

Anwendungsbeschreibung

EtherCAT Master App

EtherCAT Master für ctrlX CORE 01VRS

Schutzvermerk

© Bosch Rexroth AG Bosch Rexroth AG© 2023

Alle Rechte vorbehalten, auch bezüglich jeder Verfügung, Verwertung, Reproduktion, Bearbeitung, Weitergabe sowie für den Fall von Schutzrechtsanmeldungen.

Verbindlichkeit

Die angegebenen Daten dienen allein der Produktbeschreibung und sind nicht als zugesicherte Eigenschaften im Rechtssinne zu verstehen. Änderungen im Inhalt der Dokumentation und Liefermöglichkeiten der Produkte sind vorbehalten.

DOK-XCORE*-ETHERCATV01-AP08-DE-P

Inhaltsverzeichnis

1	Über diese Dokumentation	5
2	Wichtige Gebrauchshinweise	7
2.1	Bestimmungsgemäßer Gebrauch.	7
2.1.1	Einführung.	7
2.1.2	Einsatz- und Anwendungsbereiche	7
2.2	Nicht bestimmungsgemäßer Gebrauch.	8
3	Sicherheitshinweise	9
4	Einführung und Übersicht	11
4.1	EtherCAT Master App – Grundlagen.	11
4.2	Begriffe und Abkürzungen	11
4.3	Übertragungsarten	13
4.4	EtherCAT Grundlagen.	14
4.4.1	Topologie	14
4.4.2	Adressierungs- und Identifikationsverfahren	16
4.4.3	Zustandsmaschine	17
4.4.4	Azyklische Kommunikation (Mailbox)	19
4.4.5	Telegramm-Aufbau	21
4.4.6	WorkingCounter	22
4.4.7	SyncUnits	22
4.4.8	Hot Connect	23
4.4.9	EtherCAT Slave Controller	24
	Aufbau	24
	Kommandos	25
5	EtherCAT Synchronisation	27
5.1	Allgemein	27
5.2	Free Run	27
5.3	Synchronisation mit SM-Event	28
5.4	Synchronisation mit DC-Sync	30
6	EtherCAT Konfiguration	37
6.1	Übersicht	37
7	EtherCAT Master	39
7.1	EtherCAT Master – Features	39
7.2	EtherCAT Master konfigurieren.	40
7.3	EtherCAT Slave.	41
7.3.1	EtherCAT Feldbusteilnehmer konfigurieren.	41
8	ctrlX Bedienoberfläche – Elemente	43
8.1	Navigation.	43
8.1.1	Seitennavigation – Knoten EtherCAT Master.	43
8.2	Fenster.	43
8.2.1	Fenster – „EtherCAT Master“.	43
8.2.2	Fenster – „EtherCAT Master“ ^{Detailansicht}	44
8.3	Header.	45
8.3.1	Header – „EtherCAT Master“.	45
8.3.2	Header – „EtherCAT Slave“.	46

8.4	Registerkarten – EtherCAT Master.	48
8.4.1	Registerkarte – „Slaves“.	48
8.4.2	Registerkarte – „Distributed Clocks“.	48
8.4.3	Registerkarte – „Master-Statistik“.	50
8.4.4	Registerkarte – „Slave-Statistik“.	51
8.4.5	Registerkarte – „AoE“ (EtherCAT Master-Instanz).	52
8.4.6	Registerkarte – „EoE“ (EtherCAT Master-Instanz).	53
8.5	Registerkarten – EtherCAT Slave.	54
8.5.1	Registerkarte – „Allgemeines“ (EtherCAT Slave).	54
9	EtherCAT Diagnose	55
9.1	Allgemein	55
9.2	AL Status Codes	56
9.3	CoE SDO Abortcodes	58
9.4	FoE Fehlercodes	59
9.5	SoE Fehlercodes	60
10	FAQs	63
10.1	ctrlX DRIVE: Wie konfiguriert man SoE mit Free Run?	63
10.2	ctrlX DRIVE: Wie verbindet man ctrlX DRIVE Engineering über EtherCAT?	63
11	Weiterführende Dokumentationen	65
11.1	Übersicht.	65
11.2	ctrlX AUTOMATION.	65
11.3	ctrlX WORKS.	65
11.4	ctrlX CORE.	66
11.5	ctrlX CORE Apps.	66
12	Service und Support	70
13	Index	71

1 Über diese Dokumentation

Ausgaben dieser Dokumentation

Ausgabe	Stand	Bemerkung
01	2020-06	Erstausgabe zur EtherCAT Master App (Version ECM-V-0102 & ECM-V-0102)
02	2020-12	Ausgabe zur EtherCAT Master App (Version ECM-V-0106) Überarbeitung: <ul style="list-style-type: none"> ➔ Begriffe und Abkürzungen ➔ EtherCAT Master – Features ➔ EtherCAT Master konfigurieren ➔ EtherCAT Feldbusteilnehmer konfigurieren ➔ CoE SDO Abortcodes ➔ FoE Fehlercodes ➔ SoE Fehlercodes Entfernt: <ul style="list-style-type: none"> • Kapitel 11 "Feldbus übergreifende Features"
03	2021-06	Ausgabe zur EtherCAT Master App (Version ECM-V-0108) Überarbeitung: <ul style="list-style-type: none"> ➔ Technische Daten
04	2021-08	Ausgabe zur EtherCAT Master App (Version ECM-V-0110) Überarbeitung: <ul style="list-style-type: none"> ➔ Technische Daten
05	2022-02	Ausgabe zur EtherCAT Master App (Version ECM-V-0112) Neu: <ul style="list-style-type: none"> • Fenster "EtherCAT Slave Statistics" (in Version ECM-V-0116 in Registerkarte umgewandelt, siehe Ausgabe 06)
06	2022-08	Ausgabe zur EtherCAT Master App (Version ECM-V-0116) Neu: <ul style="list-style-type: none"> • ➔ Fenster – „EtherCAT Master“ • ➔ Fenster – „EtherCAT Master“ Detailansicht • ➔ Registerkarte – „Slaves“ • ➔ Registerkarte – „Distributed Clocks“ • ➔ Registerkarte – „Slave-Statistik“ Überarbeitung: <ul style="list-style-type: none"> • ➔ EtherCAT Master – Features

Ausgabe	Stand	Bemerkung
07	2023-01	<p>Ausgabe zur EtherCAT Master App (Version ECM-V-0118)</p> <p>Neu:</p> <ul style="list-style-type: none"> ➔ Registerkarte – „Master-Statistik“ ➔ Registerkarte – „Allgemeines“ (EtherCAT Slave) ➔ Registerkarte – „AoE“ (EtherCAT Master-Instanz) ➔ Registerkarte – „EoE“ (EtherCAT Master-Instanz) ➔ Header – „EtherCAT Master“ ➔ Header – „EtherCAT Slave“ <p>Überarbeitung:</p> <ul style="list-style-type: none"> ➔ EtherCAT Master – Features
08	2023-05	<p>Ausgabe zur EtherCAT Master App (Version ECM-V-0120)</p> <p>Überarbeitung:</p> <ul style="list-style-type: none"> ➔ EtherCAT Master – Features ➔ Allgemein ➔ WorkingCounter ➔ Registerkarte – „Distributed Clocks“

2 Wichtige Gebrauchshinweise

2.1 Bestimmungsgemäßer Gebrauch

2.1.1 Einführung

Produkte von Rexroth werden nach dem jeweiligen Stand der Technik entwickelt und gefertigt.

Vor ihrer Auslieferung werden die Produkte auf ihren betriebssicheren Zustand hin überprüft.

⚠️ WARNUNG

Personen- und Sachschäden durch falschen Gebrauch der Produkte!

Die Produkte dürfen nur bestimmungsgemäß eingesetzt werden.

Wenn die Produkte nicht bestimmungsgemäß eingesetzt werden, dann können Situationen entstehen, die Sach- und Personenbeschädigung nach sich ziehen.

HINWEIS

Schäden bei nicht bestimmungsgemäßem Gebrauch

Für Schäden bei nicht bestimmungsgemäßem Gebrauch der Produkte leistet Rexroth als Hersteller keinerlei Gewährleistung, Haftung oder Schadensersatz. Die Risiken bei nicht bestimmungsgemäßem Gebrauch der Produkte liegen allein beim Anwender.

Bevor Sie die Produkte der Firma Rexroth einsetzen, müssen die folgenden Voraussetzungen erfüllt sein, um einen bestimmungsgemäßen Gebrauch der Produkte zu gewährleisten:

- Jeder, der in irgendeiner Weise mit Rexroth Produkten umgeht, muss die entsprechenden Sicherheitsvorschriften und den bestimmungsgemäßen Gebrauch lesen und verstehen
- Sofern es sich bei den Produkten um Hardware handelt, müssen die Produkte in ihrem Originalzustand belassen werden; d. h. es dürfen keine baulichen Veränderungen an den Produkten vorgenommen werden. Softwareprodukte dürfen nicht dekompiert werden und ihre Quellcodes dürfen nicht verändert werden
- Beschädigte oder fehlerhafte Produkte dürfen nicht eingebaut oder in Betrieb genommen werden
- Es muss gewährleistet sein, dass die Produkte entsprechend den in der Dokumentation genannten Vorschriften installiert sind

2.1.2 Einsatz- und Anwendungsbereiche

Produkte der ctrlX Baureihe sind für Motion-/Logic-Anwendungen geeignet.

HINWEIS

Produkte der ctrlX Baureihe dürfen nur mit den in dieser Dokumentation angegebenen Zubehör- und Anbauteilen benutzt werden. Nicht ausdrücklich genannte Komponenten dürfen weder angebaut noch angeschlossen werden. Gleiches gilt für Kabel und Leitungen.

Der Betrieb darf nur in den ausdrücklich angegebenen Konfigurationen und Kombinationen der Hardware-Komponenten und mit der in den jeweiligen Dokumentationen und den Funktionsbeschreibungen angegebenen und spezifizierten Soft- und Firmware erfolgen.

Produkte der ctrlX Baureihe sind für den Einsatz in ein- und mehrachsigen Antriebs- und Steuerungsaufgaben geeignet. Für den applikationsspezifischen Einsatz des Systems stehen Gerätetypen mit unterschiedlicher Ausstattung und unterschiedlichen Schnittstellen zur Verfügung.

Typische Anwendungsbereiche:

- Gebäudeautomatisierung
- IoT und Security Gateway bzw. Device
- Handling & Robotic

Steuerungen der ctrlX CORE Baureihe dürfen nur unter den in den weiterführenden Dokumentationen angegebenen Montage- und Installationsbedingungen, in der angegebenen Gebrauchslage und unter den angegebenen Umweltbedingungen (Temperatur, Schutzart, Feuchte, EMV u. a.) betrieben werden.

2.2 Nicht bestimmungsgemäßer Gebrauch

Die Verwendung von ctrlX-Produkten außerhalb der vorgenannten Anwendungsgebiete oder unter anderen als den in der Dokumentation beschriebenen Betriebsbedingungen und angegebenen technischen Daten gilt als "nicht bestimmungsgemäß".

ctrlX-Produkte dürfen nicht eingesetzt werden, wenn sie den folgenden Bedingungen ausgesetzt sind:

- Betriebsbedingungen, die die vorgeschriebenen Umgebungsbedingungen nicht erfüllen. Untersagt sind z. B. der Betrieb unter Wasser, unter extremen Temperaturschwankungen oder extremen Maximaltemperaturen
- Bei Anwendungen, die von Rexroth nicht ausdrücklich freigegeben sind




3 Sicherheitshinweise

Die Sicherheitshinweise, soweit in der vorliegenden Anwendungsdokumentation vorhanden, beinhalten bestimmte Signalwörter ("Gefahr", "Warnung", "Vorsicht", "Hinweis") und ggf. eine Signalgrafik (nach ANSI Z535.6-2006).

Das Signalwort soll die Aufmerksamkeit auf den Sicherheitshinweis lenken und bezeichnet die Schwere der Gefährdung.

Die Signalgrafik (Warndreieck mit Ausrufezeichen), welche den Signalwörtern "Gefahr", "Warnung" und "Vorsicht" vorangestellt wird, weist auf Gefährdungen für Personen hin.

Die Sicherheitshinweise in dieser Dokumentation werden wie folgt dargestellt:

 GEFAHR	Bei Nichtbeachtung dieses Sicherheitshinweises werden Tod oder schwere Körperverletzung eintreten.
 WARNUNG	Bei Nichtbeachtung dieses Sicherheitshinweises können Tod oder schwere Körperverletzung eintreten.
 VORSICHT	Bei Nichtbeachtung dieses Sicherheitshinweises können mittelschwere oder leichte Körperverletzung eintreten.
HINWEIS	Bei Nichtbeachtung dieses Sicherheitshinweises können Sachschäden eintreten.

4 Einführung und Übersicht

4.1 EtherCAT Master App – Grundlagen

Durch die Installation der „EtherCAT Master App“ wird die ctrlX CORE Steuerung um die EtherCAT Master Funktionalität erweitert:

- EtherCAT-Stack zum Betrieb des Feldbusses auf der Steuerung
- Ergänzungen in der ctrlX CORE Oberfläche zur Feldbus-Konfiguration und zur Anzeige bzw. Steuerung des Betriebszustands, siehe [↗ Kapitel 8.1.1 Seitenavigation – Knoten EtherCAT Master auf Seite 43](#)

Lizenz

Zum Betrieb des EtherCAT Master auf der ctrlX CORE Steuerung ist folgende Lizenz erforderlich:

Typschlüssel	Materialnummer
SWL-XC*-ECM-ETHERCATMAS**-BANN	R911400508

Lizenz erwerben

[↗ Im ctrlX App Store](#)

Konfiguration der EtherCAT Slaves

Die Konfiguration der EtherCAT Slaves erfolgt im Software-Tool ctrlX I/O Engineering, das im Installationsumfang von ctrlX WORKS enthalten ist (optionale Installation auf einem Engineering-PC).

Verwandte Themen

- [↗ App Installation](#)
- [↗ Seitenavigation – Knoten EtherCAT Master](#)
- [↗ Fenster – „EtherCAT Master“](#)
- [↗ Fenster – „EtherCAT Master“](#) Detailansicht

Web-Links

- [↗ ctrlX CORE - Community](#)
- [↗ ctrlX CORE - How-to](#)
- [↗ ctrlX CORE - Forum](#)

4.2 Begriffe und Abkürzungen

„EtherCAT“	„Ethernet for Control Automation Technology“ EtherCAT® ist eine eingetragene Marke und patentierte Technologie, lizenziert durch die Beckhoff Automation GmbH, Deutschland EtherCAT ist ein offener Standard, der international genormt ist und von der EtherCAT Technology Group (ETG) weiterentwickelt wird.
„EC / ECAT“ „ECM“	Abkürzungen für EtherCAT (EC / ECAT) bzw. EtherCAT-Master (ECM)
„ENI“	„EtherCAT Network Information“: Beschreibung der Buskonfiguration als XML-Datei
„ESI“	„EtherCAT Slave Information“: Gerätebeschreibungsdatei als XML-Datei
„ESC“	„EtherCAT Slave Controller“: in Hardware integrierte Kommunikationsanschaltung (ASIC, FPGA) oder als reine Softwarelösung
„ETG“	„EtherCAT Technology Group“: Nutzerorganisation
„EAP“	„EtherCAT Automation Protocol“: Protokoll zur Vernetzung auf der Leitebene

„DC“	„Distributed Clocks“ Zur Synchronisation der Busteilnehmer wird das Verfahren "Verteilte Uhren" angewendet. Der Rexroth EtherCAT-Master unterstützt hierzu verschiedene Synchronisationsverfahren.
„FSOE“	„FailSafe over EtherCAT“: Safety-Protokoll für EtherCAT
„HC“	„Hot Connect“: Ab- und Zuschalten von Slaves im laufenden Betrieb
„HC-Group“	„Hot Connect Group“: Slaves können in einer Gruppe zusammengefasst werden. Die Gruppe kann als optional deklariert werden (flexible Topologien).
„Mailbox“	Azyklische Kommunikation über EtherCAT
„CoE“	„CAN application protocol over EtherCAT“ (Mailbox)
„SoE“	„Servo drive profile over EtherCAT“: Adaptiertes Sercos II Profil für Servoantriebe (Mailbox)
„FoE“	„File access over EtherCAT“ (Mailbox)
„EoE“	„Ethernet over EtherCAT“: Ethernet Kommunikation z.B. TCP/IP (Mailbox)
„VoE“	„Vendor specific protocol over EtherCAT“: Hersteller-spezifische Mailbox-Protokolle
„AoE“	„Automation Device Specification (ADS) over EtherCAT“: Geräte- und feldbusunabhängige Schnittstelle der Firma Beckhoff (Mailbox)
„WC / WKC“ „WcState“	Abkürzungen für WorkingCounter (WC / WKC) bzw. WorkingCounter-Status (WcState): Gültigkeit von Kommando-Ausführungen bzw. von Daten (ein WcState Wert 0 = gültig, 1 = ungültig).

Die folgende Abbildung stellt die feldbusabhängigen Bezeichnungen der Master bzw. Slave-Schnittstellen der Feldbusse dar:

Tab. 1: Begriffe der Feldbus-Schnittstellen

Feldbus	Master-Schnittstelle	Slave-Schnittstelle
EtherCAT	Master	Slave

Tab. 2: Begriffe der Datenübertragungsmechanismen

Feldbus	zyklische Übertragung	azyklische Übertragung
EtherCAT	zyklische Daten / Prozessdaten	verschiedene Mailbox Protokolle

Gerätebeschreibungsdateien

Die Gerätebeschreibungsdatei beschreibt die Eigenschaften des Feldbusgeräts. In dieser Datei sind alle relevanten Daten enthalten, die sowohl für das Engineering als auch für den Datenaustausch mit dem I/O-Gerät von Bedeutung sind.

ESI-Datei

Gerätebeschreibungsdatei, mittels derer EtherCAT Slaves in ctrlX I/O Engineering integriert und konfiguriert werden können (ESI: EtherCAT Slave Information).

Buskoppler

Feldbus-Anschaltung (Slave, Device, Adapter),
kompakt

- Bei einem kompakten Slave ist der Modulaufbau fest vorgegeben. Die Module unterhalb des Slave-Objektknotens sind bereits vollständig vorhanden. In der Bibliothek sind keine weiteren Module vorhanden

modular

- Der Modulaufbau des Slaves ist variabel. Die Module (Klemmen) können individuell – aber nach Bestückungsvorschrift des Gerätes – angeordnet werden.

Modul, Klemme

Kleinste I/O-Einheit zur Bestückung einer Feldbus-Anschaltung und/oder des Inlinebusses.

4.3 Übertragungsarten

Die Übertragungsarten lassen sich in folgende Kategorien einteilen:

- zyklischer Übertragungskanal
- azyklischer Übertragungskanal

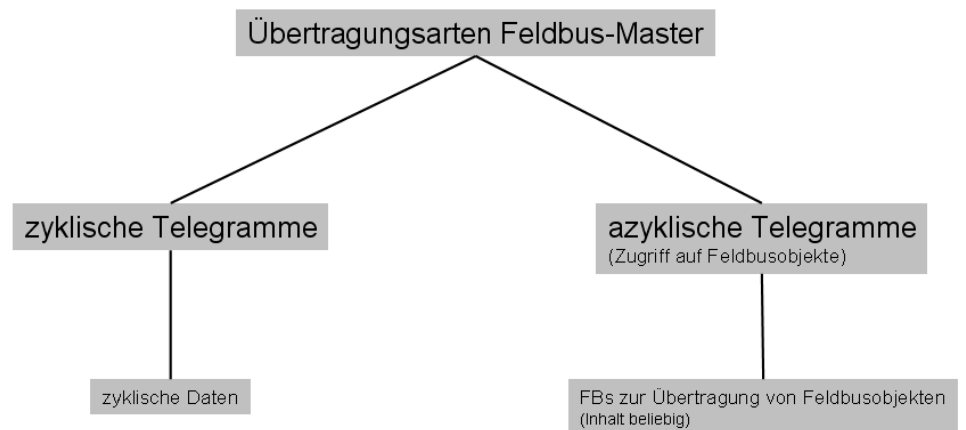


Abb. 1: Überblick über die Übertragungsarten der Feldbus-Master

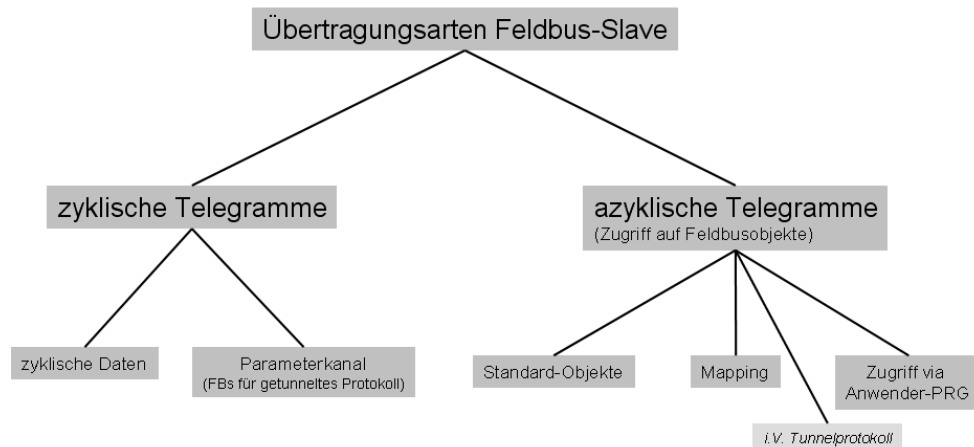


Abb. 2: Überblick über die Übertragungsarten der Feldbus-Slaves

4.4 EtherCAT Grundlagen

4.4.1 Topologie

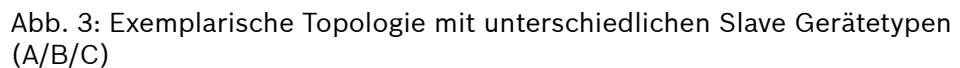
Ein EtherCAT Netzwerk besteht aus einem Master und bis zu 65535 Slaves.

Topologien können beliebig realisiert werden, z.B. durch den Einsatz von Abzweigklemmen.

Bei der Verkabelung ist zu beachten, dass Slave Geräte genau einen definierten Eingangsport haben. Dies ist in der Regel der Port 0, der üblicherweise mit „IN“ beschriftet ist. Die meisten Slaves besitzen 2 Ports, d.h. neben dem „IN-Port“ noch einen „OUT-Port“ (Port 1). Bei einem Buskoppler mit E-BUS als Rückwandbus werden die Ethernet-Telegramme über diesen Rückwandbus (E-BUS) weitergeleitet, d.h. jedes Modul (Klemme) stellt einen Slave mit IN-Port und OUT-Port dar. Der E-BUS Buskoppler oder Abzweigklemmen haben dementsprechend 3 bzw. 4 Ports (= mehrere Ausgangsports).

Nachfolgende Abbildung zeigt eine exemplarische Topologie mit unterschiedlichen Gerätetypen:

- Slave **(A)** ist ein modularer Buskoppler nach dem „Modular Device Profile“ (z.B. Rexroth S20-EC-BK): die Erweiterungsmodule sind mit einem gerätespezifischen Rückwandbus mit dem Buskoppler verbunden. Der Buskoppler stellt ein Slave Gerät im Netzwerk dar und besitzt alle E/A-Daten der Module.
- Slave **(B)** ist ein modulares System mit E-BUS (z.B. Beckhoff EK1100): jedes Modul als auch der Buskoppler und mögliche Abzweig- oder Verlängerungsklemmen stellen jeweils ein eigenes Slave Gerät am EtherCAT Netzwerk dar.
- Slave **(C)** ist ein Antrieb.



Kabelredundanz



The diagram illustrates a network configuration where a loopback is attempted. A switch on the left has two ports, XF50 and XF51. A yellow line connects XF50 to the 'IN' port of the first of two intermediate switch modules. The output of the first module connects to the 'IN' port of the second module. The output of the second module is labeled 'OUT' and has a red lightning bolt symbol next to it, indicating a fault or error. A yellow line also connects the 'OUT' port of the second module back to the XF51 port on the first switch, completing the loopback. The label 'OUT' is also present on the far right of the diagram.

Bosch Rexroth AG

4.4.2 Adressierungs- und Identifikationsverfahren

In einem EtherCAT Netzwerk gibt es verschiedene Verfahren zur Adressierung und Identifizierung eines Slave Gerätes.

Auto-Increment Adresse

Die Auto-Increment Adresse entspricht einer topologischen Adresse, d.h. das Slave Gerät wird anhand seiner physikalischen Position im Netzwerk adressiert. Die Auto-Increment Adresse wird vom Master hauptsächlich bei der Initialisierung der Buskommunikation verwendet. Ab dem Zustand „Pre-Operational“ wird die EtherCAT Adresse („Fixed Physical Address“) für die Kommunikation verwendet.

Der erste Slave nach dem Master besitzt die Auto-Increment Adresse 0. Für jeden weiteren Slave wird der Wert dekrementiert, d.h. der zweite Slave hat die Adresse -1 (0xFFFF), der dritte Slave hat die Adresse -2 (0xFFFE) usw.

Bei der Kommunikation inkrementiert jeder Slave die Auto-Increment Adresse im EtherCAT-Telegramm. Der Slave, der den Wert „0“ im Telegramm liest, wird entsprechend adressiert.

EtherCAT Adresse

Die EtherCAT Adresse wird in der ETG Spezifikation auch als „Fixed Physical Address“ bezeichnet. Bei anderen Herstellern synonym verwendete Begriffe sind zum Beispiel „Slave Address“ oder „Station Address“.

Die EtherCAT Adresse wird in den Slave Geräten nicht gespeichert, sondern vom Master bei der Bus-Initialisierung jedem Slave zugewiesen. Die Zuordnung, welche EtherCAT Adresse ein bestimmtes Slave Gerät erhält, erfolgt in der EtherCAT Konfiguration entweder über die Auto-Increment Adresse oder bei Hot Connect Gruppen (flexible Topologien) über einen Identifikationswert.

Die EtherCAT Adresse 0 ist für den Master reserviert. Slaves können die Adresse 1 bis 65535 (16 bit) haben. EtherCAT Adressen für die Slaves werden oft ab dem Wert 1001 verwendet.

Identifikationswert

Bei Hot Connect Gruppen (flexible Topologien) oder um Verkabelungsfehler auszuschließen ist es erforderlich, Slave Geräte unabhängig von deren topologischer Position eindeutig zu identifizieren.

Damit einem bestimmten Slave Gerät immer eine festgelegte EtherCAT Adresse zugewiesen werden kann, gibt es drei verschiedene Identifikationsverfahren. Welche Identifikationsverfahren von einem Slave Gerät unterstützt werden (keine Unterstützung, ein oder mehrere Verfahren werden unterstützt) ist in dessen Gerätebeschreibung bzw. Produktdokumentation beschrieben. Die Konfiguration des Identifikationswertes erfolgt beispielsweise über DIP- oder Drehschalter am Slave Gerät.

Identifikationsverfahren:

- (Configured) **Station-Alias**: der Station-Alias wird oft auch als „Second-Slave-Address“ (SSA) oder „Second-Station-Address“ bezeichnet. Der Wert kann im ESC Register 0x0012 ausgelesen werden. Die Konfiguration erfolgt i.d.R.

über das EtherCAT Konfigurationstool, in dem der gewünschte Identifikationswert in das Slave EEPROM geschrieben wird und der EtherCAT Slave Controller dann beim (Neu-) Start diesen Wert in sein Register übernimmt.

- **Explicit Device Identification:** hierbei teilt der Slave auf Anforderung durch den Master seinen Identifizierungswert im AL Status Code Register („0x0134“) mit. Dieses Verfahren wird in der „ETG.1020“ auch als „Requesting ID“ bezeichnet.
- **Data Word (/ Input Word):** hierbei steht der Identifizierungswert in einem vorgegebenen Speicherbereich des EtherCAT Slaves (z.B. Prozessdatenbereich ab ADO 0x1000). Der Address-Offset (ADO) wird aus der Gerätebeschreibung entnommen (Attribut „IdentificationAdo“) oder kann ggf. auch eingestellt werden. Dieses Verfahren wird in der „ETG.1020“ auch als „Direct ID“ bezeichnet.

Beispiel

In folgendem Beispiel haben die ersten zwei Slaves (EtherCAT-Adresse 1001 u. 1005) keinen Identifikationswert, d.h. die EtherCAT-Adresse wird nur anhand der Topologie-Position (AutoInc-Adresse) vergeben. Wird die physikalische Reihenfolge dieser beiden Geräte (identischer Geräte-Typ) getauscht, z.B. Verkabelungsfehler oder ggf. beabsichtigter Gerätetausch, wird die EtherCAT-Adresse weiterhin gültig entsprechend der Position zugewiesen.

Bei den folgenden zwei Slaves (EtherCAT-Adresse 1042 u. 1043) ist ein Identifikationsverfahren konfiguriert, d.h. nur wenn die Identifikationswerte (hier: 2 u. 3) mit der konfigurierten Topologie-Position (AutoInc-Adresse) übereinstimmen, werden die Slaves vom Master in die zyklische Kommunikation aufgenommen. Auf diese Weise wird ein Verkabelungsfehler zwischen diesen beiden Slaves ausgeschlossen. Bei einem beabsichtigten Kreuztausch oder Gerätetausch (Ersatzgerät) muss der Identifikationswert entsprechend gesetzt werden.

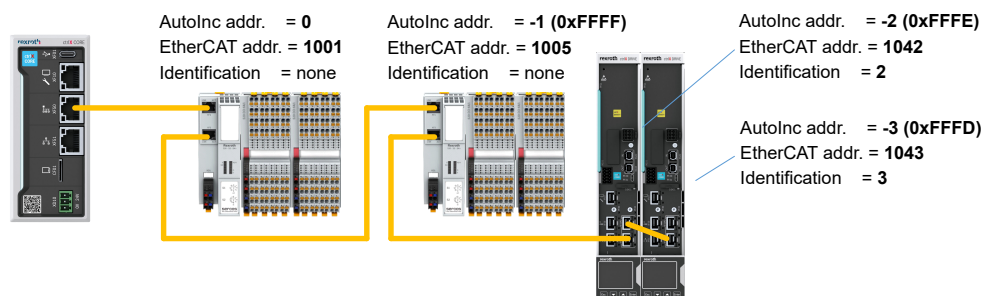


Abb. 5: Beispiel für eine EtherCAT Adressierung

4.4.3 Zustandsmaschine

Bei EtherCAT gibt es folgende definierte Kommunikationszustände:

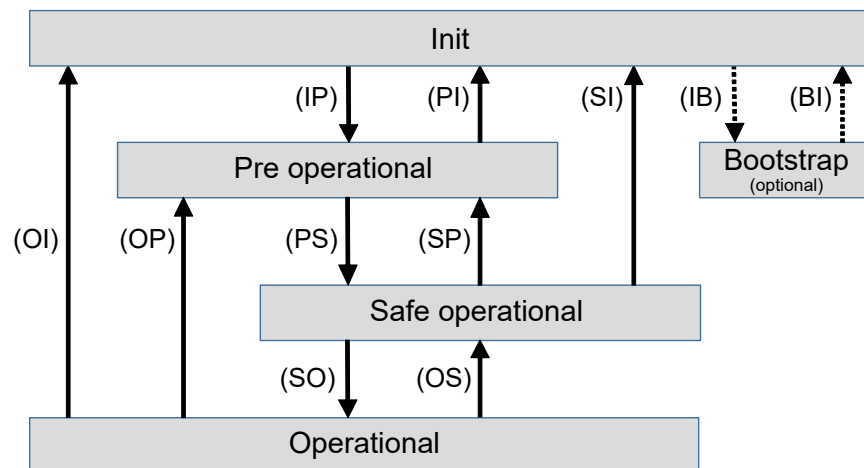


Abb. 6: EtherCAT Zustandsmaschine

Zustand	Beschreibung
Init (I)	<ul style="list-style-type: none"> keine Kommunikation zur Applikationsschicht Übergang zu ‚PreOP‘ (IP): Master konfiguriert Slave-Register, u.a. Adress-Register und Sync-Manager für Mailbox-Kommunikation
Pre-Operational (P)	<ul style="list-style-type: none"> ab ‚PreOP‘ ist Mailbox-Kommunikation zur Applikationsschicht möglich hier noch keine Prozessdaten-Kommunikation Übergang zu ‚SafeOP‘ (PS): Master konfiguriert u.a. Prozessdaten-Mapping (FMMU) und Sync-Manager für Prozessdaten-Kommunikation
Safe-Operational (S)	<ul style="list-style-type: none"> Prozessdaten-Kommunikation, allerdings werden nur die Eingangsdaten ausgewertet, die Ausgangsdaten bleiben hier noch im "sicheren Zustand" DC-Synchronisation wird eingeregelt
Operational (O)	<ul style="list-style-type: none"> Prozessdaten-Kommunikation: Ein- und Ausgangsdaten sind gültig DC-Synchronisation
Bootstrap (B)	<ul style="list-style-type: none"> optionaler Zustand in EtherCAT Slave Geräten (empfohlen für Firmware-Update) der Zustand ‚Bootstrap‘ kann nur von ‚Init‘ aus erreicht werden keine Prozessdaten-Kommunikation Mailbox-Kommunikation über FoE (ggf. ist FoE auch nur in ‚Bootstrap‘ möglich => geräte-spezifisch)

Zusätzliche Hinweise:

- ein EtherCAT Slave Gerät kann in einen "kleineren" Zustand als der (Master-) Buszustand geschaltet werden, z.B. ist der EtherCAT Buszustand ‚OP‘, kann ein Slave individuell in ‚PreOP‘ geschaltet werden.
- ein Zustandswechsel kann nur vom Master initiiert werden, ansonsten kann ein Slave selbst nur im Fehlerfall seinen Zustand herunterschalten.

4.4.4 Azyklische Kommunikation (Mailbox)

Azyklische Kommunikation wird bei EtherCAT auch als Mailbox Kommunikation bezeichnet und ist erst ab Zustand „Pre-Operational“ möglich. Es gibt verschiedene Protokolle, die über EtherCAT getunnelt werden können. Welche Protokolle ein Slave unterstützt, ist der Dokumentation (oder Gerätebeschreibung) des Slave Gerätes zu entnehmen.

Übersicht Mailbox Protokolle

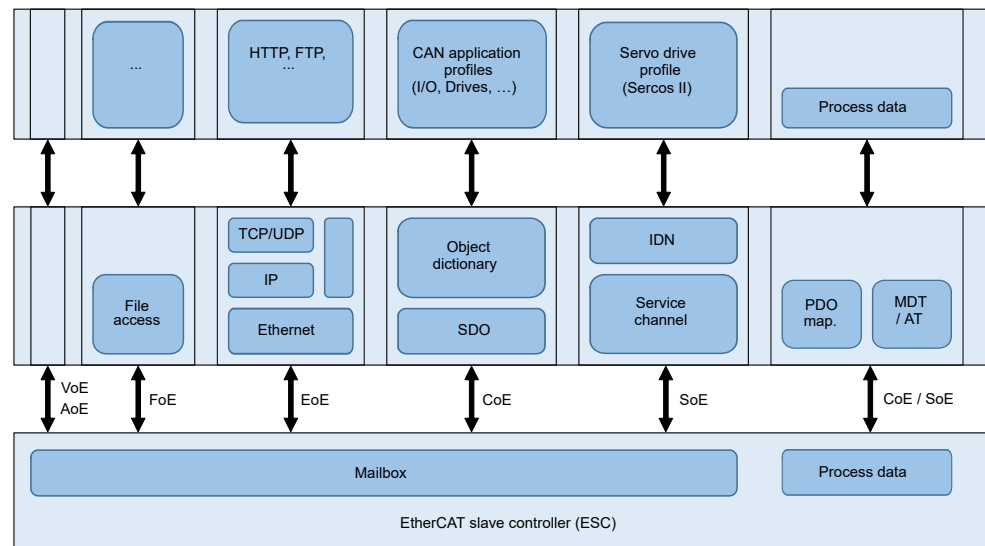


Abb. 7: Übersicht Mailbox Protokolle

CoE: CAN application protocol over EtherCAT

CoE Objekte werden auch als „Prozess Daten Objekte“ (PDO) oder „Service Daten Objekte“ (SDO) bezeichnet. Ein Objekt wird mit „Index“ und „Subindex“ adressiert, z.B. $0 \times 1018:01$. Hat ein Objekt mehr als einen Subindex, dann steht im Subindex 0 die Anzahl der Subindices.

Mit den optionalen CoE Diensten „SDO Information“ kann das Objektverzeichnis eines Slaves ausgelesen werden sowie Objektbeschreibungen und Elementbeschreibungen der Subindices.

Siehe auch [Kapitel 9.3 CoE SDO Abortcodes auf Seite 58](#).

SoE: Servo drive profile over EtherCAT

SoE ist ein adaptiertes Sercos II Profil für Servoantriebe, d.h. manche Sercos II Parameter haben bei SoE keine Bedeutung oder eine teilweise geänderte Bedeutung. Ein Parameter wird adressiert mit einer 16 bit Identifikationsnummer (IDN: Parametertyp S/P, Parametersatz, Datenblock-Nummer).

Im Unterschied zu Sercos stellen Mehrachsgeräte bei SoE nur ein Slave Gerät dar. Die unterlagerten Antriebe unterscheidet man durch eine Antriebsnummer (DriveNo: 0 / 1 / ... / 7), die alternativ auch als Kanal (Channel: A / B / ... / H) bezeichnet wird.

Siehe auch [Kapitel 9.5 SoE Fehlercodes auf Seite 60](#).

Folgende IDN haben bei SoE keine Relevanz:	
S-0-0003	Sende-Reaktionszeit AT (T1min)
S-0-0004	Umschaltzeit Senden-Empfangen (TATMT)
S-0-0005	Mindestzeit Istwerterfassung (T4min)
S-0-0009	Anfangsadresse im Master-Daten-Telegramm

S-0-0010	Länge Master-Daten-Telegramm
S-0-0088	TMTSY Erholzeit Empfangen-Empfangen
S-0-0090	TMTSG Kopierzeit Sollwerte
S-0-0127	C0100 Umschaltvorbereitung auf Komm.-Phase 3 Wird beim EtherCAT Zustandsübergang PreOP => SafeOP durchgeführt. Bei einem Fehler kann die Ursache mittels S-0-0021 ausgewertet werden.
S-0-0128	C5200 Umschaltvorbereitung auf Komm.-Phase 4 Wird beim EtherCAT Zustandsübergang SafeOP => OP durchgeführt. Bei einem Fehler kann die Ursache mittels S-0-0022 ausgewertet werden.
Folgende IDN haben bei SoE eine angepasste Bedeutung:	
S-0-0006	Sendezeitpunkt Antriebs-Telegramm (T1) T1 legt die Zeitverschiebung fest vom EtherCAT Sync-Signal bis zum Zeitpunkt, in dem die Applikation neue AT-Daten im ESC-Speicher zur Verfügung stellen soll.
S-0-0014	Schnittstellen-Status Dieser Parameter soll den DL Status, AL Status und AL Status Code von EtherCAT anzeigen.
S-0-0028	Fehlerzähler MST Zähler für fehlende Datagramme der zyklischen Daten, als auch RX-Fehlerzähler und Lost-Link Zähler von EtherCAT.
S-0-0089	T2 Sendezeitpunkt MDT T2 legt die Zeitverschiebung fest vom EtherCAT Sync-Signal bis zum Zeitpunkt, in dem neue Master Daten im ESC-Speicher verfügbar sind.

FoE: File access over EtherCAT

FoE ist ein Protokoll, vergleichbar zu TFTP, um Dateien übertragen zu können. Üblicherweise wird FoE für Firmware-Update verwendet, daher ist FoE ggf. auch nur im EtherCAT Zustand ‚Bootstrap‘ möglich (geräte-spezifisch).

Siehe auch ➔ [Kapitel 9.4 FoE Fehlercodes auf Seite 59.](#)

EoE: Ethernet over EtherCAT

Mittels EoE kann Standard Ethernet Kommunikation (z.B. TCP/IP) über EtherCAT getunnelt werden. Hierzu wird ein „virtueller Ethernet Switch“ im EtherCAT Master benötigt.

Ein IP-fähiges Gerät (z.B. Laptop) kann nicht an einer beliebigen Stelle im EtherCAT Netzwerk angeschlossen werden, sondern hierfür wird ein „Switch Port“ benötigt (z.B. spezielle Klemme mit „Switch Port“ ⇔ ist kein EtherCAT Port).

Alternativ kann bei einem EtherCAT Slave Gerät auch ein IP-Port konfiguriert werden, sofern das Gerät EoE unterstützt, um beispielsweise die Adressierung eines integrierten Webservers über einen konfigurierten IP-Port zu ermöglichen.



In beiden Fällen benötigt der EtherCAT Slave innerhalb des EtherCAT Netzwerkes eine „virtuelle MAC Adresse“, die in der EtherCAT Konfiguration festgelegt wird. Die EoE-Einstellungen werden dann beim Bus-Hochlauf (Übergang Init => PreOP) vom Master in die Slave Geräte geschrieben. EoE ist wie alle Mailbox-Protokolle erst ab Zustand PreOP möglich.

Um IP-Routing zwischen Engineering-Port und EtherCAT-Master zu nutzen, muss IP-Forwarding am Engineering-Port aktiviert sein.

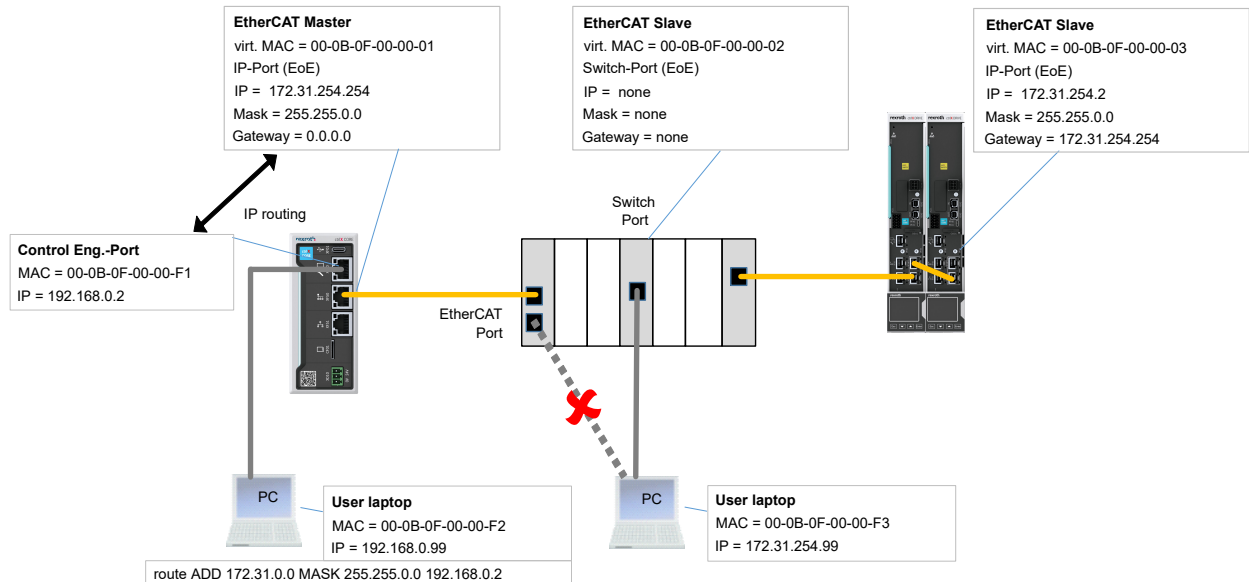


Abb. 8: EoE Konfigurations-Beispiel

VoE: Vendor specific protocol over EtherCAT

VoE dient zur Übertragung von Hersteller-spezifischen Protokollen.

AoE: ADS over EtherCAT

AoE ist eine geräte- und feldbusunabhängige Schnittstelle von Beckhoff (ADS: Automation Device Specification).

Slaves ohne Mailbox Kommunikation

EtherCAT Slave Geräte müssen kein Mailbox Protokoll unterstützen. Zum Beispiel unterstützen einfache digitale Ein-/Ausgangsklemmen überhaupt kein Mailbox Protokoll. Bei diesen Slave Geräten können dann nur die Register im EtherCAT Slave Controller (ESC) gelesen und geschrieben werden, was jeder EtherCAT Slave unterstützen muss.

4.4.5 Telegramm-Aufbau

EtherCAT verwendet Standard Ethernet Frames (IEEE 802.3) mit einem eigenen EtherType 0x88A4. Ethernet Telegramme eines anderen Typs werden vom EtherCAT Slave Controller nicht unterstützt und müssen durch EtherCAT getunnelt werden (z.B. EoE).

Der EtherCAT Frame besteht aus mehreren Sub-Telegrammen bzw. Datagrammen, die verschiedene Kommandos enthalten. Beispielsweise ist jede SyncUnit ein eigenes Datagramm als auch azyklische Kommunikation (Mailbox) oder sonstige Kommandos sind eigene Datagramme.

Siehe ➔ Kapitel 4.4.7 SyncUnits auf Seite 22.

In einem EtherCAT Frame können bis zu 15 Datagramme zusammengefasst werden. Bei mehr als 15 Datagrammen werden mehrere Frames benötigt.

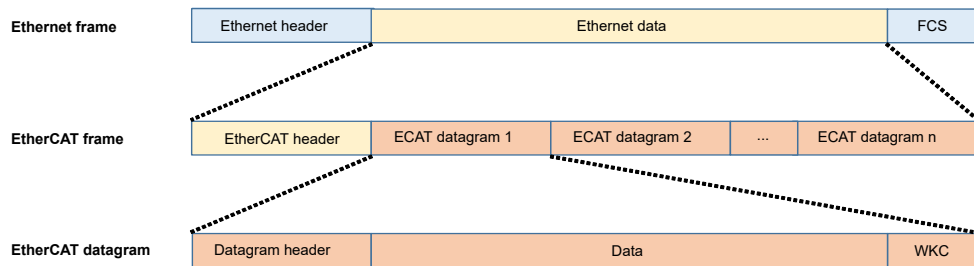


Abb. 9: EtherCAT Telegramm-Aufbau

4.4.6 WorkingCounter

Durch die Checksumme des Ethernet Frames (FCS : Frame Check Sequence) wird die korrekte Datenübertragung auf dem Bus sichergestellt.

Zur Überprüfung der korrekten Bearbeitung eines Kommandos in den Slave Geräten besitzt jedes Datagramm einen **WorkingCounter** (WKC). Der WKC wird vom Master mit dem Wert 0 abgeschickt. Jeder adressierte Slave inkrementiert den WKC dann abhängig vom Kommando. Nachdem der Frame dann alle Slaves durchlaufen hat und wieder im Master ankommt, kann der Master den Erwartungswert (Sollwert) des WKC mit dem Istwert vergleichen und somit die korrekte Bearbeitung des Kommandos überprüfen.



Für die Gültigkeit der zyklischen Prozessdaten wird der Status der WorkingCounter berücksichtigt.

Siehe hierzu auch ➔ [Allgemein](#)

4.4.7 SyncUnits

SyncUnit

Eine SyncUnit beschreibt eine logische Gruppierung von zyklischen Daten, die konsistent und synchron ausgetauscht werden sollen. Im einfachsten Fall sind alle EtherCAT Slaves in einer SyncUnit zusammengefasst oder es könnte auch einzelne SyncUnits für jeden Slave geben.

In der Regel werden optionale Slaves (z.B. Slaves, die im laufenden Betrieb ausfallen dürfen) oder Maschinenmodule in einer eigenen SyncUnit gruppiert. Auf diese Weise besitzen diese Teilnehmer bzw. diese SyncUnit ein eigenes Datagramm und damit einen eigenen WorkingCounter. Zum Beispiel führt ein fehlender Teilnehmer dann nur zu einem ungültigen WorkingCounter dieser SyncUnit und alle anderen SyncUnits haben weiterhin einen gültigen Datenstatus.

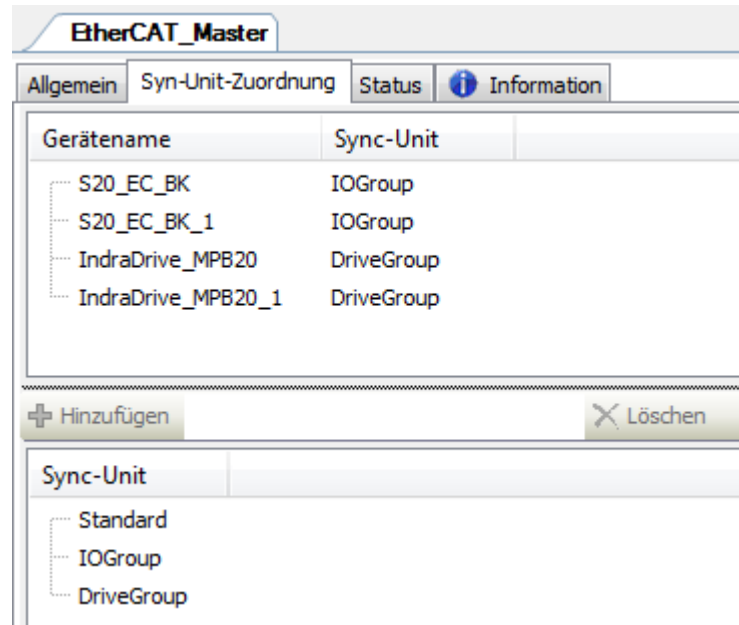


Abb. 10: Beispiel einer SyncUnit Konfiguration

Unterschiedliche EtherCAT Zykluszeiten

Mit SyncUnits können auch unterschiedliche EtherCAT Zykluszeiten realisiert werden. Hierbei werden die SyncUnits (= Datagramme) dann in eigenen Frames mit unterschiedlichen Zykluszeiten gesendet. Zum Beispiel werden schnelle I/O-Baugruppen und Antriebsgruppen in einzelnen SyncUnits mit unterschiedlichen Zykluszeiten gruppiert.



Der Rexroth EtherCAT Master unterstützt keine unterschiedlichen EtherCAT Zykluszeiten!

4.4.8 Hot Connect



Zum aktuellen Zeitpunkt wird "Hot Connect" nicht von ctrlX CORE unterstützt.

Hot Connect

Unter Hot Connect (HC) versteht man allgemein, dass während dem laufenden Betrieb (Bus-Zustand "Operational") Slaves ab- und angesteckt werden können. Hierbei sind auch die Anwendungsfälle enthalten, dass mehrere Teilnehmer bzw. Maschinenmodule nicht vorhanden sein müssen bzw. eine Inbetriebnahme eines Teils der Anlage bereits möglich ist.

HC-Gruppe

Durch die Konfiguration von HC-Gruppen werden Slaves automatisch in einer eigenen SyncUnit gruppiert. Eine HC-Gruppe kann als "optional" deklariert werden, so dass der EtherCAT Master eine nicht vorhandene HC-Gruppe nicht als Fehler erkennt (flexible Topologien). Der erste Slave einer HC-Gruppe wird durch ein Identifikationsverfahren erkannt, siehe [Kapitel 4.4.2 Adressierungs- und Identifikationsverfahren auf Seite 16](#).

Des Weiteren kann bei einer HC-Gruppe auch festgelegt werden, ob diese Gruppe nur an einer bestimmten Topologie-Position (Port des Vorgänger EtherCAT Gerätes) angeschlossen werden darf oder eine beliebige Position haben kann.



Der Rexroth EtherCAT Master unterstützt nicht die Konfiguration von HC-Gruppen!

4.4.9 EtherCAT Slave Controller

Aufbau

Nachfolgende Abbildung zeigt eine schematische Darstellung eines EtherCAT Slave Controllers (ESC).

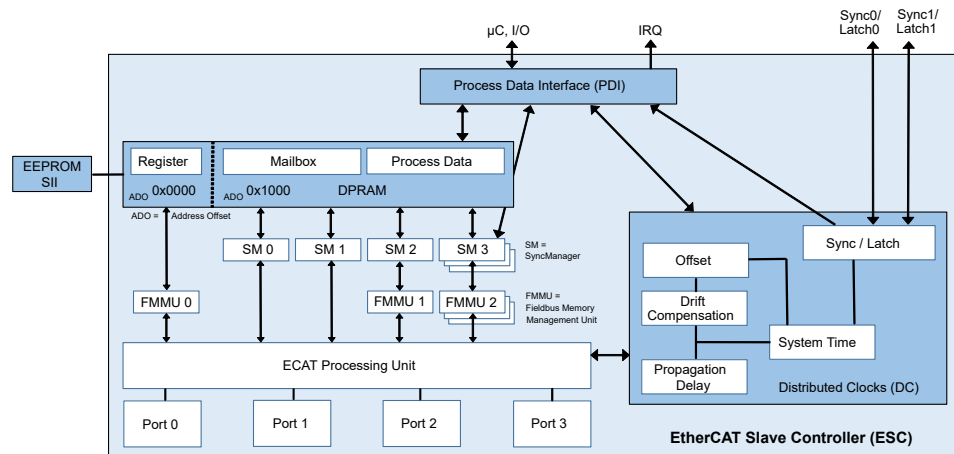


Abb. 11: EtherCAT Slave Controller Aufbau

Register

Die Register liegen im lokalen Adressbereich 0x0000 bis 0x0FFF des ESC und werden u.a. verwendet für

- Konfiguration des ESC, z.B. EtherCAT Adresse („Fixed Physical Address“), SyncManager, FMMU, DC.
- Zugriff auf EEPROM / SII (Slave Information Interface), dieses enthält u.a. die Geräte-Identifikation sowie Defaultwerte für die ESC Konfiguration.
- Steuer- und Status-Daten der EtherCAT Zustandsmaschine.

SyncManager (SM)

Die SyncManager stellen die Datenkonsistenz im DPRAM des ESC sicher, d.h. gleichzeitiger Zugriff auf Speicherbereiche wird verhindert; z.B. wenn die ECAT Processing Unit (Telegrammverarbeitung) und die lokale Applikation gleichzeitig auf einen Speicherbereich im DPRAM zugreifen wollen.

Die SM Konfiguration ist ab Register ADO 0x0800.

Es sind bis zu 16 unabhängige SyncManager Kanäle möglich:

- ein Slave ohne Mailbox hat typischerweise 2 SyncManager:
 - SM0 = Prozessdaten-Ausgänge (oder Eingänge, wenn es keine Ausgänge gibt),
 - SM1 = Prozessdaten-Eingänge.
- ein Slave mit Mailbox hat typischerweise 4 SyncManager:
 - SM0 = Mailbox-Out,
 - SM1 = Mailbox-In,

- SM2 = Prozessdaten-Ausgänge (oder Eingänge, wenn es keine Ausgänge gibt),
- SM3 = Prozessdaten-Eingänge.
- Prozessdaten einer Richtung (Eingänge oder Ausgänge) können auch in mehrere SyncManager aufgeteilt sein; z.B. wenn die Datenkonsistenz einzelner Speicherbereiche separat überwacht werden soll. Hierbei spricht man dann auch von "SyncUnits" im Slave.

Es werden verschiedene Typen unterschieden:

- Mailbox (azyklische Kommunikation): 1 Puffer System mit Handshake Mechanismus.
- Prozessdaten: 1 oder 3 Puffer System, mit 3 Puffer können unabhängige Lese- und Schreibzugriffe stattfinden; z.B. für Free Run Kommunikation, wenn die lokale Applikation und die ECAT Telegrammverarbeitung unabhängig voneinander laufen.

Fieldbus Memory Management Unit (FMMU)

Die Prozessdaten werden auf dem EtherCAT Bus einem logischen Adressraum zugeordnet (bis zu 4 Gigabyte), der sich über beliebig viele Slaves verteilen kann. Die FMMU ist für das Mapping einer globalen logischen Adresse auf die lokale physikalische Adresse des ESC zuständig.

Die FMMU Konfiguration erfolgt ab Register ADO 0x0600.

Es sind bis zu 16 unabhängige FMMU Kanäle möglich, z.B. separate FMMUs für Datenrichtung und Speicherbereiche.

Distributed Clocks (DC)

Zur Synchronisation der EtherCAT Teilnehmer wird das Verfahren der "Verteilten Uhren" mit Festlegung einer globalen Referenzuhr angewendet.

Die Systemzeit, die am 1. Jan. 2000 um 00:00 Uhr beginnt (Einheit 1 ns / 64 Bit), wird in allen Geräten eingeregelt, d.h. der Drift und der Offset der lokalen ESC Uhr zur Referenzuhr wird entsprechend kompensiert (Genauigkeit < 100 ns).

Die DC Konfiguration erfolgt ab Register ADO 0x0900.

Die DC Einheit generiert entsprechende Events (Sync / Latch) für die Ein-/Ausgangsdaten-Verarbeitung. Das zweite Sync Signal (Sync1) kann mit einer definierten Verschiebung nach dem Sync0 Event konfiguriert werden. Die Latch Signale werden z.B. für "Time Stamping" der Systemzeit verwendet.

Kommandos

Es gibt verschiedene EtherCAT Kommandos für alle Adressierungsverfahren. Neben separaten Lese- und Schreib-Kommandos gibt es auch kombinierte Befehle, in denen Lese- u. Schreibzugriffe mit einem Datagramm ausgeführt werden können. Diese kombinierten Zugriffe sind effizienter, werden aber nicht von allen Slave Geräten unterstützt.

Wert	Code	Name	Beschreibung
0	NOP	No Operation	Slave ignoriert den Befehl.
1	APRD	Auto Increment Read	Slave Adressierung mit Auto-Inkrement Adresse. Der Slave, der Adresse 0 empfängt, verarbeitet die Daten. Jeder Slave inkrementiert die Adresse um 1.
2	APWR	Auto Increment Write	
3	APRW	Auto Increment Read Write	
4	FPRD	Fixed Physical Read	Slave Adressierung mit konfigurierter EtherCAT Adresse („Fixed Physical Address“).
5	FPWR	Fixed Physical Write	
6	FPRW	Fixed Physical Read Write	

Wert	Code	Name	Beschreibung
7	BRD	Broadcast Read	Alle Slaves führen jeweils eine OR –Verknüpfung der empfangenen Datagrammdaten mit dem adressierten Speicherbereich durch und Schreiben das Ergebnis an die gleiche Stelle des Datagramms.
8	BWR	Broadcast Write	Die Daten werden in alle Slaves geschrieben.
9	BRW	Broadcast Read Write	Alle Slaves führen jeweils eine OR –Verknüpfung der empfangenen Datagrammdaten mit dem adressierten Speicherbereich durch und Schreiben das Ergebnis an die gleiche Stelle des Datagramms und in den Speicher. (Dieser Befehl wird gewöhnlich nicht verwendet).
10	LRD	Logical Memory Read	Slave Adressierung mit logischer Speicheradresse, d.h. alle Slaves, deren FMMU mit der Adresse des logischen Speicherabbaus aus dem Datagramm konfiguriert sind, verarbeiten die Daten.
11	LWR	Logical Memory Write	
12	LRW	Logical Memory Read Write	
13	ARMW	Auto Increment Read Multiple Write	Slave Adressierung mit Auto-Inkrement Adresse. Der Slave, der Adresse 0 empfängt, fügt dem Datagramm Eingangsdaten hinzu („Read“), alle anderen Slaves entnehmen dem Datagramm an derselben Stelle Ausgangsdaten („Multiple Write“). Jeder Slave inkrementiert die Adresse um 1.
14	FRMW	Fixed Physical Read Multiple Write	Der Slave mit der konfigurierten EtherCAT Adresse („Fixed Physical Address“) fügt dem Datagramm Eingangsdaten hinzu („Read“), alle anderen Slaves entnehmen dem Datagramm an derselben Stelle Ausgangsdaten („Multiple Write“).
15-255	-	-	Reserviert.

Abhängig vom Kommandotyp wird der WorkingCounter des Datagramms inkrementiert:

Kommandotyp	Ergebnis	Inkrementierung
Lesen	Lesen nicht erfolgreich (oder Slave nicht adressiert)	nein
	Lesen erfolgreich	+ 1
Schreiben	Schreiben nicht erfolgreich (oder Slave nicht adressiert)	nein
	Schreiben erfolgreich	+ 1
Lesen + Schreiben	Lesen und Schreiben nicht erfolgreich (oder Slave nicht adressiert)	nein
	Lesen erfolgreich	+ 1
	Schreiben erfolgreich	+ 2
	Lesen und Schreiben erfolgreich	+ 3

5 EtherCAT Synchronisation

5.1 Allgemein

Für jeden EtherCAT Slave kann individuell ein Synchronisations-Modus eingestellt werden. Welche Modi von einem Slave Gerät unterstützt werden, ist der Dokumentation bzw. der Gerätebeschreibung des Slave Gerätes zu entnehmen. Man unterscheidet allgemein folgende Synchronisations-Modi:

- Free Run (keine Synchronisation)
- Synchronisation mit SM-Event (SM : SyncManager)
Die Hauptaufgabe des SyncManagers (SM) ist es, die Daten-Konsistenz im DPRAM des ESC sicherzustellen. Es kann bis zu 16 unabhängige SyncManager Kanäle geben. In der Standard-Konfiguration werden 4 verwendet: SM0 = Mailbox- Out, SM1 = Mailbox-In, SM2 = Prozessdaten-Out, SM3 = Prozessdaten-In.)
- Synchronisation mit DC-Sync (DC : Distributed Clocks / "Verteilten Uhren")

Folgendes Beispiel zeigt die DC-Konfiguration des Rexroth IndraDrive MPx20 SoE. Der Konfigurator zeigt die vordefinierten Konfigurationen aus der Gerätebeschreibung (ESI) an. Neben dem "Free Run" Modus werden drei DC-Modi mit unterschiedlichen "Sync0Shift"-Zeiten angeboten (Zeitverschiebung im Slave zum Sync0 Event).

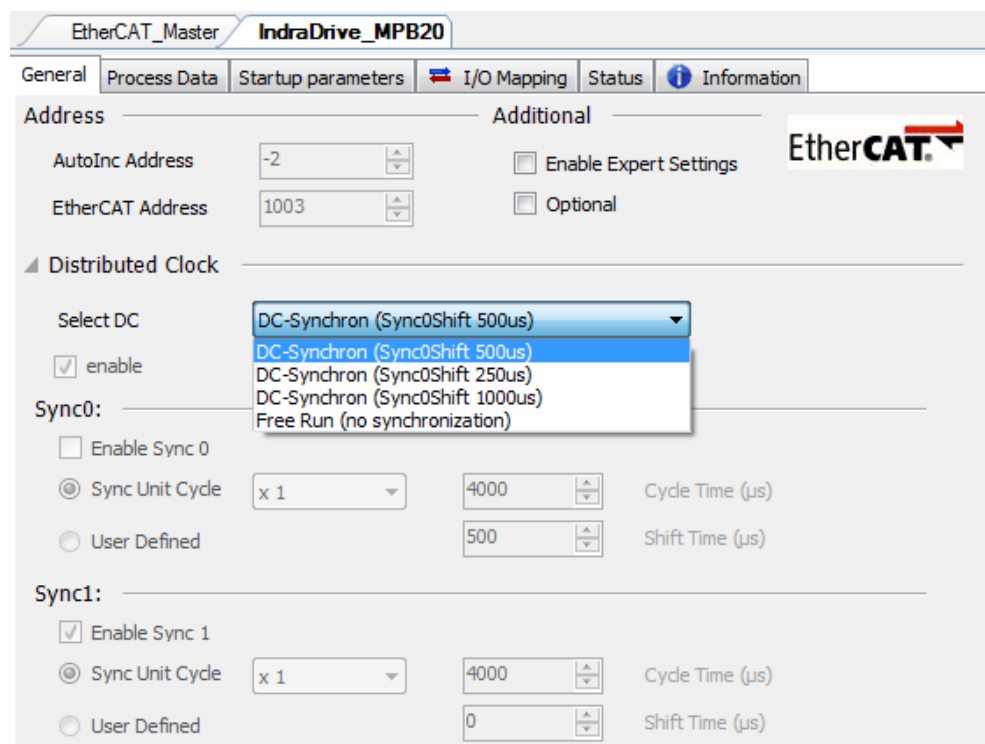


Abb. 12: DC-Konfiguration mit Rexroth IndraDrive MPx20 SoE

5.2 Free Run

Der EtherCAT Kommunikationszyklus und die Ein-/Ausgangsdaten-Verarbeitung im Slave Gerät laufen unabhängig voneinander. D.h. der lokale Applikationszyklus im Slave Gerät kann schneller oder langsamer sein als der EtherCAT Buszyklus.



In folgenden Abbildungen werden mit den „In“ / „Out“-Balken jeweils die Kopier- und Verarbeitungszeiten in den Slave-Geräten dargestellt.

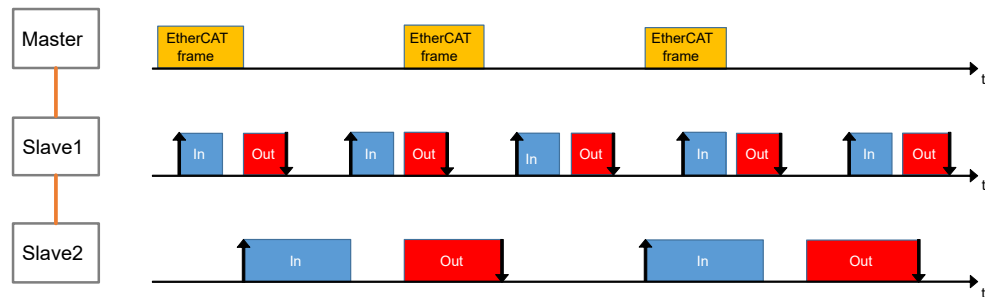


Abb. 13: Modus „Free Run“

Bei CoE sind u.a. folgende Parameter des SyncManager Objekts 0x1C3m relevant (mit der Standard-Konfiguration "SM2/3" für die Prozessdaten gilt: 0x1C32 für Ausgänge u. 0x1C33 für Eingänge):

SI	Beschreibung	Zugriff	Verwendung	Erklärung / Defaultwert
1	Synchronization Type	R / RW	M	Synchronisations-Typ: 0x00: Free Run
2	Cycle Time	R / RW	O	Lokale Zykluszeit des Applikations-Controllers in ns
4	Synchronization Types supported	R	M	Unterstützte Synchronisations-Typen: Bit 0 = 1 : Free Run unterstützt
5	Minimum Cycle Time	R	C	Minimale Zykluszeit: maximale Laufzeit des lokalen Applikations-zyklus (erforderlich, wenn SI=2 geschrieben werden kann)

SI = Subindex, R = Read only, RW = Read/Write, M = erforderlich, O = optional, C = bedingt

5.3 Synchronisation mit SM-Event

Diese Synchronisation wird oft auch als "SM2/3"-Event bezeichnet, da die SyncManager 0/1 üblicherweise für Mailbox Out/In verwendet werden und SyncManager 2/3 dann für die Prozessdaten Out/In.

Die Ausgangsdaten-Verarbeitung wird im Slave durch den Empfang des EtherCAT Telegramms getriggert durch den SyncManager. Der Zeitpunkt, wann die Ausgangsdaten gültig werden, kann variieren aufgrund von Jitter beim Sendezeitpunkt des Masters, durch die Laufzeit des Telegramms bei der Übertragung, als auch durch Verarbeitungszeiten in den Slaves. Für die Übernahme der Eingangsdaten kann optional ein "Input Shift" konfiguriert werden (geräte-individuell), damit die Eingangsdaten beim nächsten SM-Event (nächster Telegramm-Durchlauf) möglichst "aktuell" sind.

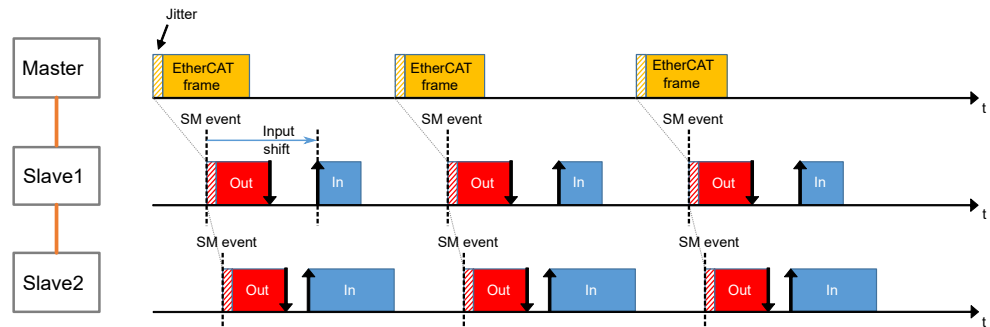


Abb. 14: Modus "Synchronisation mit SM-Event"

Bei CoE sind u.a. folgende Parameter des SyncManager Objekts 0x1C3m relevant (mit der Standard-Konfiguration "SM2/3" für die Prozessdaten gilt: 0x1C32 für Ausgänge und 0x1C33 für Eingänge):

SI	Beschreibung	Zugriff	Verwendung	Erklärung / Defaultwert
1	Synchronization Type	RW	M	Synchronisations-Typ: Ausgänge (0x1C32): 0x01 : Synchron mit SM2 Event Eingänge (0x1C33): 0x01 : Synchron mit SM3 Event (wenn keine Ausgänge verfügbar sind) 0x22 : Synchron mit SM2 Event (wenn Ausgänge verfügbar sind)
2	Cycle Time	R / RW	O	EtherCAT Kommunikationszyklus in ns (Zeit zwischen zwei SM Events)
3	Shift Time	RW	-/ M	Zeitverschiebung Eingangsdaten-Übernahme ("Shift Input Latch") (nur Eingänge / 0x1C33) in ns
4	Synchronization Types supported	R	M	Unterstützte Synchronisations-Typen: Bit 1 = 1: Synchron SM unterstützt
5	Minimum Cycle Time	R	M	Minimale Zykluszeit, die vom Slave unterstützt wird (maximale Zeit des lokalen Applikationszyklus in ns)
8	Get Cycle Time	RW	C	wird im SM- und DC-Modus bei einer variablen Zykluszeit verwendet: Bit 0: Messung der lokalen Zeit gestartet Bit 1: Reset der Fehlerzähler
11	SM-Event missed	R	O	Dieser Fehlerzähler wird inkrementiert, wenn das SM Event nicht innerhalb der erwarteten Zeit ankommt (in der Folge können die Daten nicht mehr kopiert werden)

SI	Beschreibung	Zugriff	Verwendung	Erklärung / Defaultwert
12	Cycle Time Too Small	R	M	Dieser Fehlerzähler wird inkrementiert, wenn die Zykluszeit zu klein ist, so dass der lokale Applikationszyklus nicht fertig ausgeführt werden kann (Eingangsdaten können bis zum nächsten SM Event nicht aufbereitet werden)
14	RxPDO Toggle Failed	R	O	Dieser Fehlerzähler wird inkrementiert, wenn der Slave RxPDO Toggle unterstützt und keine Statusänderung vom Master mehr empfangen wurde
32	Sync Error	R	C	Synchronisations-Fehler: sollte unterstützt werden, wenn SI = 11 vorhanden ist; kann in TxPDO gemappt werden. Wert 0 : kein Sync Fehler oder Sync nicht unterstützt Wert 1 : Sync Fehler

SI = Subindex, R = Read only, RW = Read/Write, M = erforderlich, O = optional, C = bedingt

5.4 Synchronisation mit DC-Sync

Für das Synchronisations-Verfahren "Verteilte Uhren" (DC : Distributed Clocks) gibt es allgemein verschiedene Möglichkeiten die Referenzuhr festzulegen:

- **Master Shift**
Der erste DC-Slave am Bus dient als Referenzuhr und Nachregelung des Timers des EtherCAT Masters.
- **Bus Shift**
Der EtherCAT Master stellt die Systemzeit und schreibt diese in den ersten DC-Slave am Bus (Schieben der Buszeit).
- **Externe Referenzuhr** (IEEE 1588)

Auf Basis der Referenzuhr wird in allen Teilnehmern eine globale Systemzeit eingeregelt. Mit Hilfe dieser Systemzeit werden in den EtherCAT Slave Controllern dann entsprechende "DC Sync" Events generiert zur Verarbeitung der Ein- u. Ausgangsdaten.

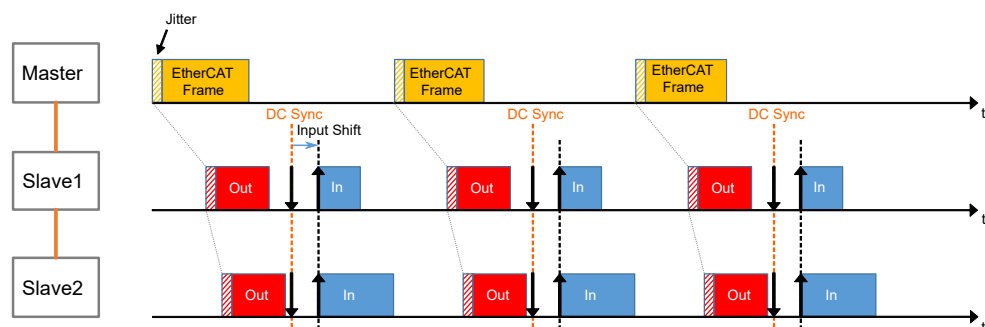


Abb. 15: Modus "Synchronisation mit DC-Sync" (vereinfachte Darstellung)

Neben dem Sync-Event "Sync0" wird i.d.R. noch ein untergeordnetes Sync-Event "Sync1" unterstützt (Funktionalität eines Standard EtherCAT Slave Controllers (ESC)). Welche Sync-Modi ein Slave Gerät unterstützt, sollte in dessen Gerätebeschreibungsdatei (ESI) enthalten sein oder ist in der Produkt-Dokumentation nachzuschlagen. Man unterscheidet allgemein folgende DC-Modi:

- Synchron mit Sync0 Event
- Synchron mit Sync0 Event, Input Shift mit Sync1 Event
- Synchron mit Sync0 Event, Output Shift mit Sync1 Event
- Synchron mit Sync1 Event
- DC mit untergeordneten Zyklen (Sync1 Event nur jeden n-ten Sync0 Zyklus)

Der Zeitpunkt des Sync-Events wird im EtherCAT Master global eingestellt als "Sync Shift Time" (oder "Sync Offset") bezogen auf den Master Applikationszyklus. Für jeden Slave kann dann individuell eine Sync0 und Sync1 Zykluszeit eingestellt werden. Diese Sync-Zykluszeit ist ein Vielfaches des EtherCAT Kommunikationszyklus. Zudem ist auch eine Verschiebung ("Shift Time") bezogen auf den globalen Sync-Zeitpunkt möglich.



Die DC Einstellungen sind normalerweise vordefinierte Konfigurationen aus der Gerätebeschreibung (ESI) eines Slaves und sollten nur von Experten geändert werden!

Abb. 16: Slave Konfiguration der DC Einstellungen

Bei CoE sind u.a. folgende Parameter des SyncManager Objekts 0x1C3m relevant (mit der Standard-Konfiguration "SM2/3" für die Prozessdaten gilt: 0x1C32 für Ausgänge und 0x1C33 für Eingänge):

SI	Beschreibung	Zugriff	Verwendung	Erklärung / Defaultwert
1	Synchronization Type	RW	M	Synchronisations-Typ: 0x02: DC Synchron mit Sync0 Event 0x03: DC Synchron mit Sync1 Event

SI	Beschreibung	Zugriff	Verwendung	Erklärung / Defaultwert
2	Cycle Time	R	O	<p>Zykluszeit Sync0 Event, Zeit zwischen zwei Sync0 Events in ns (Register 0x09A3:0x09A0)</p> <p>Bei untergeordneten Zyklen: Zeit zwischen zwei Sync1 Events in ns (Register 0x09A7:0x09A4)</p> <p>Die Festlegung der Zykluszeiten Sync0/1 und ggf. Shift-Zeiten von Sync0/1 erfolgt im Konfigurator.</p>
3	Shift Time	RW	M	<p>Zeitverschiebung in ns:</p> <p>Ausgänge: Zeit zwischen Sync0 Event und "Outputs valid" (Initialisiert mit 0x1C32:09)</p> <p>Eingänge: Zeit zwischen Sync0 Event und "Input Latch" (initialisiert mit 0x1C33:05 – 0x1C33:06)</p>
4	Synchronization Types supported	R	M	<p>Unterstützte Synchronisations-Typen:</p> <p>Bit 4-2 : unterstützte DC-Modi</p> <p>001 = DC Sync0</p> <p>010 = DC Sync1</p> <p>100 = untergeordneter Zyklus</p> <p>Bit 5-6: Shift Einstellungen</p> <p>00 = Output Shift nicht unterstützt</p> <p>01 = Input / Output Shift mit lokalem Timer</p> <p>10 = Input / Output Shift mit Sync1</p>
5	Minimum Cycle Time	R	M	<p>Minimale Zykluszeit, die vom Slave unterstützt wird</p> <p>(maximale Zeit des lokalen Applikationszyklus in ns)</p>
6	Calc and Copy Time	R	M	<p>Berechnungs- und Kopier-Zeit in ns:</p> <p>minimale Zeit zw. SM2 Event und Sync0 Event.</p> <p>bei Sync1: minimale Zeit zw. Sync0 und Setzen der Ausgänge</p>
8	Get Cycle Time	RW	C	<p>wird im SM- und DC-Modus bei einer variablen Zykluszeit verwendet</p> <p>Bit 0: Messung der lokalen Zeit gestartet</p> <p>Bit 1: Reset der Fehlerzähler</p>
9	Delay Time	R	M	<p>Hardware Verzögerungszeit in ns:</p> <p>Zeit zwischen Trigger (Sync0 oder Sync1 Event) und Gültigkeit der Daten im Prozess (z.B. elektrische Signale)</p>
11	SM-Event missed	R	O	<p>Dieser Fehlerzähler wird inkrementiert, wenn das SM Event nicht innerhalb der erwarteten Zeit ankommt</p> <p>(in der Folge können die Daten nicht mehr kopiert werden)</p>

SI	Beschreibung	Zugriff	Verwendung	Erklärung / Defaultwert
12	Cycle Time Too Small	R	M	Dieser Fehlerzähler wird inkrementiert, wenn die Zykluszeit zu klein ist, so dass der lokale Applikationszyklus nicht fertig ausgeführt werden kann (Eingangsdaten können bis zum nächsten SM Event nicht aufbereitet werden)
13	Shift Time Too Short	R	O	Dieser Fehlerzähler wird inkrementiert, wenn das SM Event zu spät empfangen wird, so dass das Sync0 Event empfangen wird bevor "Calc+Copy Time" fertig ist (Master Telegramm zu spät; Konfigurations-Problem)
14	RxPDO Toggle Failed	R	O	Dieser Fehlerzähler wird inkrementiert, wenn der Slave RxPDO Toggle unterstützt und keine Statusänderung vom Master mehr empfangen wurde.
15	Minimum cycle distance	R	O	Minimaler Abstand zwischen zwei SM Events in ns (um Jitter zu überwachen)
16	Maximum cycle distance	R	O	Maximaler Abstand zwischen zwei SM Events in ns (um Jitter zu überwachen)
17	Minimum SM Sync distance	R	O	Minimaler Abstand zwischen SM Event und Sync0 Event in ns (um Jitter zu überwachen)
18	Maximum SM Sync distance	R	O	Maximaler Abstand zwischen SM Event und Sync0 Event in ns (um Jitter zu überwachen)
32	Sync Error	R	C	Synchronisations-Fehler: sollte unterstützt werden, wenn SI = 11 oder SI = 13 vorhanden ist; kann in TxPDO gemappt werden. Wert 0 : kein Sync Fehler oder Sync nicht unterstützt Wert 1 : Sync Fehler

SI = Subindex, R = Read only, RW = Read/Write, M = erforderlich, O = optional, C = bedingt

Es gelten folgende Randbedingungen bei den DC Modi:

(1) Synchron mit Sync0 Event

Optionale Zeitverschiebungen von "Outputs valid" und "Input Latch" durch Shift Time (0x1C3m:03).

Synchronization Type:

- Outputs (0x1C32:01) = 2 (Sync0 Event)
- Inputs (0x1C33:01) = 2 (Sync0 Event)

Cycle Time:

- Sync0 Cycle Time (0x1C3m:02) > Min Cycle Time (0x1C3m:05)

Shift Time:

- Output Shift Time (0x1C32:03) > Output Delay Time (0x1C32:09)
- Input Shift Time (0x1C33:03) > [Min Cycle time (0x1C32:05) – Input Calc and Copy Time (0x1C33:06)]
- Input Shift Time (0x1C33:03) < [Sync0 Cycle Time (0x1C33:02) – Input Calc and Copy Time (0x1C33:06)]

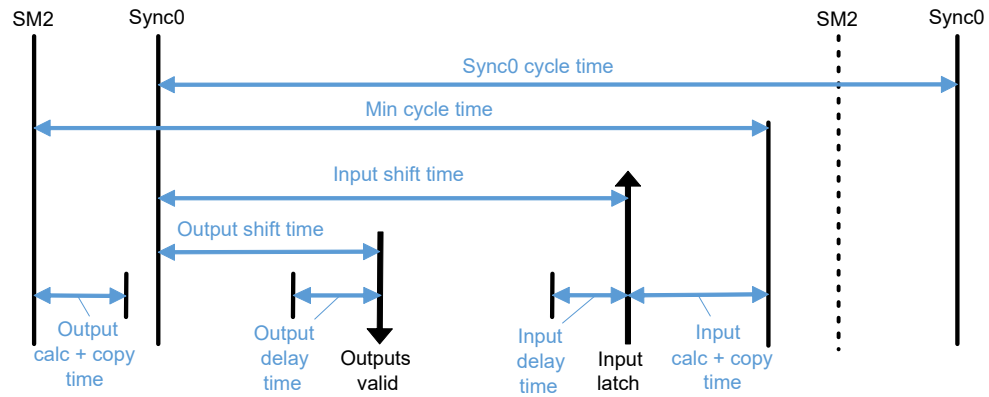


Abb. 17: Randbedingungen (1) Synchron mit Sync0 Event

(2) Synchron mit Sync0 Event, Input Shift mit Sync1 Event

Zeitverschiebung von "Input Latch" mit Sync1 Event (im Vergleich zur "Shift Time" wird hier der "Start Latch" Zeitpunkt beschrieben, d.h. ohne "Input Delay Time").

Synchronization Type:

- Outputs (0x1C32:01) = 2 (Sync0 Event)
- Inputs (0x1C33:01) = 3 (Sync1 Event)

Cycle Time:

- Sync0 Cycle Time (0x1C3m:02) > Min Cycle Time (0x1C3m:05)

Shift Time:

- Shift Sync0 <> Sync1 (Konfig.) \geq [Min Cycle Time (0x1C3m:05) – Output Calc and Copy Time (0x1C32:06) – Input Calc and Copy Time (0x1C33:06) – Input Delay Time (0x1C33:09)]

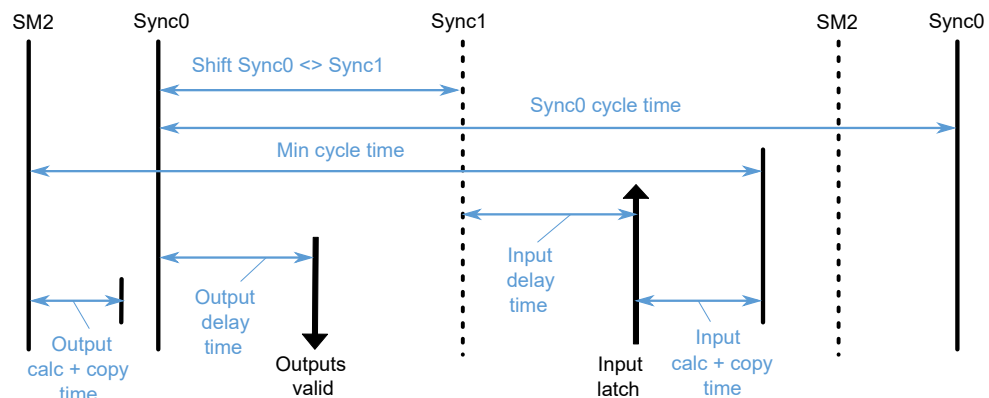


Abb. 18: Randbedingungen (2) Synchron mit Sync0 Event, Input Shift mit Sync1 Event

(3) Synchron mit Sync0 Event, Output Shift mit Sync1 Event

Zeitverschiebung von "Outputs valid" mit Sync1 Event (im Vergleich zur "Shift Time" wird hier der "Start Outputs" Zeitpunkt beschrieben, d.h. ohne "Output Delay Time").

Synchronization Type:

- Outputs (0x1C32:01) = 3 (Sync1 Event)
- Inputs (0x1C33:01) = 2 (Sync0 Event)



Für weitere Details zur Konfiguration "Output Shift mit Sync1 Event" bitte an die ETG wenden.

(4) Synchron mit Sync1 Event

Zeitverschiebungen von "Outputs valid" und "Input Latch" durch Sync1 Event.

Synchronization Type

- Outputs (0x1C32:01) = 3 (Sync1 Event)
- Inputs (0x1C33:01) = 3 (Sync1 Event)

Cycle Time

- Sync0 Cycle Time (0x1C3m:02) > Min Cycle Time (0x1C3m:05)

Shift Time

- Shift Sync0 <> Sync1 (Konfig.) >= Output Calc and Copy Time (0x1C32:06)

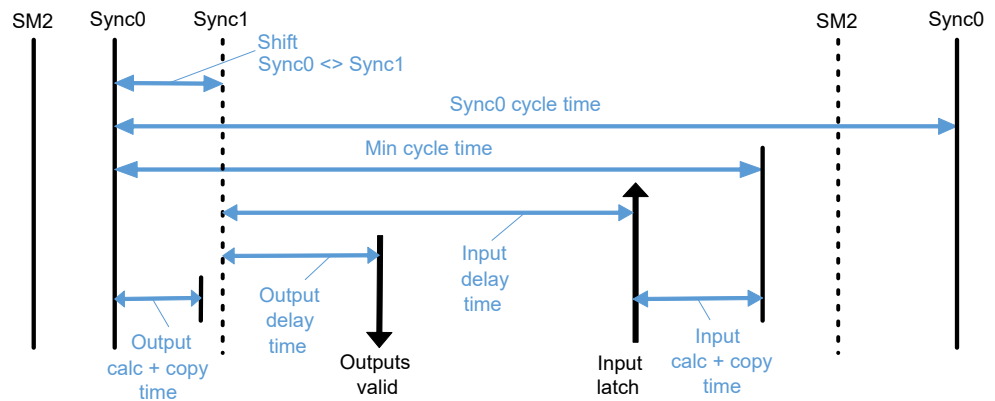


Abb. 19: Randbedingungen (4) Synchron mit Sync1 Event

(5) DC mit untergeordneten Zyklen

Das Sync0 Event dient als Trigger für den lokalen Applikationszyklus (z.B. schnelle Regelungen), während das Sync1 Event für einen (langsameren) Kommunikationszyklus verwendet wird. D.h. die Sync1 Zykluszeit ist ein Vielfaches der Sync0 Zykluszeit (Konfig.).

Synchronization Type:

- Outputs (0x1C32:01) = 3 (Sync1 Event)
- Inputs (0x1C33:01) = 3 (Sync1 Event)

Cycle Time:

- Cycle Time (0x1C3m:02) ist Sync1 Zykluszeit

Shift Time:

- Die Zeitverschiebungen von "Outputs valid" und "Input Latch" können nur durch "Input / Output Shift Time" (0x1C3m:03) festgelegt werden.

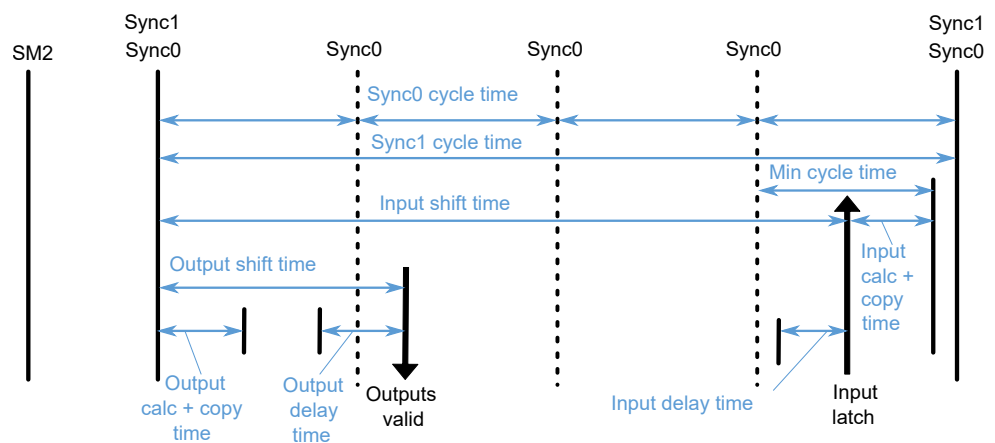


Abb. 20: Randbedingungen (5) DC mit ungeordneten Zyklen

6 EtherCAT Konfiguration

6.1 Übersicht

Anschlüsse

Der Rexroth EtherCAT-Master unterstützt keine Kabelredundanz. Es kann am EtherCAT-Master nur 1 Ethernet-Port für die EtherCAT Verkabelung verwendet werden:



Abb. 21: Anschlüsse an einer ctrlX CORE Steuerung

- ① XF10 Engineering Port
- ② XF50 EtherCAT Anschluss

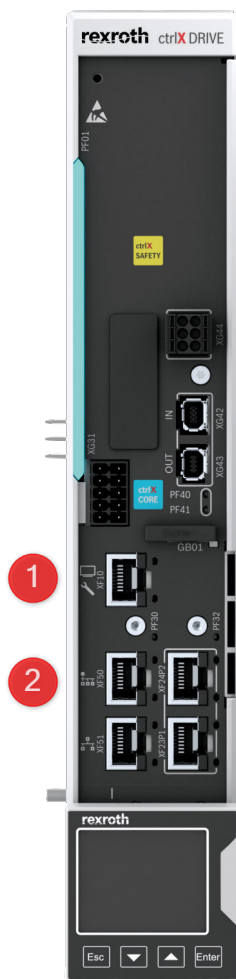


Abb. 22: Anschlüsse an einem ctrlX DRIVE mit integrierter ctrlX CORE Steuerung

- ① XF10 Engineering Port
- ② XF50 Ausgang des ersten EtherCAT Slave (ctrlX DRIVE)

EtherCAT-Master konfigurieren

Siehe ➔ [Kapitel 7.2 EtherCAT Master konfigurieren auf Seite 40](#)

EtherCAT Feldbusteilnehmer konfigurieren

Siehe ➔ [Kapitel 7.3.1 EtherCAT Feldbusteilnehmer konfigurieren auf Seite 41](#)

7 EtherCAT Master

7.1 EtherCAT Master – Features



Lizenzierung

Die EtherCAT Master-App und die Nutzung der enthaltenen Funktionen ist Lizenzpflichtig!

Für jede EtherCAT Master-Instanz ist folgende Lizenz erforderlich:

Typschlüssel: **SWL-XC*-ECM-ETHERCATMAS**-BANN**

Bestellnummer: **R911400508**

Funktionsumfang

Die Implementation des EtherCAT Master beinhaltet folgende Funktionalität:

- zyklische Kommunikation „Free Run“
- Synchronbetrieb („Distributed Clocks“ / Verteilte Uhren)
- Import der Gerätebeschreibungsdatei (ESI) in die ctrlX I/O Engineering Gerätedatenbank
- Offline Konfiguration der Slaves
- Manuelle Konfiguration von SyncUnits
- Online Diagnose in der ctrlX I/O Engineering Oberfläche
- Bus Scan und Konfiguration im ctrlX I/O Engineering Gerätebaum
- SPS-Funktionsbausteine (CXA_Ethercat Bibliothek)
 - Funktionsbausteine für Zustandsmaschine
 - Funktionsbausteine für azyklische Kommunikation
 - Funktionsbausteine für Diagnose (u.a. Master-Status, Slave-Status, Konfiguration-/Online-Info)

Steuerungs-Hardware

Unterstützte Steuerungstypen:

- ctrlX CORE
- ctrlX COREplus
- ctrlX DRIVE mit integrierter ctrlX CORE
- ctrlX IPC mit integrierter ctrlX CORE

Technische Daten

ctrlX CORE Gerätevariante	X3 X3plus DRIVEplus	X7
Max. Anzahl Teilnehmer (Slaves)	128	256
Max. Anzahl Sync Units	Unlimitiert (Begrenzung auf einen zyklischen Frame)	
Max. Anzahl zyklischer Eingangsdaten des Masters (ein zyklischer Frame)	1 kB	1,5 kB
Max. Anzahl zyklischer Ausgangsdaten des Masters (ein zyklischer Frame)	1 kB	1,5 kB
Baudrate	100 MBit/s	
Auto negotiation / autocrossing	Ja	

ctrlX CORE Gerätevariante	X3 X3plus DRIVEplus	X7
Zykluszeit Einstellbar in 1ms Schritten	Min: 2ms Max: 20ms	Min: 500µs, (oder Vielfache von 1 ms) Max: 20ms
Unterstützung mehrerer unterschiedlicher Zykluszeiten	Nein	
Synchronbetrieb (Distributed Clocks / Verteilte Uhren)	Ja	
Hot Connect Gruppen (flexible Topologien)	Nein	
Mailbox CoE (CAN application layer over EtherCAT)	Ja	
Mailbox SoE (Sercos drive profile over EtherCAT)	Ja	
Mailbox EoE (Ethernet over EtherCAT)	Ja	
Mailbox FoE (File access over EtherCAT)	Ja	
Mailbox VoE (Vendor specific protocol over EtherCAT)	Nein	
Mailbox AoE (ADS over EtherCAT)	Ja	
Slave-to-Slave Kommunikation	Nein	Ja (über Master geroutet)
Kabel-Redundanz	Nein	
Deaktivieren von Slaves oder Modulen in der Projektkonfiguration (Maximalprojekt)	Nein	
Deaktivieren von Slaves oder Modulen in der Runtime (Maximalprojekt)	Nein	
Engineering und Diagnose	ctrlX I/O Engineering	
IEC-API (Programmierschnittstelle)	Basic Subset	
Datalayer Interface NRT	Ja	
Datalayer Interface RT	Ja	
Datalayer Interface RT additional Information	Nein	
EtherCAT Schnittstellenkonfiguration	Defaultkonfiguration (Template)	

7.2 EtherCAT Master konfigurieren



Nach der Installation der EtherCAT Master-App ist das EtherCAT Bus-System auf dem ctrlX-Gerät noch nicht aktiv!


Um das System zu aktivieren, muss erst der EtherCAT Master auf dem ctrlX-Gerät aktiviert werden.

Im Anschluss muss die Bus-Topologie im Engineering-Tool ctrlX I/O Engineering vorgenommen werden.

ctrlX I/O Engineering ist ein Teil der ctrlX WORKS-Installation für Ihren Engineering-PC.

Um den EtherCAT Master auf dem ctrlX-Core Gerät zu aktivieren führen Sie bitte folgende Schritte durch:

1. Navigieren Sie in der ctrlX CORE Seitennavigation zum Fenster „EtherCAT“ → *EtherCAT Master*.

2. Im Fenster klicken Sie auf die Schaltfläche , um den EtherCAT Master hinzuzufügen.
 - ➔ Der Dialog „Add EtherCAT-Master“ öffnet sich.
3. Belassen Sie die Default-Einstellungen in den Eingabefeldern „Name“ und „Port“ und bestätigen Sie den Dialog mit „OK“.
 - ➔ Der EtherCAT Master wird hinzugefügt und in der Tabelle angezeigt.

Weiterführende Informationen








➔ [EtherCAT Feldbusteilnehmer konfigurieren](#)

7.3 EtherCAT Slave

7.3.1 EtherCAT Feldbusteilnehmer konfigurieren

Die Konfiguration der EtherCAT-Feldbusteilnehmer wird im Engineering-Tool ctrlX I/O Engineering durchgeführt. ctrlX I/O Engineering ist ein Teil der ctrlX WORKS-Installation für Ihren Engineering-PC.

Zur Konfiguration der EtherCAT-Feldbusteilnehmer führen Sie bitte folgende Schritte durch:

1. „ctrlX I/O Engineering“ aufrufen:
 - Rufen Sie über die ctrlX CORE Seitennavigation das Fenster „EtherCAT Master“ auf
2. Klicken Sie im Header des Fensters auf die Schaltfläche 
 - ➔ Das Tool „ctrlX I/O Engineering“ wird auf Ihrem Engineering-PC geöffnet und am linken Bildrand wird der Gerätebaum angezeigt.
Im Initialzustand enthält der Gerätebaum drei Objekte:
 - : Projektknoten
 - : Geräteknoten der Steuerung
 - : Geräteknoten des EtherCAT-Master
3. Selektieren Sie den Geräteknoten des EtherCAT-Master und öffnen Sie dessen Kontextmenü per Rechtsklick
4. Führen Sie den Befehl „Geräte suchen...“ aus
 - ➔ Der Dialog „Bus scannen“ öffnet sich und zeigt alle am Bus verfügbaren Slaves an
5. Um die gefundenen Slaves in das Projekt zu übernehmen klicken Sie auf die Schaltfläche  und bestätigen Sie den Dialog durch Klick auf die Schaltfläche „Übernehmen“
 - ➔ Der EtherCAT Slave wird unterhalb des EtherCAT-Master im Gerätebaum hinzugefügt
6. Übertragen und aktivieren Sie jetzt die Feldbuskonfiguration auf die Steuerung indem Sie auf die Schaltfläche  in der Befehlsleiste klicken
 - ➔ Die Feldbuskonfiguration wird auf die Steuerung übertragen und der Bus wird in den Zustand „OP“ geschaltet
 - ℹ Wenn am Bus weitere Komponenten wie beispielsweise Buskoppler enthalten sind, muss der Befehl „Geräte suchen...“ ein zweites mal durchgeführt werden, damit auch die am Koppler angeschlossenen Module am Bus erkannt werden und in die Konfiguration übernommen werden können.
7. Speichern Sie die Konfiguration in ctrlX I/O Engineering über die Schaltfläche 

8 ctrlX Bedienoberfläche – Elemente

8.1 Navigation

8.1.1 Seitennavigation – Knoten EtherCAT Master

Mit der Installation der App „EtherCAT Master“ auf der Steuerung wird die ctrlX CORE Seitennavigation um den Knoten „EtherCAT Master“ ergänzt, siehe:

➔ [Fenster – „EtherCAT Master“](#)

EtherCAT Widget

Ergänzend zum Knoten „EtherCAT Master“ in der Seitennavigation wird nach der App-Installation ein EtherCAT Widget im Fenster „Start“ angezeigt, siehe ➔ [Dokumentation](#).

Das Widget dient zur Information und bietet folgende Inhalte:

- Informationen zum aktuellen EtherCAT Betriebszustand
- Diverse Links zum Aufruf von EtherCAT relevanten Einstellungen und Engineering-Tools zur Konfiguration von Master und Slaves

8.2 Fenster

8.2.1 Fenster – „EtherCAT Master“



Das Fenster dient zur Konfiguration des EtherCAT Masters auf dem ctrlX-Gerät. Nach der Installation der EtherCAT Master-App ist der EtherCAT-Bus noch nicht direkt betriebsbereit.

Es sind noch folgende Schritte notwendig, um den EtherCAT Master auf dem ctrlX-Gerät zu aktivieren und die Bus-Topologie zu konfigurieren:

- ➔ [EtherCAT Master konfigurieren](#)
- ➔ [EtherCAT Feldbusteilnehmer konfigurieren](#)



Zur gegenwärtigen Version wird nur ein EtherCAT Master auf dem ctrlX-Gerät unterstützt.


Fensteraufruf

In der ctrlX CORE Web-Oberfläche:

„*Seitennavigation* → *EtherCAT Master*“

Fensterbeschreibung

Oberflächenelement	Beschreibung
Header	 Aufruf zum Starten von ctrlX I/O Engineering zur Konfiguration der Busteilnehmer, siehe ➔ Header – „EtherCAT Master“
Befehlsleiste	<ul style="list-style-type: none">• „[x] Elemente“ Anzahl der EtherCAT Master-Instanzen•  EtherCAT Master-Instanz hinzufügen, siehe ➔ EtherCAT Master konfigurieren

Oberflächenelement	Beschreibung
Tabelle (bei aktiviertem EtherCAT Master)	„Name“ Name der EtherCAT Master-Instanz. Mit Klick auf den Namen wird die Detailansicht des Masters geöffnet, siehe: ↗ Fenster – „EtherCAT Master“ <small>Detailansicht</small>
	„Port“ Bus-Anschluss am ctrlX-Gerät, über den der EtherCAT Master kommuniziert
	„Zustand“ Anzeige des EtherCAT Master Betriebszustands: <ul style="list-style-type: none"> • „Init“ = Init • „Pre-OP“ = Pre-Operational • „Safe-OP“ = Safe-Operational • „OP“ = Operational Siehe ↗ Zustandsmaschine
	„Diagnosen“ Diagnoseinformation EtherCAT Master
	„Aktionen“ <ul style="list-style-type: none"> •  EtherCAT Master auf der Steuerung löschen

Verwandte Themen

[↗ EtherCAT Master App – Grundlagen](#)

[↗ Seitennavigation – Knoten EtherCAT Master](#)

[↗ Header – „EtherCAT Master“](#)

8.2.2 Fenster – „EtherCAT Master“ Detailansicht

Die „EtherCAT Master“ Detailansicht ist eine erweiterte Ansicht des EtherCAT-Master und bietet folgende Funktionen und Anzeigen:

- **Header**
Zeigt den gegenwärtigen Betriebsstatus des EtherCAT Masters und bietet Schaltflächen zum Umschalten des Betriebszustands und zum Aufruf von ctrlX I/O Engineering, siehe [↗ Header – „EtherCAT Master“](#)
- **Registerkarte „Slaves“**
Enthält eine tabellarische Anzeige der angeschlossenen Slaves und deren Betriebszustände, siehe:
[↗ Registerkarte – „Slaves“](#)
- **Registerkarte „Distributed Clocks“**
Zeigt Informationen zur Zykluszeiteinstellung und der Distributed Clocks, siehe:
[↗ Registerkarte – „Distributed Clocks“](#)
- **Registerkarte „Master-Statistik“**
Liefert Informationen zu den übertragenen Daten auf dem EtherCAT-Bus, siehe:
[↗ Registerkarte – „Master-Statistik“](#)
- **Registerkarte „Slave-Statistik“**
Enthält Fehlerzähler der projektierten und erreichbaren EtherCAT-Slaves (z.B. zur Fehleranalyse), siehe:
[↗ Registerkarte – „Slave-Statistik“](#)

- **Registerkarte „AoE“** (EtherCAT Master-Instanz)
Übersicht der AoE-Konfiguration (ADS over EtherCAT) im EtherCAT-Netzwerk, siehe:
➔ [Registerkarte – „AoE“](#) (EtherCAT Master-Instanz)
- **Registerkarte „EoE“** (EtherCAT Master-Instanz)
Übersicht der EoE-Konfiguration (Ethernet over EtherCAT), siehe:
➔ [Registerkarte – „EoE“](#) (EtherCAT Master-Instanz)

Fensteraufruf

In der ctrlX CORE Web-Oberfläche:

„Seitennavigation → EtherCAT Master → Klick auf den Namen der EtherCAT-Instanz in der Übersichtstabelle“

Verwandte Themen

- ➔ [EtherCAT Master App – Grundlagen](#)
- ➔ [Seitennavigation – Knoten EtherCAT Master](#)

8.3 Header

8.3.1 Header – „EtherCAT Master“

Der Header dient zur Steuerung des EtherCAT-Masters auf dem ctrlX-Gerät und ist in die EtherCAT-Master Detailansicht der Web-Oberfläche integriert, siehe:

➔ [Fenster – „EtherCAT Master“](#) Detailansicht

Funktionen




- Anzeige des Master-Betriebsstatus
- Anzeige und Schalfläche zum Wechseln des Master-Betriebszustands
- Link zum Aufruf von ctrlX I/O Engineering (Tool zur Konfiguration der Busteilnehmer)

Aufruf

In der ctrlX CORE Web-Oberfläche:

„Seitennavigation → EtherCAT Master → Klick auf den EtherCAT Master in der Übersichtstabelle“

Beschreibung

Oberflächenelement	Beschreibung
Statusanzeige (grafisch / textuell)	<ul style="list-style-type: none"> •  = Bus in OP / RUN •  = Bus ist gestört bzw. die Konfiguration ist fehlerhaft <p>Die grafische Statusmeldung wird durch eine textuelle Statusmeldung ergänzt. Die Textausgabe hat die Aufgabe, detailliertere Informationen zum Bus-Betriebszustand zu liefern, beispielsweise für den Fall, wenn sich der Bus nicht in Betrieb nehmen lässt. In diesem Fall wird das nächstliegende Kriterium angezeigt, welches nicht erfüllt ist, damit der Bus in Betrieb gehen kann. Durch Klick auf die Textausgabe wird eine Box mit allen Kriterien ausgeklappt und deren Status grafisch angezeigt.</p> <p>Erforderliche Kriterien:</p> <ul style="list-style-type: none"> • EtherCAT Port existiert • Lizenz zum Betrieb des Bus-Systems vorhanden • Einstellungen gültig • Links vorhanden • Bus-Topologie OK • Master im erforderlichen Betriebszustand • Slaves im Master-Betriebszustand • Slaves ohne Fehler • Master in Betriebszustand OP • DC in Sync • Bus läuft
Betriebszustand	<p>Anzeige des aktuellen Betriebszustands und Möglichkeit zur Statusumschaltung.</p> <p>Betriebszustände:</p> <ul style="list-style-type: none"> • „Init“ = Init • „Pre-OP“ = Pre-Operational • „Safe-OP“ = Safe-Operational • „OP“ = Operational <p>Siehe ↗ Zustandsmaschine</p>
	Schaltfläche zum Aufruf von ctrlX I/O Engineering, zur Konfiguration der Busteilnehmer

Verwandte Themen

↗ EtherCAT Master App – Grundlagen

↗ Seitennavigation – Knoten EtherCAT Master

↗ Header – „EtherCAT Slave“

8.3.2 Header – „EtherCAT Slave“

Der Header dient zur Steuerung der EtherCAT Slaves und ist in den Slave-Detailansichten der Web-Oberfläche integriert, siehe: [↗ Registerkarte – „Allgemeines“](#) (EtherCAT Slave)

Funktionen



- Anzeige des Slave-Betriebsstatus
- Anzeige und Umschaltmöglichkeit des Slave-Betriebszustandes
- Hinweise, z.B. falls die Konfiguration geändert wurde und eine manuelle Aktualisierung erforderlich ist

Aufruf

In der ctrlX CORE Web-Oberfläche:

„Seitennavigation → EtherCAT Master → Klick auf die EtherCAT Master-Instanz in der Übersichtstabelle → Klick auf Registerkarte "Slaves" → Klick auf den gewünschten Slave“

Beschreibung

Oberflächenelement	Beschreibung
Name des Slaves	Individuell zu benennen
Visuelle Statusanzeige	<ul style="list-style-type: none">•  = Slave in OP / RUN•  = Der Slave meldet einen Fehler oder ist nicht in OP
Textuelle Statusanzeige	<p>Zeigt den aktuellen Gerätestatus (Betriebszustand, Verbindungsstatus, Anstehende Fehler). Über die Schaltfläche [∨] wird eine Tabelle mit den wichtigsten Kriterien eingeblendet:</p> <p>Mögliche Meldungen:</p> <ul style="list-style-type: none">• Slave in OP• Slave nicht in OP• Verbunden• Nicht verbunden• Kein Fehler• Bei anstehendem Fehler wird der Fehler-Code angezeigt <p>Diagnose-Codes siehe: ↗ AL Status Codes</p>
Betriebszustand (Anzeige / Umschaltung)	<p>Anzeige des aktuellen Betriebszustands und Möglichkeit zur Statusumschaltung.</p> <p>Betriebszustände:</p> <ul style="list-style-type: none">• „Init“ Transition in den Zustand Init wird ausgeführt:<ul style="list-style-type: none">– Keine Kommunikation auf dem Application-Layer– Master hat Zugriff auf die DL-Informationsregister• „Pre-OP“ Transition in den Zustand Pre-Operational wird durchgeführt.• „Safe-OP“ Transition in den Zustand Safe-Operational wird durchgeführt.• „OP“ Transition in den Zustand Operational wird durchgeführt <p>Siehe ↗ Zustandsmaschine</p>

Verwandte Themen

[↗ EtherCAT Master App – Grundlagen](#)

➔ [Header – „EtherCAT Master“](#)

8.4 Registerkarten – EtherCAT Master

8.4.1 Registerkarte – „Slaves“

Die Registerkarte dient zur Anzeige der konfigurierten Slaves in tabellarischer Darstellung. In der Tabelle können Sie durch Klick auf einen der Slave-Namen eine Detailansicht zum betreffenden Slave öffnen. Die Detailansicht bietet weitere Informationen zum Slave und einen spezifischen Header zur Diagnose und zur Steuerung des Slave-Betriebszustands, siehe:

➔ [Registerkarte – „Allgemeines“](#) (EtherCAT Slave)

➔ [Header – „EtherCAT Slave“](#)

Aufruf

In der ctrlX CORE Web-Oberfläche:

„Seitennavigation → EtherCAT Master → Übersichtstabelle → Klick auf den Namen der EtherCAT Master-Instanz → Klick auf Registerkarte "Slaves"“

Verwandte Themen

➔ [Fenster – „EtherCAT Master“](#)

➔ [Fenster – „EtherCAT Master“](#) Detailansicht

➔ [EtherCAT Master App – Grundlagen](#)

➔ [Seitennavigation – Knoten EtherCAT Master](#)

8.4.2 Registerkarte – „Distributed Clocks“

Die Registerkarte zeigt Informationen zur Zykluszeiteinstellung und der Distributed Clocks an, siehe unten.

Aufruf

In der ctrlX CORE Web-Oberfläche:

„Seitennavigation → EtherCAT Master → Klick auf die EtherCAT Master-Instanz in der Übersichtstabelle → Klick auf Registerkarte "Distributed Clocks"“

Angezeigte Informationen

Tabelleneintrag	Beschreibung
„DC-Modus“	<p>eingestellter und aktiver (in Klammern) DC Modus:</p> <ul style="list-style-type: none"> • „Auto“ <ul style="list-style-type: none"> – Standardeinstellung – Automatische Auswahl des Synchronisations-Modus • „Free Run“ <ul style="list-style-type: none"> – Zyklische Kommunikation, ohne Uhrensynchronisation – Es findet keine Regelung der Distributed Clocks statt • „Bus Shift“ <ul style="list-style-type: none"> – Distributed Clocks aktiv – Slaves werden mit dem EtherCAT-Master synchronisiert – Der EtherCAT-Master stellt die Systemzeit bereit und schreibt diese in die Bus-Referenzuhr (Schieben der Buszeit) • „Link Layer Reference Clock“ <ul style="list-style-type: none"> – Distributed Clocks aktiv – Slaves werden mit dem EtherCAT-Master synchronisiert – Die, im EtherCAT-Master-Link-Layer integrierte Uhr wird als Referenzuhr zur Synchronisation verwendet. Diese Einstellung garantiert die höchste Synchronisationsgenauigkeit, erfordert aber spezielle Hardware. <p>WARNUNG Verwenden Sie die Einstellung „Link Layer Reference Clock“ NICHT, wenn Sie gleichzeitig auf eine externe Uhr, z. B. einen NTP-Server, synchronisieren. Dies kann zu unberechenbarem Verhalten führen!</p>
„Zykluszeit“	Eingestellte Zykluszeit
„Sync Offset“	Anzeige des Sync Offset in Prozent und als absolut wirksamer Wert
„Sync-Fensterüberwachung“	<p>Zeigt an, ob die Sync Fensterüberwachung aktiv ist, inklusive dem aktiven Überwachungsfenster.</p> <p>Bei aktivierter Option, wird die Synchronisation der verteilten Uhren in allen Slaves überwacht.</p> <p>Die „Systemzeitdifferenz“ (Register 0x092C) wird zyklisch von jedem Slave ausgelesen.</p> <p>Wenn bei allen Slaves die Systemzeit-Differenz kleiner ist als das Überwachungsfenster, gilt die Synchronisation als eingeregelt (Status „DC in Sync“).</p>
„Kontinuierliche Laufzeitkompensation“	Zeigt an, ob die kontinuierliche Laufzeitkompensation aktiv ist
Slave-Übersichtstabelle	<p>„Status“</p> <ul style="list-style-type: none"> • Grün: Der Slave ist korrekt verbunden und meldet keinen Fehler • Rot: Der Slave ist nicht verbunden oder meldet einen Fehler
	<p>„Name“</p> <p>Konfigurierter Geräteiname des Slaves</p>

Tabelleneintrag	Beschreibung
	„Adresse“ EtherCAT-Adresse
	„Zustand“ Betriebszustand des EtherCAT-Slaves
	„Systemzeitabweichung“ Systemzeit-Differenz in ns (ESC Register 0x09C2) Der Wert wird nur angezeigt, wenn Sync Window Monitoring aktiv ist und ist nur dann gültig, wenn „Distributed Clocks“ konfiguriert ist. Der erste Slave am Bus, bei dem „Distributed Clocks“ aktiviert ist, stellt die Referenzuhr bereit.
	„DC-Unterstützung“ Zeigt an, ob der Controller des EtherCAT-Slave „Distributed Clocks“ unterstützt (32 / 64 Bit).

Verwandte Themen

- ➔ [Fenster – „EtherCAT Master“](#)
- ➔ [EtherCAT Master App – Grundlagen](#)
- ➔ [Seitennavigation – Knoten EtherCAT Master](#)

8.4.3 Registerkarte – „Master-Statistik“

In der Registerkarte werden Informationen zu den übertragenen Daten auf dem EtherCAT-Bus angezeigt.


Die Registerkarte ist in zwei Tabellenbereiche unterteilt:

- **Frames**
Zeigt gesendete und verlorengegangene Frames, unterteilt nach zyklischer und azyklischer Kommunikation
- **Mailbox-Anfragen**
Zeigt sämtliche, gesendete Mailbox-Anfragen, unterteilt nach Mailbox und Übertragungsrichtung

Zählerstände zurücksetzen

Über die Schaltfläche „Zähler zurücksetzen“ können die Zähler in den jeweiligen Tabellenbereichen separat zurückgesetzt werden.

Zählerstände kopieren / sichern

Über die Schaltfläche  werden die Zählerwerte des jeweiligen Tabellenbereichs in die Zwischenablage des Engineering-PC übertragen, beispielsweise um die Werte lokal auf dem PC zu sichern.

Hinweise zur Auswertung

Verlorengegangene Frames sind ein erster Indikator für Kommunikationsprobleme, können aber auch durch Stecken oder Ziehen eines Bus-Kabels oder durch das Aus- / Einschalten eines Slaves bzw. der Steuerung auftreten.

Wenn die Anzahl an verlorenen Frames im regulären Betrieb kontinuierlich ansteigt, sollte die Slave-Statistik überprüft werden, siehe:

→ [Registerkarte – „Slave-Statistik“](#)

Weitere Hinweise und Details zum Vorgehen entnehmen Sie bitte der Dokumentation der "EtherCAT Technology Group":

→ https://www.ethercat.org/.../EtherCAT_Diagnosis_For_Users.pdf

Aufruf der Registerkarte über die Navigationsleiste

In der ctrlX CORE Web-Oberfläche:

„Seitennavigation → EtherCAT Master → Klick auf den Namen der EtherCAT Master-Instanz in der Übersichtstabelle → Klick auf Registerkarte Master-Statistik“

Verwandte Themen

→ [EtherCAT Master App – Grundlagen](#)

→ [Seitennavigation – Knoten EtherCAT Master](#)

→ [Header – „EtherCAT Master“](#)

8.4.4 Registerkarte – „Slave-Statistik“

In der Registerkarte werden Fehlerzähler zu den projizierten und erreichbaren EtherCAT-Slaves angezeigt. Durch die Analyse von Fehlerzählern und der zugrundeliegenden Bus-Topologie, kann auf eine potentielle Fehlerstelle im Bus-System geschlossen werden.

Empfehlungen zur Fehlersuche

- Setzen Sie die Fehlerzähler zurück, wenn sich die Anlage im Betriebszustand "Master in OP" befindet, siehe:
→ [Header – „EtherCAT Master“](#)
- Beobachten Sie anschließend die Fehlerzähler über einen längeren Zeitraum
- Analysieren Sie die Fehlerzähler entsprechend der aktuellen Topologie (Reihenfolge des Frame-Durchlaufes)
- Wenn bei einem Slave alle Fehlerzähler 0 aufweisen und beim darauffolgenden Slave Fehler erkannt wurden, analysieren Sie die Verbindung zwischen den Slaves

Fehlerzähler zurücksetzen

Die Fehlerzähler zählen bis zu einem maximalen Wert von 255 (max. Registerbreite im Slave) und frieren dann bei diesem Wert ein.

Über die Schaltfläche „Fehlerzähler zurücksetzen“ können die Fehlerzähler wieder auf 0 zurückgestellt werden.

Die Fehlerzähler detektieren auch fehlerhafte Frames, welche beispielsweise durch Stecken oder Ziehen eines Kabels oder durch Aus- Einschalten eines Slaves bzw. der Steuerung entstehen können.

Aufruf

In der ctrlX CORE Web-Oberfläche:

„Seitennavigation → EtherCAT Master → Übersichtstabelle → Klick auf den Namen der EtherCAT Master-Instanz → Klick auf Registerkarte Slave-Statistik“

Beschreibung

Details zum Vorgehen und zur Bedeutung der angezeigten Register entnehmen Sie bitte der Dokumentation der "EtherCAT Technology Group", zu der Sie über die Schaltfläche [EtherCAT diagnostics for users](#) gelangen.

Die Anzeige erfolgt in tabellarischer Form und liefert folgende Informationen:

- **Status**
Slave Betriebszustand
- **Name**
Slave Bezeichnung
- **Adresse**
Slave Adresse
- **Invalid Frames / Port**
Inhalt der ESC Register 0x0300, 0x0302, 0x0304, 0x0306
- **RX Errors / Port**
Inhalt der ESC Register 0x0301, 0x0303, 0x0305, 0x0307
- **Forwarded RX Errors / Port**
Inhalt der ESC Register 0x0308, 0x0309, 0x030A, 0x030B
- **Proc. Unit** (processing unit error counter)
Inhalt des ESC Registers 0x030C
- **PDI Errors**
Inhalt des ESC Registers 0x030D

Verwandte Themen

➔ [Fenster – „EtherCAT Master“](#)

➔ [EtherCAT Master App – Grundlagen](#)

8.4.5 Registerkarte – „AoE“ (EtherCAT Master-Instanz)

Die Registerkarte dient zur Übersicht der AoE-Konfiguration (ADS over EtherCAT) im EtherCAT-Netzwerk.

Aufruf

In der ctrlX CORE Web-Oberfläche:

„Seitennavigation → EtherCAT Master → Klick auf den Namen der EtherCAT Master-Instanz in der Übersichtstabelle → Klick auf Registerkarte "AoE"“

Beschreibung

Oberflächen- element	Beschreibung
„Master-Konfiguration“	Net-ID des Masters
Tabellarische Auflistung der Slaves	<p>In der Tabelle werden alle Slaves aufgelistet, die AoE unterstützen und für die AoE in der Konfiguration aktiv ist. Wenn kein Slave AoE unterstützt, bleibt die Liste leer.</p> <ul style="list-style-type: none"> • „Name“: Name des Slaves • „Adresse“: Adresse des Slaves • „Net ID“: Net ID des Slaves

Verwandte Themen

➔ [EtherCAT Master App – Grundlagen](#)

➔ [Seitennavigation – Knoten EtherCAT Master](#)

➔ Header – „EtherCAT Master“

8.4.6 Registerkarte – „EoE“ (EtherCAT Master-Instanz)

Die Registerkarte gibt eine Übersicht der EoE-Konfiguration (Ethernet over EtherCAT).

Aufruf

In der ctrlX CORE Web-Oberfläche:

„Seitennavigation → EtherCAT Master → Klick auf den Namen der EtherCAT Master-Instanz in der Übersichtstabelle → Klick auf Registerkarte "EoE"“

Beschreibung

Oberflächen- element	Beschreibung
„Master-Konfiguration“	<p>EoE-Konfiguration des Masters:</p> <ul style="list-style-type: none"> • „IP-Adresse“: IP-Adresse des Masters (leer, falls EoE am Master nicht aktiv ist). • „Subnetzmaske“: Subnetzmaske des Masters (leer, falls EoE am Master nicht aktiv ist)
Tabellarische Auflistung der Slaves	<p>In der Tabelle werden alle Slaves aufgelistet, die EoE unterstützen. Wenn kein Slave konfiguriert ist, welches EoE unterstützt, bleibt die Liste leer.</p> <ul style="list-style-type: none"> • „Name“: Name des Slaves • „Adresse“: Adresse des Slaves • „MAC-Adresse“: MAC-Adresse des Slaves (leer, wenn EoE nicht unterstützt wird) • „IP-Adresse“: IP-Adresse des Slaves (leer, wenn EoE nicht aktiv ist oder als Switch-Port konfiguriert ist) • „Subnetzmaske“: Subnetzmaske des Slaves (leer, wenn EoE nicht aktiv ist oder als Switch-Port konfiguriert ist) • „Standard Gateway“: Standard-Gateway des Slaves (leer, wenn EoE nicht aktiv ist oder als Switch-Port konfiguriert ist) • „DNS-Name“: DNS-Name des Slaves (leer, wenn EoE nicht aktiv ist oder als Switch-Port konfiguriert ist)

Verwandte Themen

➔ EtherCAT Master App – Grundlagen

➔ Seitennavigation – Knoten EtherCAT Master

➔ Header – „EtherCAT Master“

8.5 Registerkarten – EtherCAT Slave

8.5.1 Registerkarte – „Allgemeines“ (EtherCAT Slave)

Die Registerkarte dient zur Anzeige allgemeiner Konfigurationsinformation des Slaves.

Aufruf

In der ctrlX CORE Web-Oberfläche:

„Seitennavigation → EtherCAT Master → Klick auf die EtherCAT Master-Instanz in der Übersichtstabelle → Klick auf Registerkarte "Slaves" → Klick auf den gewünschten Slave → Registerkarte "Allgemeines"“

Angezeigte Informationen

Oberflächen- element	Beschreibung
„Identität“	Zeigt die Identitätsinformation des Slaves an <ul style="list-style-type: none"> • „Name“: Name des Slaves aus der Konfiguration • „Hersteller“: Name des Herstellers, abgeleitet aus der VendorID • „Produktcode“: Produkt Code des Slaves • „Revision“: Revision des Slaves
„Mailbox“	Zeigt Informationen zur Mailbox-Konfiguration an. Falls keine Mailbox unterstützt wird, sind die nachfolgenden Felder leer. <ul style="list-style-type: none"> • „Protokolle“: Auflistung der unterstützten Protokolle • „Standard Out / Standard In“: Mailbox-Größe im Normalbetrieb • „Bootstrap Out / Bootstrap In“: Mailbox-Größe im Bootstrap (z.B. für Firmware-download)

Verwandte Themen

➔ [EtherCAT Master App – Grundlagen](#)

➔ [Seitennavigation – Knoten EtherCAT Master](#)

➔ [Header – „EtherCAT Slave“](#)

9 EtherCAT Diagnose

9.1 Allgemein

EtherCAT Diagnosen stehen auf verschiedenen Kommunikationsschichten zur Verfügung.

Eine zyklusgenaue Diagnose wird im Master generiert. Dies umfasst zum einen den Master-Buszustand, als auch die Auswertung des WorkingCounter (WKC). Durch die Auswertung des WKC wird zyklusgenau detektiert, ob ein Datagramm (z.B. zyklische Prozessdaten) korrekt bearbeitet wurde und der Datenstatus gültig ist. Alle Slaves, die in einem Datagramm (z.B. SyncUnit) mit fehlerhaftem WKC adressiert sind, haben somit dann einen ungültigen Datenstatus.

HINWEIS

Bei einem ungültigen WorkingCounter reagiert die EtherCAT Master App mit einem "Null-setzen" der zyklischen Eingänge der betroffenen SyncUnit. Die Gültigkeit der zyklischen Ausgänge ist davon nicht betroffen.

Um den Zustand der Slaves und davon abgeleitet den Zustand der zyklischen Eingangsdaten in den Applikationen überprüfen zu können, bietet die EtherCAT Master App folgende Möglichkeiten:

- Datalayer NRT Knoten
- SPS Funktionsbausteine z.B.: IL_ECATOMasterState, IL_ECATRemoteSlaveState.

Eine detaillierte Diagnose, ob z.B. ein Slave einen EtherCAT Zustandsfehler hat oder nicht mehr verbunden ist bzw. evtl. ein Link-Problem vorliegt (Port-Status), wird dann vom Master azyklisch ermittelt.

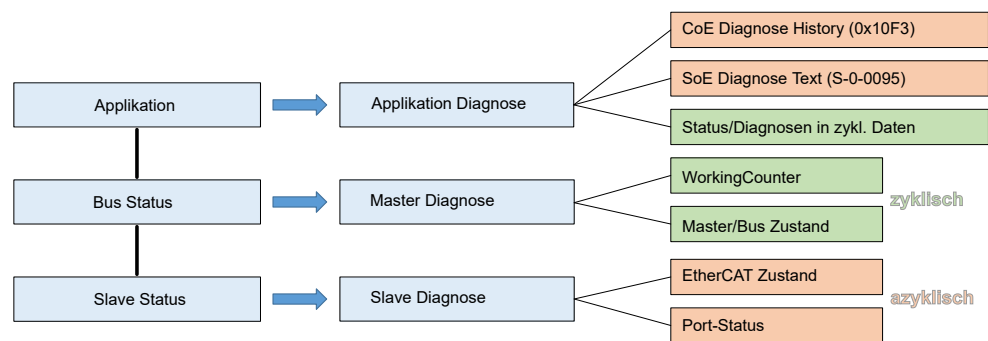


Abb. 23: Übersicht EtherCAT Diagnosen

Auf der Applikationsebene haben viele Slaves Statusmeldungen oder einfache Diagnosen-Daten bereits in den zyklischen Daten verfügbar. Detail-Diagnosen können dann azyklisch ausgelesen werden, z.B. bei CoE Slaves gibt es optional das Diagnose History Objekt (0x10F3) oder bei SoE Slaves den Diagnose-Text (S-0-0095).

Für die Auswertung der Diagnosen im SPS-Programm stehen beim Rexroth EtherCAT Master verschiedene Funktionsbausteine in der CXA_Ethercat Bibliothek zur Verfügung.

Manche CoE Slaves unterstützen auch CoE Emergency Nachrichten. Diese Emergency Nachrichten haben größtenteils aber nur geräte-spezifische Informationen und auch eine Quittierung ist nicht erforderlich. Daher werden diese i.d.R. nur ins Steuerungs-Logbuch eingetragen.

9.2 AL Status Codes

Der "AL Status Code" (AL: Application Layer) beschreibt den Fehlercode eines Slave Gerätes, wenn eine angeforderte Zustandsänderung nicht ausgeführt werden kann oder wenn der Slave einen internen Fehler entdeckt.

Der Rexroth EtherCAT Master liest im Fehlerfall automatisch den AL Status Code (ESC Register 0x0134:0x0135) aus und quittiert die Fehler-Indikation im Slave. Beim nächsten Zustandswechsel des Slaves wird der ausgelesene AL Status Code im Master zurückgesetzt.



Tritt ein unspezifizierter Fehler auf oder eine Konfiguration wird als ungültig erkannt, sind Details in der Produkt-Dokumentation des Slave Gerätes nachzuschlagen oder der Hersteller des Slave Gerätes zu kontaktieren.

In nachfolgender Tabelle werden folgende Abkürzungen verwendet:

- I = Zustand „Init“
- B = Zustand „Bootstrap“
- P = Zustand „Pre-Operational“
- S = Zustand „Safe-Operational“
- O = Zustand „Operational“

AL Status Code	Beschreibung	Aktueller Zustand oder Zustandsänderung	Resultierender Zustand
0x0000	Kein Fehler	Alle	Aktueller Zustand
0x0001	Unspezifizierter Fehler	Alle	Alle
0x0002	Kein Speicher	Alle	Alle
0x0003	Ungültige Geräte-Konfiguration	P → S	P
0x0004	Ungültige Revision	P → S	P
0x0005	Reserviert aus Kompatibilitäts-Gründen	-	-
0x0006	SII / EEPROM Daten passen nicht mit Firmware zusammen	I → P	I
0x0007	Firmware Update nicht erfolgreich	Alle	Alle
0x000E	Lizenz Fehler	Alle	I
0x0011	Ungültige Zustandsänderung angefordert	I → S, I → O, P → O, P → B, S → B, O → B	Aktueller Zustand
0x0012	Unbekannter Zustand angefordert	Alle	Aktueller Zustand
0x0013	Zustand Bootstrap nicht unterstützt	I → B	I
0x0014	Keine gültige Firmware	I → P	I
0x0015	Ungültige Mailbox Konfiguration (Bootstrap)	I → B	I
0x0016	Ungültige Mailbox Konfiguration (PreOP)	I → P	I

AL Status Code	Beschreibung	Aktueller Zustand oder Zustandsänderung	Resultierender Zustand
0x0017	Ungültige Sync Manager Konfiguration	P → S, S → O	Aktueller Zustand
0x0018	Keine gültigen Eingänge verfügbar	O, S → O	S
0x0019	Keine gültigen Ausgänge möglich	O, S → O	S
0x001A	Synchronisations-Fehler (verschiedene Ursachen)	O, S → O	S
0x001B	Sync Manager Watchdog	O, S	S
0x001C	Ungültige Sync Manager Typen	O, S, P → S	S
0x001D	Ungültige Sync Manager Konfiguration (Ausgänge)	O, S, P → S	S
0x001E	Ungültige Sync Manager Konfiguration (Eingänge)	O, S, P → S	P
0x001F	Ungültige Watchdog Konfiguration	O, S, P → S	P
0x0020	Slave benötigt Kalt-Start	Alle	Aktueller Zustand
0x0021	Slave benötigt Zustand „Init“	B, P, S, O	Aktueller Zustand
0x0022	Slave benötigt Zustand „Pre-Operational“	S, O	S, O
0x0023	Slave benötigt Zustand „Safe-Operational“	O	O
0x0024	Ungültiges Mapping Eingangsdaten	P → S	P
0x0025	Ungültiges Mapping Ausgangsdaten	P → S	P
0x0026	Inkonsistente Einstellungen (allgemein)	P → S	P
0x0027	Free Run nicht unterstützt	P → S	P
0x0028	Sync Modus nicht unterstützt	P → S	P
0x0029	Free Run benötigt Sync Manager im 3 Puffer Modus	P → S	P
0x002A	Background Watchdog	S, O	P
0x002B	Keine gültigen Ein- und Ausgangsdaten	O, S → O	S
0x002C	Fataler Sync Fehler, Sync0 oder Sync1 Signal nicht mehr empfangen.	O	S
0x002D	Kein Sync Signal, Slave wartet in „Safe-Operational“ auf Sync0/Sync1 Signal (S → O Timeout).	S → O	S
0x002E	Zykluszeit zu klein, die eingestellte EtherCAT Zykluszeit wird vom Slave nicht unterstützt.	S → O	S
0x0030	Ungültige DC Sync Konfiguration	O, S → O, P → S	S, P
0x0031	Ungültige DC Latch Konfiguration	O, S → O, P → S	S, P
0x0032	PLL Fehler, Master nicht synchronisiert, aber mind. ein DC Event empfangen.	O, S → O	S

AL Status Code	Beschreibung	Aktueller Zustand oder Zustandsänderung	Resultierender Zustand
0x0033	DC Sync IO Fehler, mehrere Synchronisations-Fehler, Master u. Slave nicht mehr synchronisiert.	O, S → O	S
0x0034	DC Sync Timeout Fehler mehrere Synchronisations-Fehler, zu viele SM Events ausgelassen	O, S → O	S
0x0035	Ungültige DC Zykluszeit	P → S	P
0x0036	DC Sync0 Zykluszeit nicht funktionsfähig mit Applikation	P → S	P
0x0037	DC Sync1 Zykluszeit nicht funktionsfähig mit Applikation	P → S	P
0x0041	Mailbox AoE	B, P, S, O	Aktueller Zustand
0x0042	Mailbox EoE	B, P, S, O	Aktueller Zustand
0x0043	Mailbox CoE	B, P, S, O	Aktueller Zustand
0x0044	Mailbox FoE	B, P, S, O	Aktueller Zustand
0x0045	Mailbox SoE	B, P, S, O	Aktueller Zustand
0x004F	Mailbox VoE	B, P, S, O	Aktueller Zustand
0x0050	EEPROM kein Zugriff	Alle	Alle
0x0051	EEPROM Fehler	Alle	Alle
0x0052	Externe Hardware nicht betriebsbereit	Alle	Alle
0x0060	Slave führte Neustart durch	Alle	I
0x0061	Geräte Identifikationswert aktualisiert	P	P
0x0070	Ungültige Modul-Konfiguration, bei einem Slave mit Modular-Device-Profile stimmen die konfigurierten Module (0xF030) nicht mit den physikalischen Modulen (0xF050) überein.	P → S	P
0x00F0	Applikations-Steuerung verfügbar	I	I
< 0x8000	Reserviert	-	-
0x8000 - 0xFFFF	Hersteller-spezifisch	-	-

9.3 CoE SDO Abortcodes

Bei azyklischer CoE Mailbox Kommunikation können folgende standardisierte SDO Abortcodes auftreten. Diese werden auf Fehlercodes der Steuerung abgebildet (→ ECM Fehlercode).

SDO Abortcode	Beschreibung	ECM Fehlercode
n/a	Ecat CoE: Protokoll nicht unterstützt	0x0C850032
0x0503 0000	Ecat SDO: Toggle Bit kein Zustandswechsel	0x0C850040
0x0504 0000	Ecat SDO: SDO Protokoll Zeitüberschreitung	0x0C850041
0x0504 0001	Ecat SDO: Client/Server ungültiger Kommandozeichner	0x0C850042

SDO Abortcode	Beschreibung	ECM Fehlercode
0x0504 0002	Ecat SDO: Ungültige Blockgröße (nur im Block Modus)	0x0C850043
0x0504 0003	Ecat SDO: Ungültige Sequenznummer (nur im Block Modus)	0x0C850044
0x0504 0004	Ecat SDO: CRC Fehler (nur im Block Modus)	0x0C850045
0x0504 0005	Ecat SDO: Kein freier Speicher	0x0C850046
0x0601 0000	Ecat SDO: Nicht unterstützter Zugriff auf ein Objekt	0x0C850047
0x0601 0001	Ecat SDO: Lesezugriff auf ein nur schreibbares Objekt	0x0C850048
0x0601 0002	Ecat SDO: Schreibzugriff auf ein nur lesbares Objekt	0x0C850049
0x0601 0003	Ecat SDO: Schreibzugriff auf Subindex fehlgeschlagen	0x0C870004
0x0601 0004	Ecat SDO: Complete Access nicht unterstützt	0x0C870005
0x0601 0005	Ecat SDO: Objekt größer als Mailbox	0x0C870006
0x0601 0006	Ecat SDO: Objekt in RxPDO gemappt, Download blockiert	0x0C870007
0x0602 0000	Ecat SDO: Objekt ist nicht im Objektverzeichnis vorhanden	0x0C85004A
0x0604 0041	Ecat SDO: Objekt kann nicht in ein PDO gemappt werden	0x0C85004B
0x0604 0042	Ecat SDO: PDO Länge überschritten	0x0C85004C
0x0604 0043	Ecat SDO: Allgemeine Inkompatibilität (Parameter)	0x0C85004D
0x0604 0047	Ecat SDO: Allgemeine Inkompatibilität (Geräteintern)	0x0C85004E
0x0606 0000	Ecat SDO: Zugriff fehlgeschlagen (Hardwarefehler)	0x0C85004F
0x0607 0010	Ecat SDO: Datentyp und Längenkennung differieren	0x0C850050
0x0607 0012	Ecat SDO: Falscher Datentyp (Parameter zu groß)	0x0C850051
0x0607 0013	Ecat SDO: Falscher Datentyp (Parameter zu klein)	0x0C850052
0x0609 0011	Ecat SDO: Subindex existiert nicht	0x0C850053
0x0609 0030	Ecat SDO: Parameterwert außerhalb gültigem Bereich	0x0C850054
0x0609 0031	Ecat SDO: Parameterwert größer als Maximalwert	0x0C850055
0x0609 0032	Ecat SDO: Parameterwert kleiner als Minimalwert	0x0C850056
0x0609 0033	Ecat SDO: Module Ident Liste unpassed	0x0C85005E
0x0609 0036	Ecat SDO: Maximalwert kleiner als Minimalwert	0x0C850057
0x0800 0000	Ecat SDO: Allgemeiner Fehler	0x0C850058
0x0800 0020	Ecat SDO: Daten übertragen oder speichern fehlgeschlagen	0x0C850059
0x0800 0021	Ecat SDO: Fehler Datenübertragung (lokale Steuerung)	0x0C85005A
0x0800 0022	Ecat SDO: Fehler Datenübertragung (Gerätezustand)	0x0C85005B
0x0800 0023	Ecat SDO: Objektverzeichnis existiert nicht	0x0C85005C
n/a	Ecat SDO: Unbekannter Fehlercode	0x0C85005D

9.4 FoE Fehlercodes

Bei azyklischer FoE Mailbox Kommunikation können folgende standardisierte FoE Fehlercodes auftreten. Diese werden auf Fehlercodes der Steuerung abgebildet (→ ECM Fehlercode).

FoE Fehlercode	Beschreibung	ECM Fehlercode
n/a	Ecat FoE: Protokoll nicht unterstützt	0x0C850034
n/a	Ecat FoE: Protokoll nicht unterstützt in Bootstrap	0x0C85010F
0x8000	Ecat FoE: Herstellerspezifischer FoE Fehler	0x0C850060
0x8001	Ecat FoE: Nicht gefunden	0x0C850061
0x8002	Ecat FoE: Zugriff verweigert	0x0C850062

FoE Fehler-code	Beschreibung	ECM Fehler-code
0x8003	Ecat FoE: Laufwerk voll	0x0C850063
0x8004	Ecat FoE: Nicht erlaubt	0x0C850064
0x8005	Ecat FoE: Falsche Paketnummer	0x0C850065
0x8006	Ecat FoE: Bereits vorhanden	0x0C850066
0x8007	Ecat FoE: Nutzer nicht vorhanden	0x0C850067
0x8008	Ecat FoE: Nur in Bootstrap	0x0C850068
0x8009	Ecat FoE: Dateiname in Bootstrap ungültig	0x0C850069
0x800A	Ecat FoE: Keine Zugriffsrechte	0x0C85006A
0x800B	Ecat FoE: Programmfehler	0x0C85006B
0x800C	Ecat FoE: Falsche Prüfsumme	0x0C85006C
0x800D	Ecat FoE: Firmware passt nicht zur Hardware	0x0C85006D
0x800E	Reserviert	n/a
0x800F	Ecat FoE: Datei nicht gefunden	0x0C85006F
0x8010	Ecat FoE: Datei-Header fehlt	0x0C870001
0x8011	Ecat FoE: Fehler Flash-Speicher	0x0C870002
0x8012	Ecat FoE: Datei inkompatibel	0x0C870003
n/a	Ecat FoE: Datei größer als maximal erlaubt	0x0C85017A

9.5 SoE Fehlercodes

Bei azyklischer SoE Mailbox Kommunikation können folgende standardisierte SoE Fehlercodes auftreten. Diese werden auf Fehlercodes der Steuerung abgebildet (→ ECM Fehlercode).

Sercos Fehler-code	Beschreibung	ECM Fehler-code
n/a	Ecat SoE: Protokoll nicht unterstützt	0x0C850035
n/a	Ecat SoE: Ungültiger Zugriff auf Element 0	0x0C850078
0x1001	Ecat SoE: Nicht vorhanden	0x0C850079
0x1009	Ecat SoE: Ungültiger Zugriff auf Element 1	0x0C85007A
0x2001	Ecat SoE: Kein Name	0x0C85007B
0x2002	Ecat SoE: Übertragungsfehler, Name zu kurz	0x0C85007C
0x2003	Ecat SoE: Übertragungsfehler, Name zu lang	0x0C85007D
0x2004	Ecat SoE: Name kann nicht geändert werden (read only)	0x0C85007E
0x2005	Ecat SoE: Name ist zur Zeit schreibgeschützt	0x0C85007F
0x3002	Ecat SoE: Übertragungsfehler, Attribut zu kurz	0x0C850080
0x3003	Ecat SoE: Übertragungsfehler, Attribut zu lang	0x0C850081
0x3004	Ecat SoE: Attribut kann nicht geändert werden (read only)	0x0C850082
0x3005	Ecat SoE: Attribut ist zur Zeit schreibgeschützt	0x0C850083
0x4001	Ecat SoE: Keine Einheit	0x0C850084
0x4002	Ecat SoE: Übertragungsfehler, Einheit zu kurz	0x0C850085
0x4003	Ecat SoE: Übertragungsfehler, Einheit zu lang	0x0C850086
0x4004	Ecat SoE: Einheit kann nicht verändert werden (read only)	0x0C850087
0x4005	Ecat SoE: Einheit ist zur Zeit schreibgeschützt	0x0C850088
0x5001	Ecat SoE: Kein Min-Wert	0x0C850089
0x5002	Ecat SoE: Übertragungsfehler, Min-Wert zu kurz	0x0C85008A

Sercos Fehler-code	Beschreibung	ECM Fehler-code
0x5003	Ecat SoE: Übertragungsfehler, Min-Wert zu lang	0x0C85008B
0x5004	Ecat SoE: Min-Wert kann nicht geändert werden (read only)	0x0C85008C
0x5005	Ecat SoE: Min-Wert ist zur Zeit schreibgeschützt	0x0C85008D
0x6002	Ecat SoE: Übertragungsfehler, Max-Wert zu kurz	0x0C85008F
0x6003	Ecat SoE: Übertragungsfehler, Max-Wert zu lang	0x0C850090
0x6004	Ecat SoE: Max-Wert kann nicht geändert werden (read only)	0x0C850091
0x6005	Ecat SoE: Max-Wert ist zur Zeit schreibgeschützt	0x0C850092
0x7001	Ecat SoE: Kein Betriebsdatum	0x0C850093
0x7002	Ecat SoE: Übertragungsfehler, Betriebsdatum zu kurz	0x0C850094
0x7003	Ecat SoE: Übertragungsfehler, Betriebsdatum zu lang	0x0C850095
0x7004	Ecat SoE: Betriebsdatum nicht änderbar (read only)	0x0C850096
0x7005	Ecat SoE: Betriebsdatum ist zur Zeit schreibgeschützt	0x0C850097
0x7006	Ecat SoE: Betriebsdatum ist kleiner als Min-Wert	0x0C850098
0x7007	Ecat SoE: Betriebsdatum ist größer als Max-Wert	0x0C850099
0x7008	Ecat SoE: Ungültiges Betriebsdatum	0x0C85009A
0x7009	Ecat SoE: Betriebsdatum ist schreibgeschützt (Passwort)	0x0C85009B
0x700A	Ecat SoE: Betriebsdatum ist schreibgeschützt (zykl. konf.)	0x0C85009C
0x700B	Ecat SoE: Ungültige Adressierung (Datencontainer, Liste)	0x0C85009D
0x700C	Ecat SoE: Betriebsdatum ist schreibgeschützt (anderweitig)	0x0C85009E
0x7010	Ecat SoE: Befehlsfolge bereits aktiv	0x0C85009F
0x7011	Ecat SoE: Befehlsfolge nicht unterbrechbar	0x0C850100
0x7012	Ecat SoE: Befehlsfolge zur Zeit nicht ausführbar	0x0C850101
0x7013	Ecat SoE: Befehlsfolge nicht ausführbar, Parameter ungültig	0x0C850102
n/a	Ecat SoE: Falsche Antriebsnummer in Antwort	0x0C850103
n/a	Ecat SoE: Falsche IDN in Antwort	0x0C850104
n/a	Ecat SoE: Fragment verloren	0x0C850105
n/a	Ecat SoE: Empfangspuffer voll	0x0C850106
n/a	Ecat SoE: Kein Datenstatus	0x0C850107
n/a	Ecat SoE: Kein Standardwert	0x0C850108
n/a	Ecat SoE: Übertragungsfehler, Standardwert zu lang	0x0C850109
n/a	Ecat SoE: Standardwert kann nicht geändert werden	0x0C85010A
n/a	Ecat SoE: Ungültige Antriebsnummer	0x0C85010B
n/a	Ecat SoE: Allgemeiner Fehler	0x0C85010C
n/a	Ecat SoE: Kein Element adressiert	0x0C85010D

10 FAQs

10.1 ctrlX DRIVE: Wie konfiguriert man SoE mit Free Run?

- **ctrlX DRIVE als Slave Gerät hinzufügen**
ctrlX DRIVE mit entsprechender Version unterhalb EtherCAT Master im Projektexplorer anlegen, hier am Beispiel MPB20:
- **Einstellungen im Slave Dialog "Allgemein"**
Für Free Run Betrieb ist bei der Einstellung "Verteilte Uhren" die Auswahl "Free Run" auszuwählen. Dies ist ein Einstellwert aus der Gerätebeschreibungsdatei des Slaves (hier: aus IndraDrive MPB20).
Zudem ist in diesem Dialog noch "Experteneinstellungen aktivieren" zu setzen, damit der zusätzliche Dialog "Prozessdaten Expertenmodus" sichtbar wird.
- **Einstellungen im Slave Dialog "Prozessdaten Expertenmodus"**
In diesem Dialog können die gewünschten Prozessdaten für den Drive manuell konfiguriert werden.
Im Feld rechts oben "PDO-Liste" wird entweder S-0-0016 (AT) oder S-0-0024 (MDT) ausgewählt, abhängig davon werden rechts unten "PDO-Inhalt" dann die aktuell konfigurierten zyklischen Parameter angezeigt. Mittels den Funktionen Einfügen / Bearbeiten / Löschen können im Dialog rechts unten die gewünschten Parameter für das AT (Antriebstelegramm) bzw. für das MDT (Master-Daten-Telegramm) manuell konfiguriert werden.
Wenn in der Gerätebeschreibung des Slaves keine vordefinierten Parameter vorhanden sind (wie im nachfolgendem Screenshot dargestellt), dann müssen Name, Parameternummer (IDN) und Datentyp manuell richtig eingegeben werden.
- **Einstellungen im Slave Dialog "Startparameter"**
Für Free Run Betrieb ist als Startparameter S-0-0002 ("Sercos Zykluszeit") mit dem gleichen Wert anzulegen wie S-0-0001 ("NC-Zykluszeit"). Während S-0-0001 bereits über die Gerätebeschreibung als Startparameter vordefiniert ist, muss S-0-0002 als 16 Bit Parameter manuell hinzugefügt werden. Der Wert von S-0-0001 entspricht der EtherCAT Zykluszeit und wird automatisch vom Konfigurator gesetzt. Für S-0-0002 ist der gleiche Wert einzustellen.
- **Experteneinstellungen "SyncManager"**
Die SyncManager-Konfiguration steht bei SoE als Default auf "1 Puffer System".
Für den Free Run Betrieb wird die Einstellung "3 Puffer" empfohlen, siehe [EtherCAT Feldbusteilnehmer konfigurieren](#).

10.2 ctrlX DRIVE: Wie verbindet man ctrlX DRIVE Engineering über EtherCAT?

ctrlX DRIVE Engineering kann eine Kommunikationsverbindung zum ctrlX DRIVE beim Rexroth EtherCAT-Master entweder mit SoE oder EoE über den ctrlX Data Layer herstellen.

Variante A

ctrlX DRIVE Engineering über den ctrlX Data Layer mit dem EtherCAT-Master verbinden

In ctrlX DRIVE Engineering ist im Dialog Verbindungsauswahl, Registerkarte Steuerung, als Typ „ctrlX CORE“ auszuwählen.

Als IP-Adresse ist der Engineering-Port der ctrlX CORE Steuerung einzugeben. ctrlX DRIVE Engineering nutzt hierbei eine ctrlX Data Layer Verbindung zur ctrlX CORE Steuerung, die mittels Mailbox SoE azyklisch mit den Antrieben kommuniziert.



Die Variante kann nur für Antriebe mit SoE Profil verwendet werden. Eine EoE Konfiguration oder IP-Routing Einstellungen auf dem Engineering-PC sind hierbei nicht erforderlich.

Für ctrlX DRIVE mit CoE Profil werden nachfolgende Varianten B.) und C.) mit EoE empfohlen.

Variante B

ctrlX DRIVE Engineering mit EoE und Switch Port

Mit einem Switch Port, z.B. Beckhoff EL6601, können dezentral beliebige Ethernet Geräte an ein EtherCAT Netzwerk angeschlossen werden. Auf diese Weise ist IP Kommunikation mit den EtherCAT Slaves möglich.

Beim ctrlX DRIVE muss eine EoE Konfiguration eingestellt werden, bei denen die IP-Einstellungen im gleichen Netzwerk sind wie der Engineering-PC, siehe [Abb. 8](#).

In ctrlX DRIVE Engineering kann dann mittels Netzwerk-Suche oder IP-Adress-Suche eine Verbindung hergestellt werden. Ein IP-Routing auf dem Engineering-PC ist nicht erforderlich.

Variante C

ctrlX DRIVE Engineering mit EoE und IP-Routing im EtherCAT-Master

Der Engineering-PC mit ctrlX DRIVE Engineering ist mit dem Engineering-Port der Steuerung verbunden und die Steuerung führt ein IP Routing auf EoE in das EtherCAT Netzwerk aus.

Beim ctrlX DRIVE und beim EtherCAT-Master muss eine EoE Konfiguration eingestellt werden, so dass eine gültige IP Konfiguration im EtherCAT Netzwerk entsteht, siehe auch [Abb. 8](#).

Bevor in ctrlX DRIVE Engineering dann mittels Netzwerk-Suche oder IP-Adress-Suche eine Verbindung hergestellt werden kann, muss manuell noch die IP Routing-Tabelle des Engineering-PC's mit einem entsprechenden Routing-Eintrag erweitert werden (siehe z.B. Windows Kommandozeile, Befehl „route help“).

11 Weiterführende Dokumentationen

11.1 Übersicht

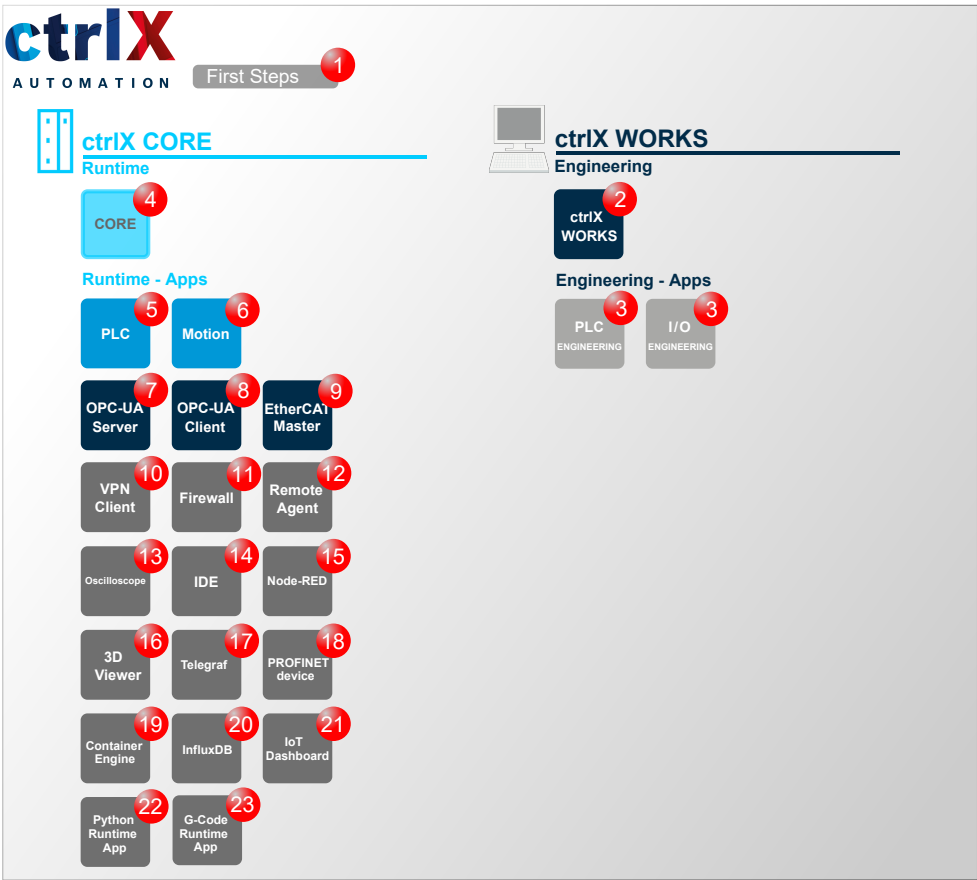


Abb. 24: Übersicht der weiterführenden Dokumentationen

11.2 ctrlX AUTOMATION

Nr.	Dokumentation
1	ctrlX WORKS - Erste Schritte 01VRS Quick Start Guide ↪ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none">• DOK-XWORKS-F*STEP**V01-QURS-DE-P• R911403759

11.3 ctrlX WORKS

Nr.	Dokumentation
2	ctrlX WORKS - Basissystem 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XWORKS-*****V01-APRS-DE-P • R911403762
3	ctrlX PLC Engineering - SPS-Programmiersystem 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XPLC**-ENG*****V01-APRS-DE-P • R911403763
3	ctrlX PLC Engineering - SPS-Bibliotheken 01VRS Referenz ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XPLC**-LIBRARY*V01-RERS-DE-P • R911403765

11.4 ctrlX CORE

Nr.	Dokumentation
4	ctrlX CORE - Runtime 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-BASE****V01-APRS-DE-P • R911403767
	ctrlX CORE - Knoten des Data Layer 01VRS Referenz ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-BASE*DL*V01-RERS-DE-P • R911420071
	ctrlX CORE - Diagnosen 01VRS Referenz ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-DIAG****V01-RERS-DE-P • R911403769

11.5 ctrlX CORE Apps

Nr.	Dokumentation
5	PLC App - SPS-Laufzeitumgebung für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-PLC*****V01-APRS-DE-P • R911403786
6	Motion App - Motion-Laufzeitumgebung für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-MOTION**V01-APRS-DE-P • R911403790
7	OPC UA Server App - OPC UA Server für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-OPCSERV*V01-APRS-DE-P • R911403776
8	OPC UA Client App - OPC UA Client für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-OPCCLIENV01-APRS-DE-P • R911403779
9	EtherCAT Master App - EtherCAT Master für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-ETHERCATV01-APRS-DE-P • R911403771
10	VPN Client App - Fernwartungssoftware für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-VPN*****V01-APRS-DE-P • R911403774
11	Firewall App - Security Funktionen für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-FIREWALLV01-APRS-DE-P • R911403782

Nr.	Dokumentation
12	Remote Agent App - ctrlX Device Portal-Anbindung für ctrlX-Geräte 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-REMOTE**V01-APRS-DE-P • R911403784
13	Oscilloscope App - Oszilloskopfunktion für ctrlX-Geräte 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-OSCI****V01-APRS-DE-P • R911409805
14	IDE App - Integrated Development Environment 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-IDE*****V01-APRS-DE-P • R911410624
15	Node-RED App - Grafische Programmierung für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-NODERED*V01-APRS-DE-P • R911403788
16	3D Viewer App - Browserbasierte 3D-Kinematik-Simulation für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-3D*VIEW*V01-APRS-DE-P • R911416123
17	Telegraf App - Server-Agent zum Sammeln von Daten im Data Layer 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-TELEGRAFV01-APRS-DE-P • R911416837
18	PROFINET Device App - PROFINET device für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-PROFINETV01-APRS-DE-P • R911417858

Nr.	Dokumentation
19	Container Engine App - Verwendung von Docker® Images auf der ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-DOCKER**V01-APRS-DE-P • R911417856
20	InfluxDB App - Influx-Datenbankanbindung für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-INFLUXD*V01-APRS-DE-P • R911418737
21	IoT Dashboard App - Datenvisualisierung in dynamischen, interaktiven Dashboards 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-GDB*****V01-APRS-DE-P • R911420427
22	Python Runtime App - Python-Laufzeitumgebung für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-PYR*****V01-APRS-DE-P • R911420431
23	G-Code Runtime App - G-Code Interpreter für ctrlX CORE 01VRS Anwendungsbeschreibung ↗ Link zur Web-Dokumentation Bestellinformationen: <ul style="list-style-type: none"> • DOK-XCORE*-GCO*****V01-APRS-DE-P • R911420429

12 Service und Support

Für Ihre schnelle und optimale Unterstützung verfügen wir über ein dichtes weltweites Servicenetz. Unsere Experten stehen Ihnen mit Rat und Tat zur Seite. Sie erreichen uns täglich **rund um die Uhr – auch an Wochenenden und Feiertagen**.

Service Deutschland

Unser technologieorientiertes Competence Center in Lohr deckt alle Belange rund um den Service für elektrische Antriebe und Steuerungen ab.

Sie erreichen unsere **Service-Hotline** und unseren **Service-Helpdesk** unter:

Telefon: **+49 9352 40 5060**

Fax: **+49 9352 18 4941**

E-Mail: [↗ service.svc@boschrexroth.de](mailto:service.svc@boschrexroth.de)

Internet: [↗ http://www.boschrexroth.com](http://www.boschrexroth.com)

Auf unseren Internetseiten finden Sie ergänzende Hinweise zu Service, Reparatur (z. B. Anlieferadressen) und Training.

Service weltweit

Außerhalb Deutschlands nehmen Sie bitte zuerst Kontakt mit Ihrem Ansprechpartner auf. Die Hotline-Rufnummern entnehmen Sie bitte den Vertriebsadressen im Internet.

Vorbereitung der Informationen

Wir können Ihnen schnell und effizient helfen, wenn Sie folgende Informationen bereithalten:

- Eine detaillierte Beschreibung der Störung und der Umstände
- Angaben auf dem Typenschild der betreffenden Produkte, insbesondere Typenschlüssel und Seriennummern
- Ihre Kontaktdaten (Telefon-, Faxnummer und E-Mail-Adresse)

13 Index

B

Bestimmungsgemäßer Gebrauch

Anwendungsbereiche.	7
Einleitung.	7
Einsatzfälle.	7

C

ctrlX AUTOMATION

Weiterführende Dokumentationen.	65
--------------------------------------	----

E

EtherCAT

Feldbusteilnehmer konfigurieren.	41
Konfiguration Übersicht.	37

EtherCAT FAQs

Wie konfiguriert man SoE mit Free Run?. .	63
Wie verbindet man ctrlX DRIVE Engineering über EtherCAT?.	63

EtherCAT Grundlagen

Adressierungs- und Identifikationsverfahren	16
Azyklische Kommunikation (Mailbox).	19
Diagnose AL Status Codes.	56
Diagnose Allgemein.	55
Diagnose CoE SDO Abortcodes.	58
Diagnose FoE Fehlercodes.	59
Diagnose SoE Fehlercodes.	60
EtherCAT Slave Controller.	24
EtherCAT Slave Controller / Aufbau.	24
EtherCAT Slave Controller / Kommandos. .	25
Hot Connect.	23
Synchronisation / Allgemein.	27
Synchronisation / Free Run.	27
Synchronisation mit DC-Sync.	30
Synchronisation mit SM-Event.	28
SyncUnits.	22
Telegramm-Aufbau.	21
Topologie.	14
Topologie / Kabelredundanz.	15
WorkingCounter.	22
Zustandsmaschine.	17

EtherCAT Konfiguration Übersicht

Anschlüsse.	37
------------------	----

EtherCAT Master

Features.	39
Grundlagen.	11
konfigurieren.	40

F

Fenster

Distributed Clocks.	48
EtherCAT Master.	43
EtherCAT Master Detailansicht.	44
Slaves.	48

H

Header

EtherCAT Master.	45
EtherCAT Slave.	46

Helpdesk. 70 |

Hotline. 70 |

K

Kommunikation und Feldbusse

Begriffe und Abkürzungen.	11
--------------------------------	----

N

Nicht bestimmungsgemäßer Gebrauch. 8 |

Folgen, Haftungsausschluss.	7
----------------------------------	---

R

Registerkarte

AoE.	52
EoE.	53
EtherCAT Slave "Allgemeines".	54
Master-Statistik.	50
Slave-Statistik.	51

S

Seitennavigation

EtherCAT Master.	43
Fenster – EtherCAT Master.	43
Fenster – EtherCAT Master Detailansicht. .	44
Header – EtherCAT Master.	45
Header – EtherCAT Slave.	46
Registerkarte – Distributed Clocks.	48
Registerkarte – EtherCAT Slave "Allgemeines"	54
Registerkarte – Master-Statistik.	50
Registerkarte – Slave-Statistik.	51
Registerkarte – Slaves.	48

Service-Hotline. 70 |

Sicherheitshinweise. 9 |

Support. 70 |

U

Übertragungsarten

Feldbus-Master.	13
Feldbus-Slaves.	13

Bosch Rexroth AG
Bgm.-Dr.-Nebel-Str. 2
97816 Lohr a.Main
Germany
Tel. +49 9352 18 0
Fax +49 9352 18 8400
www.boschrexroth.com/electrics



R911403771