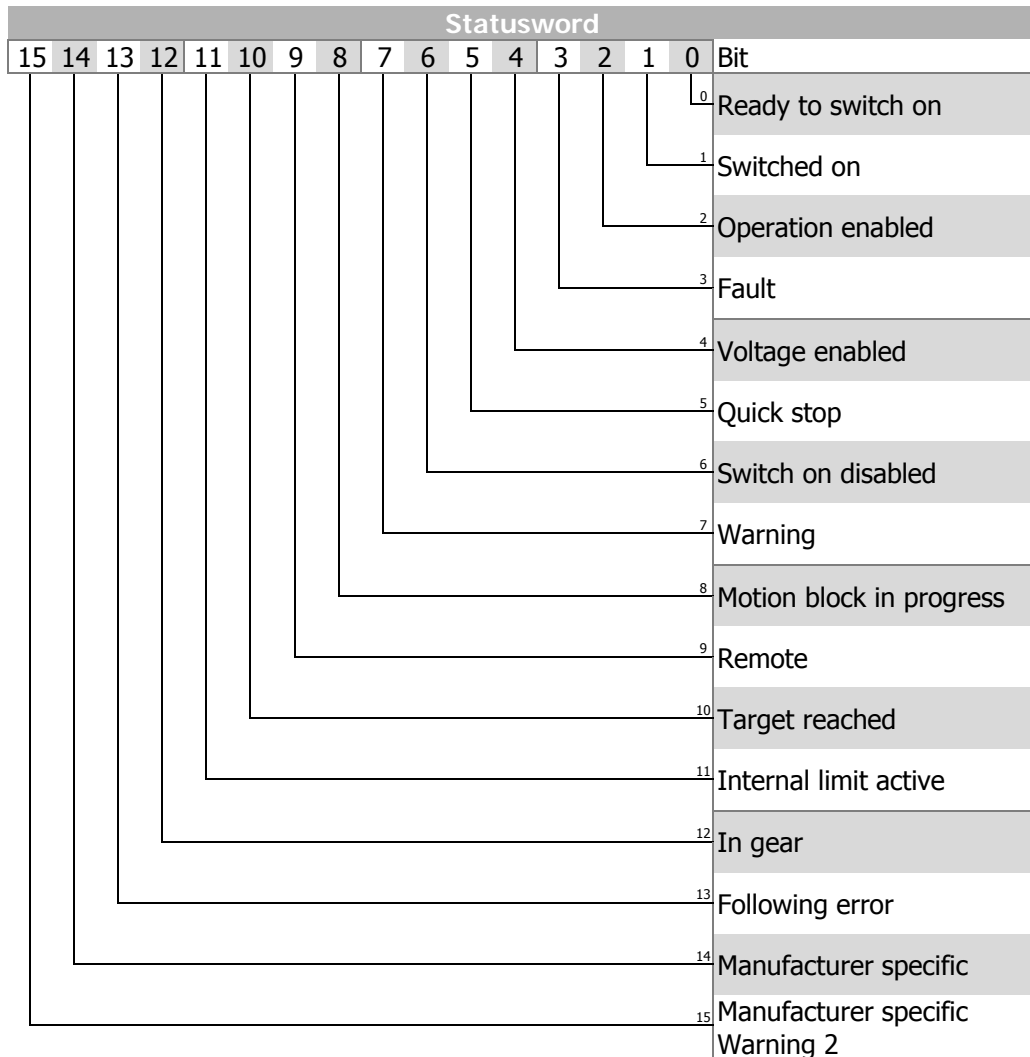
For an exact description of table travel record mode options refer to the application manual "Positioning".

Related objects:

0x6040	Controlword
0x6041	Statusword
0x6060	Modes of operation
0x6061	Modes of operation display
0x5FF0	Active motion block
0x5FF1	Motion block to resume
0x6064	Position actual value
0x6065	Following error window
0x6066	Following error time
0x6067	Position window
0x6068	Position window time
0x6085	Quick stop deceleration

In table travel mode the "operation mode specific" and "manufacturer specific" bits of *controlword* and *statusword* are used as shown:





Controlword

Name	Value	Description
Sequence mode	0	Single motion block
	1	Sequence of motion blocks
Resume	0	Start motion block = motion block select
	1	Start motion block = last active motion block
Halt	0	Execute instruction of bit 4 "sequence mode"
	1	Stop axle with ramp of actual motion block, inverter remains in state "operation enabled"
Start motion block	0	Stop axle with ramp of actual motion block
	0 → 1	Start execution of motion block(s)
Motion block select 0...4	n	Start motion block = n + 1

Motion block select:

Controlword															
15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Motion block select															
4	3	2	1	0											

Start motion block = motion block select + 1:

Motion block select					resulting
4	3	2	1	0	start motion block
0	0	0	0	0	1
0	0	0	1	1	4
1	0	0	0	0	17
1	1	1	1	1	32

Statusword

Name	Value	Description
Motion block in progress	0	Single motion block: motion block completed Sequence of motion blocks: sequence completed
	1	Single motion block/sequence of motion blocks active
Target reached	0	Halt = 0: target position not reached yet (motion blocks with positioning only)
		Halt = 1: axle decelerates
	1	Halt = 0: target position reached (motion blocks with positioning only)
		Halt = 1: axle has velocity 0
In gear	0	Electronic gear not coupled
	1	Electronic gear coupled
Following error	0	No following error
	1	Following error

Basic functions

The control bit *sequence mode* decides between execution of one single motion block (*sequence mode* = 0) and execution of a sequence of motion blocks (*sequence mode* = 1).

In both cases the selection of the desired motion block (motion block number of single motion block or start motion block number of motion block sequence) is done by calculating the motion block number from *motion block select* with the rising edge of *start motion block*

While the selected motion block or motion block sequence is processed *motion block in progress* is set to 1. *Motion block in progress* remains 1 until the motion block processing is finished. When using a single motion block *motion block in progress* is set to 0 with completion of the single motion block. Otherwise when using a motion block sequence *motion block in progress* is set to 0 when reaching a next motion block setting of 0 (end of motion block) or -1 (error) -2 (stop and error) or -3 (quick stop and error).

While processing a motion block sequence the actual processed motion block is displayed by object 0x5FF0 *active motion block*.

If motion block processing is interrupted by setting *start motion block* to 0 the axle stops with the ramp defined by the actual motion block. The interrupted motion block/motion block sequence can be restarted again by setting *resume* and a rising edge of *start motion block*.

If *resume* is set to 1 and there is no valid motion block available the motion block defined by *motion block select* will be used. A valid motion block is displayed by object 0x5FF1 *motion block to resume*. *Motion block to resume* displays -1 in case there is no valid motion block or the last motion block/motion block sequence was not interrupted.

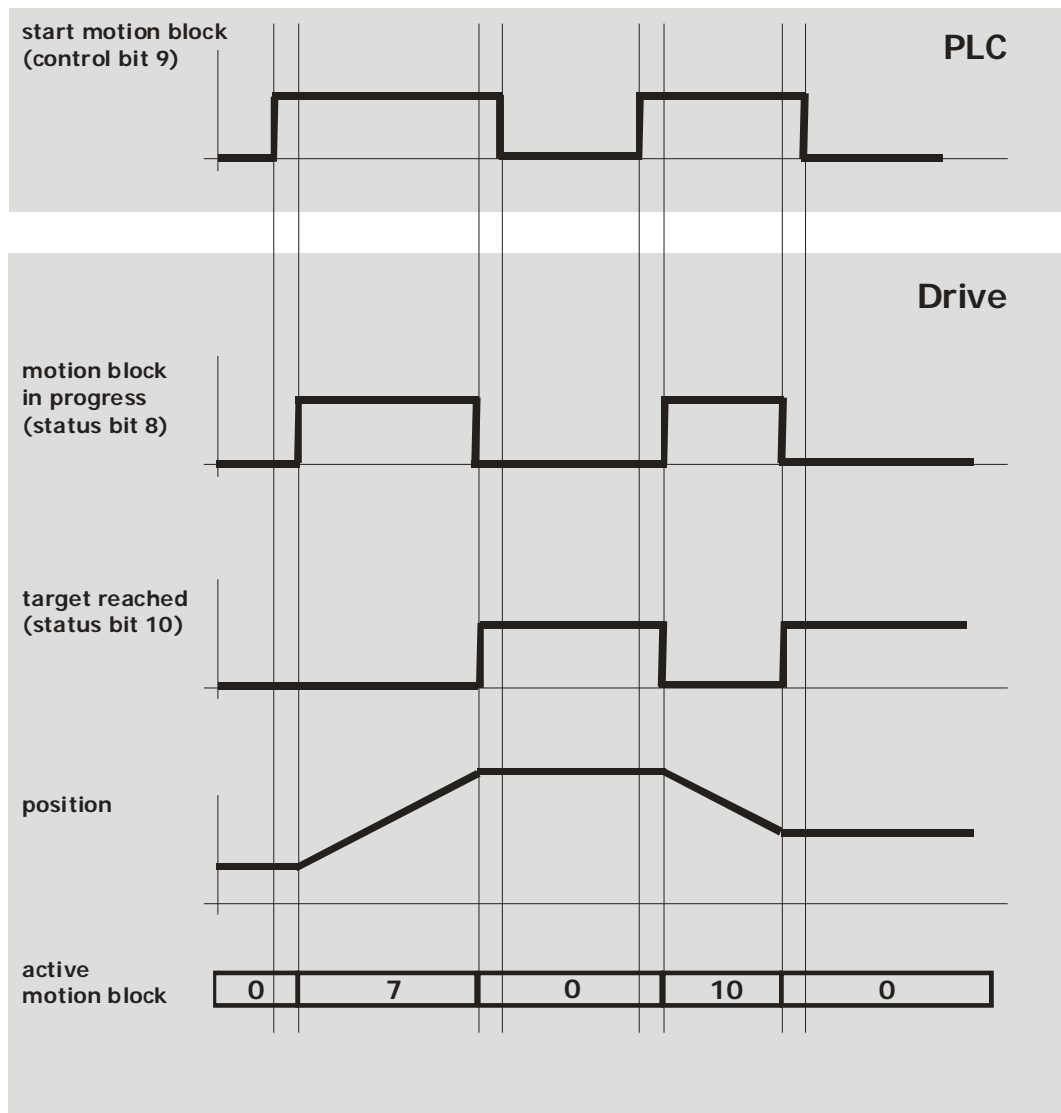
Target reached is set by motion blocks with absolute or relative positioning when the actual position reaches the *position window*.

In gear is set when the function electronic gear is used and the gear is coupled.

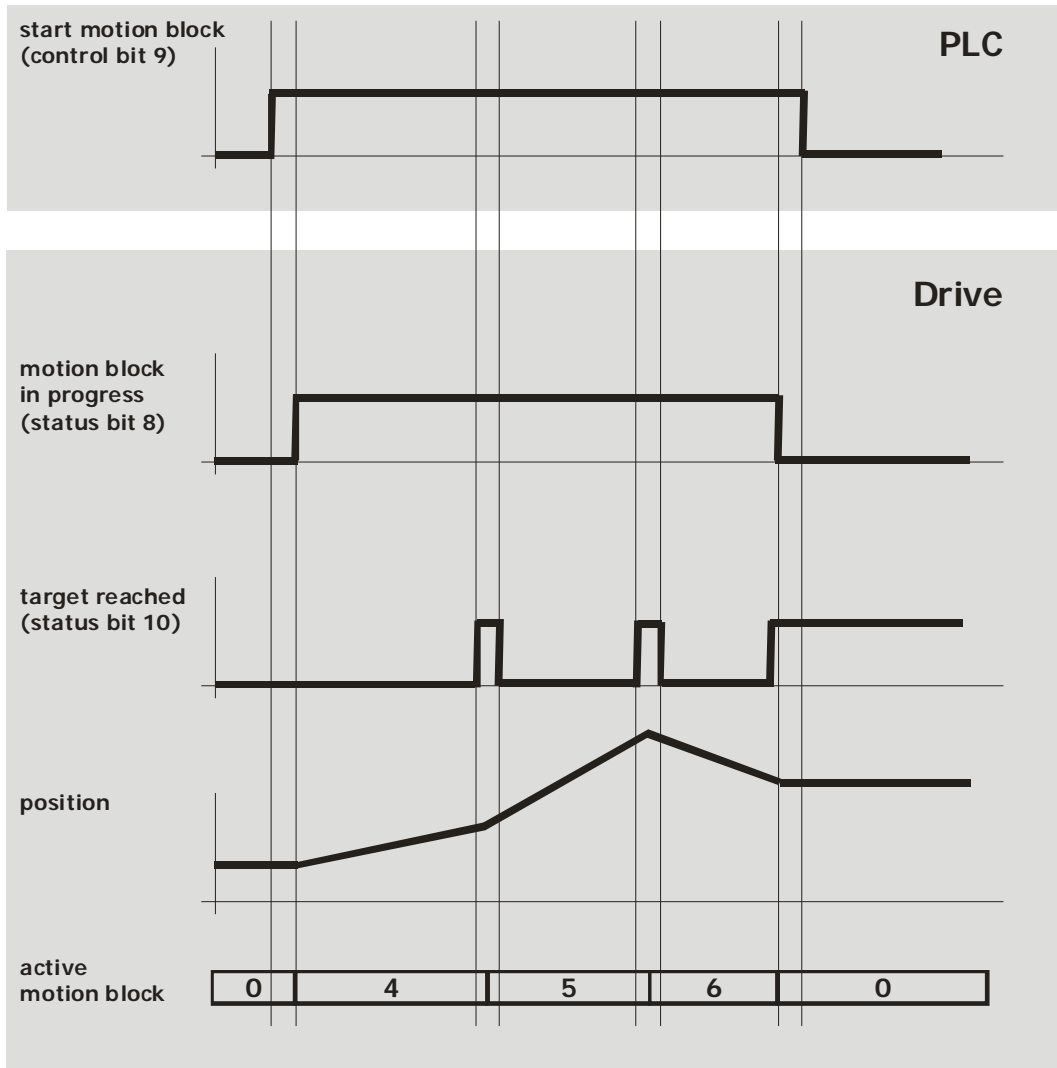
Setting *Halt* to 1 interrupts an actual processed motion block. The axle is stopped with the ramp defined by the actual motion block. When reaching velocity 0 *target reached* is set to 1. The drive remains in state *operation enabled*. Resetting *Halt* to 0 restarts processing of the interrupted motion block.

Examples:

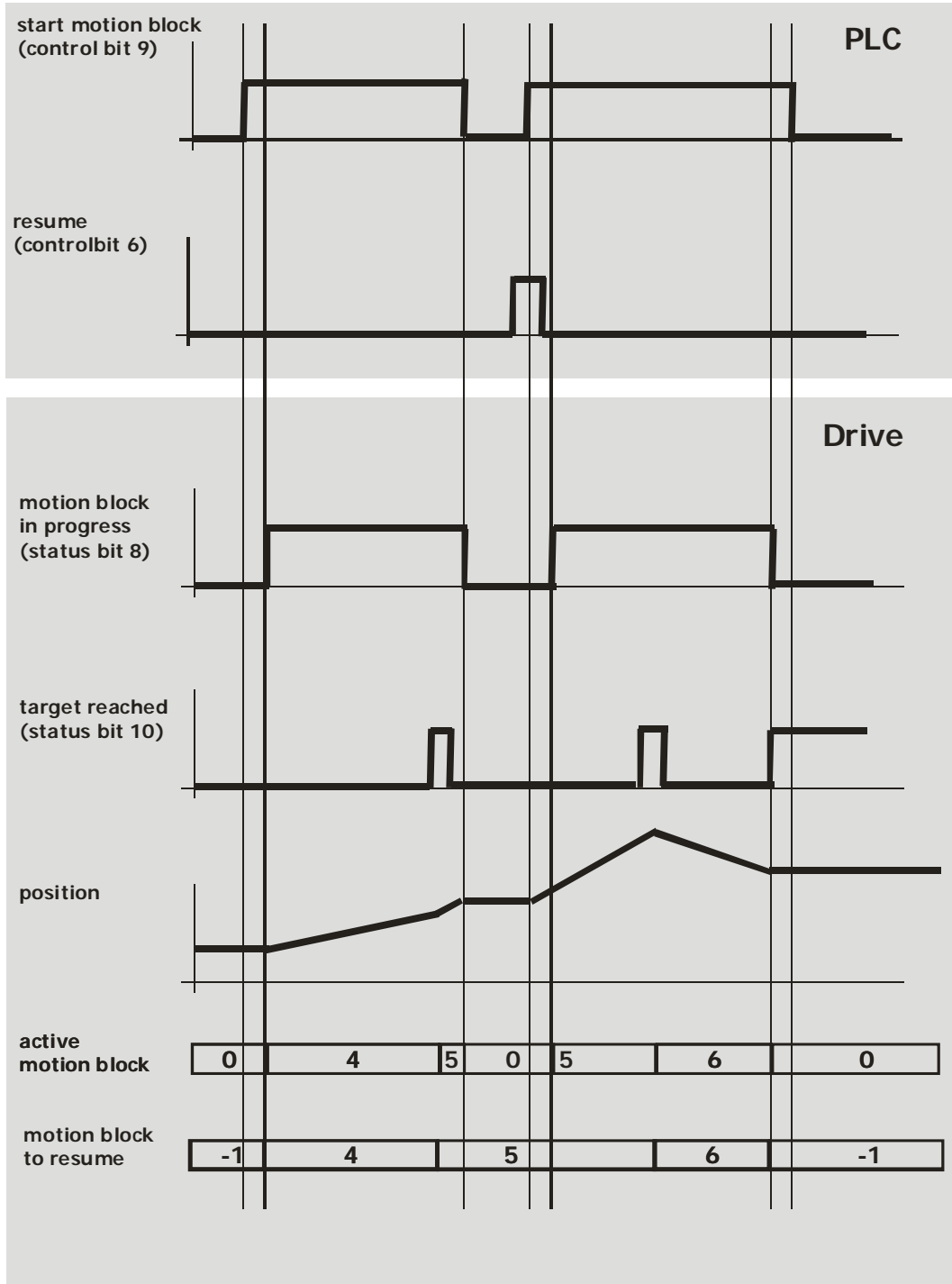
“single motion block” sequence mode (control bit 4) = 0
2 motion blocks 7 + 10



“ 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



12 Parameter list

The parameter list is structured according to the menu branches of the operating unit. For better clarity, the parameters have been marked with pictograms:

-  The parameter is available in the four data sets.
- The parameter value is set by the SET-UP routine.
- This parameter cannot be written when the frequency inverter is in operation.









12.1 Actual values



No.	Description	Unit	Display range	Chapter
Actual values of the frequency inverter				
228	Internal Reference Frequency	Hz	-1000.00 ... 1000.00	11.3.3
249	Active Data Set	-	1 ... 4	11
260	Current error	-	0 ... 0xFFFF	10.5.2
270	Warnings	-	0 ... 0xFFFF	13.1
282	Reference Bus Frequency	Hz	-1000.00 ... 1000.00	11.3.3
283	Reference Ramp Frequency	Hz	-1000.00 ... 1000.00	11.3.3
1290	Node State (NMT)	-	0 ... 127	9.7
1291	CAN State (physical layer)	-	0 ... 4	4

Note:

The parameters *Current error 260* and *Warnings 270* are only accessible via the manufacturer objects 0x2nnn. They cannot be accessed via the VPlus program or the KP500 control unit.

12.2 Parameters

No.	Description	Unit	Display range	Factory setting	Chapter
CAN bus					
276	CAN Interface	-	Selection	1 - CM-CAN	7
385	CAN Baud Rate	-	Selection	6 - 250 kBit/s	5
387	CAN Node Number	-	-1 ... 127	-1	6
388	CAN Error Behavior	-	Selection	1 - Error	8, 10.5.1
Rated motor parameters					
 373	No. of Pole Pairs	-	1 ... 24	2	10.5
Bus control					
392	State-transition 5	-	Selection	2 - Ramp	11.3.2
 412	Local/Remote	-	Selection	44 - Ctrl. Cont.+KP, Dir. Cont.+KP	11
Data set change-over					
414	Data set selection	-	0 ... 4	0	11
Frequency ramps					
 420	Acceleration (Clockwise)	Hz/s	0.01 ... 9999.99	5.00	10.5.9
 421	Deceleration (Clockwise)	Hz/s	0.01 ... 9999.99	5.00	10.5.10, 10.5.11
 422	Acceleration (Anticlockwise)	Hz/s	-0.01 ... 9999.99	-0.01	10.5.9
 423	Deceleration (Anticlockwise)	Hz/s	-0.01 ... 9999.99	-0.01	10.5.10, 10.5.11
 424	Emergency stop (Clockwise)	Hz/s	0.01 ... 9999.99	5.00	10.5.11, 11.3.1
 425	Emergency stop (Anticlockwise)	Hz/s	0.01 ... 9999.99	5.00	10.5.11, 11.3.1

No.	Description	Unit	Display range	Factory setting	Chapter
Frequency ramps					
434	Ramp Setpoint	-	Selection	3 - Internal + Line Setpoint	11.3.3
Digital outputs					
549	Max. Control Deviation	%	0.01 ... 20.00	5.00	11.1, 11.2
Stopping behavior					
 637	Switch-Off Threshold	%	0.0 ... 100.0	1.0	11.3.1, 11.3.2
 638	Holding Time	s	0.0 ... 200.0	1.0	11.3.1, 11.3.2
CANopen Mux/DeMux					
1420	CANopen Mux Input Index (write) ¹⁾	-	EEPROM: 0 ... 16 RAM: 17 ... 33	1	10.4.5
1421	CANopen Mux Input Index (read) ¹⁾	-	EEPROM: 0 ... 16 RAM: 17 ... 33	1	10.4.5
1422	CANopen Mux Input	-	Selection	7 - Off	10.4.5
1423	CANopen Percentage Actual Value Source	-	Selection	52 - Analog Input MFI1A	10.4.8

1)	Non-volatile (fixed parameterization)	Volatile
	0: All indices in EEPROM	17: All indices in RAM
	1...16: One Index of 1...16 in EEPROM	18...33: One Index of 1...16 in RAM

Note:

The setting "0" for *CANopen Mux Input Index (write)* **1420** changes all data in EEPROM and RAM.

Note:

The parameter *Data set selection* **414** is only accessible via the manufacturer objects 0x2nnn. It cannot be accessed via the VPlus program or the KP500 control unit.

13 Annex

13.1 Warning messages

The various control functions and methods as well as the hardware of the frequency inverter contain functions that continuously monitor the application. In addition to the messages documented in the manual, the following warning messages are activated by the CANopen communication module CM-CAN.

The warning messages are given via parameter *Warnings 270*, bit-coded according to the following scheme:

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 MF11A
12	0x1000	Warning Analog Input A2
13	0x2000	Warning System bus
14	0x4000	Warning Udc
15	0x8000	Warning V-Belt

Note:

The meaning of the individual warnings is described in detail in the operating instructions.

13.2 Fault 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 error		
Code		Meaning
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 (number of received bytes different to mapping)
	28	RxPDO3 length error (number of received bytes different to mapping)
F23	nn	Heartbeat failure – nn = node address of the failed subscriber (hex)

In addition to the fault messages stated, there are further fault messages used for internal purposes only and which are not listed here. If you receive any fault messages which are not listed, please contact us by phone.

14 Motion-control-interface for Profibus connection

The motion control interface is the interface between the communication system and the motion control system. With factory setting the motion control interface is connected to the CANopen system. This connection can be changed to the Profibus system.

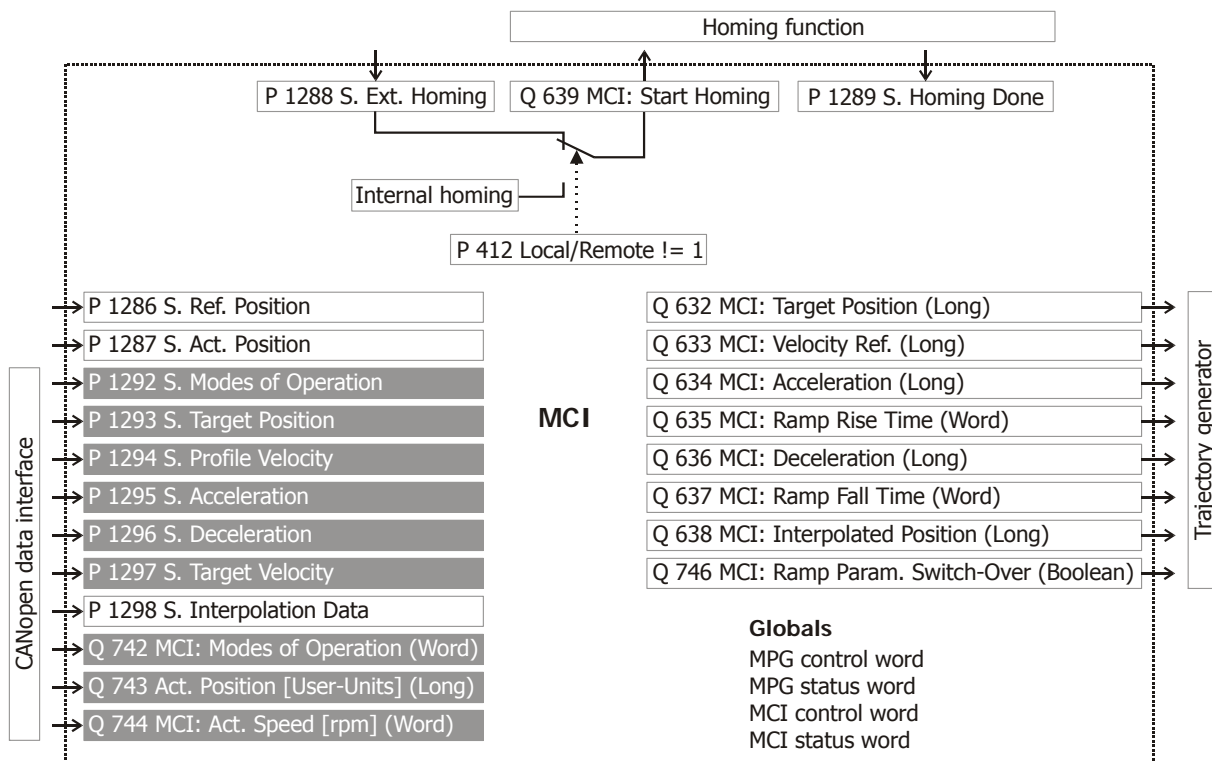
As with CANopen parameter *Local/Remote* **412** MUST be set to "1" = "control via state machine".

The motion control functions are available in configuration **x40** only.

This setting is necessary to control the inverter and motion control functions with the help of controlword (located in PZD1-OUT) and statusword (located in PZD1-IN). The functions and bit definitions are identical to the descriptions of CANopen.

Note:

With Profibus the mode of operation "interpolated position mode" can **NOT** be used. In motion control configurations Profibus objects PZD2-OUT (reference speed) and PZD2-IN (actual speed) have no function.



MCI: motion control interface,
 MPG: motion profile generator (trajectory generator),
 P: parameter,
 S: source,
 rpm: revolution per minute
 Only the grey marked parameters are necessary for controlling the motion control system by Profibus.

Input parameters **P xxxx** must be connected to Profibus OUT sources. These input parameters are direct accessible with parameter *Control Level 28* set to **4**.

Output sources **S xxx** must be connected to Profibus IN parameters (PZD-IN objects).



Bonfiglioli Worldwide & BEST Partners

- AUSTRALIA**
BONFIGLIOLI TRANSMISSION (Aust) Pty Ltd.
101, Plumpton Road, Glendenning NSW 2761, Australia
Locked Bag 1000 Plumpton NSW 2761
Tel. (+61) 2 8811 8000 - Fax (+61) 2 9675 6605
www.bonfiglioli.com.au - sales@bonfiglioli.com.au
- AUSTRIA *BEST***
MOLL MOTOR GmbH
Industriestrasse 8 - 2000 Stockerau
Tel. (+43) 2266 63421+DW - Fax (+43) 6342 180
www.mollmotor.at - office@mollmotor.at
- BELGIUM *BEST***
ESCO TRANSMISSION N.V./S.A.
Culliganlaan 3 - 1831 Machelem Diegem
Tel. (+32) 2 7176460 - Fax (+32) 2 7176461
www.esco-transmissions.be - info@esco-transmissions.be
- BRASIL *BEST***
ATI BRASIL
Rua Omlio Monteiro Soares, 260 - Vila Fanny - 81030-000
Tel. (+41) 334 2091 - Fax (+41) 332 8669
www.atibrasil.com.br - vendas@atibrasil.com.br
- CANADA**
BONFIGLIOLI CANADA INC.
2-7941 Janie Street - Concord, ONTARIO L4K 4L6
Tel. (+1) 905 7384466 - Fax (+1) 905 7389833
www.bonfigliolicanada.com - sales@bonfigliolicanada.com
- CHINA**
BONFIGLIOLI DRIVES (SHANGHAI) CO. LTD.
No. 8 Building, 98 Tian Ying Road
Qingpu District, Shanghai, PRC 201712
Tel. +86 21 69225500 - Fax +86 21 69225511
www.bonfiglioli.cn - bds@bonfiglioli.com.cn
- FRANCE**
BONFIGLIOLI TRANSMISSIONS S.A.
14 Rue Eugène Pottier BP 19
Zone Industrielle de Moimont II - 95670 Marly La Ville
Tel. (+33) 1 34474510 - Fax (+33) 1 34688800
www.bonfiglioli.fr - btf@bonfiglioli.fr
- GERMANY**
BONFIGLIOLI DEUTSCHLAND GmbH
Sperberweg 12 - 41468 Neuss
Tel. (+49) 02131 2988-0 - Fax (+49) 02131 2988-100
www.bonfiglioli.de - info@bonfiglioli.de
- GREAT BRITAIN**
BONFIGLIOLI UK Ltd
Industrial Equipment - Unit 3 Colemeadow Road
North Moons Moat - Redditch, Worcestershire B98 9PB
Tel. (+44) 1527 65022 - Fax (+44) 1527 61995
www.bonfiglioli-uk.com - uksales@bonfiglioli-uk.com
- Mobile Equipment
5 Grosvenor Grange - Woolston - Warrington, Cheshire WA1 4SF
Tel. (+44) 1925 852667 - Fax (+44) 1925 852668
www.bonfiglioli-uk.com - salesmobile@bonfiglioli-uk.com
- GREECE *BEST***
B.E.S.T. HELLAS S.A.
O.T. 48A T.O. 230 - C.P. 570 22 Industrial Area - Thessaloniki
Tel. (+30) 2310 796456 - Fax (+30) 2310 795903
www.bonfiglioli.gr - info@bonfiglioli.gr
- HOLLAND *BEST***
ELSTO AANDRIJFTECHNIEK
Loostenweg, 7 - 2215 TL Voorhout
Tel. (+31) 252 219 123 - Fax (+31) 252 231 660
www.elsto.nl - imfo@elsto.nl
- HUNGARY *BEST***
AGISYS AGITATORS & TRANSMISSIONS Ltd
2045 Törökbálint, Tó u.2. Hungary
Tel. +36 23 50 11 50 - Fax +36 23 50 11 59
www.agisys.hu - info@agisys.hu
- INDIA**
BONFIGLIOLI TRANSMISSIONS PVT Ltd.
PLOT AC7-AC11 Sidco Industrial Estate
Thirumudivakkam - Chennai 600 044
Tel. +91(0)44 24781035 / 24781036 / 24781037
Fax +91(0)44 24780091 / 24781904
www.bonfiglioli.co.in - bonfig@vsnl.com
- ITALY**
BONFIGLIOLI ITALIA S.p.A.
Via Sandro Pertini lotto 7b - 20080 Carpiano (Milano)
Tel. (+39) 02 985081 - Fax (+39) 02 985085817
www.bonfiglioli.it - customerservice.italia@bonfiglioli.it
- NEW ZEALAND *BEST***
SAECO BEARINGS TRANSMISSION
36 Hastie Avenue, Mangere
Po Box 22256, Otahuhu - Auckland
Tel. +64 9 634 7540 - Fax +64 9 634 7552
mark@saeco.co.nz
- POLAND *BEST***
POLPACK Sp. z o.o. - Ul. Chrobrego 135/137 - 87100 Torun
Tel. (+48) 56 6559235 - 6559236 - Fax (+48) 56 6559238
www.polpack.com.pl - polpack@polpack.com.pl
- PORTUGAL *BEST***
BT BONFITEC Equipamentos Industriais, Lda.
Largo do Colegio de Ermesinde, 70 - Formiga 4445-382 Ermesinde
Tel. (+351) 229759634/5/6 - Fax (+351) 229752211
www.bonfitec.pt - bonfitec@bonfitec.pt
- RUSSIA *BEST***
FAM
57, Maly prospekt, V.O. - 199048, St. Petersburg
Tel. +7 812 3319333 - Fax +7 812 3271454
www.fam-drive.ru - info@fam-drive.ru
- SPAIN**
TECNOTRANS BONFIGLIOLI S.A.
Pol. Ind. Zona Franca sector C, calle F, nº6 08040 Barcelona
Tel. (+34) 93 4478400 - Fax (+34) 93 3360402
www.tecnotrans.com - tecnotrans@tecnotrans.com
- SOUTH AFRICA**
BONFIGLIOLI POWER TRANSMISSION Pty Ltd.
55 Galaxy Avenue, Linbro Business Park - Sandton
Tel. (+27) 11 608 2030 OR - Fax (+27) 11 608 2631
www.bonfiglioli.co.za - bonfigsales@bonfiglioli.co.za
- SWEDEN**
BONFIGLIOLI SKANDINAVIEN AB
Koppargatan 8 - 234 35 Lomma, Sweden
Tel. (+46) 40418230 - Fax (+46) 40414508
www.bonfiglioli.se - info@bonfiglioli.se
- THAILAND *BEST***
K.P.T MACHINERY (1993) CO.LTD.
259/83 Soi Phiboovues, Sukhumvit 71 Rd. Phrakonong-nur,
Wattana, Bangkok 10110
Tel. 0066.2.3913030/7111998
Fax 0066.2.7112852/3811308/3814905
www.kpt-group.com - sales@kpt-group.com
- USA**
BONFIGLIOLI USA INC
3541 Hargrave Drive Hebron, Kentucky 41048
Tel.: (+1) 859 334 3333 - Fax: (+1) 859 334 8888
www.bonfiglioliusa.com
industrialsales@bonfiglioliusa.com
mobilesales@bonfiglioliusa.com
- VENEZUELA *BEST***
MAICA SOLUCIONES TECNICAS C.A.
Calle 3B - Edif. Comindu - Planta Baja - Local B
La Urbina - Caracas 1070
Tel. (+58) 212 2413570 / 2425268 / 2418263
Fax (+58) 212 2424552 - Tlx 24780 Maica V
maica1@cantv.net
- HEADQUARTERS**
BONFIGLIOLI RIDUTTORI S.p.A.
Via Giovanni XXIII, 7/A
40012 Lippo di Calderara di Reno
Bologna (ITALY)
Tel. (+39) 051 6473111
Fax (+39) 051 6473126
www.bonfiglioli.com
bonfiglioli@bonfiglioli.com
- SPARE PARTS BONFIGLIOLI**
B.R.T.
Via Castagnini, 2-4
Z.I. Bargellino - 40012
Calderara di Reno - Bologna (ITALY)
Tel. (+39) 051 727844
Fax (+39) 051 727066
www.brtbonfiglioliricambi.it
brt@bonfiglioli.com



ACTIVE Cube

The image shows a small, black, rectangular industrial device labeled 'ACTIVE Cube'. It is positioned in the center of a graphic that features a grid of squares on a light-colored surface. The background is filled with faint, overlapping binary code (0s and 1s) and several glowing, white, circular lines that create a sense of motion or data flow. The overall aesthetic is technical and futuristic.

www.bonfiglioli.com

 **BONFIGLIOLI**

The logo consists of a stylized, circular emblem with three interlocking shapes, followed by the brand name 'BONFIGLIOLI' in a bold, sans-serif font.