miControl®

EtherCAT Interface

Digital Servo Amplifiers with EtherCAT Interface

Version 0.01

miControl®
Blankenfelder Chaussee 1
15831 Großbeeren (Diedersdorf)

Tel. +49 (3379) 31259-0
Fax +49 (3379) 31259-19

http://www.miControl.de
mailto:mc@miControl.de

Table of contents

1 Introduction

1.1 Used symbols

Symbol	Description	Meaning
	**Danger !**	Warns against immediate human health risk. Failure to comply can result in death, grievous body harm and/or considerable damage if the necessary precautions are not been taken.
	**Warning !**	Warns against possible danger. Failure to comply can result in death, grievous body harm and/or considerable damage if the necessary precautions are not been taken.
	**Caution !**	Warns against a possibly dangerous situation. Failure to comply can result in grievous body harm and/or considerable damage if the necessary precautions are not been taken.
	**Important !**	Symbolizes important information about the device or its method of operating.
	**Note**	Symbolizes a comment about the device, its parameters or its method of operating.
	**Hardware**	Hints related to hardware.
	**Tip**	Symbolizes a useful hint.
	**Help**	Support help
	**FAQ**	Frequently asked questions.
	**Source code**	Marks a section with source code.
	**Example**	Symbolizes an example.

1.2 Abbreviations

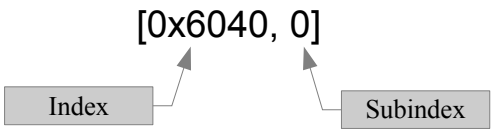
Term	Meaning
MPU	Motion Process Unit. Unit (developed by miControl) to programming logical connections and sequences. (SPS functionality).
MPU2	Motion Process Unit. Unit (developed by miControl) to programming logical connections and sequences. (SPS functionality). Programmable in a Python script.
CAN	Controller Area Network
CANopen	Communication protocol for CAN-Bus System. CANopen based on CAL (CAN Application Layer).
COB	Communications Object
COB-ID	CAN Communication Object Identifier
Node-ID	CAN Node Identifier
PDO	Process Data Object
RxPDO	Receive-PDO
TxPDO	Transmit-PDO
SDO	Service Data Object
EDS	Electronic Data Sheet
PWM	Pulse Wide Modulation
AIN	Analog input
AOUT	Analog output
DIN	Digital input
DOUT	Digital output
Baudrate	Transmission speed of the conveyance of communication
CAN Master	„Command center" of the CAN Bus
Bus	Communication network where all participants can be reached by passive connections and where communication is possible in both directions.
Default settings	Default settings
DSP 402	CANopen device profile for drivers and controllers
Hall sensors	Sensors for determining the position of a rotor
Incremental encoder	Digital encoder. An integrated logic processes the signal of the photo diodes and creates two square-wave signals with a phase shift of 90°.

1.3 Convention of document

1.3.1 Parameters

Device parameter is address through Index (16-Bit value) and Subindex (8-Bit value).

Notation (Index = 0x6040, Subindex = 0):



Prefix „0x“ or suffix „h“ next to number means HEX-Notation.

Example: 0x6040 or 6040h

Index of parameter will be usually written in HEX-Notation.

1.3.2 Hints

Hints will be marked in following way:

> TIP
> „Description of useful tip”

1.3.3 Source code - Listings

Listings of program will be marked in following way:

```
from mc.dsa import *

d = Dsa(1)

d.Enable()
d.ModeVel()
d.Vel(1000)
```

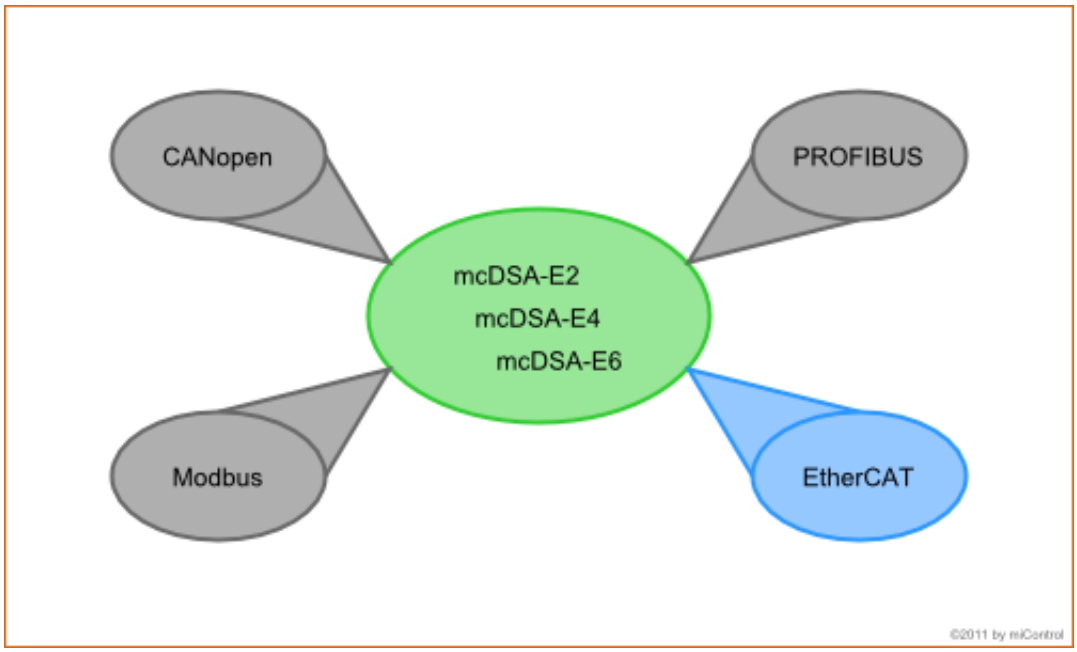
2 System description

Ethernet for Control Automation Technology (EtherCAT) is a control system protocol typically used to communicate with controllers and devices in the manufacturing sector with very high performance requirements. The EtherCAT protocol is a special, officially assigned Ether-type in the Ethernet protocol. This allows EtherCAT to place control data directly into the Ethernet frame without requiring a higher layer protocol.

The master unit has a standard Ethernet connection with a standard Ethernet chipset. It can be a gateway device to any other protocol that passes over the Ethernet network. The slave units sit behind the master unit and are not on the typical Ethernet network. The master and slave units can be considered one device on an Ethernet subnet. The EtherCAT 'bus' can be almost any topology: line, tree or star. It will work with Ethernet switches, but does not require Ethernet switches

The telegram (EtherCAT message) can include data for multiple slave stations, which increases the efficiency of the protocol. A slave evaluates the EtherCAT frame, determines if any part of the telegram is for it, and extracts the appropriate data. The evaluation takes place 'on the fly' as the frame is being processed. It is not necessary to send the packet to the device CPU to extract the information destined for the slave station.

EtherCAT telegrams are sent from master to each slave … until the last slave returns the telegram to the Master as a response telegram. In a typical EtherCAT network this is all occurring over a physical Ethernet network using the EtherCAT slave protocol.

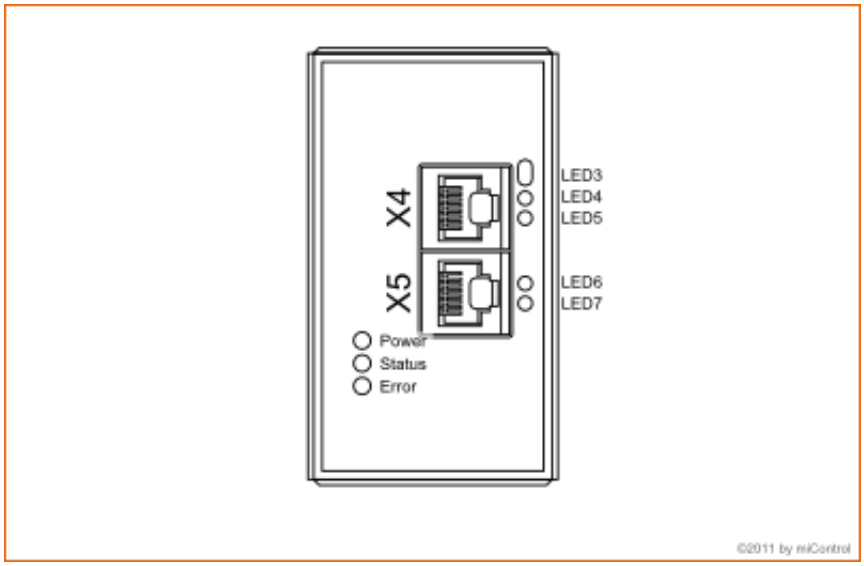


3 Specifications

Description	EtherCAT series
Physical layer	100 Base-Tx EtherCAT
Bus controller	ET1100
Baud rate	100 Mbit/s
Cable type	CAT5
Length of cable	max. 100 m
Ports	2x RJ45
Diagnostic LEDs	Link activity, Collision, EtherCAT state
Supported communication protocol	CoE (CANopen over EtherCAT)

4 Display elements

4.1 mcDSA-E4-EtherCAT



The controller has three LEDs which display the current status of the controller.

LED	Colour	Status	Description	
LED0 „Power“	green	on	Normal operation	
		off	Power supply is missing	
		flashing	Setup mode active	
LED1 „Status“	yellow	on	CANopen pre-operational mode (PDOs not active)	
		off	CANopen operational mode (all PDOs active)	
		flashing	Bootloader-mode (incoming message)	
LED2 „Error“	red	on	Error	
		off	No error, normal operation	
		flashing	Error state	
			Number of pulses	**Description**
			0	Error
			1	Short circuit on digital output
			2	Motor supply under voltage
			3	Over temperature
			4	Communication error

The EtherCAT module has five LEDs which display the current status of the module.

LED	Colour	Status	Description
LED3 „EtherCAT“	red/green	red, on	Initialising, no data exchange
		red/green, 1:1	Pre-operational, no data exchange
		red/green, 3:1	Safe operational, inputs readable
		green, on	Operational, unrestricted data exchange
LED4 „Collision“	gelb	on	Collision (Port X4)
		off	No collision (Port X4)
LED5 „Link Act.“	green	on	Connected to Ethernet (Port X4)
		off	No Ethernet connection (Port X4)
		flashing	Exchanging telegram (Port X4)
LED6 „Collision“	gelb	on	No collision (Port X5)
		off	Collision (Port X5)
LED7 „Link Act.“	green	on	Connected to Ethernet (Port X5)
		off	No Ethernet connection (Port X5)
		flashing	Exchanging telegram (Port X5)

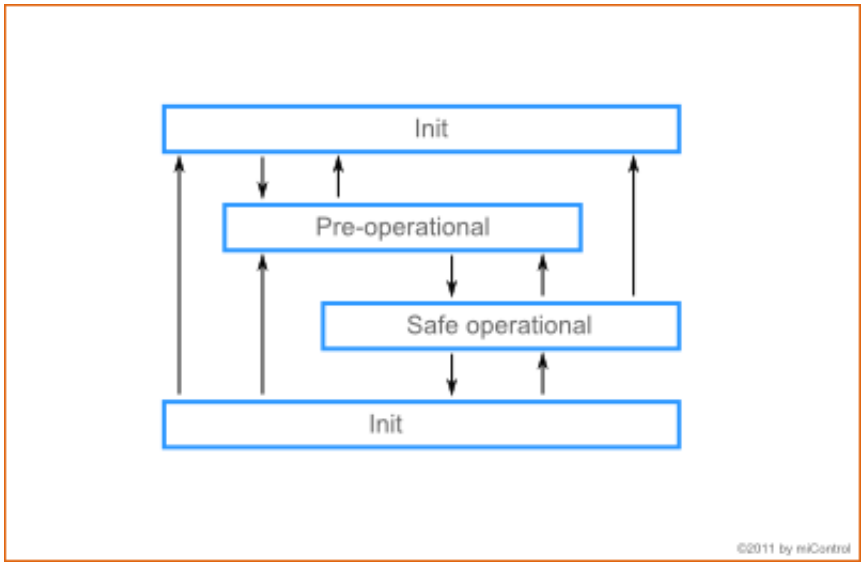
5 EtherCAT State machine

The EtherCAT State machine is responsible for the coordination of master and slave applications at start up and during operation. State changes are typically initiated by requests of the master. They are acknowledged by the slave application after the associated operations have been executed. Unsolicited state changes of the slave application are also possible.

The Slave supports four states an EtherCAT:

- Init
- Pre-operational
- Safe operational
- Operational

The states and the allowed state changes are shown below.



The active services of each state are shown in the table below.

State / State change	Services
INIT	• No communication on Application Layer • Master has access to the DL-Information registers
INIT TO PREOP	• Master configures registers • Master initializes DC clock synchronization • Master requests 'Pre-Operational' state • Slave confirms state
PREOP	• Mailbox communication on the Application Layer • No Process Data communication
PREOP TO SAFEOP	• Master configures parameters using the Mailbox (e.g., Process Data Mapping) • Master configures DL Register (SyncManager and FMMU channels) • Master requests 'Safe-Operational' state • Slave confirms state
SAFEOP	• Mailbox communication on the Application Layer • Process Data communication, but only Inputs are evaluated – Outputs remain in 'Safe' state

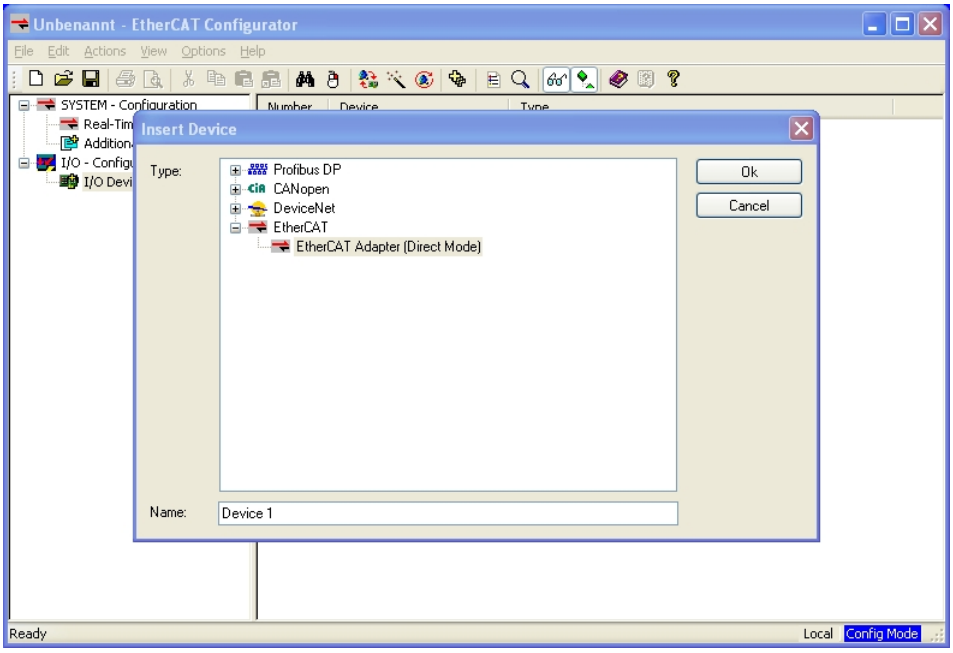
State / State change	Services
SAFEOP TO OP	• Master sends valid Outputs • Master requests ‘Operational’ state • Slave confirms state
OP	• Inputs and Outputs are valid

6 Configuration

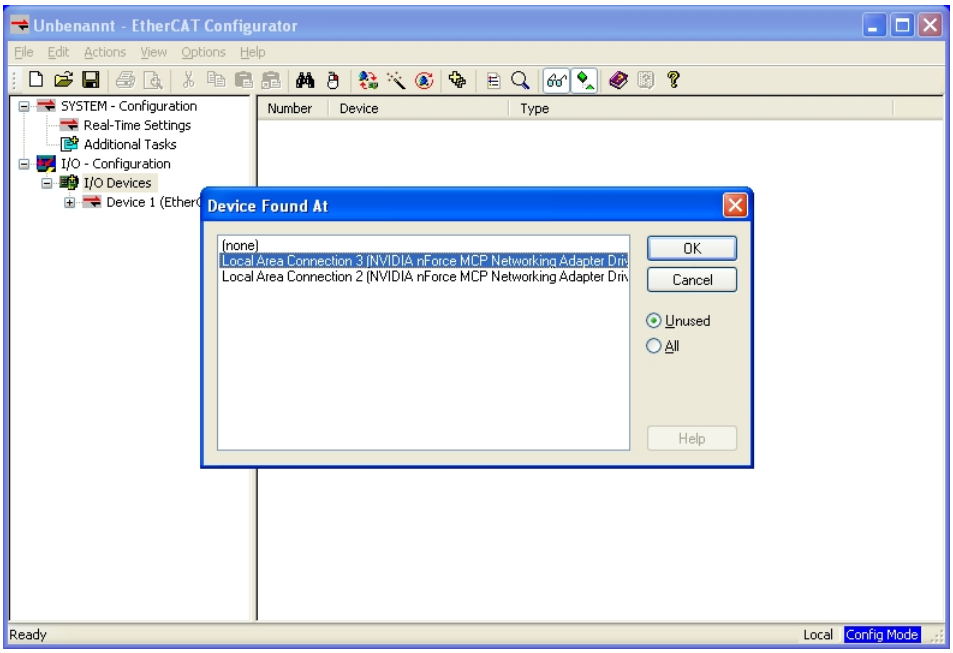
The EtherCAT master needs to be configured to drive the EtherCAT network. One major part of the configuration is to specify the EtherCAT slave stations.
The XML device files provide EtherCAT administrators with convenient options. EtherCAT allows both, a configuration offline and the scanning of station data via an Ethernet line (online configuration).
The examples below are based around the standard ETG configuration tool (EtherCAT configuration tool supplied by Beckhoff Automation GmbH) which accesses the XML device files for both offline and online configuration.
The file to use for mcDSA controller with EtherCAT interface is called „mcDSA-xx-EtherCAT.xml“.
Copy file „mcDSA-xx-EtherCAT.xml“ to folder „C:\Program Files\EtherCAT Configurator\EtherCAT“ or, if you are using another tool, to the folder set for that tool.

6.1 Offline configuration

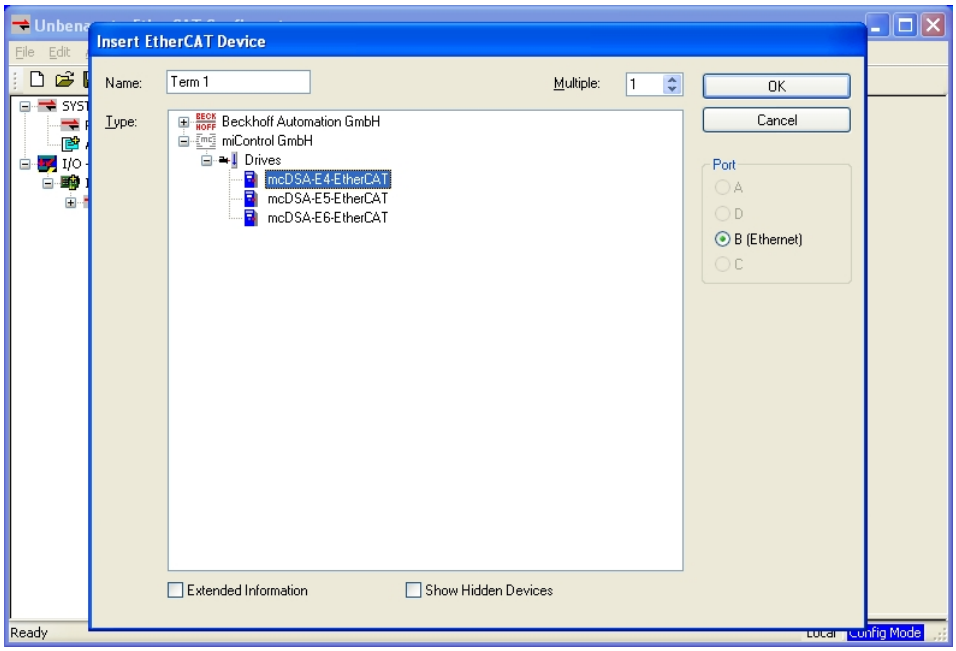
1. Start the EtherCAT configuration tool.
2. Choose File, New to create a new I/O Configuration.
3. Select „I/O Devices“ and run „Append Device...“. This will add „Device 1“ to your configuration.



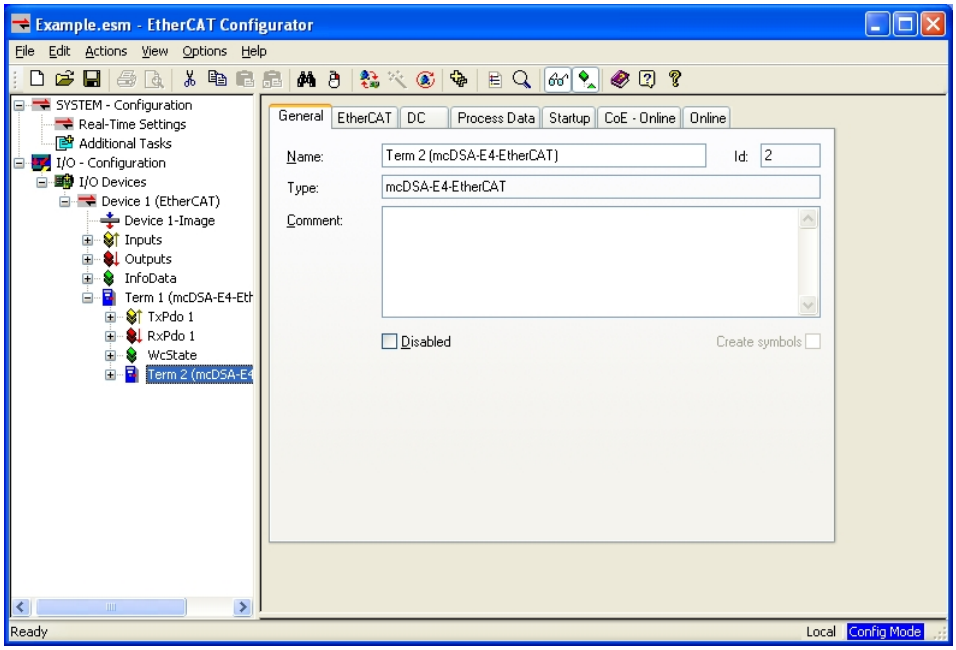
4. Assuming your PC has several Ethernet ports you must first select the one that the EtherCAT line is attached to.



5. Select „Device 1“ and run „Append Box...“.
6. Expand the branch labelled „miControl GmbH“ and its sub-branch and select e.g. „mcDSA-E4-EtherCAT“.



7. Repeat the step 5 and 6 if you want to add more devices.
8. Save the finished configuration as an *.esm type file.

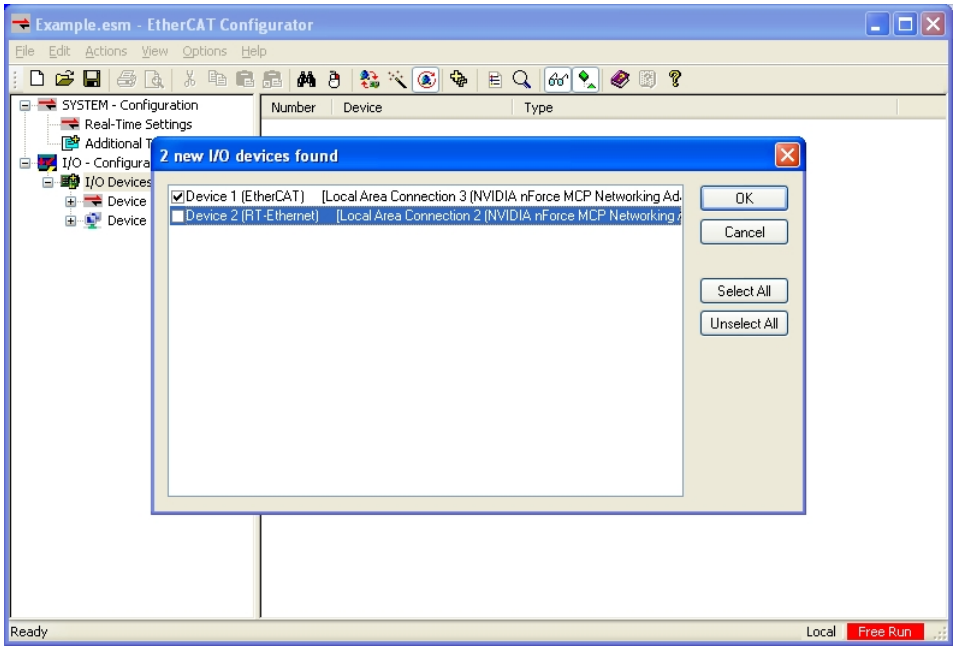


If your PC is connected to the EtherCAT module via the Ethernet, you can already run an online test of the mcDSA controller.

6.2 Online configuration

EtherCAT allows you to scan the stations connected to an Ethernet line. The task of the example below is to retrieve the configuration of an mcDSA-E4-EtherCAT unit.

1. Turn on the power supply (electronic) of mcDSA-E4-EtherCAT.
2. Attach a CAT5 cable to the Ethernet port of your PC to connect it to the mcDSA-E4-EtherCAT. You are free to attach either a patch cable or a cross cable.
3. Start the EtherCAT configuration tool.
4. Choose File, New to create a new I/O Configuration.
5. Select „I/O Devices“ and run „Scan Devices...“. Assuming your PC has several Ethernet ports you must first select the one that the EtherCAT line is attached to.



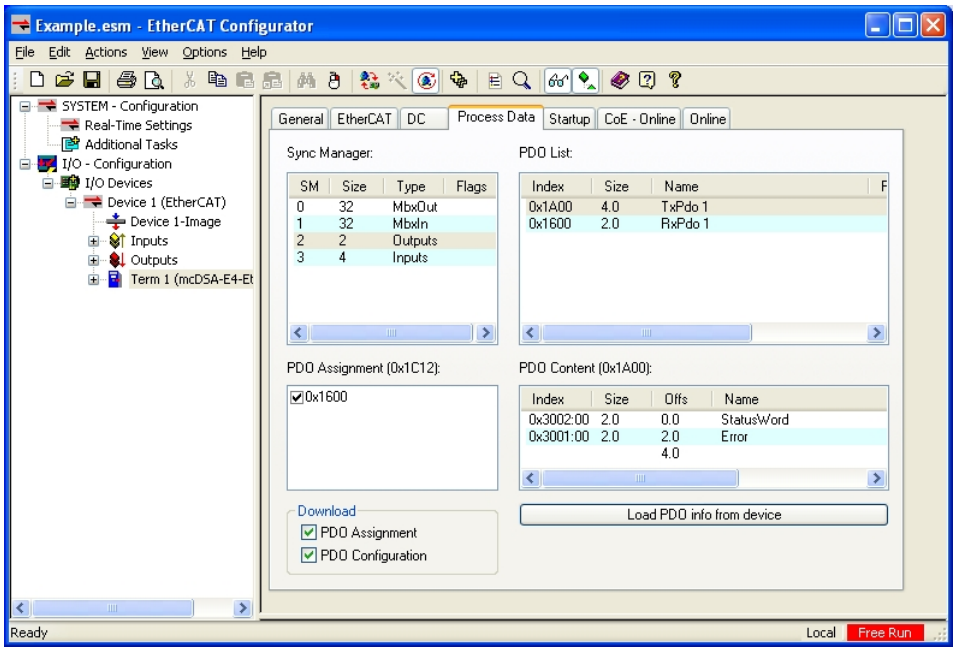
6. Click on "Yes" to answer prompt "Scan for boxes".
7. Save the finished configuration as an *.esm type file.

6.3 EthetCAT Slave Device „Process Data“ tab

This tab displays the configuration of the process data. The input and output variables of the slave device are displayed as CANopen Process Data Objects(PDO). The device enables the user to select a different PDO via PDO Assignment and to alter the contents of an individual PDO. Each mcDSA controller supports 8 TxPDOs and 8 RxPDOs:

- RxPDO1 - Index 0x1600
- RxPDO2 - Index 0x1601
- RxPDO3 - Index 0x1602
- RxPDO4 - Index 0x1603
- RxPDO5 - Index 0x1604
- RxPDO6 - Index 0x1605
- RxPDO7 - Index 0x1606
- RxPDO8 - Index 0x1614

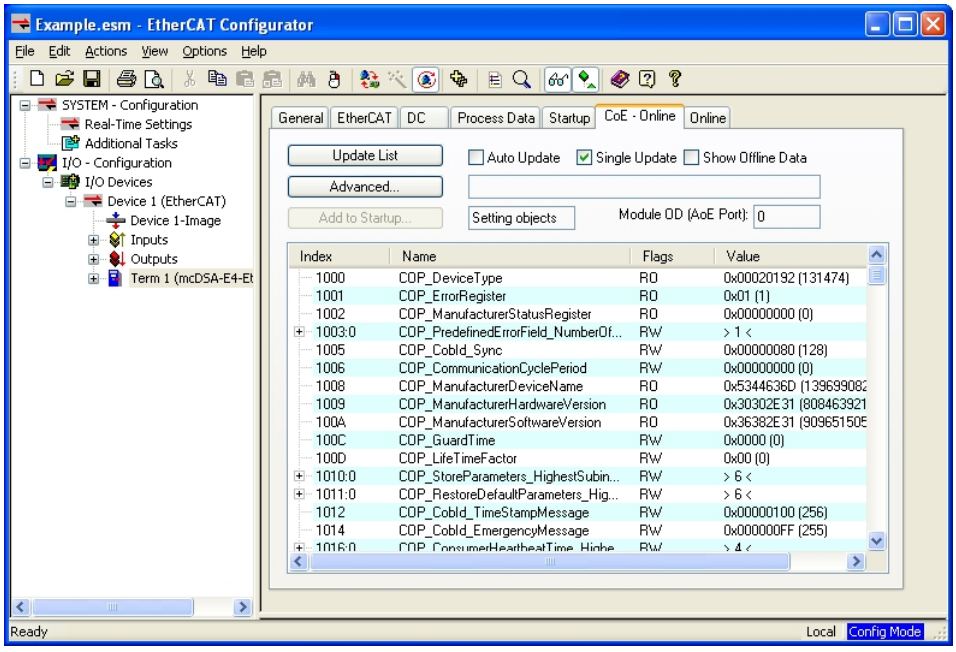
- TxPDO1 - Index 0x1A00
- TxPDO2 - Index 0x1A01
- TxPDO3 - Index 0x1A02
- TxPDO4 - Index 0x1A03
- TxPDO5 - Index 0x1A04
- TxPDO6 - Index 0x1A05
- TxPDO7 - Index 0x1A06
- TxPDO8 - Index 0x1A14



For more information about the Mapping see description of the parameters 0x1600 and 0x1A00 in „DsaPar_en.chm“

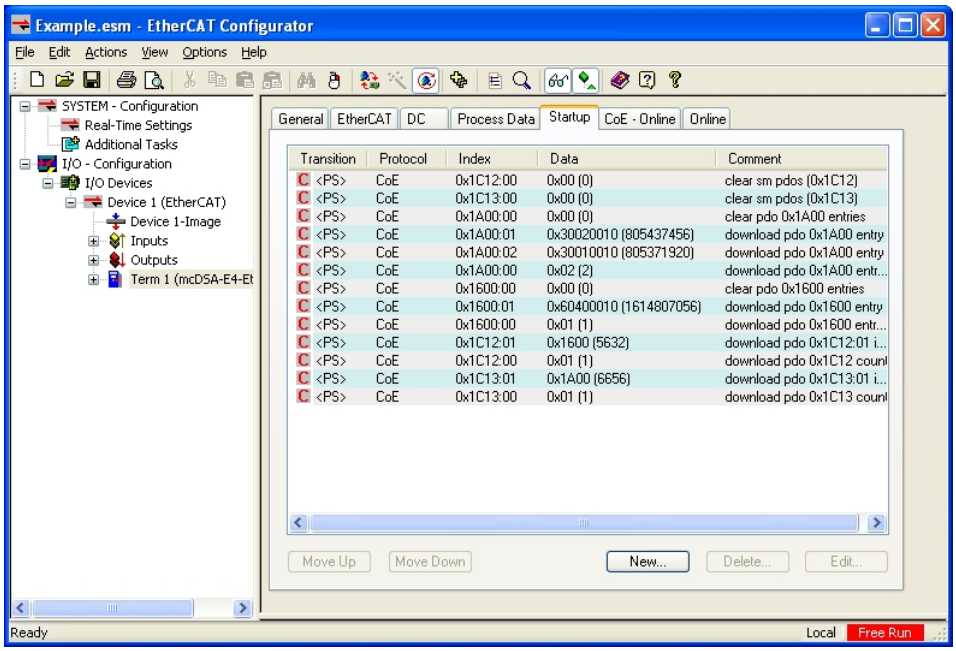
6.4 EtherCAT Slave Device „CoE-Online“ tab

This tab enables the user to view the contents of the object dictionary of the slave(SDO-Read) and to change the contents of an object in the dictionary(SDO-Write).



6.5 EtherCAT Slave Device „Startup“ tab

With the help of the "Startup" tab the user can view which download requests are sent to the mailbox during startup. It is also possible to add new mailbox requests to the list view. The download requests are sent to the slave in the same order as displayed in the list view.



The meaning and description of each parameter you will find in file „DsaPar_en.chm“.

7 Maintenance and Service

7.1 Service

miControl
Blankenfelder Chaussee 1
15831 Großbeeren (Diedersdorf)

www.miControl.de
email: mc@miControl.de

7.2 Warranty

The warranty expires with the end of the legal warranty term and/or opening the module housing. miControl does not give any warranty for damage on the module which result from incorrect wiring.

7.3 Storage

Please store the module protected against dust, dirt and humidity. Pay care that the storage temperature and/or humidity is within the specified range (see „Performance data “).

7.4 Shipment

Please ship the modules as 7.3 Storage but shock protected.

7.5 Disposal

The module with its high integrated printed circuit board is special electronic waste and must be disposed according to the regulations of your country.

Important information