AKD™

EtherCAT Communication

Edition: H, May 2013

Valid for firmware version 1.11

Part Number 903-200005-00

Original Documentation

Keep all manuals as a product component during the life span of the product.
Pass all manuals to future users and owners of the product.

KOLLMORGEN

Because Motion Matters™

Record of Document Revisions

Revision	Remarks
...	Table with lifecycle information of this document see "Record of Document Revisions" (➜ p. 54)
F, 03/2012	Minor corrections
G, 11/2012	New chapter EEProm content
H, 05/2013	Fixed mapping, supported cyclic values, FBUS.PARAM05 added, several updates, formatting according to 82079

Trademarks

- AKD is a registered trademark of Kollmorgen Corporation
- EnDat is a registered trademark of Dr. Johannes Heidenhain GmbH
- EtherCAT is a registered trademark and patented technology, licensed by Beckhoff Automation GmbH
- Ethernet/IP is a registered trademark of ODVA, Inc.
- Ethernet/IP Communication Stack: copyright (c) 2009, Rockwell Automation
- sercos® is a registered trademark of sercos® international e.V.
- HIPERFACE is a registered trademark of Max Stegmann GmbH
- PROFINET is a registered trademark of PROFIBUS and PROFINET International (PI)
- SIMATIC is a registered trademark of SIEMENS AG
- Windows is a registered trademark of Microsoft Corporation

Current patents

- US Patent 5,162,798 (used in control card R/D)
- US Patent 5,646,496 (used in control card R/D and 1 Vp-p feedback interface)
- US Patent 6,118,241 (used in control card simple dynamic braking)
- US Patent 8,154,228 (Dynamic Braking For Electric Motors)
- US Patent 8,214,063 (Auto-tune of a Control System Based on Frequency Response)

Technical changes which improve the performance of the device may be made without prior notice!

Printed in the United States of America

1 Table of Contents

2 General

2.1 About this Manual

This manual, *AKD EtherCAT Communication*, describes the installation, setup, range of functions, and software protocol for the EtherCAT AKD product series. All AKD EtherCAT drives have built-in EtherCAT functionality; therefore an additional option card is not required.

A digital version of this manual (pdf format) is available on the DVD included with your drive. Manual updates can be downloaded from the Kollmorgen website.

Related documents for the AKD series include:

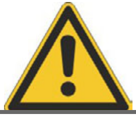
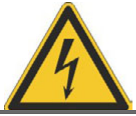
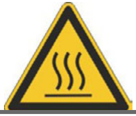
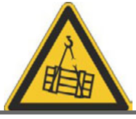
- *AKD Installation Manual*This manual provides instructions for installation and drive setup.
- *AKD User Guide*. This manual describes how to use your drive in common applications. It also provides tips for maximizing your system performance with the AKD. The *User Guide* includes the *Parameter and Command Reference Guide* which provides documentation for the parameters and commands used to program the AKD.
- *AKD CAN-BUS Communication*. This manual describes the CAN communication and delivers a lot of information for CAN over EtherCAT communication.
- *Accessories Manual*. This manual provides documentation for accessories like cables and regen resistors used with AKD. Regional versions of this manual exist.

Additionally, an EtherCAT XML file, entitled *AKD EtherCAT Device Description*, describes the drive SDO and PDO. This file is available on the Kollmorgen website (part of the firmware zip file):

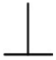
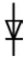
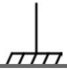
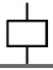
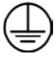
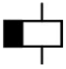
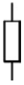
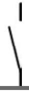
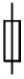
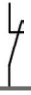
http://www.kollmorgen.com/en-us/products/drives/servo/akd/

2.2 Symbols Used

Warning Symbols

Symbol	Indication
⚠ DANGER	Indicates a hazardous situation which, if not avoided, will result in death or serious injury.
⚠ WARNING	Indicates a hazardous situation which, if not avoided, could result in death or serious injury.
⚠ CAUTION	Indicates a hazardous situation which, if not avoided, could result in minor or moderate injury.
NOTICE	This is not a safety symbol. Indicates situations which, if not avoided, could result in property damage.
NOTE	This is not a safety symbol. This symbol indicates important notes.
	Warning of a danger (general). The type of danger is specified by the text next to the symbol.
	Warning of danger from electricity and its effects.
	Warning of hot surfaces
	Warning of suspended loads.

Drawing symbols

Symbol	Description	Symbol	Description
	Signal ground		Diode
	Chassis ground		Relays
	Protective earth		Relays switch off delayed
	Resistor		Normal open contact
	Fuse		Normal closed contact

2.3 Abbreviations Used

Abbreviation	Meaning
AL	Application Layer: the protocol that directly used by the process entities.
Cat	Category – classification for cables that is also used in Ethernet.
DC	Distributed Clocks Mechanism to synchronize EtherCAT slaves and master
DL	Data Link(=Layer 2). EtherCAT uses Ethernet, which is standardized as IEEE 802.3.
FPGA	Field Programmable Gate Array
FTP	File Transfer Protocol
HW	Hardware
ICMP	Internet Control Message Protocol: Mechanisms for signaling IP errors.
IEC	International Electrotechnical Commission: The international standards
IEEE	Institute of Electrical and Electronics Engineers, Inc.
LLDP	Link Layer Discovery Protocol
MAC	Media Access Control
MII	Media Independent Interface: Standardized interface Ethernet controller <-> routing equipment.
MDI	Media Dependant Interface: Use of connector Pins and Signaling.
MDI-X	Media Dependant Interface (crossed): Use of connector Pins and Signaling with crossed lines.
OSI	Open System Interconnect
OUI	Organizationally Unique Identifier – the first 3 Bytes of an Ethernet-Address, that will be assign to companies or organizations and can be used for protocoll identifiers as well (e.g. LLDP)
PDI	Physical Device Interface: set of elements that allows access to ESC from the process side.
PDO	Process Data Object
PDU	Protocol Data Unit: Contains protocol information transferred from a protocol instance of transparent data to a subordinate level
PHY	Physical interface that converts data from the Ethernet controller to electric or optical signals.
PLL	Phase Locked Loop
PTP	Precision Time Protocol in accordance with IEEE 1588
RSTP	Rapid Spanning Tree Protocol
RT	Real-time, can be run in Ethernet controllers without special support.
RX	Receive
RXPDO	Receive PDO
SNMP	Simple Network Management Protocol
SPI	Serial Peripheral Interface
Src Addr	Source Address: Source address of a message.
STP	Shielded Twisted Pair
TCP	Transmission Control Protocol
TX	Transmit
TXPDO	Transmit PDO
UDP	User Datagram Protocol: Non-secure multicast/broadcast frame.
UTP	Unshielded Twisted Pair
ZA ECAT	Access mode EtherCAT
ZA Drive	Acces mode drive

3 Safety

3.1 Safety Instructions

3.2 You should pay attention to this

This section helps you to recognize and avoid dangers to people and objects.

Read the documentation!

Read the available documentation before installation and commissioning. Improper handling of the drive can cause harm to people or damage to property. The operator of systems using the AKD must require that all personnel who work with the drive read and understand the manual before using the drive.

Install the drive as described in the *Installation Manual*. The wiring for the analog setpoint input and the positioning interface, as shown in the wiring diagram in the *Installation Manual*, is not required.

Check Firmware Revision!

Check the Firmware Revision of the product. This number is the link between your product and the fieldbus manual. It must match the Firmware Revision on the manual's cover page.

Perform a risk assessment!

The manufacturer of the machine must generate a risk assessment for the machine, and take appropriate measures to ensure that unforeseen movements cannot cause injury or damage to any person or property. Additional requirements on specialist staff may also result from the risk assessment.

Observe remote-controlled machine behaviour!

Electronic equipment is basically not failure-proof. The user is responsible for ensuring that, in the event of a failure of the drive, the drive is set to a state that is safe for both machinery and personnel, for instance with the aid of a mechanical brake.

Drives with EtherCAT are remote-controlled machines. They can start to move at any time without previous warning. Take appropriate measures to ensure that the operating and service personnel is aware of this danger.

Implement appropriate protective measures to ensure that any unintended start-up of the machines cannot result in dangerous situations for personnel or machinery. Software limit-switches are not a substitute for the hardware limit-switches in the machine.

Specialist staff required!

Only properly qualified personnel are permitted to perform such tasks as setup and programming. Qualified specialist staff are persons who are familiar with the installation, setup and programming of drives and who bring their relevant minimum qualifications to bear on their duties:

- Installation: only by electrically qualified personnel.
- Setup : only by qualified personnel with extensive knowledge of electrical engineering and drive technology
- Programming: Software developers, project-planners

The qualified personnel must know and observe ISO 12100 / IEC 60364 / IEC 60664 and national accident prevention regulations.

Observe electrostatically sensitive components!

The drives contain electrostatically sensitive components which may be damaged by incorrect handling. Electrostatically discharge your body before touching the drive. Avoid contact with highly insulating materials (artificial fabrics, plastic film etc.). Place the drive on a conductive surface.

Hot surface!

Drives may have hot surfaces during operation. The heat sink can reach temperatures above 80°C. Risk of minor burns! Measure the temperature, and wait until the heat sink has cooled down below 40 °C before touching it.

Earthing!

It is vital that you ensure that the drive is safely earthed to the PE (protective earth) busbar in the switch cabinet. Risk of electric shock. Without low-resistance earthing no personal protection can be guaranteed and there is a risk of death from electric shock.

High voltages!

Wait at least 7 minutes after disconnecting the drive from the main supply power before touching potentially live sections of the equipment (such as contacts) or removing any connections.

Capacitors can have dangerous voltages present up to seven minutes after switching off the supply power. Always measure the voltage in the DC bus link and wait until the voltage is below 40 V before handling components.

Never modify the drive!

It is not allowed to modify the drive without permission by the manufacturer. Opening the housing causes loss of warranty.

3.3 Use as directed

Drives are components that are built into electrical plants or machines and can only be operated as integral components of these plants or machines. The manufacturer of the machine used with a drive must generate a risk assessment for the machine and take appropriate measures to ensure that unforeseen movements cannot cause personnel injury or property damage.

- Observe the chapters "Use as directed” and "Prohibited use" in the *AKD Installation Manual*.
- The EtherCAT interface serves only for the connection of the *AKD* to a master with EtherCAT connectivity.

3.4 Prohibited use

Other use than that described in chapter “Use as directed” is not intended and can lead to personnel injuries and equipment damage. The drive may not be used with a machine that does not comply with appropriate national directives or standards. The use of the drive in the following environments is also prohibited:

- potentially explosive areas
- environments with corrosive and/or electrically conductive acids, alkaline solutions, oils, vapors, dusts
- ships or offshore applications

The connectors X5 and X6 of the AKD EtherCAT drive may not be used for any ethernet protocol except EtherCAT (CoE, Can over EtherCAT).

4 Installation and Setup

4.1 Important Instructions

⚠ **DANGER**

Never undo any electrical connections to the drive while it is live. There is a danger of electrical arcing with damage to contacts and serious personal injury.
Wait at least seven minutes after disconnecting the drive from the main supply power before touching potentially live sections of the equipment (e.g. contacts) or undoing any connections.
To be sure, measure the voltage in the DC Bus link and wait until it has fallen below 40 V.

⚠ **WARNING**

Electronic equipment is basically not failure-proof. The user is responsible for ensuring that, in the event of a failure of the drive, the drive is set to a state that is safe for both machinery and personnel, for instance with the aid of a mechanical brake.

Drives with EtherCAT are remote-controlled machines. They can start to move at any time without previous warning. Take appropriate measures to ensure that the operating and service personnel is aware of this danger.

Implement appropriate protective measures to ensure that any unintended start-up of the machines cannot result in dangerous situations for personnel or machinery. Software limit-switches are not a substitute for the hardware limit-switches in the machine.

NOTICE

Install the drive as described in the *Installation Manual*. The wiring for the analog setpoint input and the positioning interface, as shown in the wiring diagram in the *Installation Manual*, is not required. Never break any of the electrical connections to the drive while it is live. This action can result in destruction of the electronics.

NOTICE

The drive's status must be monitored by the PLC to acknowledge critical situations. Wire the FAULT contact in series into the emergency stop circuit of the installation. The emergency stop circuit must operate the supply contactor.

NOTE

It is permissible to use the setup software to alter the settings of the drive. Any other alterations will invalidate the warranty. Because of the internal representation of the position-control parameters, the position controller can only be operated if the final limit speed of the drive does not exceed:

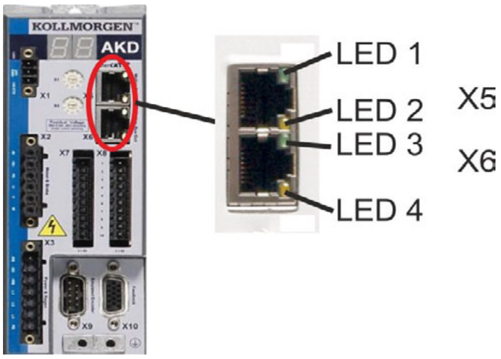
rotary	linear
at sinusoidal² commutation: 7500 rpm	at sinusoidal² commutation: 4 m/s
at trapezoidal commutation: 12000 rpm.	at trapezoidal commutation: 6.25 m/s

NOTE

All the data on resolution, step size, positioning accuracy etc. refer to calculatory values. Non-linearities in the mechanism (backlash, flexing, etc.) are not taken into account. If the final limit speed of the motor must be altered, then all the parameters that were previously entered for position control and motion blocks must be adapted.

4.2 EtherCAT Onboard

Connection to the EtherCAT Network via X5 (in port) and X6 (out port).



4.2.1 LED functions

The communication status is indicated by the built-in LEDs.

Connector	LED#	Name	Function

X5	LED1	IN port Link	ON = active
			OFF= not active

	LED2	RUN	ON = running
			OFF = not running

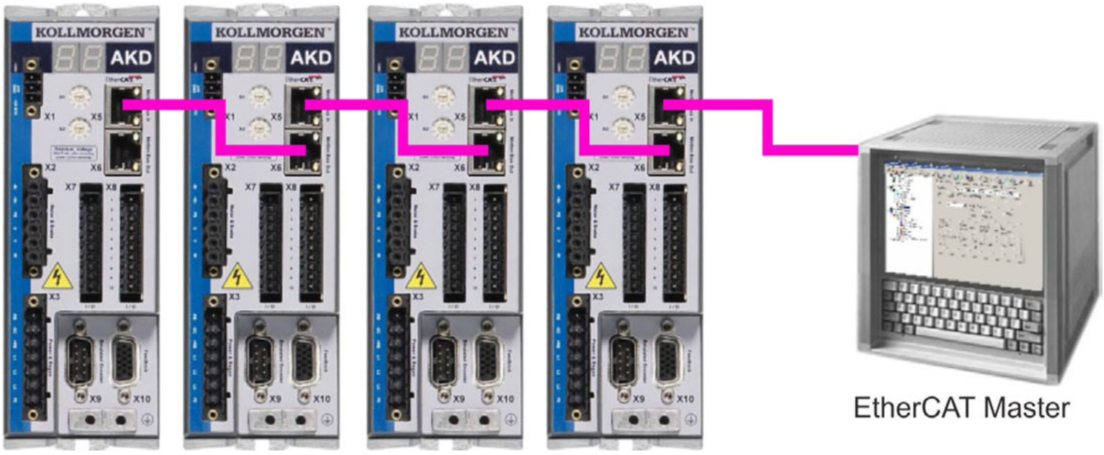
X6	LED3	OUT port Link	ON = active
			OFF = not active

	LED4	-	-

4.2.2 Connection technology

You can connect to the EtherCAT network using RJ-45 connectors.

4.2.3 Network Connection Example



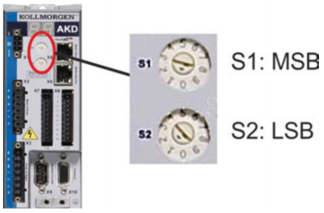
4.3 EtherCAT activation with AKD-CC models

AKD-CC drive models are Drives, which support EtherCAT and CAN fieldbus types within one common software. CC drive models are delivered with EtherCAT set active. If you must change a drive from CANopen to EtherCAT, the DRV.TYPE parameter must be changed

1. by software: connect the PC to the AKD and change the parameter DRV.TYPE in the WorkBench terminal screen (see DRV.TYPE parameter documentation) or
2. by hardware: with the rotary switches S1 & S2 at the front and the button B1 on the top side of the Drive.

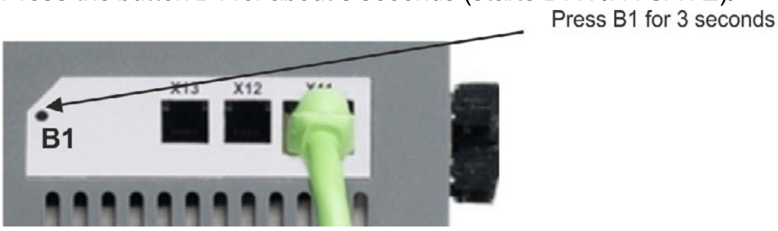
The following steps are needed for changing the fieldbus type from CAN to EtherCAT with the rotary switches.

1. Set the rotary switches on the front side of the AKD to the value of 89.



Set S1 to 8 and S2 to 9

2. Press the button B1 for about 3 seconds (starts DRV.NVSAVE).



Press B1 for 3 seconds

The display shows **En** during the process of changing DRV.TYPE to EtherCAT.
Do not switch off the 24[V] power supply while the seven segment shows En!

3. Wait until the display returns to the original state, now the drive is prepared for EtherCAT.
4. Power cycle the drive by switching the 24 V power supply **off** and then **on** again.

NOTE

The seven segment display shows Er (Error) in case that the DRV.TYPE instruction failed. In this case please power cycle the drive and contact the Kollmorgen customer support for further help.

4.4 Guide to Setup

NOTICE

Only professional personnel with extensive knowledge of control and drive technology are allowed to setup the drive.

⚠ CAUTION

Drives with EtherCAT are remote-controlled machines. They can start to move at any time without previous warning. Take appropriate measures to ensure that the operating and service personnel is aware of this danger.

Implement appropriate protective measures to ensure that any unintended start-up of the machines cannot result in dangerous situations for personnel or machinery. Software limit-switches are not a substitute for the hardware limit-switches in the machine.

1. Check assembly/installation. Check that all the safety instructions in the product manual for the drive and this manual have been observed and implemented. Check the setting for the station address and baud rate.
2. Connect PC,start WorkBench. Use the setup software WorkBench to set the parameters for the drive.
3. Setup basic functions. Start up the basic functions of the drive and optimize the current, speed and position controllers. This section of the setup is described in the in the online help of the setup software.
4. Save parameters. When the parameters have been optimized, save them in the drive.

4.5 Setup via TwinCAT NC/PTP System Manager

Before you set up the drive, make sure the following have been completed:

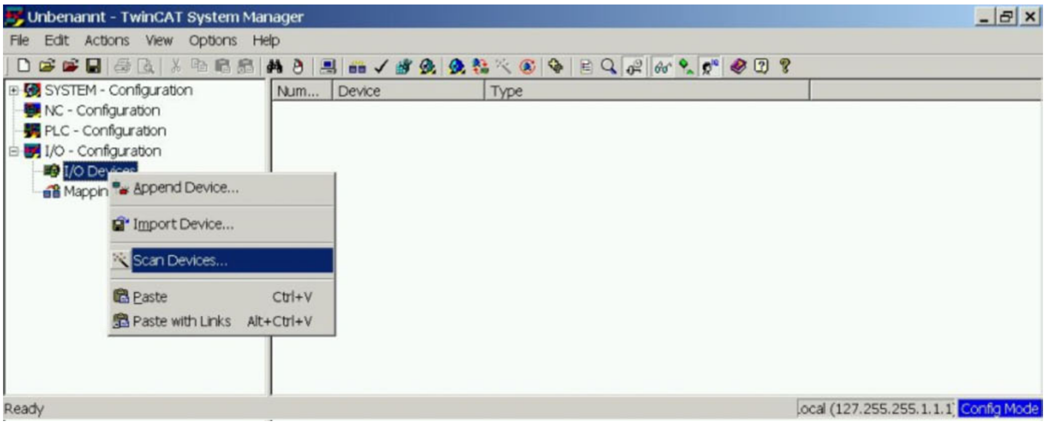
- The AKD is configured with WorkBench and the servomotor is able to move
- A correctly configured EtherCAT card is present in the master.
- TwinCAT software from Beckhoff (NC/PTP-Mode setup) is installed. Install first the TwinCAT System Manager, restart your PC, then install the option package NC/PTP-Mode.
- The XML description of the drive is available (the XML file on the DVD or on the Kollmorgen website).
- An AKD EtherCAT slave is connected to the EtherCAT master PC.
- The TwinCAT system manager resides in Config-Mode. The current mode of the system manager is displayed of the bottom right side of the TwinCAT main-screen window.

Copy the XML description of the drive to the TwinCAT system (usually to the folder c:\TwinCAT\IO\EtherCAT) and restart the TwinCAT system since TwinCAT analyzes all device description files during start-up.

The following example explains the automatic EtherCAT network setup. The network setup can also be done manually; please refer to the TwinCAT manual for more details.

4.5.1 Scan devices

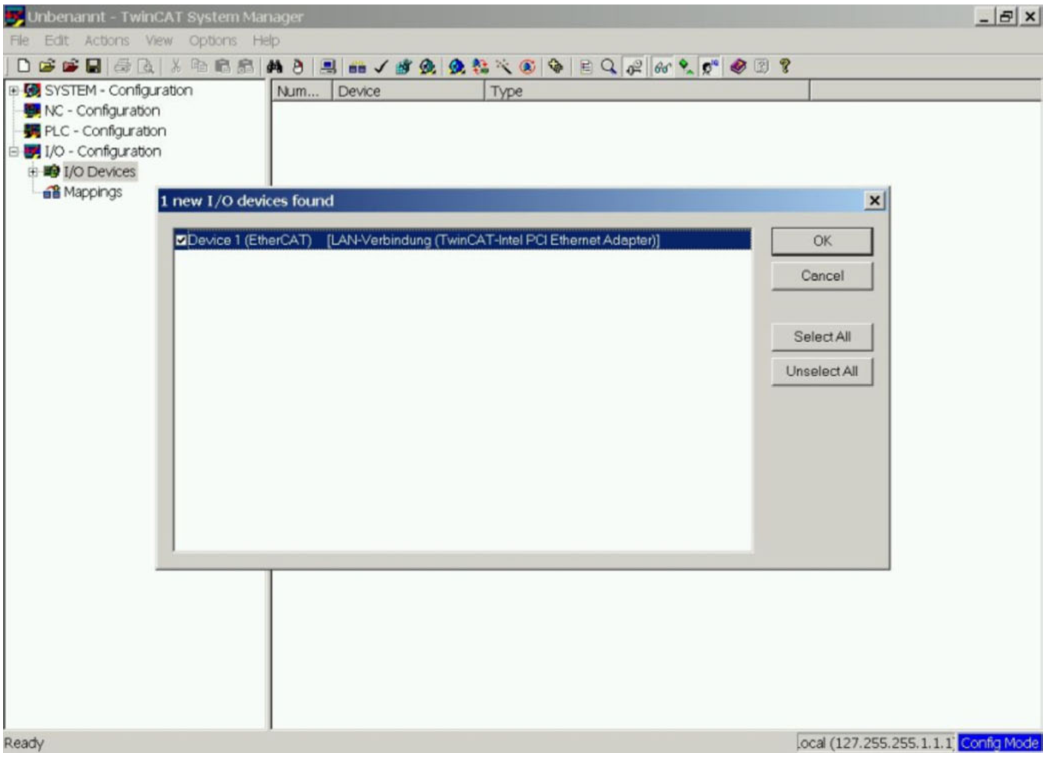
First ensure that the EtherCAT master is physically connected to the EtherCAT AKD. Create a new (empty) project. Right click I/O-Devices and scan for the devices. An example is included in the EtherCAT network card, which is plugged into the PC.



A pop-up window informs you that not all devices can be detected by the TwinCAT software. Click **OK** to continue.

4.5.2 Select the device

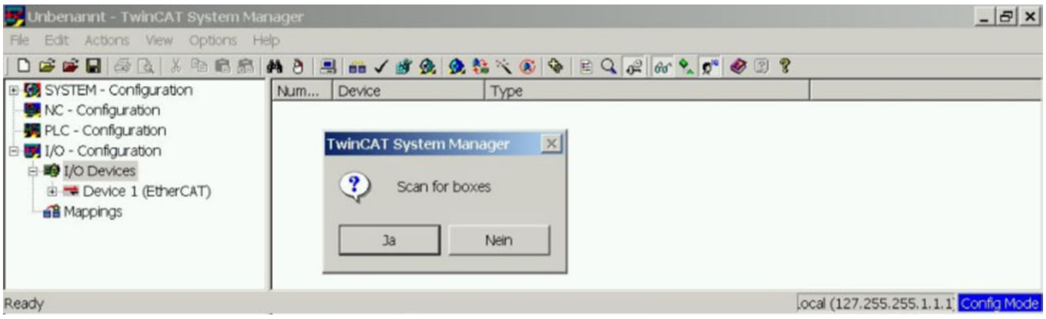
TwinCAT must be able to find the EtherCAT network card. An EtherCAT slave must be connected to the network card; otherwise TwinCAT will find a real-time EtherNET card instead of the EtherCAT card. Press the **OK** button.



☑ Device 1 (EtherCAT) [LAN-Verbindung (TwinCAT-Intel PCI Ethernet Adapter)]

4.5.3 Scan for boxes

Click **Yes** to allow TwinCat to scan for boxes. A *box* is an alias for a slave device and is always used in Beckhoff software products.



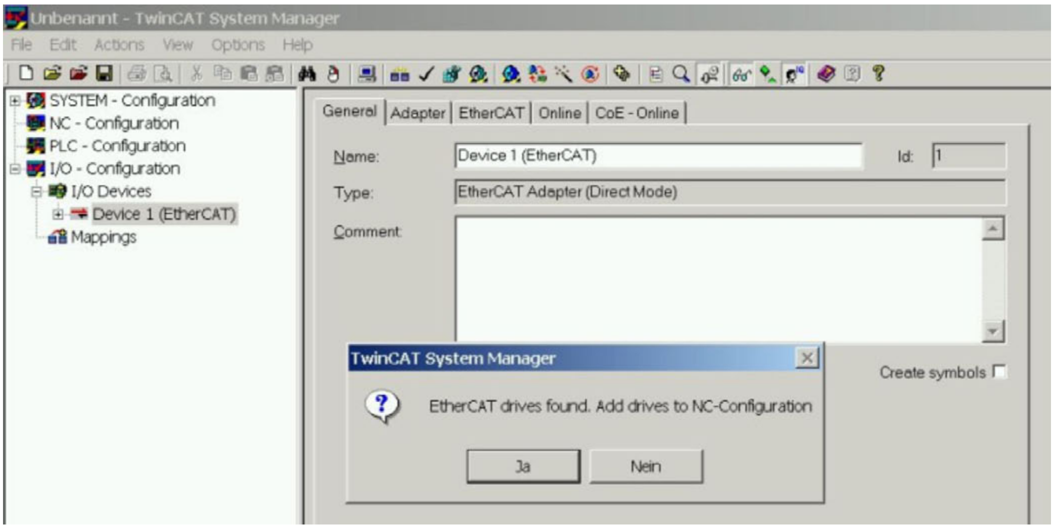
TwinCAT System Manager

Scan for boxes

local (127.255.255.1.1.1

4.5.4 Add Slaves to NC tasks

TwinCAT should now have identified the AKD according to the Device Description file. TwinCAT next asks if the slaves should be connected to NC tasks. Click **Yes** to continue. An NC task can, for example, contain a PLC program, which can be programmed by the user.



Name: Device 1 (EtherCAT) Id: 1

Type: EtherCAT Adapter (Direct Mode)

Comment:

Create symbols □

TwinCAT System Manager

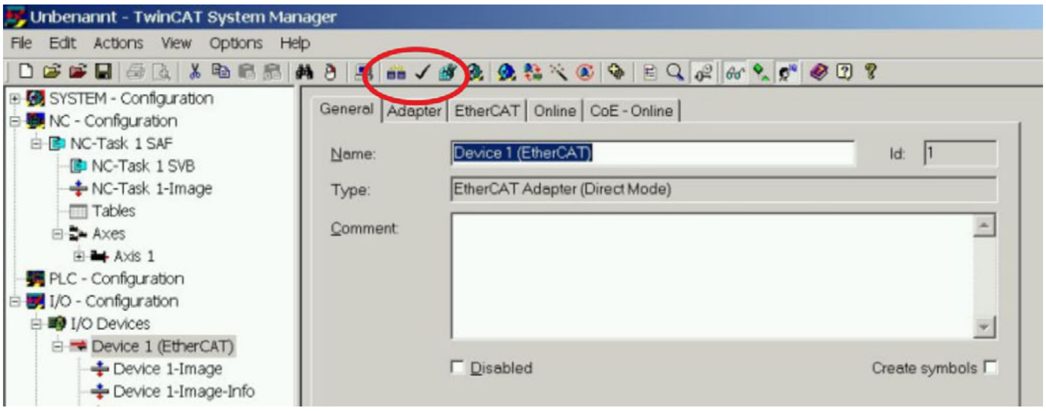
EtherCAT drives found. Add drives to NC-Configuration

4.5.5 Enable the network configuration

Confirm that the AKD appears in the device tree. Next, enable the network configuration

Press first the button in order to generate the mappings, afterwards press the ✓ button in order to let TwinCAT check the configuration and use finally the button in order to step into run-mode.

Confirm afterwards that TwinCAT is allowed to jump into run-mode.



Name: Device 1 (EtherCAT) Id: 1

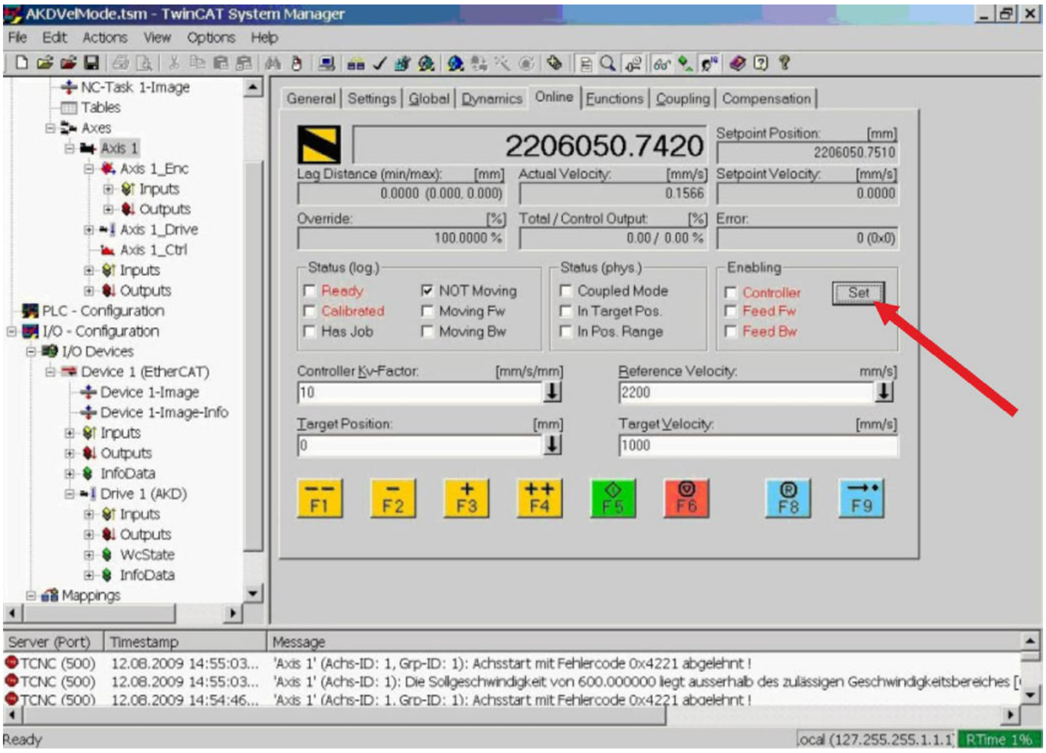
Type: EtherCAT Adapter (Direct Mode)

Comment:

□ Disabled Create symbols □

4.5.6 Enable the axis and move the axis

The Axis can be enabled by a mouse-click on the Set button within the Online window inside of each Axis, see also the next picture.



2206050.7420

Setpoint Position: [mm] 2206050.7510

Lag Distance (min/max): [mm] 0.0000 (0.000, 0.000)

Actual Velocity: [mm/s] 0.1566

Setpoint Velocity: [mm/s] 0.0000

Override: [%] 100.0000 %

Total / Control Output: [%] 0.00 / 0.00 %

Error: 0 (0x0)

Status (log.)
- ☐ Ready
- ☐ Calibrated
- ☐ Has Job
- ☑ NOT Moving
- ☐ Moving Fw
- ☐ Moving Bw

Status (phys.)
- ☐ Coupled Mode
- ☐ In Target Pos.
- ☐ In Pos. Range

Enabling
- ☐ Controller
- ☐ Feed Fw
- ☐ Feed Bw

Controller Kv-Factor: [mm/s/mm] 10

Reference Velocity: mm/s] 2200

Target Position: [mm] 0

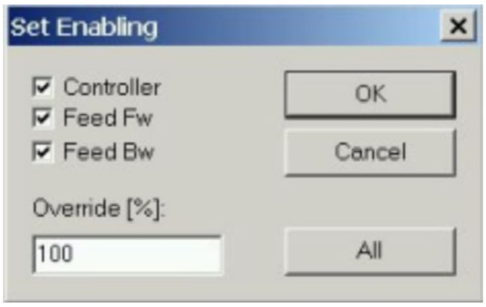
Target Velocity: [mm/s] 1000

Server (Port)	Timestamp	Message
TCNC (500)	12.08.2009 14:55:03...	'Axis 1' (Achs-ID: 1, Grp-ID: 1): Achsstart mit Fehlercode 0x4221 abgelehnt !
TCNC (500)	12.08.2009 14:55:03...	'Axis 1' (Achs-ID: 1): Die Sollgeschwindigkeit von 600.000000 liegt ausserhalb des zulässigen Geschwindigkeitsbereiches [
TCNC (500)	12.08.2009 14:54:46...	'Axis 1' (Achs-ID: 1, Grp-ID: 1): Achsstart mit Fehlercode 0x4221 abgelehnt !

Ready ocal (127.255.255.1.1.1) RTime 1%

Afterwards a pop-up window appears.

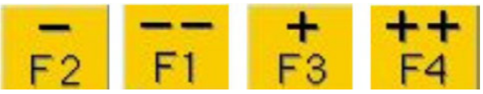
The following setting enables the drive and allows command values in both directions.



- ☑ Controller
- ☑ Feed Fw
- ☑ Feed Bw

Override [%]: 100

Afterwards the motor should move in positive or negative direction as soon as the clicks on the following yellow buttons within the Online window:



4.6 Setup WorkBench over TwinCAT

This chapter describes a quick start guide for a user to be able to setup a WorkBench over TwinCAT system and be able to make a motor spin under that system.

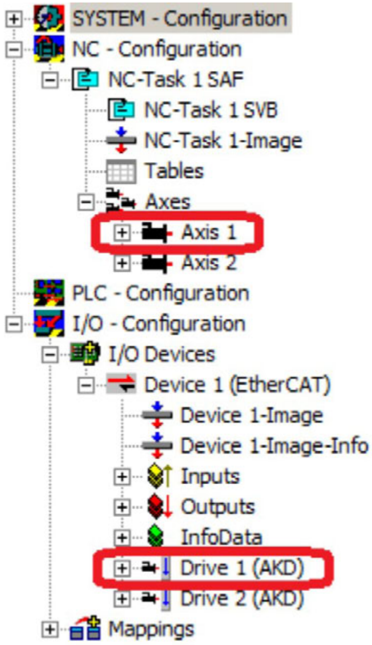
This chapter does not give any specific details on TwinCAT system or WorkBench alone but is giving guidelines and information on how TwinCAT master and WorkBench works together.

Main steps in configuring a WorkBench over TwinCAT system are:

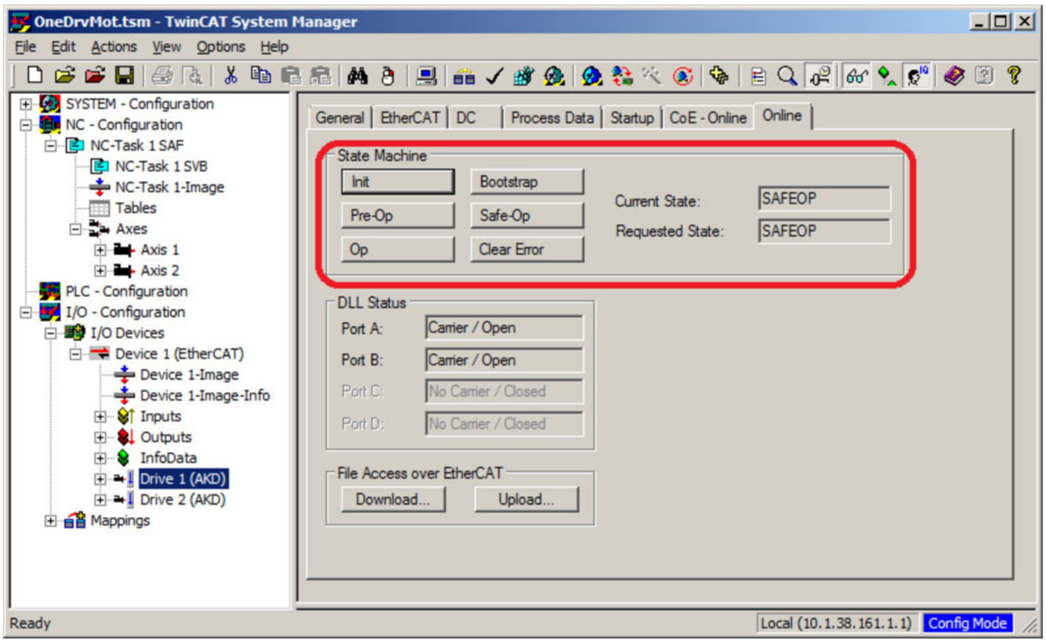
- TwinCAT and WorkBench configuration
- Connecting to a drive using WorkBench
- Configuring and enabling a drive

4.6.1 TwinCAT and WorkBenchconfiguration

The EtherCAT network must be setup and managed using TwinCAT System Manager. To be able to connect to a drive and enable it, the drive must be loaded under the I/O Devices node in TwinCAT System Manager and axis must be added to NC - Configuration as shown ➜ p. 16 "Setup via TwinCAT NC/PTP System Manager " in the EtherCAT Manual.



In order to connect to the drives using WorkBench, the drives must be either in Pre-Op, Safe-Op or Op state. State machine for a drive can be accessed from the Online tab for the corresponding drive under the I/O Configuration → I/O Devices → Device [x] → Drive [x] node (see screenshot below).



Current State: SAFEOP
Requested State: SAFEOP

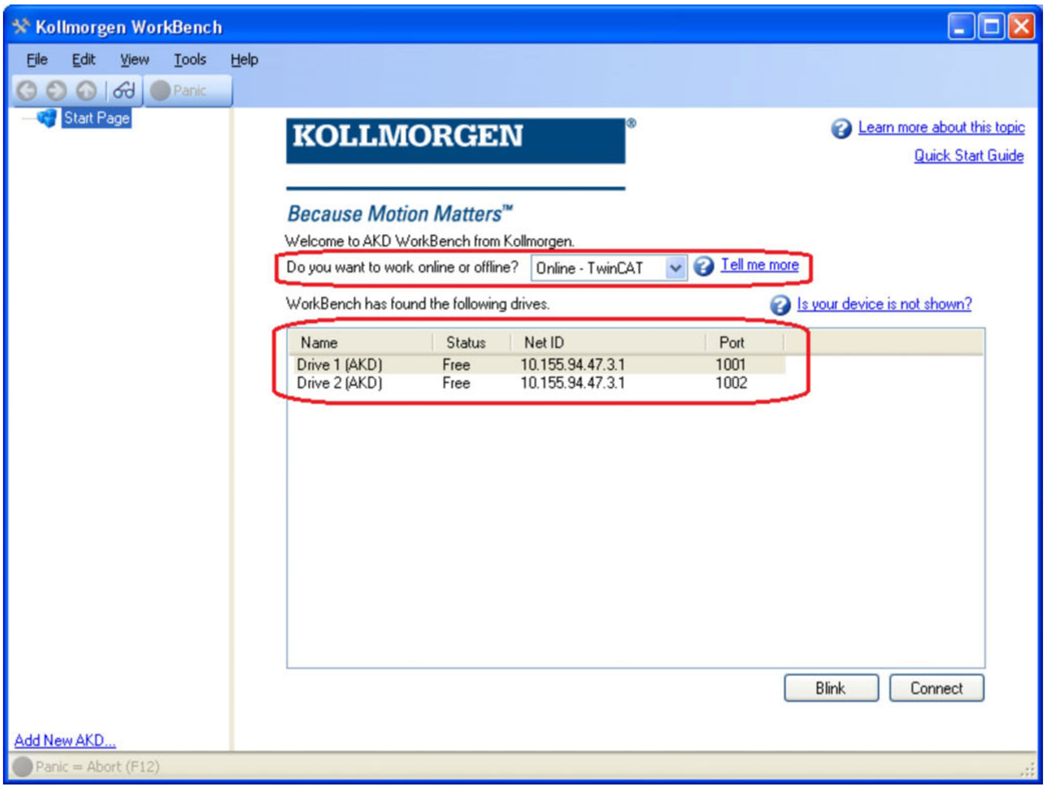
Port A: Carrier / Open
Port B: Carrier / Open
Port C: No Carrier / Closed
Port D: No Carrier / Closed

Local (10.1.38.161.1.1)

Installation process for WorkBench is the same process as normal, except that it must be installed on the same machine as TwinCAT. Communication to the drive is done thru TwinCAT master and it's not possible to connect WorkBench to the master remotely.

4.6.2 Connecting to a drive using WorkBench

In order to connect to a drive, a TwinCAT device must be added in WorkBench. The start page of WorkBench can be used to do this. First, the type of drive (Online - TwinCAT) must be specified. Then, a list of available drives will be provided.



Welcome to AKD WorkBench from Kollmorgen.

Do you want to work online or offline? Online - TwinCAT

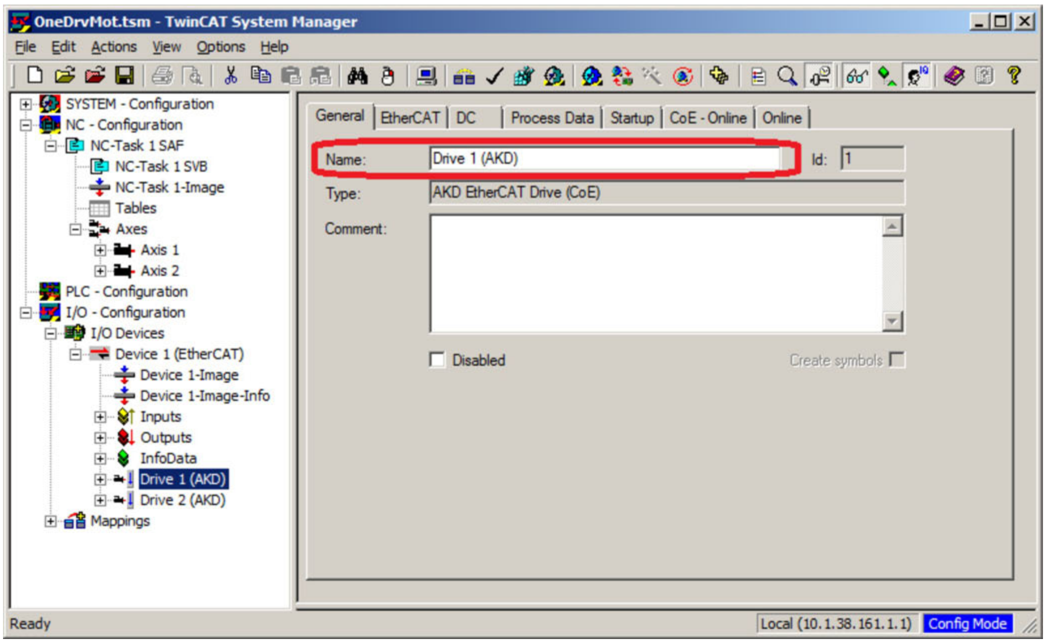
WorkBench has found the following drives.

Name	Status	Net ID	Port
Drive 1 (AKD)	Free	10.155.94.47.3.1	1001
Drive 2 (AKD)	Free	10.155.94.47.3.1	1002

The information provided for a drive are it's name, status, Net ID and Port number. After selecting a drive from the list, clicking on the "Connect" button will create a device in the left frame of WorkBenchand connect the device.

The name, Net ID and port number are information comming from the TwinCAT master configuration file (the name may be different than the drive name returned by the *DRV.NAME* command). While the status is an indicator that tells if there is already a device created within WorkBench which is already connected to that particular drive.

Using TwinCAT System Manager, the drive name and port number can be found in the General and EtherCAT tab respectively for the corresponding drive under the I/O Configuration → I/O Devices → Device [x] → Drive [x] node.



Name: Drive 1 (AKD) Id: 1

Type: AKD EtherCAT Drive (CoE)

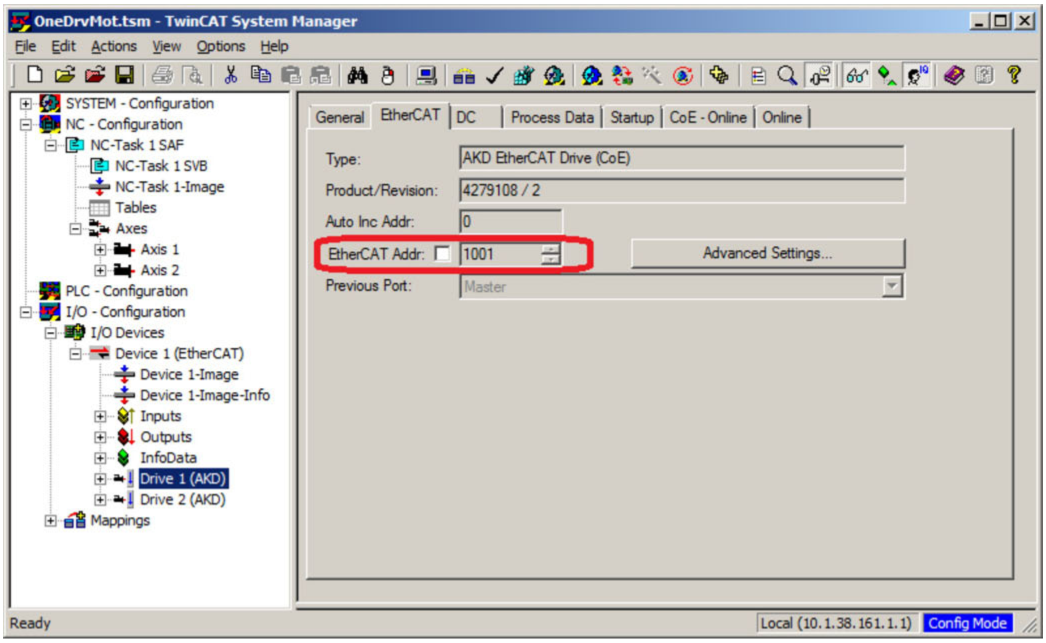
Comment:

□ Disabled

Create symbols □

Ready

Local (10.1.38.161.1.1) Config Mode



Type: AKD EtherCAT Drive (CoE)

Product/Revision: 4279108 / 2

Auto Inc Addr: 0

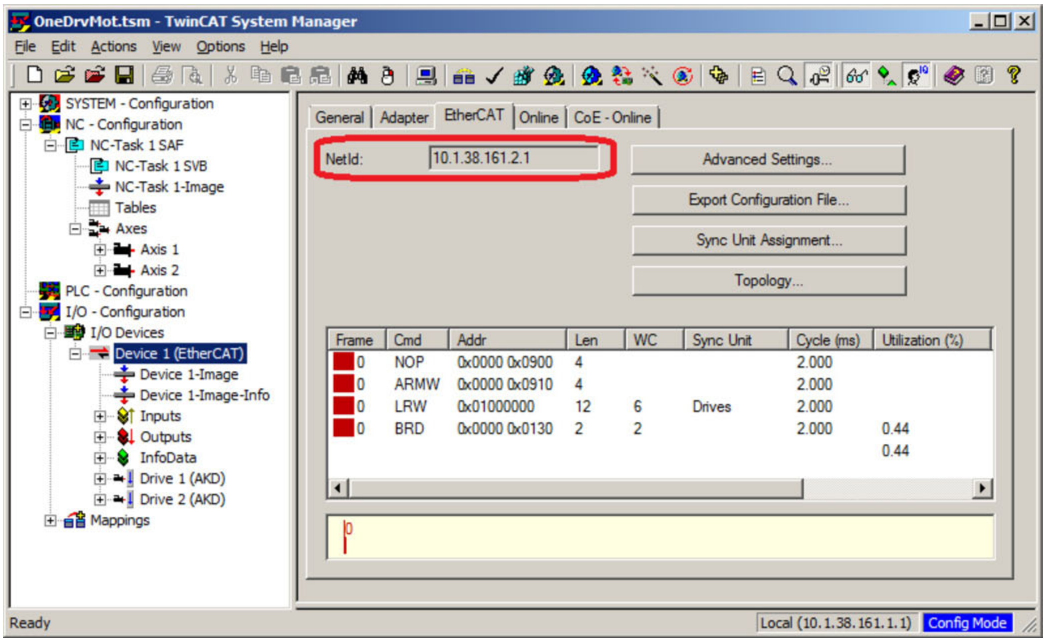
EtherCAT Addr: □ 1001

Previous Port: Master

Ready

Local (10.1.38.161.1.1) Config Mode

The Net ID can be found in the EtherCAT tab in the I/O Configuration → I/O Devices → Device [x] node.



NetId: 10.1.38.161.2.1

Frame	Cmd	Addr	Len	WC	Sync Unit	Cycle (ms)	Utilization (%)
🟥 0	NOP	0x0000 0x0900	4			2.000	
🟥 0	ARMW	0x0000 0x0910	4			2.000	
🟥 0	LRW	0x01000000	12	6	Drives	2.000	
🟥 0	BRD	0x0000 0x0130	2	2		2.000	0.44
							0.44

0

Ready | Local (10.1.38.161.1.1) | Config Mode

It is important to understand that these information are comming from the TwinCAT master and it's configuration file but not from the drive itself. Thus, if the TwinCAT configuration is not reflecting the actual network configuration,you may have a drive listed in WorkBench which is not be powered up or even connected in the EtherCAT network, or you have a drive powered up and connected to the TwinCAT network but not shown in the WorkBench list.

4.6.3 Configuring and enabling a drive

Once connected with WorkBench, a drive can be configured using all normal functionnalities of WorkBench.

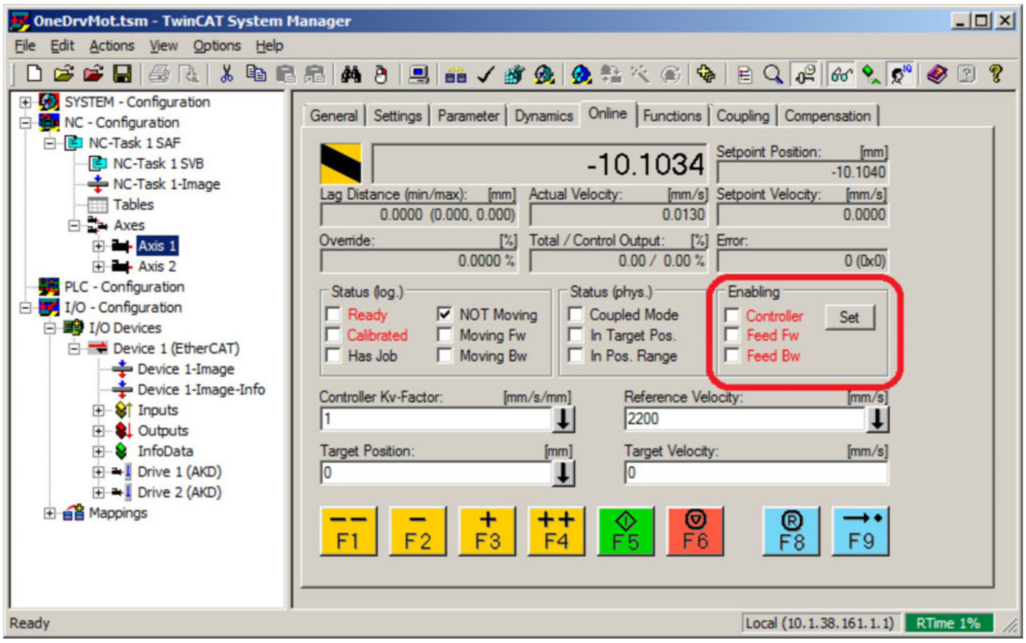
The only operation that is not possible to do using WorkBench over TwinCAT is the download of a new firmware in the drive. Downloading a new firmware in the drive must be performed using File over EtherCAT (FoE) feature of TwinCAT server.

NOTICE

If the cyclic communication of the TwinCAT master is enabled, it is possible that some commands sent by WorkBench using the ASCII channel are overwirtten by the TwinCAT master. Typically, the drive enable command will have no effect if sent from WorkBench because the control word is usually mapped.

Using TwinCAT, enabling the drive can be done with the following procedure:

1. Under NC Configuration → Axes → Axis [x] node, choose the Online tab.
2. Press the Set button within the Enabling section.



-10.1034

Setpoint Position: [mm] -10.1040

Lag Distance (min/max): [mm] 0.0000 (0.000, 0.000)

Actual Velocity: [mm/s] 0.0130

Setpoint Velocity: [mm/s] 0.0000

Override: [%] 0.0000 %

Total / Control Output: [%] 0.00 / 0.00 %

Error: 0 (0x0)

Status (log.): □ Ready ☑ NOT Moving □ Calibrated □ Moving Fw □ Has Job □ Moving Bw

Status (phys.): □ Coupled Mode □ In Target Pos. □ In Pos. Range

Enabling: □ Controller □ Feed Fw □ Feed Bw Set

Controller Kv-Factor: [mm/s/mm] 1

Reference Velocity: [mm/s] 2200

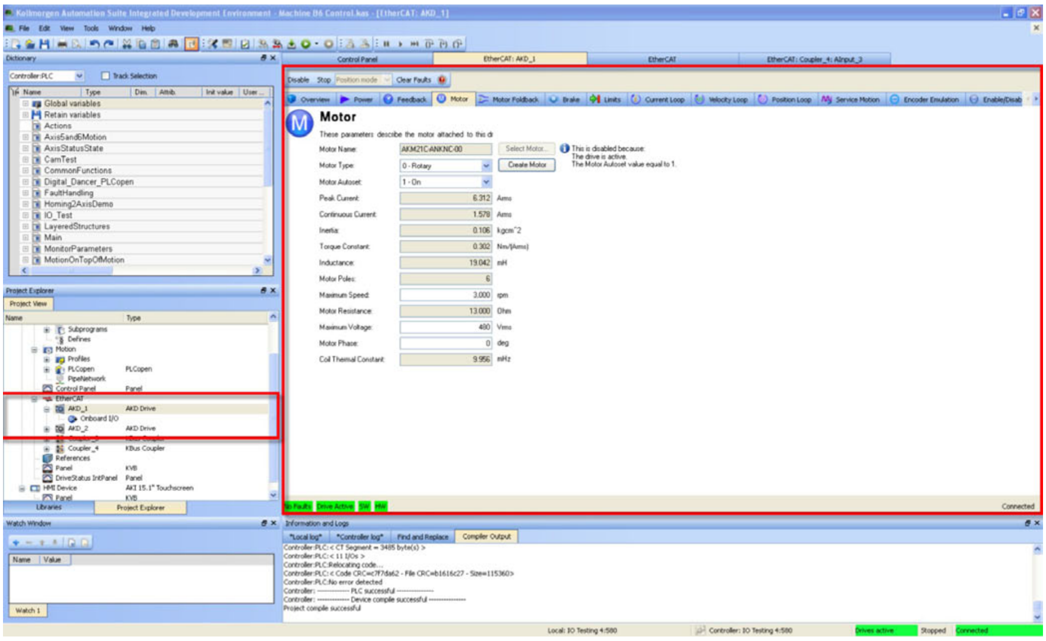
Target Position: [mm] 0

Target Velocity: [mm/s] 0

3. In the pop-up dialog box, check the Controller checkbox to enable the drive (or un-check to disable the drive) and press on the OK button.

4.7 Setup via KAS IDE

If you are using a Kollmorgen Automation Suite (KAS) system, the AKD setup is completely integrated into the KAS Integrated Development Environment (IDE), as shown below:


Kollmorgen Automation Suite Integrated Development Environment - Machine B6 Control.kas - [EtherCAT: AKD_1]
File Edit View Tools Window Help
Control Panel
EtherCAT: AKD_1
EtherCAT
EtherCAT: Coupler_4: AInput_3
Disable Stop Position mode Clear Faults
Overview Power Feedback Motor Motor Foldback Brake Limits Current Loop Velocity Loop Position Loop Service Motion Encoder Emulation Enable/Disab
Select Motor...
Create Motor
No Faults Drive Active SW HW
Connected
Information and Logs
Local log *Controller log* Find and Replace Compiler Output
Local: IO Testing 4:580
Controller: IO Testing 4:580
Drives active Stopped Connected

Motor

These parameters describe the motor attached to this di

Motor Name:	AKM21C-ANKNC-00	
Motor Type:	0 - Rotary	
Motor Autoset:	1 - On	
Peak Current:	6.312	Arms
Continuous Current:	1.578	Arms
Inertia:	0.106	kgcm^2
Torque Constant:	0.302	Nm/(Arms)
Inductance:	19.042	mH
Motor Poles:	6	
Maximum Speed:	3,000	rpm
Motor Resistance:	13.000	Ohm
Maximum Voltage:	480	Vrms
Motor Phase:	0	deg
Coil Thermal Constant:	9.956	mHz

This is disabled because:
The drive is active.
The Motor Autoset value equal to 1.

```
Controller:PLC:< CT Segment = 3485 byte(s) >
Controller:PLC:< 11 I/Os >
Controller:PLC:Relocating code....
Controller:PLC:< Code CRC=c7f7da62 - File CRC=b1616c27 - Size=115360>
Controller:PLC:No error detected
Controller: ------------ PLC successful ------------
Controller: ------------ Device compile successful ------------
Project compile successful
```

For further information on the setup for a KAS system, see the following sections in the KAS documentation:

- *KAS IDE User Manual*: See section 4.2.3 Add and Configure Drive.
- *KAS Online Help*: See **Using the KAS IDE> Creating a Project> Step 3 - Add and Configure Drive**.

5 EtherCAT Profile

5.1 Slave Register

The table below gives the addresses of individual registers in the FPGA memory. The data is provided in little-endian format, with the 'least significant byte' occupying the lowest address. A detailed description of all registers and FPGA memory locations is available in the "EtherCAT Slave Controller" description of the EtherCAT user organization (www.EtherCAT.org).

Address	Length (Byte)	Description	ZA ECAT*	ZA Drive*
0x0120	2	AL Control	R/W	R/O
0x0130	2	AL Status	R/O	R/W
0x0134	2	AL Status Code	R/O	R/W
0x0204	2	Interrupt Enable Register	R/O	R/W
0x0220	2	AL Event (IRQ Event)	R/W	R/O
0x0800	8	Sync Manager 0 (Mail Out Control Register)	R/W	R/O
0x0808	8	Sync Manager 1 (Mail In Control Register)	R/W	R/O
0x0810	8	Sync Manager 2 (Process data Output Control Register)	R/W	R/O
0x0818	8	Sync Manager 3 (Process data Input Control Register)	R/W	R/O
0x0820	8	Sync Manager 4	R/W	R/O
0x0828	8	Sync Manager 5	R/W	R/O
0x0830	8	Sync Manager 6	R/W	R/O
0x0838	8	Sync Manager 7	R/W	R/O
0x0840	8	Sync Manager 8	R/W	R/O
0x1100	Max. 64	ProOut Buffer (Process data Output, setpoints ECAT)	R/W	R/O
0x1140	Max. 64	ProIn (Process data Input, act. values ECAT)	R/O	R/W
0x1800	512	Mail Out Buffer (Object Channel Buffer ECAT, byte-length is specified in the device description file)	R/W	R/O
0x1C00	512	Mail In Buffer (Object Channel Buffer Drive, byte-length is specified in the device description file)	R/O	R/W

* ZA ECAT = Access mode EtherCAT

* ZA Drive = Access mode drive

5.2 AL Event (Interrupt Event) and Interrupt Enable

Communication between the drive and the EtherCAT FPGA can be interrupt-driven. The interrupt enable register and the AL event register are responsible for the EtherCAT interface interrupt functionality.

There are two events which lead also to a HW interrupt within the drive, the EEPROM emulation event and the SyncManager 2 event. The actual values of the drive (SyncManager 3 data) are written without any AL event request during each HW IRQ, e.g. triggered by a SyncManager 2 event. The Mailbox exchange between the master and the AKD is completely handled by polling the AL event register within the background task of the drive.

The drive activates individual EtherCAT interface events when the corresponding bit of the interrupt enable register is set to 1. When it is set to 0, the hardware interrupts for the specific events are deactivated.

5.2.1 Interrupt Enable Register (Address 0x0204:0x0205)

Parameter	Address	Bit	ZA Drive	ZA ECAT	Description
AL Control Event	0x204	0	R/W	R/O	Activation of AL control event for phase run-up
-	0x204	1	R/W	R/O	Reserved
Sync0 DC Distributed Clock	0x204	2	R/W	R/O	Activation of distributed clock (DC) sync 0 interrupts for entire communication
Sync1 DC Distributed Clock	0x204	3	R/W	R/O	Activation of distributed clock (DC) sync 1 interrupts for entire communication
SyncManager activation register change	0x204	4	R/W	R/O	Activation of ‘SyncManager activation register change’ IRQ.
EEPROM emulation event	0x204	5	R/W	R/O	Activation of the EEPROM emulation interrupts.
-	0x204	3 to 7	R/W	R/O	Reserved
Sync Manager 0 Event (Mail Out Event)	0x205	0	R/W	R/O	Activation of output event mailbox (SDO, Sync Manager 0) for object channel.
Sync Manager 1 Event (Mail In Event)	0x205	1	R/W	R/O	Activation of input event mailbox (SDO, Sync Manager 1) for object channel.
Sync Manager 2 Event (Pro Out Event)	0x205	2	R/W	R/O	Activation of output event process data (PDO, card's cyclical setpoints)
Sync Manager 3 Event (Pro In Event)	0x205	3	R/W	R/O	Activation of input event process data (PDO, drive's cyclical actual values)
-	0x205	4 to 7	R/W	R/O	Reserved

5.2.2 AL Event Request (Address 0x0220:0x0221)

When the relevant bit of the AL event request register is set to 1, the EtherCAT interface tells the drive which event it should process by the AKD.

Parameter	Address	Bit	ZA Drive	ZA ECAT	Description
AL Control Event	0x220	0	R/O	R/W	Processing of AL control event for phase run-up
Sync0 Distributed Clock (DC) Event	0x220	2	R/O	R/W	Processing of a distributed clock (DC) event
Sync1 Distributed Clock (DC) Event	0x220	3	R/O	R/W	Processing of a distributed clock (DC) event
SyncManager activation register change	0x220	4	R/O	R/W	The content of the Sync-Manager activation register has been changed.
EEPROM emulation event	0x220	5	R/O	R/W	Processing of an EEPROM emulation event in order to identify the AKD within the network.
-	0x220	6 to 7	R/O	R/W	Reserved
Sync Manager 0 Event	0x221	0	R/O	R/W	Mailbox request (SDO, Sync Manager 0) for object channel.
Sync Manager 1 Event	0x221	1	R/O	R/W	Mailbox response (SDO, Sync Manager 1) for object channel.
Sync Manager 2 Event	0x201	2	R/O	R/W	Process data output (PDO, card's cyclical setpoints)
Sync Manager 3 Event	0x201	3	R/O	R/W	Process data input (PDO, drive's cyclical actual values)
Sync Manager 4 –					
Sync Manager 7 Event	0x221	4 to 7	R/O	R/W	Reserved
Sync Manager 8 –					
Sync Manager 15 Event	0x222	0 to 7	R/O	R/W	Reserved

5.3 Phase Run-Up

The AL control, AL status and AL status code registers are responsible for communication phase run-up (also referred to as EtherCAT status change), for current status display and for any fault messages. The drive responds to every EtherCAT interface transition request made by the AL control register via the AL Status and AL Status Code registers. Any fault messages are displayed in the AL status code register.

A status change within the AL control register is polled within the AKD, which means that an AL control event does not lead to a HW interrupt within the drive.

5.3.1 AL Control (Address 0x0120:0x0121)

Parameter	Address	Bit	ZA Drive	ZA ECAT	Description
Status	0x120	3 to 0	R/O	W/O	0x01: Init Request
0x02: PreOperational Request					
0x03: Bootstrap Mode Request					
0x04: Safe Operational Request					
0x08: Operational Request					
Acknowledgement	0x120	4	R/O	W/O	0x00: No fault acknowledgement 0x01: Fault acknowledgement (positive edge)
Reserved	0x120	7 to 5	R/O	W/O	-
Applic. specific	0x120	15 to 8	R/O	W/O	-

5.3.2 AL Status (Address 0x0130:0x0131)

Parameter	Address	Bit	ZA Drive	ZA ECAT	Description
Status	0x130	3 to 0	W/O	R/O	0x01: Init
0x02: PreOperational					
0x03: Bootstrap Mode					
0x04: Safe Operational					
0x08: Operational					
Status change	0x130	4	W/O	R/O	0x00: Acknowledgement 0x01: Error, e.g. forbidden transition
Reserved	0x130	7 to 5	W/O	R/O	-
Applic. specific	0x130	15 to 8	W/O	R/O	-

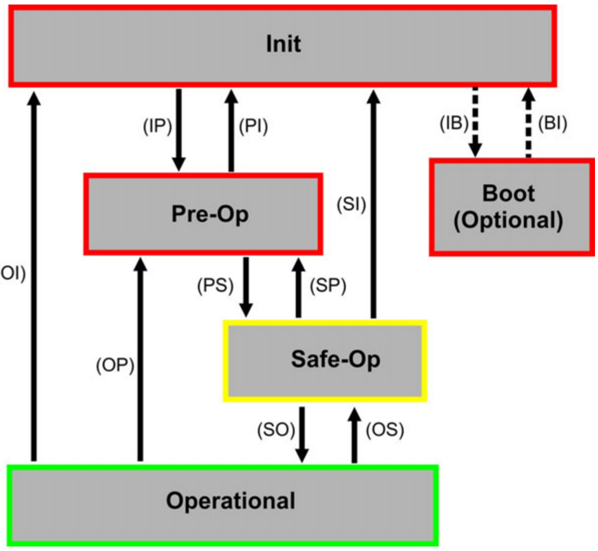
5.3.3 AL Status Code (Address 0x0134:0x0135)

Parameter	Address	Bit	ZA Drive	ZA ECAT	Description
Status	0x134	7 to 0	W/O	R/O	See table below
Status	0x135	7 to 0	W/O	R/O	See table below

Code	Description	Current Status (Status change)	Resulting Status
0x0000	No error	All	Current Status
0x0011	Invalid requested state change	I -> S, I -> O, P -> O, O -> B, S -> B, P -> B	Current Status + E
0x0017	Invalid sync manager configuration	I -> P, P -> S	Current Status + E

No other codes are supported.

5.3.4 EtherCAT communication phases



INIT:
Initialization, no communication.
EEPROM emulation will be activated.

PRE-OP:
Mailbox active, slave parameterization and startup parameters

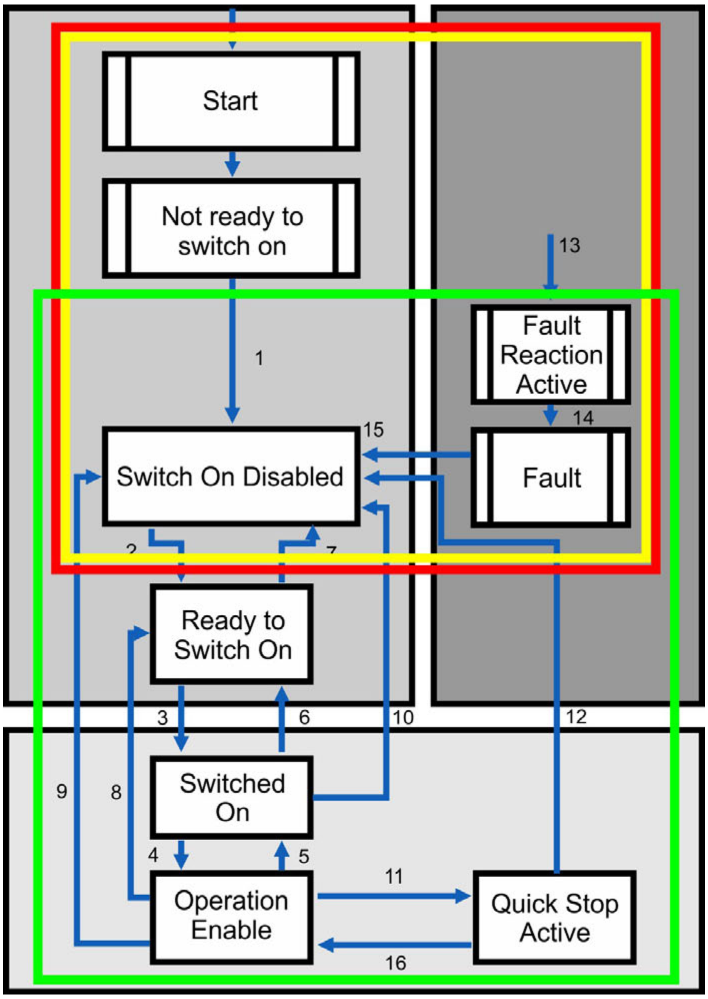
SAVE-OP:
Cyclical actual values are transferred and the drive tries to synchronize.

OPERATIONAL:
Cyclical setpoints are processed, torque enable can be activated and the drive must be synchronized.

Individual communication transitions

Transition	AL Control (Bit 3 to 0)	Description
(IB)	0x03	-
(BI)	-	-
(IP)	0x02	AKD reads the SyncManager 0 & 1 configuration and verifies the value of the start-address and the length. The AKD prepares itself for handling SyncManager 0 events.
(PI)	0x01	-
(PS)	0x04	AKD reads the SyncManager 2 & 3 configuration and verifies the value of the start-address and the length.
(SP)	0x02	-
(SI)	0x01	-
(SO)	0x08	The SnycManager 2 hardware interrupt will be enabled by the drive.
(OS)	0x04	Deactivation of SyncManager 2 hardware interrupt.
(OP)	0x02	Deactivation of SyncManager 2 hardware interrupt..
(OI)	0x01	Deactivation of SyncManager 2 hardware interrupt.

5.4 CANopen over EtherCAT (CoE) Status Machine



The status machine for the control and status words corresponds to the CANopen status machine in accordance with DS402.

CANopen control and status words are captured in every instance of fixed PDO mapping (see chapter "Fixed PDO Mappings" (➜ p. 36)).

5.4.1 Status Description

Status	Description
Not Ready to Switch On	The drive is not ready to switch on; the controller has not indicated readiness for service. The drive is still in the boot phase or in fault status.
Switch On Disable	In 'Switch On Disable' status, the amplifier cannot be enabled via the EtherCAT interface, because (for example) there is no connection to a power source.
Ready to Switch On	In 'Ready to Switch On' status, the drive can be enabled via the control word.
Switched On	In 'Switched On' status, the amplifier is enabled, but the setpoints of the EtherCAT-interface are not yet transferred. The amplifier is idle, and a positive edge in bit 3 of the control word activates setpoint transfer (transition to 'Operation Enable' status).
Operation Enable	In this status, the drive is enabled and setpoints are transferred from the EtherCAT interface.
Quick Stop Active	The drive follows a quick stop ramp.
Fault Reaction Active	The drive responds to a fault with an emergency stop ramp.
Fault	A fault is pending, the drive is stopped and disabled.

5.4.2 Commands in the Control Word

Bit assignment in the control word

Bit	Name	Bit	Name
0	Switch on	8	Pause/halt
1	Disable Voltage	9	reserved
2	Quick Stop	10	reserved
3	Enable Operation	11	reserved
4	Operation mode specific	12	reserved
5	Operation mode specific	13	Manufacturer-specific
6	Operation mode specific	14	Manufacturer-specific
7	Reset Fault (only effective for faults)	15	Manufacturer-specific

Commands in the control word

Command	Bit 7 Fault Reset	Bit 3 Enable Operation	Bit 2 Quick Stop	Bit 1 Disable Voltage	Bit 0 Switch On	Transitions
Shutdown	X	X	1	1	0	2, 6, 8
Switch on	X	X	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, 16
Fault Reset	1	X	X	X	X	15

Bits labeled **X** are irrelevant. **0** and **1** indicate the status of individual bits.

Mode-dependent bits in the control word

The following table shows the mode-dependent bits in the control word. Only manufacturer-specific modes are supported at present. The individual modes are set by Object 6060h Modes of operation.

Operation mode	No	Bit 4	Bit 5	Bit 6
Profile Position Mode (pp)	01h	new_setpoint	change_set_immediately	absolute/relative
Profile Velocity Mode (pv)	03h	reserved	reserved	reserved
Profile Torque Mode (tq)	04h	reserved	reserved	reserved
Homing Mode (hm)	06h	homing_operation_start	reserved	reserved
Interpolated Position Mode (ip)	07h		reserved	reserved
Cyclic synchronous position mode	08h	reserved	reserved	reserved

Description of the remaining bits in the control word

Bit 8: (Pause) If Bit 8 is set, then the drive halts (pauses) in all modes. The setpoints (speed for homing or jogging, motion task number, setpoints for digital mode) for the individual modes are retained.

Bit 9,10: These bits are reserved for the drive profile (DS402).

Bit 13, 14, 15: These bits are manufacturer-specific, and reserved at present.

5.4.3 Status Machine Bits (status word)

Bit assignment in the status word

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

States of the status machine

State	Bit 6 switch on disable	Bit 5 quick stop	Bit 3 fault	Bit 2 operation enable	Bit 1 switched on	Bit 0 ready to switch on
Not ready to switch on	0	X	0	0	0	0
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
Fault	0	X	1	0	0	0
Fault reaction active	0	X	1	1	1	1
Quick stop active	0	0	0	1	1	1

Bits labeled **X** are irrelevant. **0** and **1** indicate the status of individual bits.

Description of the remaining bits in the status word

Bit 4: voltage_enabled The DC-link voltage is present if this bit is set.

Bit 7: warning There are several possible reasons for Bit 7 being set and this warning being produced. The reason for this warning can be revealed by using the Object 20subindex manufacturer warnings.

Bit 9: remote is always set to 1, i.e. the drive can always communicate and be influenced via the RS232 - interface.

Bit 10: target_reached This is set when the drive has reached the target position.

Bit 11: internal_limit_active This bit specifies that a movement was or is limited. In different modes, different warnings cause the bit to be set.

5.5 Fixed PDO Mappings

Various ready-to-use mappings can be selected for cyclic data exchange via SDO's of the object 0x1C12 and 0x1C13. Using object 0x1C12 subindex 1 (Sync Manager 2 assignment), a fixed mapping for the cyclic command values can be set with the values 0x1701, 0x1702, 0x1720 to 0x1724. Using object 0x1C13 subindex 1 (Sync Manager 3 assignment), a fixed mapping for the cyclic actual values can be set via the data 0x1B01, 0x1B20 to 0x1B25.

Use the sequence below to select the fixed command value mapping 0x1701 via SDO's:

1. SDO write access to object 0x1C12Sub0 Data:0x00
2. SDO write access to object 0x1C12Sub1 Data:0x1701
3. SDO write access to object 0x1C12Sub0 Data:0x01

NOTE

Up to firmware version 1.8.x.x AKD.XML file, fixed mapping 0x1701 called out 0x6062sub0 as the "Position Command". From AKD firmware release 1.8.5.0, the AKD.XML will be changed to call out 0x60C1sub1 as the "Position Command" and an additional XML file called "AKD_TwinCAT.XML" will be added to support TwinCat 2x and older. In reality, SDO 0x6062sub0 is not supported in the AKD firmware but was called in the fixed mapping to support a TwinCat issue.

Position interface, supported fixed mappings:

0x1701	Position command value (4 bytes), Control word (2 bytes), total (6 bytes)
0x1720	Control Word (2 bytes), Interpolated position command value (4 bytes), Latch control word (2 bytes), Torque feed forward (2 bytes), Digital outputs (2 bytes)
0x1721	Interpolated position command value (4 bytes), Control Word (2 bytes), Torque feed forward (2 bytes)
0x1722	Control word (2 byte), Interpolated position command value (4 bytes), Latch control word (2 bytes), Torque feed forward (2 bytes), Digital outputs (2 bytes), max. torque (2 bytes)
0x1723	Control word (2 bytes), Interpolated position command value (4 bytes), Latch control word (2 bytes), Torque feed forward (2 bytes), Digital outputs (2 bytes), Reset of changed input information (2 bytes)
0x1724	Target position for cyclic synchronous position mode (4 bytes), Control word (2 byte), Torque feed forward (2 bytes)
0x1B01	Position actual value (4 bytes), Status word (2 bytes), total (6 bytes)
0x1B20	Position actual internal value (4 bytes), 2nd position feedback position (4 bytes), velocity actual value (4 bytes), digital inputs (4 bytes), following error (4 bytes), latch position positive (4 bytes), status word (2 bytes), torque actual value (2 bytes), latch status (2 bytes), analogue input value (2 bytes)
0x1B21	Position Actual Internal Value (4 bytes), Status word (2 bytes)
0x1B22	Position actual internal value (4 bytes), 2nd position feedback position (4 bytes), velocity actual value (4 bytes), digital inputs (4 bytes), following error (4 bytes), latch position negative (4 bytes), status word (2 bytes), torque actual value (2 bytes), latch status (2 bytes), analogue input value (2 bytes)
0x1B23	Position actual internal value (4 bytes), 2nd position feedback position (4 bytes), velocity actual value (4 bytes), digital inputs (4 bytes), following error (4 bytes), latch position positive / negative (4 bytes), status word (2 bytes), torque actual value (2 bytes), latch status (2 bytes), analogue input value (2 bytes)
0x1B24	Position actual value (4 bytes), status word (2 bytes)
0x1B25	Position actual internal value (4 bytes), 2nd position feedback position (4 bytes), latch position 2 positive / negative (4 bytes), digital inputs (4 bytes), following error (4 bytes), latch position 1 positive / negative (4 bytes), status word (2 bytes), torque actual value (2 bytes), latch status (2 bytes), analogue input value (2 bytes)

Velocity interface, supported fixed mappings:

0x1702	Velocity command value (4 bytes), Control word (2 bytes), total (6 bytes)

The objects, which are mapped into the fixed PDOs can be read via the subindices 1 to n of the above indices. The number of mapped entries is available by reading subindex 0 of the above indices.
Example: A read access to object 1702 sub 0 gives a value of 2, a read on subindex 1 gives 0x60ff0020, on subindex 2 0x60400010. The meaning of these numbers can be seen in the CANopen manual or the flexible-mapping example (➜ p. 40.).

5.6 Flexible PDO Mappings

In addition to the fixed PDO mapping the so-called flexible mapping of real-time objects is possible.

Restrictions of flexible mapping:

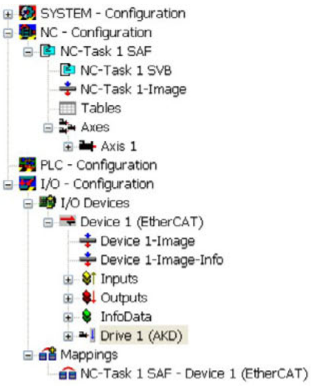
- An odd length PDO is not allowed.
  - In the Rx(=set-point)-direction the dummy-object 0x0002 sub 0 with a length of 8 bits can be used to make the PDO-length even.
  - In the Tx(=actual value)-direction one sub-index of the manufacturer status object 0x2002 sub 1..4 can be used to guarantee the even length of the Tx-PDO.
  - These special mappings may be used if the objects 0x6060 and 0x6061 have to be used in the mapping.
- The allowed PDOs have up to 32 bytes (Tx) or 22 bytes (Rx). They are built from smaller PDO modules with a maximum length of 8 bytes. These are built by using the mapping objects 0x1600 to 0x1603 and 0x1a00 to 0x1a03.

The configuration is similar to the described sequence for the fixed mappings:

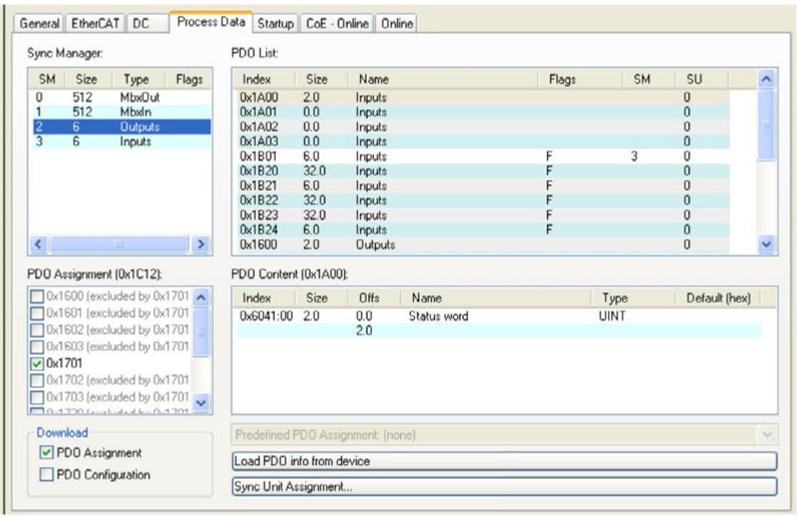
1. The mapping selection is cleared (write 0 to object 0x1C12 sub 0 and 1C13 sub 0)
2. As the AKD - implementation is based on CANopen the real-time data are build from up to 4 PDOs with 8 bytes in both directions. These PDOs are built in the same way as in a CAN-drive with the objects 0x1600 - 0x1603 and 0x1A00 - 0x1A03. Unused PDOs must be cleared with writing 0 to the subindex 0.
3. SDO write access to object 0x1C12 sub 1 .. 4 with the PDOs (0x1600 .. 0x1603), that should be used in receive direction of the AKD (set point values).
4. SDO write access to object 0x1C13 sub 1 .. 4 with the PDOs (0x1A00 .. 0x1A03), that should be used in transmit direction of the AKD (actual values).
5. SDO write access to the objects 0x1C12 sub 0 and 0x1C13 sub 0 with the number of mapped PDOs in this direction.

See an example in chapter "Flexible PDO Mappings" (➜ p. 38) .

The cyclically used data are visible in the PDO-assignment window for the Inputs and Outputs of the Sync Managers. Default setting are the fixed PDOs 0x1701 and 0x1B01 (visible contents when selected in the PDO list).



```
⊞ SYSTEM - Configuration
⊟ NC - Configuration
   ⊟ NC-Task 1 SAF
        NC-Task 1 SVB
        NC-Task 1-Image
        Tables
      ⊟ Axes
         ⊞ Axis 1
   PLC - Configuration
⊟ I/O - Configuration
   ⊟ I/O Devices
      ⊟ Device 1 (EtherCAT)
           Device 1-Image
           Device 1-Image-Info
         ⊞ Inputs
         ⊞ Outputs
         ⊞ InfoData
         ⊞ Drive 1 (AKD)
   ⊟ Mappings
        NC-Task 1 SAF - Device 1 (EtherCAT)
```



Sync Manager:

SM	Size	Type	Flags
0	512	MbxOut	
1	512	MbxIn	
2	6	Outputs	
3	6	Inputs	

PDO List:

Index	Size	Name	Flags	SM	SU
0x1A00	2.0	Inputs			0
0x1A01	0.0	Inputs			0
0x1A02	0.0	Inputs			0
0x1A03	0.0	Inputs			0
0x1B01	6.0	Inputs	F	3	0
0x1B20	32.0	Inputs	F		0
0x1B21	6.0	Inputs	F		0
0x1B22	32.0	Inputs	F		0
0x1B23	32.0	Inputs	F		0
0x1B24	6.0	Inputs	F		0
0x1600	2.0	Outputs			0

PDO Assignment (0x1C12):

□ 0x1600 (excluded by 0x1701
□ 0x1601 (excluded by 0x1701
□ 0x1602 (excluded by 0x1701
□ 0x1603 (excluded by 0x1701
☑ 0x1701
□ 0x1702 (excluded by 0x1701
□ 0x1703 (excluded by 0x1701

PDO Content (0x1A00):

Index	Size	Offs	Name	Type	Default (hex)
0x6041:00	2.0	0.0	Status word	UINT	
		2.0			

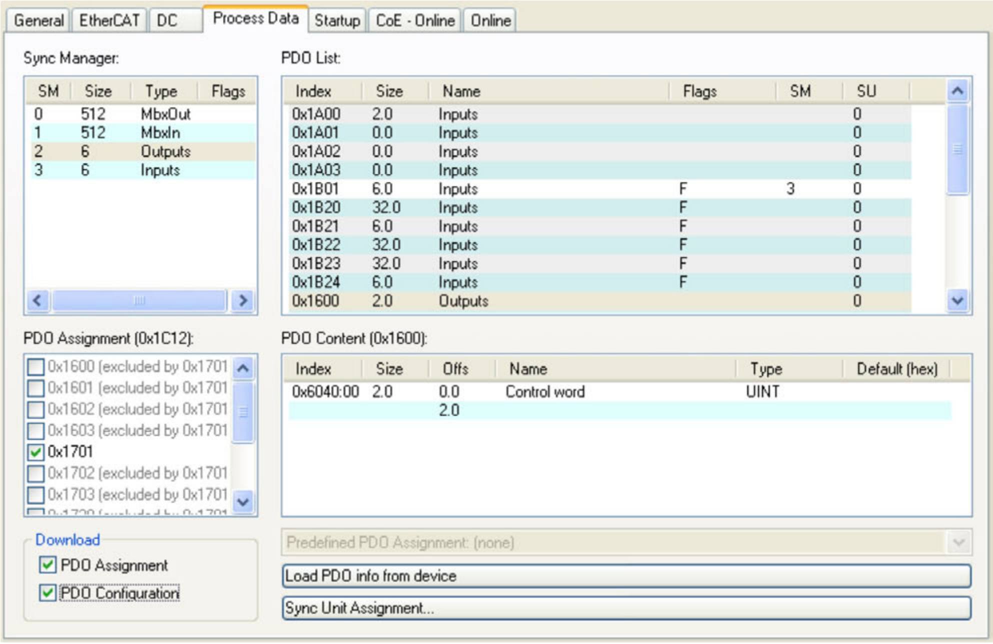
Download

☑ PDO Assignment

□ PDO Configuration

Predefined PDO Assignment: (none)

If the flexible mapping is required, the PDO configuration check box must be changed.



Sync Manager:

SM	Size	Type	Flags
0	512	MbxOut	
1	512	MbxIn	
2	6	Outputs	
3	6	Inputs	

PDO List:

Index	Size	Name	Flags	SM	SU
0x1A00	2.0	Inputs			0
0x1A01	0.0	Inputs			0
0x1A02	0.0	Inputs			0
0x1A03	0.0	Inputs			0
0x1B01	6.0	Inputs	F	3	0
0x1B20	32.0	Inputs	F		0
0x1B21	6.0	Inputs	F		0
0x1B22	32.0	Inputs	F		0
0x1B23	32.0	Inputs	F		0
0x1B24	6.0	Inputs	F		0
0x1600	2.0	Outputs			0

PDO Assignment (0x1C12):

- □ 0x1600 (excluded by 0x1701
- □ 0x1601 (excluded by 0x1701
- □ 0x1602 (excluded by 0x1701
- □ 0x1603 (excluded by 0x1701
- ☑ 0x1701
- □ 0x1702 (excluded by 0x1701
- □ 0x1703 (excluded by 0x1701

PDO Content (0x1600):

Index	Size	Offs	Name	Type	Default (hex)
0x6040:00	2.0	0.0	Control word	UINT	
		2.0			

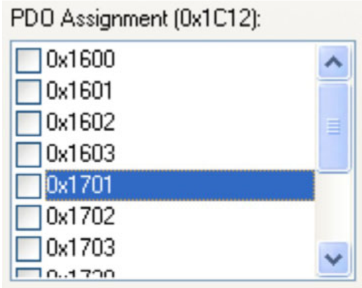
Download

- ☑ PDO Assignment
- ☑ PDO Configuration

Predefined PDO Assignment: (none)

5.6.1 Example: Flexible PDO Mapping

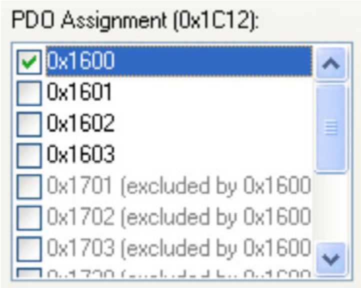
For the flexible mapping of the Outputs the fixed mapping 0x1701 has to be switched off and up to 4 free-mappable PDOs (0x1600-0x1603) can be used instead. The maximum number of bytes for each of these PDOs is 8.



PDO Assignment (0x1C12):

- □ 0x1600
- □ 0x1601
- □ 0x1602
- □ 0x1603
- □ 0x1701
- □ 0x1702
- □ 0x1703

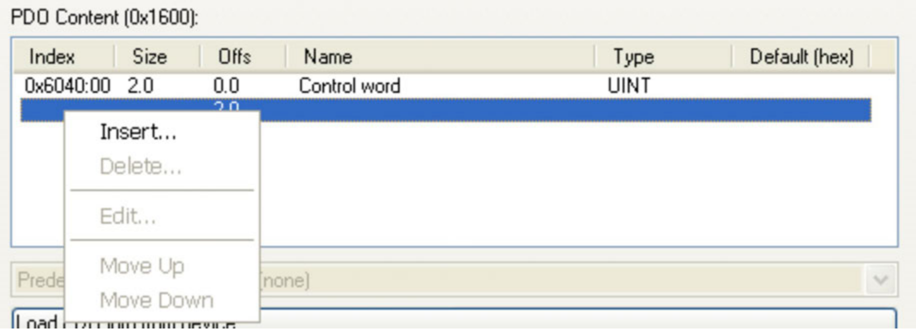
==========>



PDO Assignment (0x1C12):

- ☑ 0x1600
- □ 0x1601
- □ 0x1602
- □ 0x1603
- □ 0x1701 (excluded by 0x1600
- □ 0x1702 (excluded by 0x1600
- □ 0x1703 (excluded by 0x1600

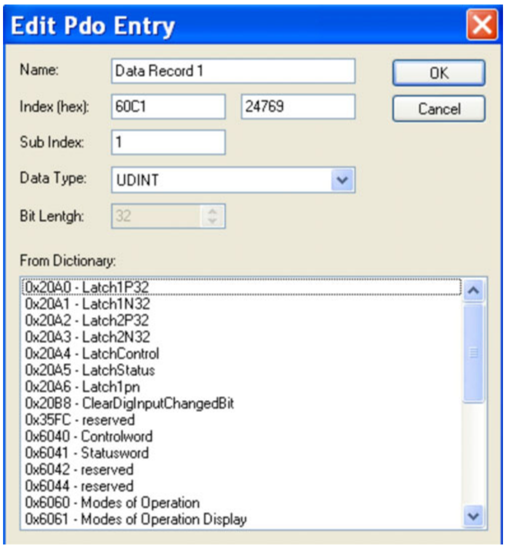
After that the default mapping of e.g. the PDO 0x1600 can be extended:



PDO Content (0x1600):

Index	Size	Offs	Name	Type	Default (hex)
0x6040:00	2.0	0.0	Control word	UINT	
		2.0			

A list of possible objects for the mapping will be shown and a new entry can be chosen.

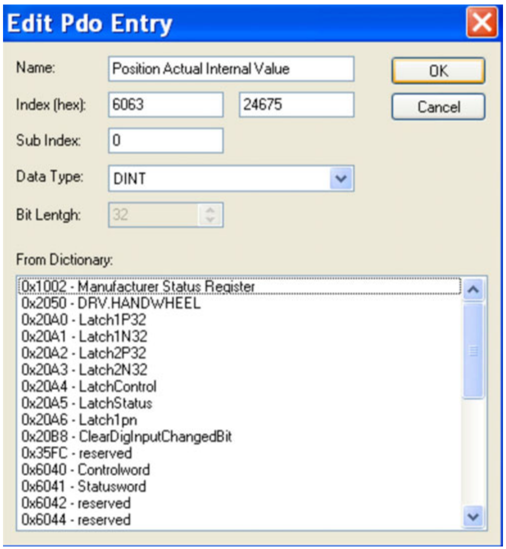


Name: Data Record 1
Index (hex): 60C1 24769
Sub Index: 1
Data Type: UDINT
Bit Lentgh: 32

From Dictionary:

0x20A0 - Latch1P32
0x20A1 - Latch1N32
0x20A2 - Latch2P32
0x20A3 - Latch2N32
0x20A4 - LatchControl
0x20A5 - LatchStatus
0x20A6 - Latch1pn
0x20B8 - ClearDigInputChangedBit
0x35FC - reserved
0x6040 - Controlword
0x6041 - Statusword
0x6042 - reserved
0x6044 - reserved
0x6060 - Modes of Operation
0x6061 - Modes of Operation Display

In this case the setpoint for the interpolated position mode is selected.



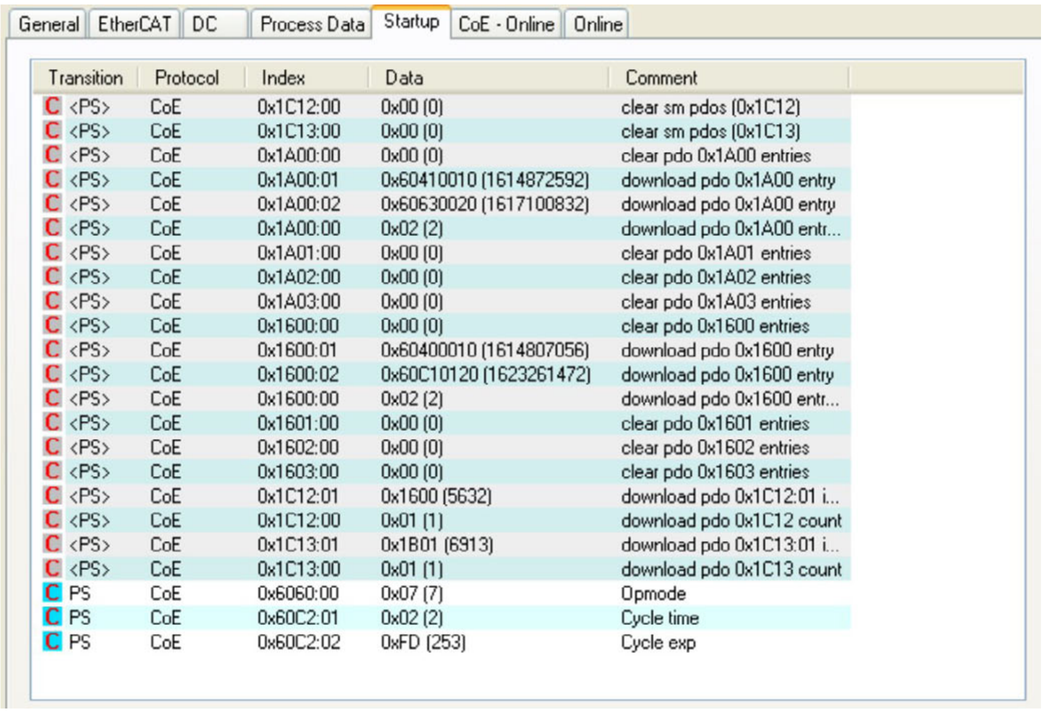
Name: Position Actual Internal Value
Index (hex): 6063 24675
Sub Index: 0
Data Type: DINT
Bit Lentgh: 32

From Dictionary:

0x1002 - Manufacturer Status Register
0x2050 - DRV.HANDWHEEL
0x20A0 - Latch1P32
0x20A1 - Latch1N32
0x20A2 - Latch2P32
0x20A3 - Latch2N32
0x20A4 - LatchControl
0x20A5 - LatchStatus
0x20A6 - Latch1pn
0x20B8 - ClearDigInputChangedBit
0x35FC - reserved
0x6040 - Controlword
0x6041 - Statusword
0x6042 - reserved
0x6044 - reserved

The same is valid for the Tx-PDO-direction. Here the value of the actual internal position is selected.

This results in the startup-SDO-list for this sample free-mapped-configuration.

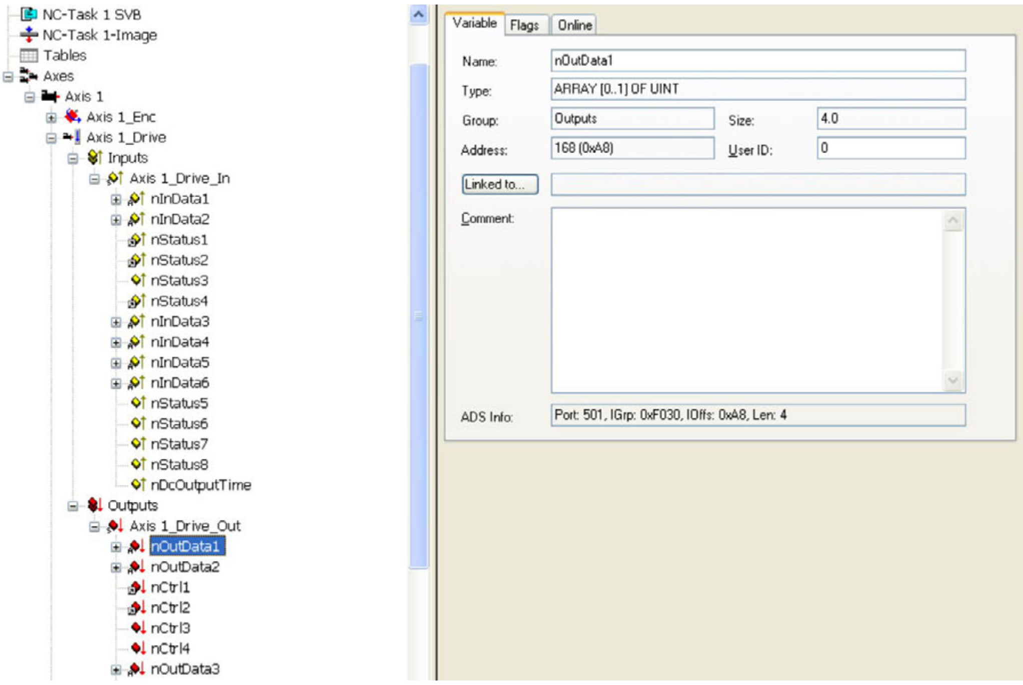


Transition	Protocol	Index	Data	Comment
C <PS>	CoE	0x1C12:00	0x00 (0)	clear sm pdos (0x1C12)
C <PS>	CoE	0x1C13:00	0x00 (0)	clear sm pdos (0x1C13)
C <PS>	CoE	0x1A00:00	0x00 (0)	clear pdo 0x1A00 entries
C <PS>	CoE	0x1A00:01	0x60410010 (1614872592)	download pdo 0x1A00 entry
C <PS>	CoE	0x1A00:02	0x60630020 (1617100832)	download pdo 0x1A00 entry
C <PS>	CoE	0x1A00:00	0x02 (2)	download pdo 0x1A00 entr...
C <PS>	CoE	0x1A01:00	0x00 (0)	clear pdo 0x1A01 entries
C <PS>	CoE	0x1A02:00	0x00 (0)	clear pdo 0x1A02 entries
C <PS>	CoE	0x1A03:00	0x00 (0)	clear pdo 0x1A03 entries
C <PS>	CoE	0x1600:00	0x00 (0)	clear pdo 0x1600 entries
C <PS>	CoE	0x1600:01	0x60400010 (1614807056)	download pdo 0x1600 entry
C <PS>	CoE	0x1600:02	0x60C10120 (1623261472)	download pdo 0x1600 entry
C <PS>	CoE	0x1600:00	0x02 (2)	download pdo 0x1600 entr...
C <PS>	CoE	0x1601:00	0x00 (0)	clear pdo 0x1601 entries
C <PS>	CoE	0x1602:00	0x00 (0)	clear pdo 0x1602 entries
C <PS>	CoE	0x1603:00	0x00 (0)	clear pdo 0x1603 entries
C <PS>	CoE	0x1C12:01	0x1600 (5632)	download pdo 0x1C12:01 i...
C <PS>	CoE	0x1C12:00	0x01 (1)	download pdo 0x1C12 count
C <PS>	CoE	0x1C13:01	0x1B01 (6913)	download pdo 0x1C13:01 i...
C <PS>	CoE	0x1C13:00	0x01 (1)	download pdo 0x1C13 count
C PS	CoE	0x6060:00	0x07 (7)	Opmode
C PS	CoE	0x60C2:01	0x02 (2)	Cycle time
C PS	CoE	0x60C2:02	0xFD (253)	Cycle exp

The meaning of the data (for example 0x60410010 in the mapping of 0x1A00 sub 1) is as follows:

- 0x6041 is the index of the DS402 status word
- 0x00 is the subindex of the DS402 status word
- 0x10 is the number of bits for this entry, i. e. 16 bits or 2 bytes.

If this shall be used in the NC, the interpolation set point position has to be linked from the axis to the NC-axis.



Name: nOutData1

Type: ARRAY [0..1] OF UINT

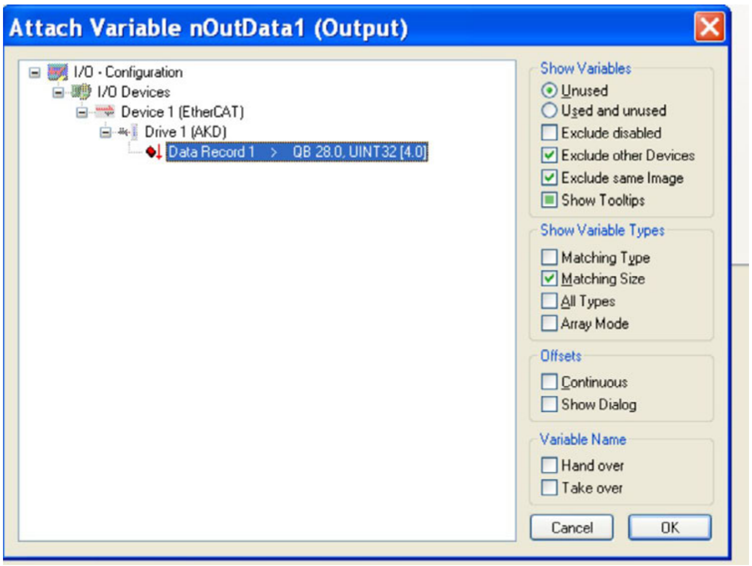
Group: Outputs

Size: 4.0

Address: 168 (0xA8)

User ID: 0

ADS Info: Port: 501, IGrp: 0xF030, IOffs: 0xA8, Len: 4

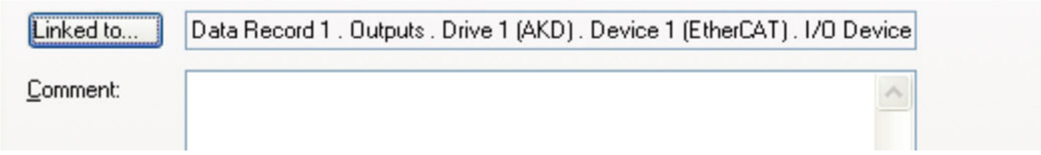

Attach Variable nOutData1 (Output)

Show Variables
- ● Unused
- ○ Used and unused
- □ Exclude disabled
- ☑ Exclude other Devices
- ☑ Exclude same Image
- ▨ Show Tooltips

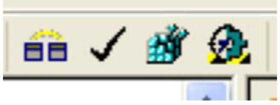
Show Variable Types
- □ Matching Type
- ☑ Matching Size
- □ All Types
- □ Array Mode

Offsets
- □ Continuous
- □ Show Dialog

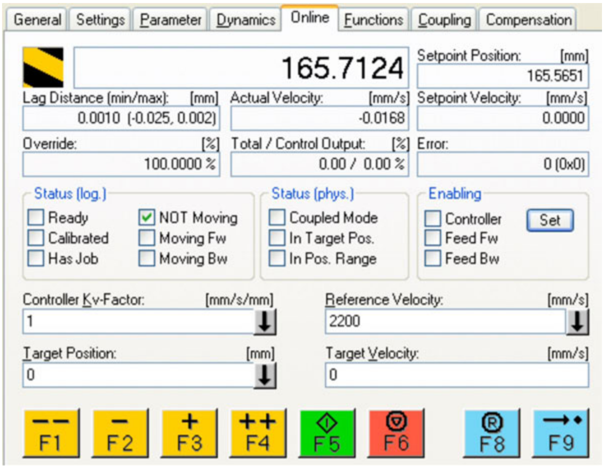
Variable Name
- □ Hand over
- □ Take over


Linked to... Data Record 1 . Outputs . Drive 1 (AKD) . Device 1 (EtherCAT) . I/O Device

After doing this configuration the mapping can be activated as seen before in this document:



Now the NC-screen should show a position in the online window, which changes a bit in the last digits.



165.7124

Setpoint Position: [mm] 165.5651

Lag Distance (min/max): [mm] 0.0010 (-0.025, 0.002)

Actual Velocity: [mm/s] -0.0168

Setpoint Velocity: [mm/s] 0.0000

Override: [%] 100.0000 %

Total / Control Output: [%] 0.00 / 0.00 %

Error: 0 (0x0)

Status (log.)
- □ Ready
- □ Calibrated
- □ Has Job
- ☑ NOT Moving
- □ Moving Fw
- □ Moving Bw

Status (phys.)
- □ Coupled Mode
- □ In Target Pos.
- □ In Pos. Range

Enabling
- □ Controller
- □ Feed Fw
- □ Feed Bw

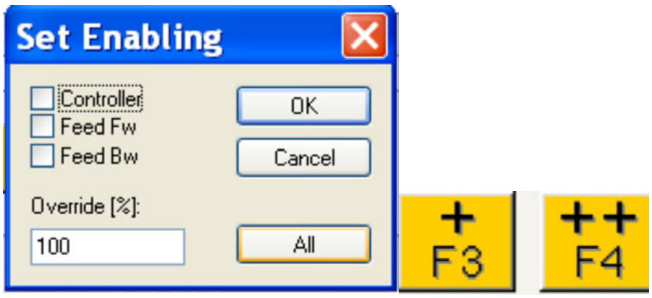
Controller Kv-Factor: [mm/s/mm] 1

Reference Velocity: [mm/s] 2200

Target Position: [mm] 0

Target Velocity: [mm/s] 0

After enabling the power stage with the All-button, the drive can be moved via the jog-buttons or via the functions in the function menu.



- □ Controller
- □ Feed Fw
- □ Feed Bw

Override [%]: 100

5.7 Supported Cyclical Setpoint and Actual Values

Supported cyclical setpoint values

Name	CANopen object	Data type	Description
Position command value	0x60C1 sub 1	INT32	Interpolation data record in IP-mode
Velocity command value	0x60FF sub 0	INT32	
CANopen control-word	0x6040 sub 0	UINT16	CANopen control word.
Latch Control word	0x20A4 sub 0	UINT16	
Torque feed forward	0x60B2 sub 0	INT16	
Digital outputs	0x60FE sub 1	UINT32	
Target current	0x2071 sub 0	32 bit	scaled in mA
Modes of Operation	0x6060 sub 0	8 bit	DS402 opmode setpoint
Velocity Window	0x606D sub 0	16 bit	
Velocity Window Time	0x606E sub 0	16 bit	
Target Torque	0x6071 sub 0	16 bit	0.1% resolution
Maximum Torque	0x6072 sub 0	16 bit	
Profile position target velocity	0x6081 sub 0	32 bit	MT.V
Profile position target acc	0x6083 sub 0	32 bit	MT.ACC
Profile position target dec	0x6084 sub 0	32 bit	MT.DEC
Velocity feed forward	0x60B1 sub 0	32 bit	
Touch probe function	0x60B8	16 bit	
Analog output value	0x3470 sub 3	16 bit	
External feedback position	0x3497 sub 0	32 bit	
Clear digital Input Change Bit	0x20B8	16 bit	

Supported cyclical actual values

Name	CANopen object	Data type	Description
Position actual internal value	0x6063 sub 0	INT32	
Velocity actual value	0x606C sub 0	INT32	
CANopen status-word	0x6041 sub 0	UINT16	CANopen status word
Second position feedback	0x2050 sub 0	INT32	
Digital inputs	0x60FD sub 0	UINT32	
Following error actual value	0x60F4 sub 0	INT32	
Latch position positive edge	0x20A0 sub 0	INT32	
Torque actual value	0x6077 sub 0	INT16	
Latch status	0x20A5 sub 0	UINT16	
Analog input value	0x3470 sub 0	INT16	
Actual Current	0x2077 sub 0	32 bit	scaled in mA
Latch1 negative edge	0x20A1 sub 0	32 bit	
Latch2 Positive	0x20A2 sub 0	32 bit	
Latch2 Negative	0x20A3 sub 0	32 bit	
Latch1 positive/negative edge	0x20A6	32 bit	
Latch 2 positive/negative edge	0x20A7	32 bit	
Modes of Operation	0x6061	8 bit	DS402 opmode status
Position Actual Value	0x6064 sub 0	32 bit	WB/ DS402 scale units
Touch probe status	0x60B9 sub 0	16 bit	
Touch probe 1 positive edge pos	0x60BA sub 0	32 bit	
Touch probe 1 negative edge pos	0x60BB sub 0	32 bit	
Touch probe 2 positive edge pos	0x60BC sub 0	32 bit	
Touch probe 2 negative edge pos	0x60BD sub 0	32 bit	
Additional Pos actual value	0x60E4 sub 0	48 bit	
Additional Pos actual value	0x60E4 sub 1	32 bit	
Motor I2t	0x3427 sub 3	32 bit	
Analog output value	0x3470 sub 2	16 bit	
Analog Input & Output value	0x3470 sub 4	16 bit	
Manufacturer status register	0x1002 sub 0	32 bit	

5.8 Supported Operation Modes

CANopen mode of operation	AKD mode of operation	Description
Profile velocity	DRV.OPMODE 1 DRC.CMDSOURCE 1	0x6060Sub0 Data: 3 In this mode, the EtherCAT master sends cyclic velocity command values to the AKD.
Interpolated position	DRV.OPMODE 2 DRV.CMDSOURCE 1	0x6060Sub0 Data: 7 In this mode of operation the EtherCAT master sends cyclic position command values to the AKD. These command values are interpolated by the AKD according to the fieldbus sample rate.
Homing mode	DRV.OPMODE 2 DRV.CMDSOURCE 0	0x6060 sub 0 data : 6 In this mode an AKD-internal homing can be done.
Profile Position	DRV.OPMODE 2 DRV.CMDSOURCE 0	0x6060sub0 Data: 1 Uses motion task 0 to execute a point to point move
Torque	DRV.OPMODE 0 DRV.CMDSOURCE 1	0x6060sub0 Data: 4 Commands torque in % of drive peak torque
Cyclic Synchronous Position	DRV.OPMODE 2 DRV.CMDSOURCE 1	0x6060sub0 Data: 8 Master calculates move profile and commands motion with position points

5.9 Adjusting EtherCAT Cycle Time

The cycle time to be used in the drive for the cyclical setpoints and actual values can either be stored in the FBUS.SAMPLEPERIOD parameter in the amplifier or configured in the startup phase.

This takes place via SDO mailbox access to CANopen objects 60C2 subindex 1 and 2.

Subindex 2, known as the interpolation time index, defines the power of ten of the time value (e.g. -3 means 10-3 or milliseconds) while subindex 1, known as interpolation time units, gives the number of units (e.g. 4 means 4 units).

You can run a 2 ms cycle using various combinations. For example,

Index = -3, Units = 2
or
Index = -4, Units = 20 etc.

The FBUS.SAMPLEPERIOD parameter is counted in multiples of 62.5us microseconds within the device. This means, for example that 2 ms equates to FBUS.SAMPLEPERIOD value of 32.

5.10 Maximum Cycle Times depending on operation mode

The minimum cycle time for the drive is largely dependent on the drive configuration (second actual position value encoder latch functionality enabled and so on)

Interface	Cycle time AKD
Position	≥ 0.25 ms (≥ 250 µs)
Velocity	≥ 0.25 ms (≥ 250 µs)
Torque	≥ 0.25 ms (≥ 250 µs)

5.11 Synchronization

On all drives, the internal PLL is theoretically able to even out an average deviation of up to 4800 ppm in the cycle time provided by the master. The drive checks once per fieldbus cycle a counter within the drive internal FPGA, which is cleared by a Sync0 (Distributed clock) event. Depending of the counter value, the drive extends or decreases the 62.5 µs MTS signal within the drive by a maximum of 300 ns.

The theoretical maximum allowed deviation can be calculated by using the following formula:

$$\max_{de}v = \frac{300[ns]}{62.5[\mu s]} \cdot 1{,}000{,}000 = 4800\ [ppm]$$

The synchronization functionality within the drive can be enabled via setting bit 0 of the FBUS.PARAM02 parameter to high. Therefore FBUS.PARAM02 must be set to the value of 1. Furthermore the distributed clock functionality must be enabled by the EtherCAT master in order to activate cyclic Sync0 events.

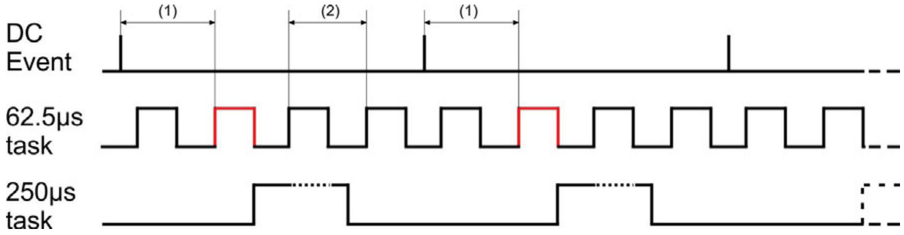
5.11.1 Synchronization behavior with distributed clocks (DC) enabled

When the EtherCAT master enables distributed clocks, a distributed clock (DC) event is created in the AKD once per fieldbus cycle. An assigned 62.5 µs real-time task in the AKD monitors the elapsed time between the DC events and the AKD System time and extends or reduces the 62.5 µs strobe to the CPU as necessary.

The following fieldbus parameters are used for the synchronization feature:

1. FBUS.SYNCDIST = Expected time delay of the AKD PLL-code to the DC event.
2. FBUS.SYNCACT = Actual time delay of the AKD PLL-code to the DC event.
3. FBUS.PLLTHRESH = Number of consecutive successful synchronized PLL cycles of the AKD before the Drive is considered as synchronized.
4. FBUS.SYNCWND = Synchronization window in which the AKD is considered to be synchronized. The Drive is considered synchronized as long as the following statement is true is true for FBUS.PLLTHRESH consecutive cycles:
   FBUS.SYNCDIST-FBUS.SYNCWND < FBUS.SYNCACT < FBUS.SYNCDIST+FBUS.SYNCWND

Example with a 4kHz fieldbus sample rate:



Explanation: The red-marked 62.5[µs] real-time task displays the AKD 62.5 µs real-time task within one fieldbus cycle which is responsible for calling the AKD PLL-code. The time delay (1) shows the actual delay to the previous DC event, which is ideally close to the adjusted FBUS.SYNCDIST parameter. Depending on (1) the AKD slightly extends or reduce the 62.5 [µs] IRQ generation of the high-priority real-time task in order to either increase or decrease the measured time delay to the DC event (1) for the next PLL cycle. The time distance (2) shows the 62.5[µs] ± x[ms] realtime task of the AKD.

5.11.2 Synchronization behavior with distributed clocks (DC) disabled

The AKD fieldbus synchronization algorithm is similar to that used by Distributed Clocks. The difference is that the AKD synchronizes to a SyncManager2 event instead of the DC event. A SyncManager2 event is created when the EtherCAT Master sends a new package of command values to the drive while the network is in the Operational state. This occurs once per fieldbus cycle.

5.12 Latch Control Word and Latch Status Word

Latch Control word (2 Byte)

Bit	Value (bin)	Value (hex)	Description
0	00000000 00000001	zz01	Enable extern latch 1 (positive rise)
1	00000000 00000010	zz02	Enable extern latch 1 (negative rise)
2	00000000 00000100	zz04	Enable extern latch 2 (positive rise)
3	00000000 00001000	zz08	Enable extern latch 2 (negative rise)
4			
5-7			Reserve
8-12	00000001 00000000	01zz	Read external latch 1 (positive rise)
	00000010 00000000	02zz	Read external latch 1 (negative rise)
	00000011 00000000	03zz	Read external latch 2 (positive rise)
	00000100 00000000	04zz	Read external latch 2 (negative rise)
13-15			Reserve

Latch Status word (2 Byte)

Bit	Value (bin)	Value (hex)	Description
0	00000000 00000001	zz01	External latch 1 valid (positive rise)
1	00000000 00000010	zz02	External latch 1 valid (negative rise)
2	00000000 00000100	zz04	External latch 2 valid (positive rise)
3	00000000 00001000	zz08	External latch 2 valid (negative rise)
4			
5-7			Reserve
8-11	00000001 00000000	z1zz	Acknowledge value external latch 1 (positive rise)
	00000010 00000000	z2zz	Acknowledge value external latch 1 (negative rise)
	00000011 00000000	z3zz	Acknowledge value external latch 2 (positive rise)
	00000100 00000000	z4zz	Acknowledge value external latch 2 (negative rise)
12-15	00010000 00000000	1zzz	Zustand Digital Input 4
	00100000 00000000	2zzz	Zustand Digital Input 3
	01000000 00000000	4zzz	Zustand Digital Input 2
	10000000 00000000	8zzz	Zustand Digital Input 1

5.13 Mailbox Handling

With EtherCAT, acyclical data traffic (object channel or SDO channel) is called mailbox. This system is based around the master:

Mailbox Output:

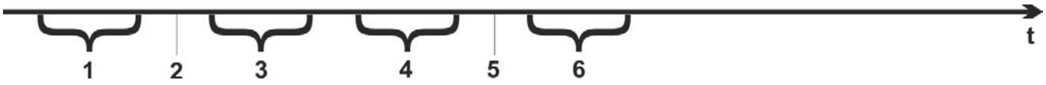
The master (EtherCAT controller) sends data to the slave (drive). This is essentially a (read/write) request from the master. Mailbox output operates via Sync Manager 0.

Mailbox Input:

The slave (drive) sends data to the master (EtherCAT controller). The master reads the slave's response. Mailbox input operates via Sync Manager 1.

Timing diagram

The timing diagram illustrates the mailbox access process:



1. The EtherCAT master writes the mailbox request to the mail-out buffer.
2. On the next interrupt, the EtherCAT interface activates a Sync Manager 0 event (mailbox output event) in the AL event register.
3. The drive reads 16 bytes from the mail-out buffer and copies them to the internal mailbox output array.
4. The drive identifies new data in the internal mailbox output array and performs an SDO access to the object requested by the EtherCAT interface. The response from the drive is written to an internal mailbox input array.
5. The drive deletes all data in the internal mailbox output array so that a new mailbox access attempt can be made.
6. The drive copies the response telegram from the internal mailbox input array to the mail-in buffer of the EtherCAT interface.

5.13.1 Mailbox Output

An interrupt by the EtherCAT-interface with a Sync Manager 0 - Event starts a Mailbox Output Process. A 1 in the Mail Out Event-Bit of the AL Event register signalizes the drive, that the EtherCAT-interface wants to send a Mailbox message and that it has already stored the required data in the Mail Out Buffer. Now 16 Byte data are read by the drive with the IRQ process. The bytes are defined as follows

Address 0x1800								Address 0x180F							
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CAN over EtherCAT specific data (CoE Header)								CAN specific data (standard CAN SDO)							

Byte 0	Length of the data (Low Byte)	
Byte 1	Length of the data (High Byte)	
Byte 2	Address (Low Byte)	
Byte 3	Address (High Byte)	
Byte 4	Bit 0 to 5: Channel Bit 6 to 7: Priority	
Byte 5	Bit 0 to 3: Type	1 = Reserved: ADS over EtherCAT 2 = Reserved: Ethernet over EtherCAT 3 = Can over EtherCAT…)
	Bit 4 to 7: Reserved	
Byte 6	PDO Number (with PDO transmissions only, Bit 0 = LSB of the PDO number, see Byte 7 for MSB)	
Byte 7	Bit 0: MSB of the PDO number, see Byte 6	
	Bit 1 to 3: Reserved	
	Bit 4 to 7: CoE specific type	0: Reserved
		1: Emergency message
		2: SDO request
		3: SDO answer
		4: TXPDO
		5: RxPDO
		6: Remote transmission request of a TxPDO
		7: Remote transmission request of a RxPDO
		8…15: reserved
Byte 8	Control-Byte in the CAN telegram:	
	write access:	0x23=4Byte, 0x27=3Byte, 0x2B=2Byte, 0x2F=1Byte
	read access:	0x40
Byte 9	Low Byte of the CAN object number (Index)	
Byte 10	High Byte of the CAN object number (Index)	
Byte 11	Subindex according to CANopen Specification for the drive	
Byte 12	Data with a write access (Low Byte)	
Byte 13	Data with a write access	
Byte 14	Data with a write access	
Byte 15	Data with a write access (High Byte)	

The drive answers every telegram with an answer in the Mailbox Input buffer.

5.13.2 Mailbox Input

The drive answers every CoE telegram with a 16 byte answer telegram in the Mailbox Input buffer. The bytes are defined as follows:

Address 0x1C00								Address 0x1C0F							
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CAN over EtherCAT specific data (CoE Header)								CAN specific data (standard CAN SDO)							

Byte 0	Length of the data (Low Byte)	
Byte 1	Length of the data (High Byte)	
Byte 2	Address (Low Byte)	
Byte 3	Address (High Byte)	
Byte 4	Bit 0 to 5: Channel Bit 6 to 7: Priority	
Byte 5	Bit 0 to 3: Type	1 = Reserved: ADS over EtherCAT 2 = Reserved: Ethernet over EtherCAT 3 = Can over EtherCAT…)
	Bit 4 to 7: Reserved	
Byte 6	PDO Number (with PDO transmissions only, Bit 0 = LSB of the PDO number, see Byte 7 for MSB)	
Byte 7	Bit 0: MSB of the PDO number, see Byte 6	
	Bit 1 to 3: Reserved	
	Bit 4 to 7: CoE specific type	0: Reserved
		1: Emergency message
		2: SDO request
		3: SDO answer
		4: TXPDO
		5: RxPDO
		6: Remote transmission request of a TxPDO
		7: Remote transmission request of a RxPDO
		8…15: reserved
Byte 8	Control-Byte in the CAN telegram:	
	write access OK:	0x60
	read access OK + length of answer:	0x43 (4 Byte), 0x47 (3 Byte), 0x4B (2Byte), 0x4F (1Byte)
	error with read- or write access:	0x80
Byte 9	Low Byte of the CAN object number (Index)	
Byte 10	High Byte of the CAN object number (Index)	
Byte 11	Subindex according to CANopen Specification for Kollmorgen drive	
Byte 12	Data (Low Byte)	
Byte 13	Data	error code Fehlercode according to CAN-open Specification in case of an error
Byte 14	Data	data value of the object in case of successfull read access
Byte 15	Data (High Byte)	

5.13.3 Example: Mailbox Access

In the example below, PDOs 0x1704 are mapped (see Chapter # “Fixed PDO Mappings”):

The master sends this mailbox output message:

Byte 0	0x0A	The next 10 Bytes contain data (Byte 2 to Byte 11)
Byte 1	0x00	The next 10 Bytes contain data (Byte 2 to Byte 11)
Byte 2	0x00	Address 0
Byte 3	0x00	Address 0
Byte 4	0x00	Channel 0 and Priority 0
Byte 5	0x03	CoE Object
Byte 6	0x00	PDO Number 0
Byte 7	0x20	PDO Number 0 and SDO-Request
Byte 8	0x2B	2 Byte write access
Byte 9	0x12	SDO-Object 0x1C12
Byte 10	0x1C	SDO-Object 0x1C12
Byte 11	0x01	Subindex 1
Byte 12	0x04	Data value 0x00001704
Byte 13	0x17	Data value 0x00001704
Byte 14	0x00	Data value 0x00001704
Byte 15	0x00	Data value 0x00001704

The drive returns the following message:

Byte 0	0x0E	The next 14 Bytes contain data (Byte 2 to Byte 15)
Byte 1	0x00	The next 14 Bytes contain data (Byte 2 to Byte 15)
Byte 2	0x00	Address 0
Byte 3	0x00	Address 0
Byte 4	0x00	Channel 0 and Priority 0
Byte 5	0x03	CoE Object
Byte 6	0x00	PDO Number 0
Byte 7	0x20	PDO Number 0 and SDO-Answer
Byte 8	0x60	Successful write access
Byte 9	0x12	SDO-Object 0x1C12
Byte 10	0x1C	SDO-Object 0x1C12
Byte 11	0x01	Subindex 1
Byte 12	0x00	Data value 0x00000000
Byte 13	0x00	Data value 0x00000000
Byte 14	0x00	Data value 0x00000000
Byte 15	0x00	Data value 0x00000000

5.14 Fieldbus Parameters

The AKD holds several fieldbus-specific, general purpose parameters. Some of them contain the following EtherCAT relevant data:

- **FBUS.PARAM02**: This parameter activates the synchronization feature of the AKD. The DC feature must be activated in order to allow the AKD to get synchronized with the master. A value of 1 enables the drive internal PLL functionality, a value of 0 deactivates this feature.
- **FBUS.PARAM03**: This parameter contains the Configured Station Alias address of the AKD. An EEPROM emulation write access to the Configured Station Alias address forces the AKD to store the drive parameters automatically using the DRV.NVSAVE command.
- **FBUS.PARAM04**: This parameter enables (1) or disables(0) the synchronization supervision of the CANOpen or EtherCAT fieldbus.
  Default values for this parameter are as follows:
  CANopen drive: disabled (0)
  EtherCAT drive: enabled (1)
  Synchronization supervision is active when FBUS.PARAM 04 = 1 and the first CANOpen Sync message or first EtherCAT frame is received. When more than three CANOpen sync messages or seven EtherCAT frames have not been received and the drive is enabled, fault F125 ("Synchronization lost"), occurs.
- **FBUS.PARAM05**

Bit 0	1	Faults can only be reset using DS402 control word bit 7.
	0	The reset can also be done via telnet or digital input and the DS402 state machine reflects this condition.
Bit 1	1	The state of the hardware enable does not change the state machine state Operation Enable.
	0	If the state Operation Enable or Switched on is active it falls back to the state switched On Disabled, if the Hardware enable goes to 0.
Bit 2	1	Workbench/Telnet can not software enable the drive, when CANopen/EtherCAT are Operational.
	0	Workbench/Telnet can software enable the drive.
Bit 3	1	DS402-state machine is not influenced, if the software-enable is taken away via Telnet.
	0	DS402-state machine is influenced, if the software-enable is taken away via Telnet.
Bit 4	1	Scaling is done via special DS402 - objects (independent on units)
	0	Scaling for position, velocity and acceleration objects is done via UNIT parameters
Bit 5	1	FBUS.PARAM03 defines the station alias address if not 0. If FBUS.PARAM03 set to 0, the address will be taken from rotary switches instead, if they are not 0. The EtherCAT master has the ability to use the alias address, selected by the drive, or issue its own.
	0	The rotary switches define the station alias address if not 0. If the rotary switches are set to 0, the address will be taken from FBUS.PARAM03 instead, if it is not 0.
Bit 6	1	Bit 0 of parameter MT.CNTL (object 35D9 sub 0) can be accessed
	0	Bit 0 of parameter MT.CNTL (object 35D9 sub 0) is exclusively used for DS402 controlword
Bit 7		reserved
Bit 8	1	DS402-state SWITCHED ON means power stage disabled
	0	DS402-state SWITCHED ON means power stage enabled

5.15 EEProm Content

AKD has a built-in emulated EEProm. This EEProm can be read by the EtherCAT master to get some information about drive properties, like PDO-information, drive name, serial numbers and communication-specific attributes.

They are organized in categories. There are two manufacturer-specific categories implemented in the AKD:

- Category 0x0800: Holds a string with the model type in the format AKD-P00000-NxxC-0000
- Category 0x0801: Holds the firmware version in the format 0x_xx-xx-yyy

6 Record of Document Revisions

Revision	Remarks
-, 11/2009	Beta launch version
-, 12/2009	Minor formatting changes
A, 07/2010	FBUS.PARAM04 added, part number added, page format, release information
B, 10/2010	Setup for KAS added
C, 01/2011	HW Rev. C
D, 04/2011	WoE, corrections
E, 10/2011	Flexible mapping, cover page layout
F, 03/2012	Minor corrections
G, 11/2012	New chapter EEProm content
H, 05/2013	Fixed mapping, supported cyclic values, FBUS.PARAM05 added, several updates, formatting according to 82079

7 Index

This page intentionally left blank.

This page intentionally left blank.

About KOLLMORGEN

Kollmorgen is a leading provider of motion systems and components for machine builders. Through world-class knowledge in motion, industry-leading quality and deep expertise in linking and integrating standard and custom products, Kollmorgen delivers breakthrough solutions that are unmatched in performance, reliability and ease-of-use, giving machine builders an irrefutable marketplace advantage.

For assistance with your application needs, visit www.kollmorgen.com or contact us at:

North America
KOLLMORGEN
203A West Rock Road
Radford, VA 24141 USA

Web: www.kollmorgen.com
Mail: support@kollmorgen.com
Tel.: +1 - 540 - 633 - 3545
Fax: +1 - 540 - 639 - 4162

Europe
KOLLMORGEN Europe GmbH
Pempelfurtstraße 1
40880 Ratingen, Germany

Web: www.kollmorgen.com
Mail: technik@kollmorgen.com
Tel.: +49 - 2102 - 9394 - 0
Fax: +49 - 2102 - 9394 - 3155

Asia
KOLLMORGEN
Rm 2205, Scitech Tower, China
22 Jianguomen Wai Street

Web: www.kollmorgen.com
Mail: sales.asia@kollmorgen.com
Tel.: +86 - 400 666 1802
Fax: +86 - 10 6515 0263

KOLLMORGEN®

Because Motion Matters™