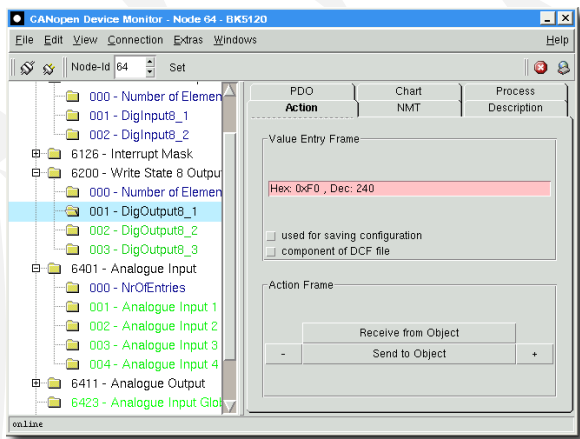


# CANopen Device Monitor

## Overview

The *CANopen Device Monitor* is a CANopen Master tool for the graphical inspection and configuration of CANopen devices in a CANopen network. The embedded scripting ability provides access to the implemented CANopen services and makes it possible to write test or control applications with a minimum of effort.



The information on how to display the device object dictionary is obtained from the EDS file of the device (EDS - Electronic Data Sheet, text- or XML format according to ISO 15745) or can be scanned from the device directly. Special device configurations can be loaded, stored and scanned from DCF files. Additionally, configurations of complete networks can be saved in project files.

## Description

The program's graphical user interface consists of

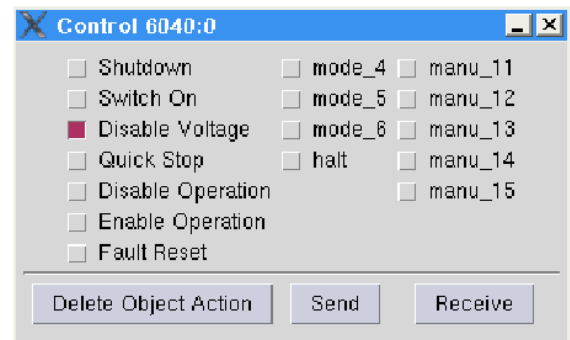
- a hierarchical tree representation of the object dictionary
- various action fields
- an optional command console interface

Displaying all implemented objects as a tree facilitates the configuration by hand.

Devices with up to 8 device profiles are supported.

The *tab* Action is used for reading and writing device objects with CANopen SDO transfer messages.

Optionally a slider with a user defined value range can be assigned to every object to achieve a more useful configuration. For each object the meaning of each bit can be displayed by special bit boxes. There are already prepared bit boxes for special objects like the CiA 402 control word.



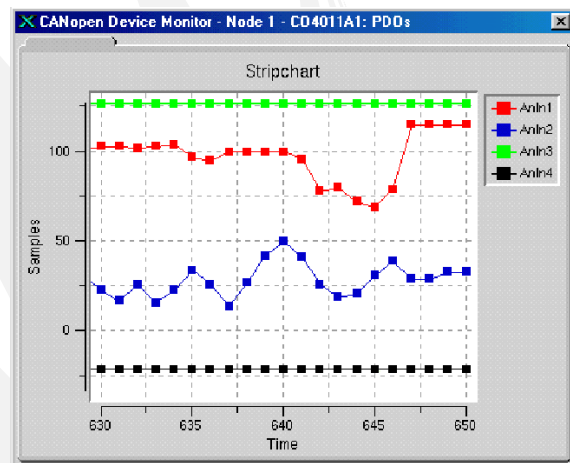
Values of manually configured device objects can be stored to a file and reloaded into the device.

The *NMT* *tab* makes several fields available for the execution of NMT services like *Start All*, *Start Node*, *Preop All*, *Preop Node* and *Scan Network*. The device properties of the currently present nodes in the network are shown as well.

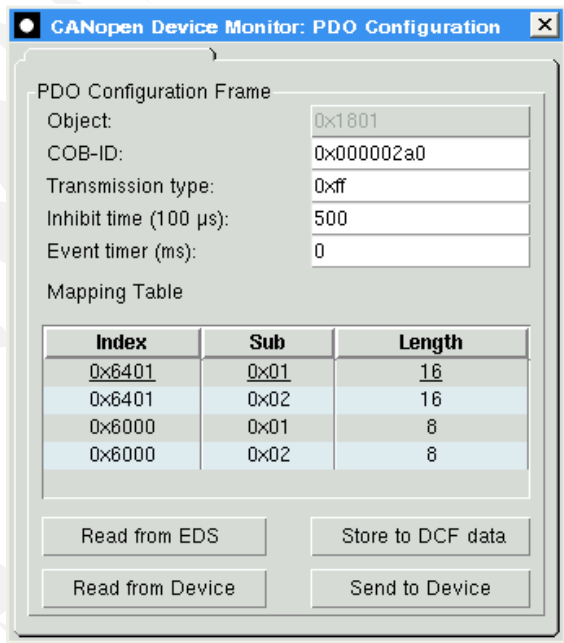
A graphical network overview displays all devices in a CANopen network.

Network configurations can be saved in project files.

Under the *tab* Description an object description of the currently selected object in the object tree is shown. The description is taken from a text file which is created by the *CANopen Design Tool* or manually written with a common text editor.



The integrated process image summarizes the most important process variables of the network. The data can be received via SDO and PDO. The *PDO* *tab* allows the simple configuration of the PDO-Mapping by Drag&Drop. Additionally PDOs can be sent and received and data from received PDOs can be plotted in a stripchart with one click.



*PDO configuration mask*

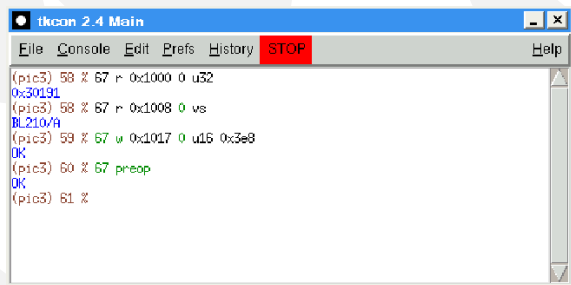
Additional tabs are available for further graphical representations of CAN data and can be defined by the user.

Extensions to control drives according to CiA 402, a CANopen-Safety-Plugin (CiA 304) and a LSS-module (CiA 305) are additionally available.

The CAN communication can be logged, saved and evaluated afterwards.

The console is an extra feature. With the console an interactive access to the CANopen network is possible by using commands for:

- SDO and PDO transfers
- network management services
- Heartbeat and Node Guarding
- Time Stamp, SYNC, Emergency



Complete sequences of commands can be combined to procedures by the embedded scripting language. The scripting language is available for different operating systems.

By using the *CANopen Library* and the *CANopen Design Tool* from **port** the developer gets a closed tool chain for creating CANopen compatible devices.

Through the scripting language access to elements of the graphical interface is also available. The graphical representation of values or test results is therefore application specific extensible.

**port** GmbH can perform special adaptations to the GUI or behavior to meet CANopen device characteristics for service or assembly testing.

## CAN-Interfaces

The usage of the *CANopen Device Monitor* requires a CAN interface. CAN interfaces are currently available for USB, ISA, PCI, parallel port, serial port and PC-104 or an Ethernet interface like the standalone CANopen-Gateway-Server EtherCAN of **port**.

## System Requirements

- Operating System: Windows™ 2000, XP, Vista  
LINUX™
- Processor: Pentium IV or later
- RAM: 256 MByte
- Harddisk Space: >20 MByte available

## Ordering Information

- 0642/10 CANopen-DM-W (Full Version, Windows™)
- 0642/20 CANopen-DM-L (Full Version, LINUX™)
- 0642/11 CANopen-DM-Safety
- 0642/12 CANopen-DM-LSS
- 0642/13 CANopen-DM-402

Functional demo versions of the software tools are available for download on [www.canopen-tools.com](http://www.canopen-tools.com).

### Engineering Services

*port* is providing engineering services and trainings for our business activities:

- CAN and CAN-based protocols: CANopen, J1939, DeviceNet
- Industrial Ethernet Protocols: POWERLINK, EtherNet/IP, EtherCAT
- Implementation of devices according to CANopen device profiles
- VHDL based solutions for industrial applications
- application specific implementations or enhancements
- embedded LINUX projects

### Notice

Brands and product names are trademarks or registered trademarks of their respective companies. The product will be continuously improved. *port* therefore reserves the right to change technical properties at any time without appointment.



*port* GmbH  
Regensburger Straße 7b  
D-06132 Halle/Saale  
+49 345 777 55 0  
service@port.de