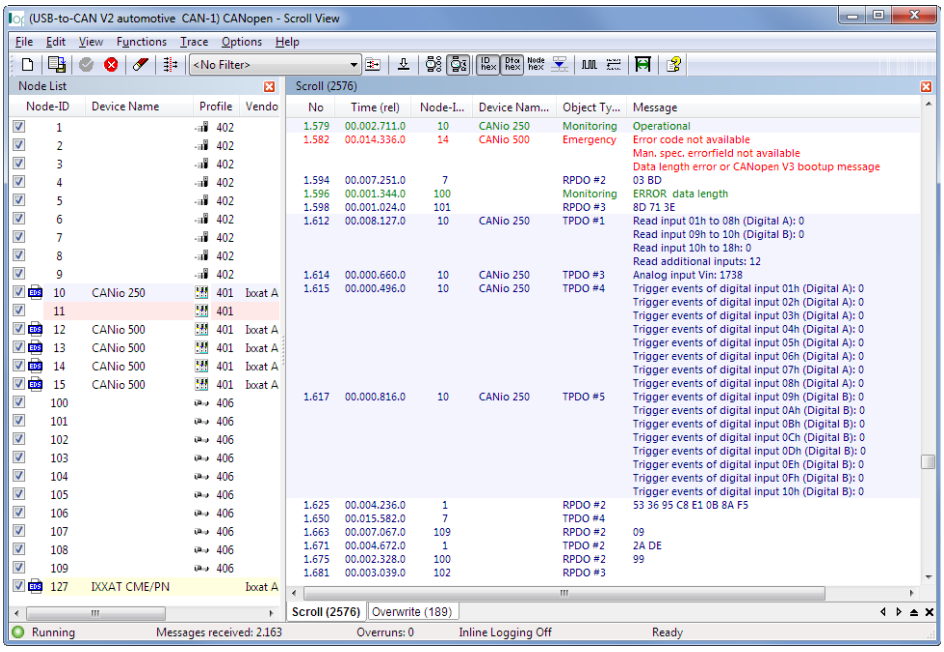
Software Manual

CANopen-Module

canAnalyser3 Module
for CANopen Protocol Interpretation

Software Version 3.2



HMS Technology Center Ravensburg GmbH
Helmut-Vetter-Straße 2
88213 Ravensburg
Germany

Tel.: +49 751 56146-0
Fax: +49 751 56146-29
Internet: www.hms-networks.de
E-Mail: info-ravensburg@hms-networks.de

Support

In case of unsolvable problems with this product or other HMS products please contact HMS in written form:

Fax: +49 751 56146-29
E-Mail: support@ixxat.de

Further international support contacts can be found on our webpage www.hms-networks.de

Copyright

Registered trademarks

Document number: 4.02.0145.20000
Version: 3.8

Contents

Chapter 1

Overview

1.1 General

The CANopen-Module (Fig. 1.1) is an add-on module for the canAnalyser3 and provides the display of received layer-2 messages in CANopen notation. The messages are displayed in accordance with the configuration and the CAN-in-Automation [1] A.2 [7] A.2 specified communication objects, assigned to individual nodes, their content extracted and output in plaintext. In addition, it is possible to display the corresponding layer-2 messages of the protocol-specific interpretation.

The interpretation is based on a network model (analysis configuration) in which the individual nodes connected to the CANopen network to be analyzed are described. This description can be carried out by reading in device description files (EDS-, DCF- and XDD-files)[5] A.2 [6] A.2, by simple profile assignment (default configuration), by online node scan, or completely manually.

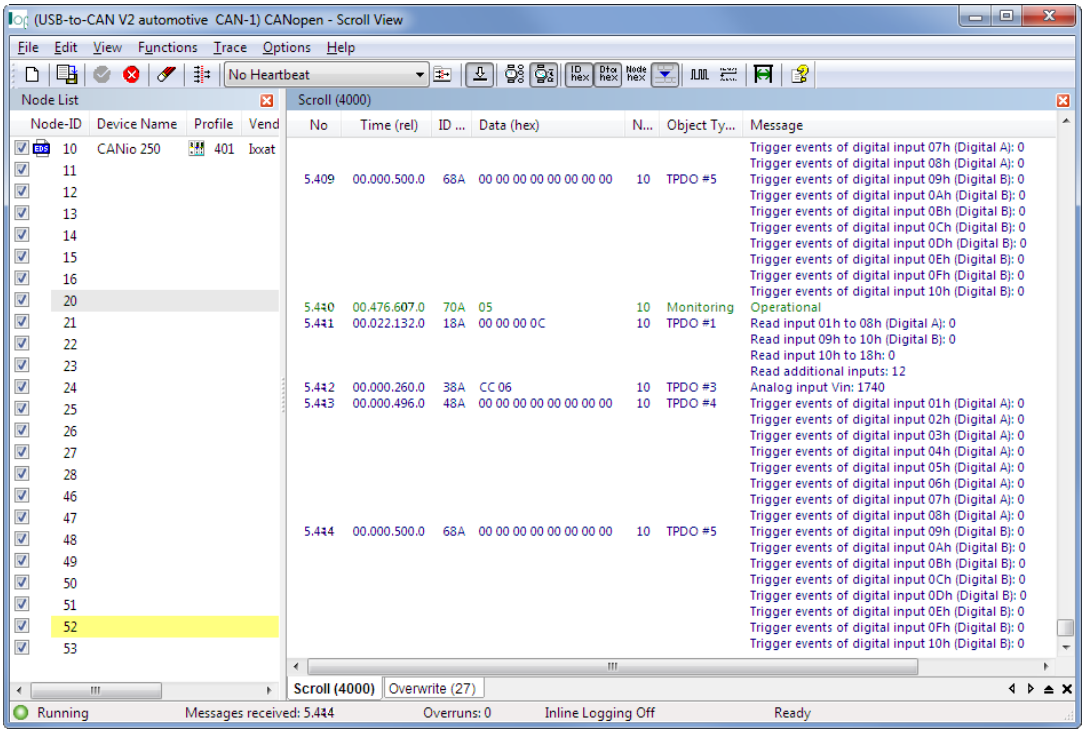


Figure 1.1: CANopen-Module

1.2 System requirements

- Installed canAnalyser3

1.3 Highlights

- Interpretation of layer-2 messages in CANopen format
- Continuous display of the bus traffic in real-time
- Cumulated display sorted by CAN identifier in ascending order with highlighted changed data and statistical data
- Output of data contents and descriptive names of objects transmitted in PDOs
- Nondestructive display filter according to node-ID and object type (SDO, PDO, NMT etc.)
- Output of segmented SDO transfers as complete message or individual messages
- Freely configurable CAN identifiers for every node with import of EDS/DCF/XDD files
- Support of SDO block transfer protocols, multiplex PDOs and SRDOs
- Indication of non-interpretable messages
- Display of the raw (non-interpreted) CAN telegrams
- Analysis of the SDO command and protocol bytes in plaintext
- Interpretation of the LSS protocol [4] A.2
- Interpretation of the Flying Master protocol [3] A.2
- On-line logging of the interpreted messages to file
- Configurable display color for each object type
- Colored background of all messages related to a certain node
- Network scan with automatic identification of device profile and device identity
- Changing of device profiles during runtime

Chapter 2

Installation and start-up

2.1 System requirements

The requirement for installation of CANopen-Modules is an installed, functional canAnalyser3 or higher.

2.2 Installation

To install the CANopen-Module, insert the provided program CD into the drive of your computer. Windows will automatically start the installation program of the CANopen-Module. If this is not the case, please run the file "CANopenModule32.exe".

2.3 Starting the CANopen-Module

In order to start the CANopen-Module, the canAnalyser3 must first be called. The module now appears in the `Modules Window` of the Control Panel (Fig. 2.1). From here, the module can be dragged to a CAN line. For more information, see the canAnalyser3 user's manual.
If the configuration is saved, the canAnalyser3 starts the CANopen-Module automatically by loading this analysis configuration the next time.
Starting separate instances of CANopen-Module on different CAN lines is supported (not possible with canAnalyser3 lite). These instances work independently and can therefore be configured individually (with regard to filter settings, device description files, colors etc.)

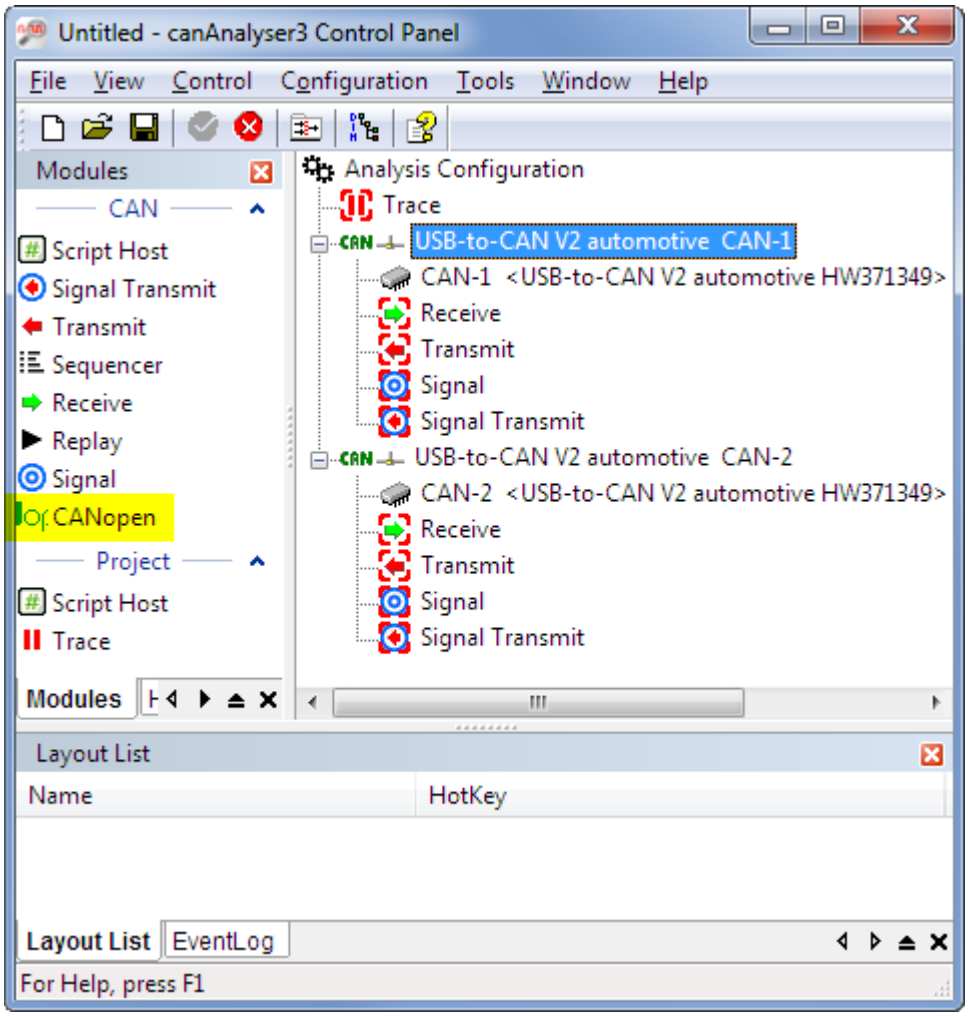


Figure 2.1: Control Panel with CANopen-Module

Chapter 3

Use of the CANopen-Module

3.1 Message display

Interpreted messages can be displayed one by one in the order of their reception, or cumulated sorted by CAN identifier in ascending order.
The one by one display is referred to as scroll view and can be found on the `Scroll` tab, whereas the cumulated view is displayed on the `Overwrite` tab. Use hotkey `Ctrl-TAB` to switch between the views.
For each of the views, the following display options can be set:

- Various column values can be displayed in hexadecimal or decimal format. The display format for the individual columns is selected via the menu `View`, the toolbar or by right mouse click on the corresponding column header.
- To display the corresponding layer-2 messages in two extra columns named `ID` and `Data`, the menu command `View | Show also raw CAN data` has to be enabled.
- The font and face can be freely selected. This is done using the menu command `Options | Font...`

3.1.1 Scroll View

In this view, the messages are listed in order of their reception with the following information (Fig. 1.1):

Column	Meaning
No	Continuous distinct number of received message.
Time (rel/abs)	Time stamp of the reception, either absolute in UTC time format or relative to the previously received message; the display of hours can be switched on and off by right-clicking on the column heading
ID (hex/dec)	CAN-identifier of received message. This column is visible only if layer-2 messages display is enabled.
Data (hex/dec)	Received CAN data bytes readout. This column is visible only if layer-2 messages display is enabled.
Node-ID (dec/hex)	CANopen Node-ID in decimal or hexadecimal notation, depending on whether the View menu item Node-ID hex is checked or not.
Device Name	Name of the device according to its device description file.
Object Type	Type of CANopen communication object. The following object types are distinguished: Server-SDO (SSDO), Client-SDO (CSDO), Transmit-PDO (TPDO), Receive-PDO (RPDO), Multiplex-RPDO (M-RPDO), Multiplex-TPDO (M-TPDO), Network management (NMT), Emergency object (Emergency), Node error control (Monitoring), Time stamp object (Timestamp), Sync Object (Sync), Layer setting services (LSS), Flying master message (FLY MASTER), Transmit safety relevant data object (TSRDO), Receive safety relevant data object (RSRDO). The relative consecutive number of the object is appended where applicable.
Message	Interpreted CANopen message. This multi-line column finally shows the plaintext CANopen message.

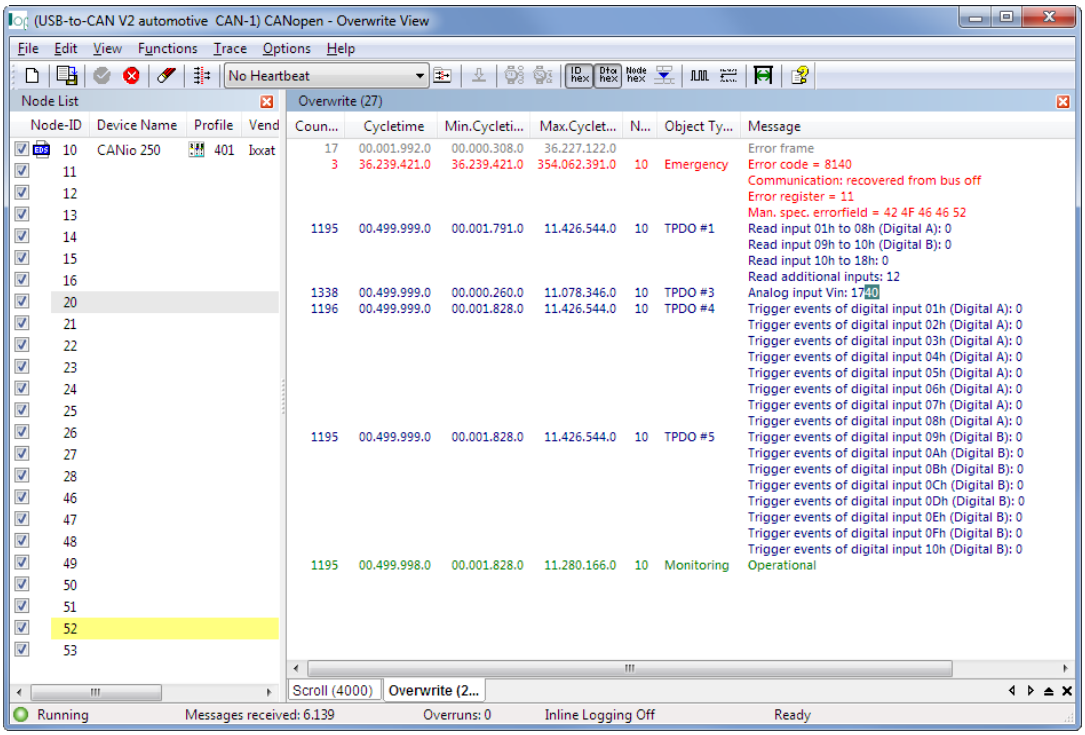


Figure 3.1: Overwrite View

3.1.2 Overwrite View

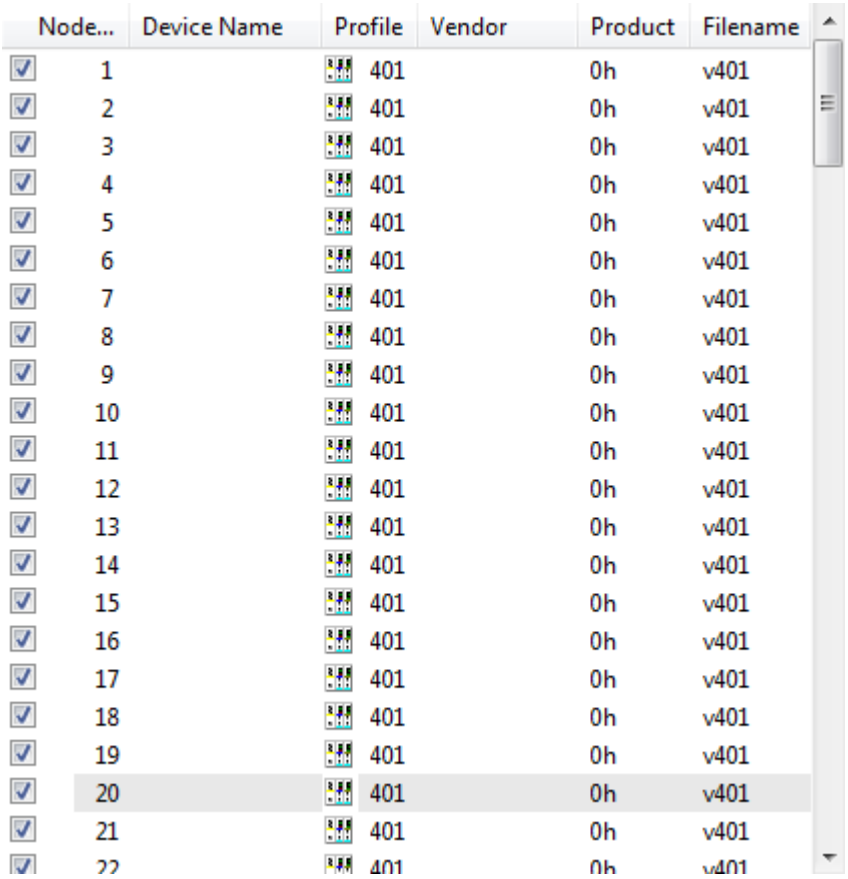
On the Overwrite tab (Fig. 3.1), the messages are listed by their CAN-identifier in ascending order. The last received information of a message is always displayed. A change detection working character by character highlights changed data in color. The change detection compares to the most recently received message contents. Additional columns show reception statistics. The following information is listed in detail:

Column	Meaning
Counter	Total number of received messages of this identifier.
Cycletime	Time elapsed since the last reception of this message; by right-clicking on the column heading, the display of hours and minutes can be switched on or off.
Min.Cycletime	Minimum cycletime occurred, i.e. measured so far; by right-clicking on the column heading, the display of hours and minutes can be switched on or off.
Max.Cycletime	Maximum cycletime occurred, i.e. measured so far; by right-clicking on the column heading, the display of hours and minutes can be switched on or off.
Time (rel/abs)	Time stamp of the reception, optionally absolute in UTC time format or relative to the previously received message; the display of hours can be switched on and off by right-clicking on the column heading. This column is visible only if layer-2 messages display is enabled.
ID (hex/dec)	CAN-identifier of received message. This column is visible only if layer-2 messages display is enabled.

Column	Meaning
Data (hex/dec)	Received CAN data bytes readout. This column is visible only if layer-2 messages display is enabled.
Node-ID (dec/hex)	CANopen Node-ID in decimal or hexadecimal notation, depending on whether View menu item Node-ID hex is checked or not.
Device Name	Name of the device according to its device description file.
Object Type	Type of CANopen communication object. The following object types are distinguished: Server-SDO (SSDO), Client-SDO (CSDO), Transmit-PDO (TPDO), Receive-PDO (RPDO), Multiplex-RPDO (M-RPDO), Multiplex-TPDO (M-TPDO), Network management (NMT), Emergency object (Emergency), Node error control (Monitoring), time stamp object (Timestamp), Sync Object (Sync), Layer setting services (LSS), Flying master message (FLY MASTER), Transmit safety relevant data object (TSRDO), Receive safety relevant data object (RSRDO). The relative consecutive number of the object is appended where applicable.
Message	Interpreted CANopen message. This multi-line column finally shows the plaintext CANopen message.

3.2 Node list

To the left of the messages view there is the node list (Fig. 3.2). It is essential for CANopen-Module usage, not only for its comprehensive context menu (Fig. 3.3), which includes commands not present anywhere else in the application, but also - as the node list represents the CANopen network - because it shows a tabular overview of the names, Node-IDs, profiles and identities of all 127 possible CANopen nodes.
In addition, an individual background color can be assigned to node-related messages. The selected background color will also be applied to the Node Properties dialog (Abb. 3.6).
Moreover, the node list allows for node-centered message filtering. By clicking on the checkbox, the display of all node individual messages can be enabled or disabled.
Node list supports multiple selection. Using key sequence `Ctrl-A`, all nodes can be selected - or, by clicking single CANopen nodes while holding the `Ctrl`-key, a group of particular CANopen nodes can be selected. The subsequent context menu command, e.g. `Color Reset` or `Change Profile...`, then applies to all currently selected nodes.



Node...	Device Name	Profile	Vendor	Product	Filename
1		401		0h	v401
2		401		0h	v401
3		401		0h	v401
4		401		0h	v401
5		401		0h	v401
6		401		0h	v401
7		401		0h	v401
8		401		0h	v401
9		401		0h	v401
10		401		0h	v401
11		401		0h	v401
12		401		0h	v401
13		401		0h	v401
14		401		0h	v401
15		401		0h	v401
16		401		0h	v401
17		401		0h	v401
18		401		0h	v401
19		401		0h	v401
20		401		0h	v401
21		401		0h	v401
22		401		0h	v401

Figure 3.2: Node list (Showing all use of standardized CiA device profile 401)

3.2.1 Description of columns

Column	Meaning
-Check-	Enable node individual messages display.
Node-ID	CANopen Node-ID in decimal or hexadecimal notation, depending on whether View menu item Node-ID hex is checked or not.
Device Name	Name of the device according to its object dictionary entry [1008].
Profile	Number of the standardized CiA device profile which the node complies with. Using context menu command `Change Profile...` it can be switched anytime. An empty column means that CANopen base profile CiA-301 [1] A.2 is active.
Vendor	Manufacturer of the device according to OD-entry [1018.1] Provided that the Vendor ID is registered with CiA, the name will be shown accordingly, otherwise just the number.
Product	Manufacturer-specific (hexadecimal) product code according to OD-entry [1018.2]
Filename	Name of the underlying device description file or of the default profile description.

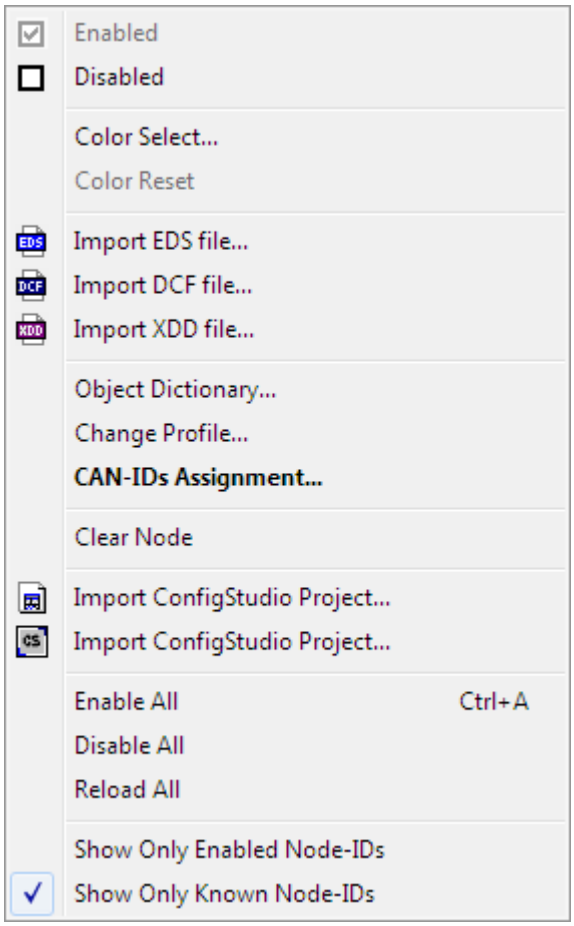


Figure 3.3: Context menu of node list

3.2.2 Context menu of node list

Menu item	Command
Enabled	Enables node-related messages display.
Disabled	Disables node-related messages display.
Color Select...	Selects node background color. Opens up the Colors dialog that already contains 16 custom pastel colors, but also allows to define own colors. The selected colors are used throughout the CANopen-Module to accentuate all the node-specific messages resp settings.
Color Reset	Clears the node background color.
Import EDS file...	With the Open File dialog a custom EDS file can be assigned to a node whose Object Dictionary entries and CAN-identifiers are read from it.
Import DCF file...	With the Open File dialog a custom DCF file can be assigned to a node whose Object Dictionary entries and CAN-identifiers are read from it.
Import XDD file...	With the Open file dialog a custom XDD file can be assigned to a node whose Object Dictionary entries and CAN-identifiers are read from it.
Object Dictionary...	Opens the Object Dictionary window.
Change Profile...	Opens the Node Properties dialog to choose the standardized CiA device profile for the node. Like the EDS/DCF/XDD-Import this has an instant effect on the Object Dictionary entries and potentially the CAN-identifiers.
CAN-IDs Assignment...	Opens the Node Properties dialog, to change and edit the single communication objects of a node.
Clear Node	Reset the node configuration to CANopen base profile after confirmation.
Import ConfigStudio Project...	With the Open File dialog an IXXAT CANopen Configuration Studio Project can be selected for import. This project file contains binary device descriptions of all network nodes. After the project import, command Show only Known Node-IDs will be applied, thereby only the nodes from the project are shown in the node list. In addition, the imported nodes are Enabled in the node list and all others Disabled. This causes UNDEFINED to be shown for all CANIDs not defined in the imported project so the user can easily see inconsistencies.
Enable All	Enables the node-related messages display for all nodes at once.
Disable All	Disables the node-related messages display for all nodes at once.
Reload All	Re-Import all currently assigned device description files.
Show only Enabled Node-IDs	Switch to narrow down the entries that are visible in the node list. If checked, there are only those nodes visible which are Enabled for individual messages display.
Show only Known Node-IDs	Switch to narrow down the entries that are visible in the node list. If checked, only those nodes having an individual device description file are visible.

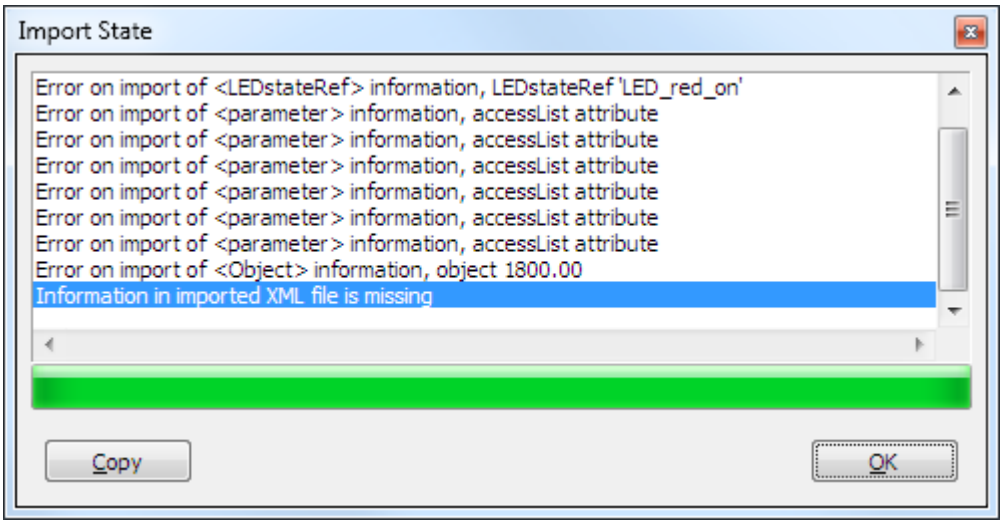


Figure 3.4: Status window when importing device description files

3.2.3 Import of device description files

When importing a device description file (using the menu commands `Import EDS file...`, `Import DCF file...`, `Import XDD file...` and also `Import ConfigStudio Project...`) there is a binary conversion into a CANopen-Module-specific format. The data imported will be stored to the canAnalyser3 configuration file, so that the original device description file(s) are not needed for message interpretation.
With the import, node-specific interpretation rules are gathered from the different object description files and formats that provide for full real-time decoding of all the CANopen communication objects transmitted and received by an individual node.
An explicit re-import of device descriptions from the above mentioned files and formats is made possible with the menu command `Reload All`.
Even after an import, that overwrites a node configuration after confirmation, any node can be switched back to a standardized CiA device profile (`Change Profile...`) or to the CANopen base profile (`Clear Node`).

EDS (Electronic Data Sheet) files [5] A.2 contain all the object dictionary entries of a CANopen node. They are provided by the device manufacturer/vendor.
DCF (Device Configuration File) files contain the same data as the EDS files, plus actual object values such as configured PDOs, etc.
XDD (Extended Device Description) files [6] A.2 are electronic data sheets in XML format. It is the EDS replacement.
FBR is the file format of IXXAT CANopen Configuration Studio. Contrary to the device description formats listed above, it is a network description rather than a device description. Therefore, **not only a single node but all network participants will be replaced when this file format is opened/imported.**
COPPRJ is the file format of the next generation IXXAT CANopen Configuration Studio. Contrary to the device description formats listed above, it is a network description rather than a device description. Therefore, **not only a single node but all network participants will be replaced when this file format is opened/imported.**

During the import process of (device) description files the above progress form is shown (Fig. 3.4). In the event of critical errors, the dialog remains open to allow the user to read the error message and checking the file to import where appropriate.

By pressing the `Pause`-key during import process, the window also remains open regardless of the import success.
Button `Copy` allows for copying the listed messages to the clipboard.

When importing device description files, the node number of the following object dictionary entries is assigned according to the rules of the so-called Predefined Connection Set:

Index	Subindex	Designation	CAN-identifier
1014h	0	Emergency	80h
1200h	1	1. Server-SDO (rx)	600h
1200h	2	1. Server-SDO (tx)	580h
1400h	1	1. Receive-PDO	200h
1401h	1	2. Receive-PDO	300h
1402h	1	3. Receive-PDO	400h
1403h	1	4. Receive-PDO	500h
1800h	1	1. Transmit-PDO	180h
1801h	1	2. Transmit-PDO	280h
1802h	1	3. Transmit-PDO	380h
1803h	1	4. Transmit-PDO	480h

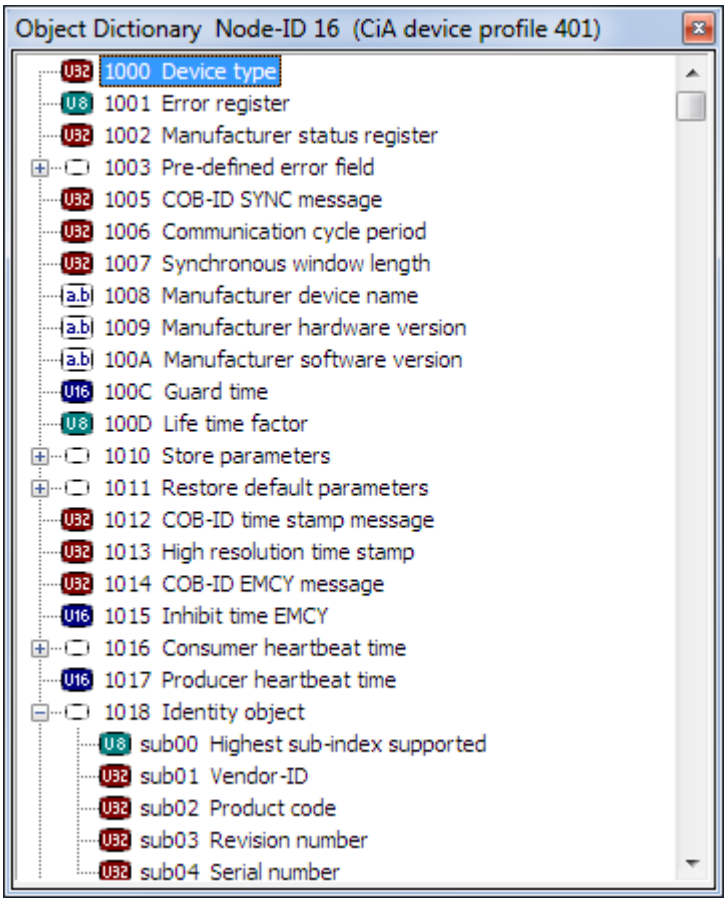


Figure 3.5: Readout of an Object Dictionary

3.2.4 Object Dictionary window

Using the command `Object Dictionary...` of the node list context menu the object dictionary of the selected node can be examined (Fig. 3.5).
For standardized device profiles (e.g. CiA-401) the complete object dictionary (including CANopen base profile entries) is presented. For nodes that are assigned an individual device description file, only the objects from that device description are shown.

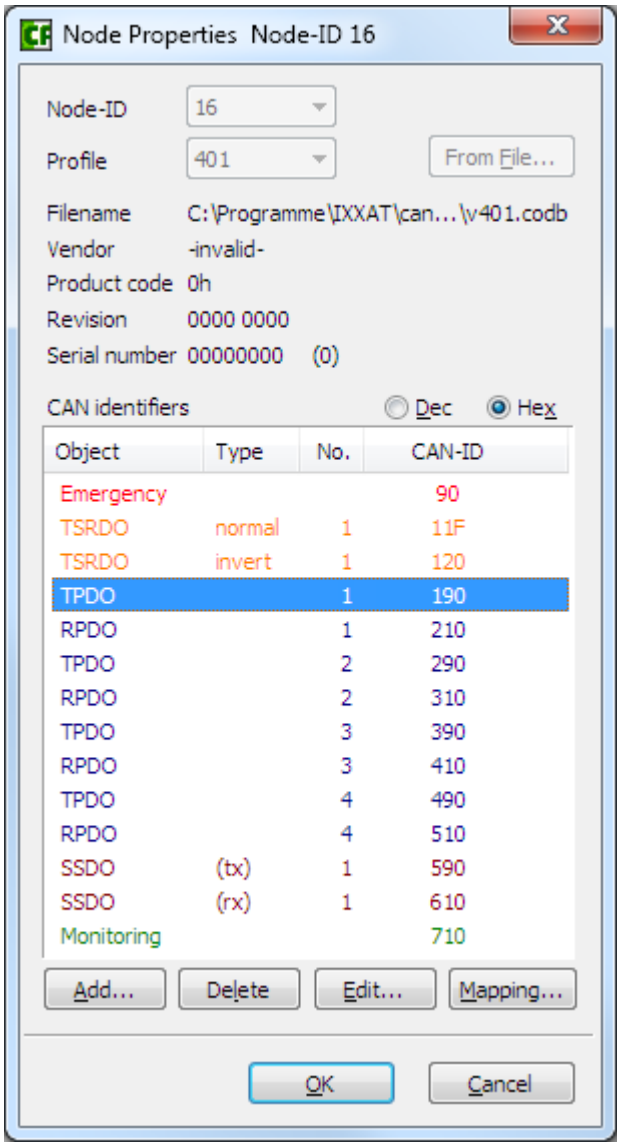


Figure 3.6: Node Properties dialog

3.2.5 Node Properties dialog

In Node Properties dialog (Fig. 3.6) all communication objects in use by a node are listed. They are colored according to the application global filter and color settings.
This dialog is also used to switch over the appropriate standardized device profile to be used and it even supports the import of device description files.

When the device profile is changed or a device description file is imported, the list of used CAN identifiers is recalculated according to Predefined Connection Set rules or DCF contents.

Control	Meaning
Node-ID	Current Node-ID of the device.
Profile	Selection of a provided standardized device profile for the node. An empty entry means that CANopen base profile CiA-301 [1] A.2 is active.
From File...	Shows the Open File dialog for assignment of a particular device description file (EDS, DCF, XDD).
Filename	Full name and path of the underlying device description file or of the default profile description.
Vendor	Manufacturer of the device according to OD-entry [1018.1] Provided that the Vendor ID is registered with CiA, the name will be shown accordingly, otherwise just the number.
Product code	Manufacturer specific product code according to OD-entry [1018.2]
Revision	Manufacturer specific revision number according to OD-entry [1018.3]
Serial number	Serial number of the device according to OD-entry [1018.4]
CAN identifiers	List of all used communication objects (column Object / Type), relative consecutive number of the object (column No.) and the CAN identifier it takes. (column ID).
Dec/Hex	Switch between hexadecimal and decimal CAN identifier notation.
Add...	Create a new communication object.
Delete	Remove the selected communication object.
Edit...	Change CAN identifier and other attributes of the selected communication object.
Mapping...	If the PDO is valid and not empty, the PDO Mapping dialog is opened, which shows the mapped object addresses, names, and lengths (Fig. 3.10).
OK	Accept changes
Cancel	Discard inputs

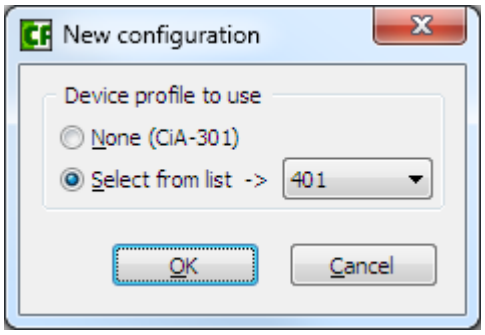


Figure 3.7: Create new configuration

3.3 Analysis configuration

The CANopen-Module provides flexible configuration possibilities in order to adapt it optimally to an existing CANopen network. Analysis and interpretation rests upon a network modeling called analysis configuration which is stored to the canAnalyser3s configuration file.
The network modeling is to be fit to the physical network as close as possible. For this purpose each node is assigned a standardized device profile initially. For refinement, all CANopen communication objects of each node can be edited and changed freely.

Analysis configuration is managed via node list. In particular these are the context menu commands `Change profile...` and `CAN-IDs Assignment...` that directly lead to the node-specific communication objects which are handled in a separate dialog.
The individual configuration possibilities are described in more detail in the following sections.

3.3.1 New configuration

When the CANopen-Module is started for the first time, an initial configuration is active in which each node works according to CANopen base profile CiA-301. This means that it has all the following predefined communication objects according to the Predefined Connection Set:

Object type	Number	Labeling
Emergency object		Emergency
Transmit-PDO	4	TPDO #1 to #4
Receive-PDO	4	RPDO #1 to #4
Server-SDO	1	SSDO #1
Error control		Monitoring

The analysis configuration can be recreated during runtime. To do this, call the menu command `File | New Configuration...`, which will open up the New Configuration dialog (Fig. 3.7):

Control	Meaning
None (CiA-301)	Creates a new configuration consisting of CANopen base profile CiA-301 for all nodes. This is the initial state of CANopen-Module.
Select from list ->	Creates a new configuration consisting of the selected standardized device profile for all nodes.
-Device profile no-	Selection of a provided standardized CiA device profile.
OK	Accept changes
Cancel	Discard input

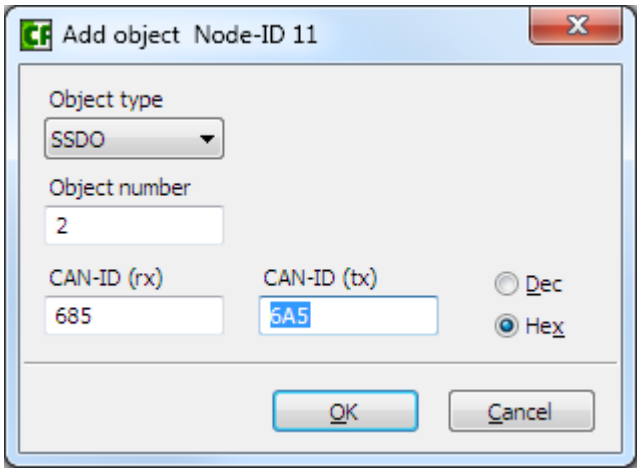


Figure 3.8: Add CANopen communication object (e.g. SDO)

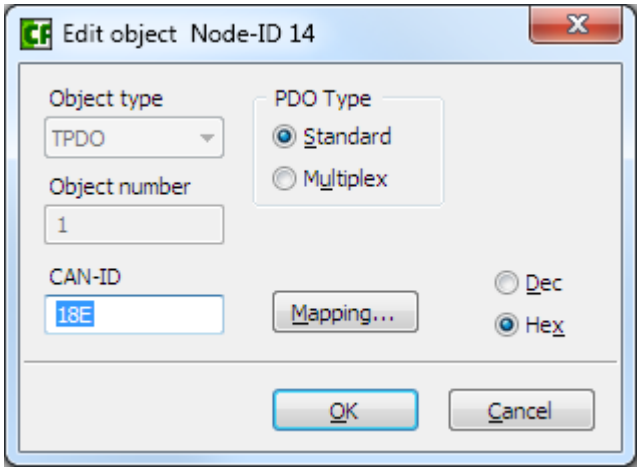


Figure 3.9: Changing object settings (e.g. PDO)

Note: The new configuration will overwrite all existing nodes. It is not possible to switch back once the `OK` button has been pressed!

3.3.2 Adding, deleting or changing communication objects

In the Node Properties dialog (Fig. 3.6), which is opened via the Context menu of node list command `CAN-IDs Assignment...`, communication objects can be added, deleted or edited using the corresponding buttons below the objects list.
A prerequisite for changing is that no specific device description file (DCF or FBR/COPPRJ) is assigned. Even with the other two possible file formats (EDS and XDD) it is not possible to add or remove communication objects, because this simply would not be congruent with the actual device outfit as documented in its description file. However, existing objects can be freely edited.

For any change, the Object Settings dialog (Fig. 3.8) is used. Depending on the desired action, parts of the dialog are invisible or deactivated. (Fig. 3.9).
A description of all the existing controls is given in the following.

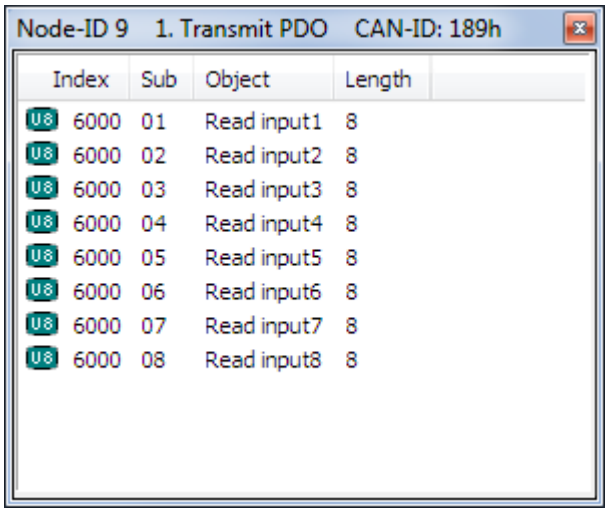


Figure 3.10: PDO data: Mapping

Control	Meaning
Object type	Used to select the object type (RPDO, TPDO, SSDO, etc.)
Object number	Relative consecutive number of the object (if applicable).
CAN-ID / CAN-ID (rx) / CAN-ID (normal)	CAN-identifier of the object. For SDOs the CAN-identifier of the SDO request, for SRDOs the CAN-identifier of the first (normal) transmission.
CAN-ID (tx) / CAN-ID (invert)	For SDOs the CAN-identifier of the SDO response, for SRDOs the CAN-identifier of the second (invert) transmission.
PDO Type Standard	Regular PDO
PDO Type Multiplex	Multiplex-PDO. No need to differentiate Source Address Mode (SAM) and Destination Address Mode (DAM) here, because the identification takes place upon actual reception.
Mapping...	If the PDO is valid, the PDO Mapping dialog is opened, which shows the mapped object addresses, names, and lengths (Fig. 3.10).
Dec/Hex	Switch between hexadecimal and decimal CAN identifier notation.
OK	Accept changes
Cancel	Discard input

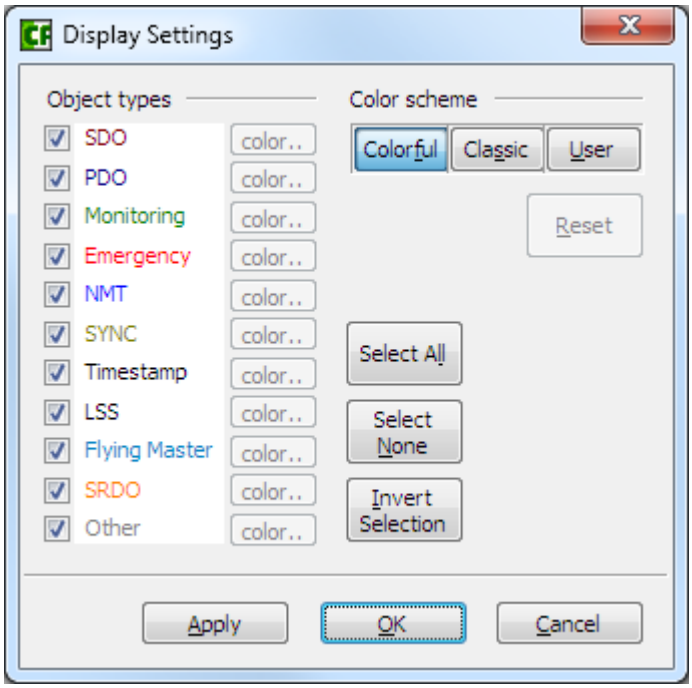


Figure 3.11: Display Settings dialog of CANopen-Module

3.4 Message filtering concept

CANopen-Module has two different superimposed display filters and an optional reception filter. Since these three different possibilities often provoke confusion in daily use and with newcomers, they will be discussed and explained in this section.

3.4.1 Display filtering according to object type

Menu command `Options | Display Settings...`, the corresponding toolbar button and the keystroke `F7` open up the non-modal Display Settings dialog (Fig. 3.11). It is used to select the object types to be shown in the message display, i.e. the views. In addition, the display color of each communication object can be set.
The buttons `Select All`, `Select None` and `Invert Selection` allow for fast (un-)checking i.e. selecting of all the available object types.

Meaning of the items in `Object types:`

SDO	Show SDOs.
PDO	Show (Multiplex-)PDOs. Please note that these can be displayed in interpreted or uninterpreted format, depending on the corresponding Options menu resp. toolbar switch state.
Monitoring	Show error control messages, that is: Bootup, Guarding and Heartbeat.
Emergency	Show emergency objects.
NMT	Show network management objects (messages sent with CAN identifier 0h).
SYNC	Show the CANopen sync object with CAN identifier 80h.
Timestamp	Show timestamp objects with CAN identifier 100h according to CANopen specification as days / milliseconds since 1.1.1984, and also as interpreted local date. Local and user settings are taken into account (e.g. 1.12.2010 in Europe, 12/01/2010 in the USA).
LSS	Show Layer Setting Services and Protocol objects acc.to CiA-305.
Flying Master	Show Flying Master objects acc.to CiA-302.
SRDO	Show safety relevant data messages, that is: SRDOs acc.to. CiA-304.
Other	Show all other messages that could not be interpreted. They are displayed as UNDEFINED in the views.

`Color schemes`

There are three linked switching buttons `Colorful`, `Classic` and `User` for quick switching over to a different color scheme.
The first color scheme, named `Colorful`, is the CANopen-Module preset. It defines different, discrete colors for each object type wherever possible.
The second color scheme, `Classic`, is equivalent to the coloring of the initial version of CANopen-Module which did not differentiate between SDOs and PDOs.
The third color scheme, `User`, allows for individual choice of a display color for each object type by clicking on the small `color...` button to the right of the control.
The user settings can be `Reset` to the `Colorful` preset by clicking on the corresponding button.

3.4.2 Display filtering according to Node-ID

In the node list, the display of node-related messages can be enabled or disabled for each node individually using the checkbox of the first column. This has been explained earlier.

3.4.3 Reception filtering

CANopen-Module utilizes *downstream* display filtering by default. This means that **all** CAN messages are always being received and the interpreted CANopen messages are only filtered when displayed on the basis of the filter options described in the previous section. The main advantage of this (non-destructive) filtering is that all interpreted messages are always available and users only see the currently relevant messages during the network analysis. It allows for real-time re-arrangement of the visible objects according to the currently selected display filters.
Unfortunately, this approach requires rather a large amount of memory. Since memory is always limited, it could happen that the internal message buffer, which is organized as a ring buffer, is overwritten quite quickly, particularly with high bus loads. Therefore, seldom sent messages might be disregarded because of their short lifetime in the message buffer. Incidentally, the capacity of the message

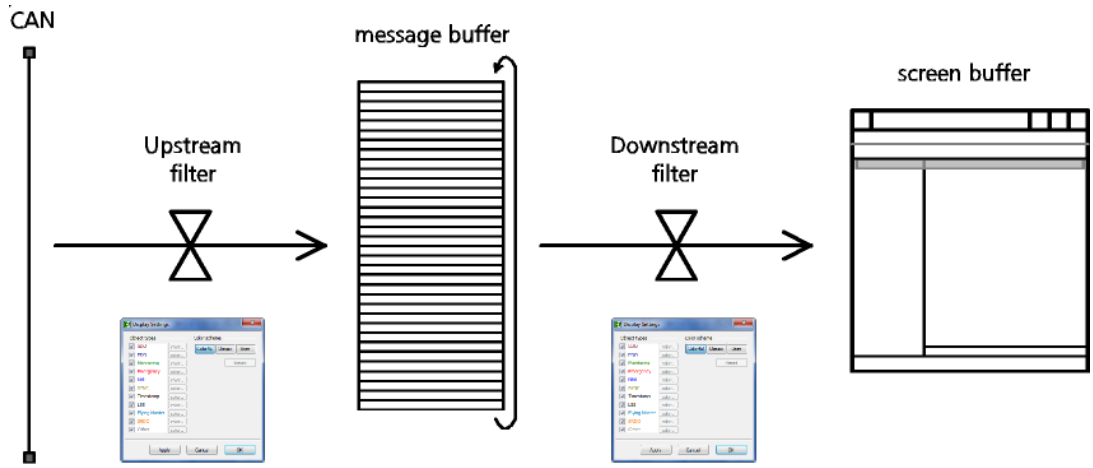


Figure 3.12: Idea of reception filtering

buffer is configured according to customs in the canAnalyser3s Control Panel under `Configuration | Preferences... | Modules | Size of scroll view in Receive-M`
CANopen-Module takes four times the configured number of messages for its internal line-oriented message buffer.

If a longer lasting analysis session is carried out, or if very rarely occurring CANopen messages are definitely to be found, the CANopen-Module can be switched to *upstream* filtering. This is done with the menu command `Options | Upstream Filtering`. In this operating mode, the CAN messages are immediately rejected at the time of reception according to their object type. Therefore only the filtered CANopen communication objects get to the message buffer, thus filling it more slowly. However, this also means that display filtering is no longer able to show hitherto "hidden" messages, because they are simply not present in the internal message buffer.
The Display Settings dialog no longer controls the display filtering in this mode, but the reception filtering instead (Fig. 3.12).
When using upstream filtering, the Node-ID is not taken into account, only the object type. Therefore, display filtering according to Node-ID will work as usual.

3.5 Menu reference

3.5.1 File menu

Menu item	Meaning
New Configuration...	Creates a new configuration.
Import Options...	Imports previously exported settings such as display filters, loaded device descriptions, window layout, etc. from a file
Export Options...	Exports the current settings such as display filters, loaded device descriptions, window layout, etc. to a file
Export Messages...	Writes the exact contents of the view to an ASCII file. Effectively, this is a screenshot of the current view incorporating all display filters.
Exit	Exits CANopen-Module

3.5.2 Edit menu

Menu item	Meaning
Copy CSV	Copies marked lines CSV formatted to clipboard
Toggle Marker *	Sets or Removes a Marker for selected message
Previous Marker *	Jumps to previous Marker (no wraparound)
Next Marker *	Jumps to next Marker (no wraparound)
Set/Release Time Reference *	Sets Timestamp Zero for selected message / Releases previously set Timestamp Zero
Jump to Time Reference *	Jumps to previously set Timestamp Zero message

* Only available in Scroll View

3.5.3 View menu

Menu item	Meaning
Time relative	Shows a message's time stamp relative to the previously received message.
ID hex	Shows the CAN-identifier in the views in hexadecimal notation (only applicable if layer-2 messages display is enabled.
ID representation	Representation options of the layer-2 CAN-identifier column: Display with leading zero or without. You can also right click the respective column header to toggle the ID representation.
Data hex	Shows the data of layer-2 messages in hexadecimal notation (only applicable if layer-2 messages display is enabled.
Data representation	Representation options of the layer-2 Data column: Display with leading zero or with leading space or neither. You can also right click the respective column header to toggle the Data representation.
Node-ID hex	Shows the CANopen Node-ID in hexadecimal notation, otherwise in decimal notation.
Draw Guides	Draws additional horizontal guides between the CANopen messages in grey
Show recent Frames	Always shows the most recent messages, scrolls down to bottom.
Show also raw CAN data	Displays additional columns in the message views named ID and Data that show the corresponding uninterpreted CAN layer-2 messages.
Node List	Shows resp hides the node list.
Scroll View	Switches to resp hides the scroll view.
Overwrite View	Switches to resp hides the overwrite view.
Toolbar	Shows the toolbar.
Status Bar	Shows the status bar.

3.5.4 Functions menu

Menu item	Meaning
Start	Starts message reception.
Stop	Stops message reception.
Available Filters...	Adjust application wide available message resepction filters
Select Filter	Selects a message reception filter. It is an upstream filter which is applied prior to all other CANopen-Module Reception filtering.
Clear All	Deletes all existing messages and resets the receive counter.
Reset Interpreter	Reset the internal protocol interpreter engine and its state machine. This way discrepancies resp. mismatches e.g. in the SDO protocol interpretation, between the protocol interpreter state of the CANopen-Module and the current bus traffic can be resolved.
Sample Devices	Initiates a node scan. This works as follows. By sending corresponding SDO messages to all 127 network nodes, the available nodes are detected and displayed in the node list. During the scanning process, the node list is displayed in grey (disabled). The detected nodes are Enabled in the node list and all others Disabled. We advise against operating the program until the scan is finished. At the end of the process, the context menu command Show only Enabled Node-IDs is called automatically, so that the node list intentionally only displays the detected nodes.
Autosize Columns	Regulate ideal column widths

Notes:
The scanning process cannot be stopped.
Because of the SDO requests transmitted during the node scan, the CANopen-Module behaves as an active participant rather than a silent observer with regard to the CANopen network. This might irritate and even disturb an existing active CANopen Master.
During the scanning process, the complete analysis configuration of the CANopen-Module is overwritten with the default settings of the detected standardized CiA device profiles. It is therefore best to perform a scan immediately after program start, with an empty analysis configuration. To sum up, the node scan should only be used sparingly and with awareness of the risk involved.

3.5.5 Trace menu

Simultaneously to the continuous display on the screen, the interpreted messages can be written in real-time and unfiltered to a CSV file. This is referred to as `Inline logging` and thus also displayed in the status bar of the CANopen-Module.
Please note that the corresponding log file grows rapidly, and that the logging itself puts a strain on the computer. Especially when using anti-virus scanners with real-time scanning, serious restrictions in the reaction time may occur. If this is the case, please use the Trace module of the canAnalyser3. A single log file is limited to 1 GiB size. Above this a self-acting segmentation takes place, by creating a new log file with contiguous segment numbering name scheme.
Before using inline logging, the file name of the log file must be entered using the menu command `Trace | File...`. Then logging can be enabled via the menu command `Trace | Active` or the corresponding toolbar button or hotkey `F2`. It is disabled again with the same command.

Menu item	Meaning
File...	Entry of a file name in .CSV format. An existing file will be overwritten.
Active	Enable/disable inline logging.

3.5.6 Options menu

Menu item	Meaning
Upstream Filtering	Switches upstream filtering. See also sect. 3.4.3
Change Detection Color...	Opens the Colors dialog to select the color with which changed data are highlighted.
Font...	Opens a dialog to select the font type in which the data are displayed in the current view.
Display Settings...	Opens the Display Settings dialog. See also sect. 3.4.1
Show Raw PDO Bytes	Always displays PDO data uninterpreted. This is only relevant for nodes which have a device description file assigned to them.
Buffered SDO	Displays segmented SDO messages in the form of an overall access at the end of SDO transmission.

3.5.7 Help menu

Menu item	Meaning
Help Topics	Opens the online help
About...	Displays the version information of the CANopen-Module.

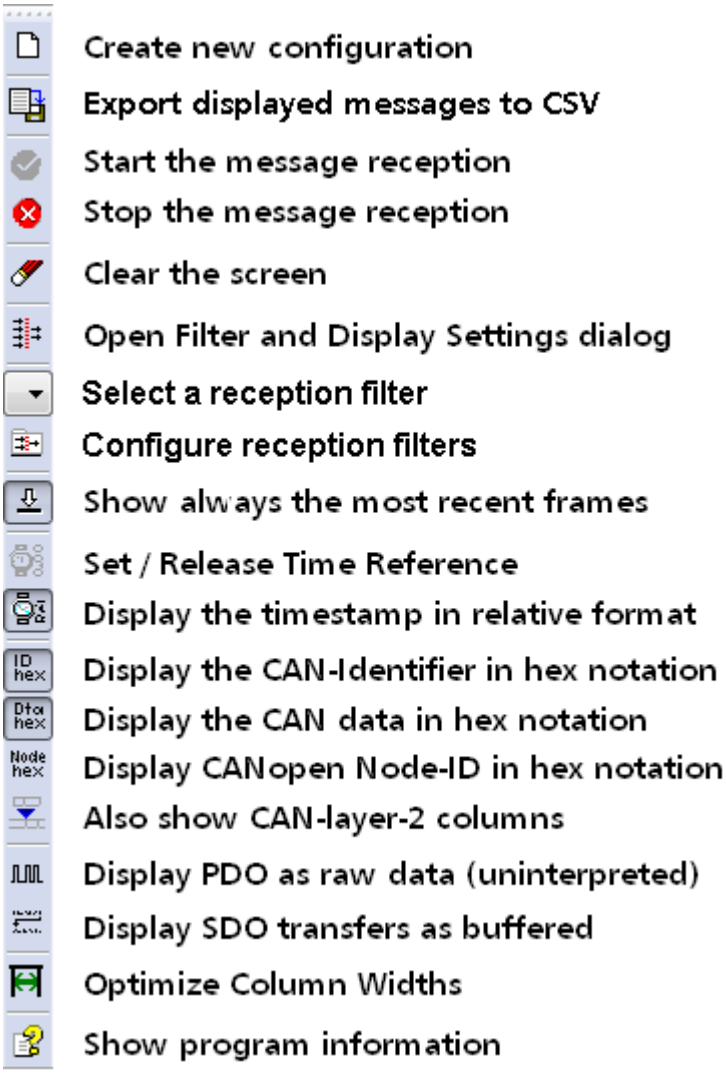


Figure 3.13: Toolbar of CANopen-Module

3.6 Toolbar

The main functions of the CANopen-Module can also be called via the toolbar (Fig. 3.13).

3.7 Status bar

The status bar contains an LED icon that displays the status of the Control Panel or of the CANopen-Module:

LED color	Meaning
Green	Control Panel and CANopen-Module are started
Flashing red	Control Panel is stopped
Red	CANopen-Module is stopped

3.8 Hotkeys

TAB	Switch between node list and message views
Ctrl+TAB	Switch between Scroll view and Overwrite view
F1	Online-Help
F2	Go to Next Marker in Scroll View
Shift+F2	Go to Previous Marker in Scroll View
Ctrl+F2	Toggle Marker in Scroll View
F5	Start message reception
Shift+F5	Stop message reception
F6	Enable/disable inline logging
F7	Open Display Settings dialog
F8	Clear all Views
Ctrl+F8	Reset all protocol Interpreters
F11	Show/hide node list
Ctrl+A	Enable all nodes in node list at once
Ctrl+C	Copy marked lines CSV formatted to clipboard
Ctrl+E	Export screen message buffer to file
Ctrl+I	Configure reception Filters
Ctrl+N	Creates a new configuration
Ctrl+O	Load all module settings from file
Ctrl+S	Save all module settings to file
Ctrl+W	Close the application window
PageDown	Scroll one page ahead in current View
PageUp	Scroll one page backward in current View
Ctrl+PageDown	Scroll 1000 messages ahead in current View
Ctrl+PageUp	Scroll 1000 messages backward in current View
Ctrl+0	Jump to Time Reference message
Ctrl+1..9	Jump to 10%..90% of current View

Appendix A

Registers

A.1 Definitions, Acronyms, Abbreviations with CANopen

Application object The device functionality provided by a device is described by application objects. Application objects can be readable or writeable device parameters, data or functions. The application object can be accessed via an unambiguous address in the object dictionary.

CANopen object The functionality of a CANopen device visible via the bus is described by CANopen objects. CANopen objects can be data, parameters or functions of a device. The object can be identified in the object dictionary via a 16-bit index and an 8-bit subindex.

CiA CAN in Automation e.V. Organization of CAN bus manufacturers and users

CiA-301 CANopen communication profile [1] A.2[7] A.2. Mandatory specification of the communication model and object dictionary structure for all CANopen devices. Starting with Version 4.0, CMS and NMT have been included, DBT has been discarded, and LMT turned to LSS.

CiA-302 General specification for programmable CANopen devices [2] A.2. Amongst other things, contains the predefinitions for CiA-405.

CiA-401 CANopen device profile for generic I/O modules .

CiA-402 CANopen device profile for drives.

CiA-405 CANopen device profile for IEC-1131 programmable devices.

CiA-406 CANopen device profile for encoders.

Client-SDO A client SDO refers to the initiator of an SDO transfer. This has access to the object dictionary entries of an "SDO server".

COB: Communication object A COB is a message which is transferred in the CAN network. Data are transported with a COB.

COB-ID / COBID The COB-ID makes the communication connection between a transmit COB and receive COBs and at the same time defines the message priority. The highest priority ID 0 is reserved for network management services.

Communication cycle period Communication cycle period defines the time interval between consecutive sync objects.

Communication parameters The attributes of a PDO are described in its communication parameters. These attributes include transmission type, inhibit time and of course COB-ID.

Device profiles The device functionality is described via standardized functions in the area of the standardized device profile, for manufacturer-specific device functions in the area of the manufacturer-specific device profile.

Dummy / Dummy entry Dummy mapping is needed to fill gaps in receive-PDO mapping.

DCF: Device configuration file The DCF file describes a real, existing, configured device in a network. The structure of the DCF file corresponds to that of the EDS file plus the project-specific configuration of this device. Amongst other things, the configuration contains the baud rate, PDO mapping, project-specific device name, set Node-ID and the parameterization of the application objects.

EDS: Electronic data sheet The EDS describes the device functionality. This file must be provided by the Vendor/ Manufacturer. It contains general and special device data, some statistical information about the file itself, and most of all the detailed complete Object Dictionary description.

Emergency object By a high-priority emergency object a device signals the occurrence of a fatal internal device error or the reset of one or all internal device errors. Support of the device error message is optional. The emergency error code specifies the error type in accordance with CiA-301.

Guard time The NMT master cyclically transmits a request to the NMT slave to transmit its current node status. This request must be answered within the node lifetime. The node lifetime of a node results from the lifetime factor multiplied with the guard time of the node. The NMT slave does not carry out monitoring of the NMT master if the guard time is parameterized with 0. However, the node guarding protocol is answered. The reactions to infringements of node guarding are described in the CANopen specification 301.

Granularity The maximum possible number of objects that can be entered in a PDO is defined by the granularity (= object length in bits) of the application objects. The maximum data field size of a PDO is 8 data bytes. So with a granularity of 8, at most 8 byte application objects can be mapped into a PDO. With a granularity of 1, even 64 Boolean application objects are supported.

Inhibit time A process data object (PDO) may only be re-transmitted after this time has expired.

NMT: Network management Service element of the application layer in the CAN reference module, which consists of configuration, initialization and error control of the network as well as network-wide process synchronization. The network management has a master/slave structure.

Node guarding Cyclic monitoring of a node.

Node-ID An individual device is unambiguously defined in the network by its node number (between 1 and 127). This number is used by the predefined connection set for the predefined identifier allocation. Node-ID 0 is reserved for NMT services.

OD, Object dictionary The object dictionary is a data structure via which all objects of a CANopen device can be addressed. The object dictionary is divided into an area with general information on the device, such as manufacturer name etc., a range which contains the communication parameters and a range which describes the specific device functionality. Via the entries (objects) of the object dictionary, the application objects of a device, such as input and output signals, device parameters, device services or network variables are made available in standardized form via the network. The object dictionary makes up the interface between the network and the application process.

OD entry See CANopen object

PDO: Process data object PDOs represent the actual means of transport for the transfer of process data. A PDO is transmitted by a "producer" and can be received by one or more "consumers". The process data transmitted by a producer in a PDO can consist of a maximum of 8 bytes. A PDO is transferred unacknowledged and requires an identifier clearly assigned to the PDO. The meaning of the transferred data is defined by the identifier it uses and by the PDO mapping assigned to a PDO. The priority and operating mode of the PDO is defined with the communication-specific parameters. For the management of PDOs, both PDO producers and PDO consumers require congruent data structures. The data required by a PDO producer are managed in the form of so-called TxPDO OD entries; the data to be received by a PDO consumer in the form of so-called Rx-PDO OD entries.

PDO linking PDO linking represents the communication connection between transmit-PDO and corresponding receive-PDOs. The communication connection emerges by the allocation of the same PDO-identifier to transmit and receive PDO(s).

PDO mapping Allocation of the data field (max. 8 bytes) of a PDO with application objects is defined by PDO mapping. It can be static (i.e. constant) or dynamic (i.e. changeable).

Predefined connection set Predefined connection set means predefined identifier assignment based on the Node-ID and on the function code. For the following communication objects, the predefined connection set regulates the COB-ID: Node guarding/heartbeat, emergency object, sync message, timestamp, server-SDO 1, RPDO 1 to 4 and TPDO 1 to 4.

RPDO Receive PDO, see also PDO

Scan timeout Time frame within which a device must answer to the network after being called in order to be recognised as present.

SDO: Service data object An SDO is a CANopen communication object used for configuration and parameterisation of CANopen devices, resp for transmission of long data. Device object dictionary entries can be accessed read or write by SDOs. The desired OD entry is addressed by index and subindex. An SDO forms a direct 1:1 communication channel between any two nodes.

SDO timeout An SDO request must be answered within the timeout time. The time is given in milliseconds.

Server SDO Each device must support at least one server SDO and thus enable access to the entries in its object dictionary. The specification of a SDO server object requires one CAN identifier defined for each transfer direction, because it is an acknowledged service. Optionally the associated client or server node (provided that dynamic creation of SDO connections is supported) can be given. The CAN identifiers of the first Server-SDO are dependent on the Node-ID, and they are strictly regulated.

Sync object The sync object is used for synchronized data collection, synchronized command strobing and cyclic transfer of process data. The reception of a SYNC object triggers updating and transmission of synchronous messages. For this, one device (sync producer) transmits the high-priority sync object cyclically. The sync object requires the specification of the communication cycle period parameter and of the synchronous window length parameter for its full description. If a parameter is initialized with 0, it has no effect.

Synchronous window length Window after a sync object for sending the synchronous transmission type PDOs.

Timestamp message Used for re-synchronization of the local timers to ensure higher requirements of synchronization basis for all devices of a system.

Transmission type The operating mode of a PDO is specified in the communication profile of a device via the transmission type parameter. CANopen provides the following transmission types for PDOs: Synchronous: Transmission depends on a SYNC object. either Acyclic: once or cyclic: with each reception or after a number of SYNC objects specifiable via the transmission rate. Asynchronous: Transmission is triggered by a manufacturer-specific event or by an event defined in the device profile. Remote: Transmission occurs only after a request by another subscriber (PDO consumer).

Transmission rate In cyclic-synchronous PDO mode, this value represents the number of synchronization messages that must have been received before re-transmission of the PDO is allowed.

TPDO Transmit PDO. See PDO

A.2 Specifications

[1] CiA-301 CANopen Application Layer and Communication Profile
Version 4.2
07 December 2007

[2] CiA-302 CANopen Additional Application Layer Functions
Part 2: Network management
Version 4.1
02 February 2009

[3] CiA-302 CANopen Additional Application Layer Functions
Part 6: Network redundancy
Version 4.1
02 February 2009

[4] CiA-305 CANopen Layer setting services (LSS) and protocols
Version 2.2
26 August 2008

[5] CiA-306 CANopen Electronic data sheet specification
Version 1.3
01 January 2005

[6] CiA-311 CANopen device description XML schema definition
Version 1.0.2
17 July 2007

[7] EN 50325-4 Industrial communications subsystem based on ISO 11898 (CAN) for controller-device interfaces
Part 4: CANopen

Appendix B

Copyrights

B.1 Copyright

© 2008-2015 HMS Technology Center Ravensburg GmbH

B.2 Additional Copyrights

This software contains material that is © 1994-2000 DUNDAS SOFTWARE LTD., all rights reserved.

This software installs or updates Microsoft OS components (MSXML3 SP5) which are copyrighted by © Microsoft Corp.

This product includes software developed by The Apache Software Foundation (`http://www.apache.org/`). Portions of this software was originally based on the following:

- software copyright (c) 1999, IBM Corporation., `http://www.ibm.com`.

This product includes SQLite (`http://www.sqlite.org/`).

This product includes muParser by Ingo Berg.