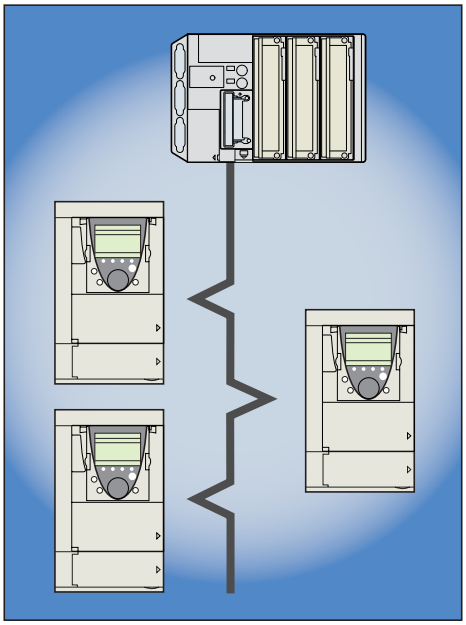
Altivar 61 / 71

CANopen

User's manual

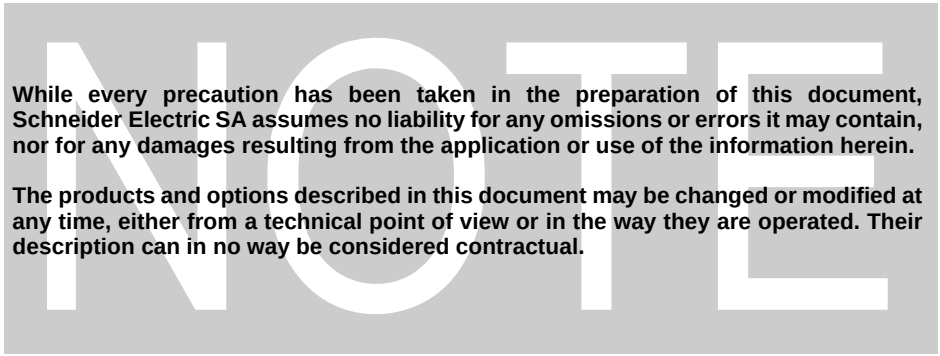
Retain for future use



a brand of Schneider Electric

Telemecanique

Contents



While every precaution has been taken in the preparation of this document, Schneider Electric SA assumes no liability for any omissions or errors it may contain, nor for any damages resulting from the application or use of the information herein.

The products and options described in this document may be changed or modified at any time, either from a technical point of view or in the way they are operated. Their description can in no way be considered contractual.

Read and understand these instructions before performing any procedure with this drive.

⚠ DANGER

HAZARDOUS VOLTAGE

- Read and understand this manual before installing or operating the Altivar drive. Installation, adjustment, repair, and maintenance must be performed by qualified personnel.
- The user is responsible for compliance with all international and national electrical standards in force concerning protective grounding of all equipment.
- Many parts in this variable speed drive, including printed wiring boards, operate at line voltage. DO NOT TOUCH. Use only electrically insulated tools.
- DO NOT touch unshielded components or terminal strip screw connections with voltage present.
- DO NOT short across terminals PA and PC or across the DC bus capacitors.
- Install and close all covers before applying power or starting and stopping the drive.
- Before servicing the variable speed drive:
  - Disconnect all power
  - Place a "DO NOT TURN ON" label on the variable speed drive disconnect
  - Lock the disconnect in the open position
- Disconnect all power including external control power that may be present before servicing the drive. WAIT 15 MINUTES for the DC bus capacitors to discharge. Then follow the DC bus voltage measurement procedure given in the Installation Manual to verify that the DC voltage is less than 45 Vdc. The drive LEDs are not accurate indicators of the absence of DC bus voltage.

Electric shock will result in death or serious injury

CAUTION

DAMAGED EQUIPMENT

Do not operate or install any drive that appears damaged.
Failure to follow this instruction can result in equipment damage.

Documentation structure

Installation manual

This manual describes:
- How to assemble the drive
- How to connect the drive

Programming manual

This manual describes:
- The functions
- The parameters
- How to use the drive display terminal (integrated display terminal and graphic display terminal)

Communication parameters manual

This manual describes:
- The drive parameters with specific information for use via a bus or communication network
- The operating modes specific to communication (state chart)
- The interaction between communication and local control

Modbus, CANopen, Ethernet, Profibus, INTERBUS, Uni-Telway, FIPIO, Modbus Plus, DeviceNet ... manuals

These manuals describe:
- Connection to the bus or network
- Diagnostics
- Configuration of the communication-specific parameters via the integrated display terminal or the graphic display terminal

They describe the protocol communication services in detail.

Altivar 58/58F migration manual

This manual describes the differences between the Altivar 71 and the Altivar 58/58F.
It explains how to replace an Altivar 58 or 58F, including how to replace drives communicating on a bus or network.

Altivar 38 migration manual

This manual describes the differences between the Altivar 61 and the Altivar 38.
It explains how to replace an Altivar 38, including how to replace drives communicating on a bus or network.

Introduction

Presentation

The CANopen protocol is available on the Altivar speed drive via a VW3 CAN A71 adapter, which must be ordered separately.

The CANopen adapter features a CANopen-compliant 9-way male SUB-D connector referred to as the “CANopen port” in this manual.

The CANopen port on the Altivar can be used for the following functions:

- Configuration
- Adjustment
- Control
- Monitoring

This manual describes how to set up the Altivar drive on CANopen and also describes the CANopen services that are available on this drive.

Notation

Drive terminal displays

The graphic display terminal menus are shown in square brackets.
Example: **[1.9 COMMUNICATION]**.

The integrated 7-segment display terminal menus are shown in round brackets.
Example: (**COM-**).

Parameter names are displayed on the graphic display terminal in square brackets.
Example: **[Fallback speed]**

Parameter codes are displayed on the integrated 7-segment display terminal in round brackets.
Example: (**LFF**).

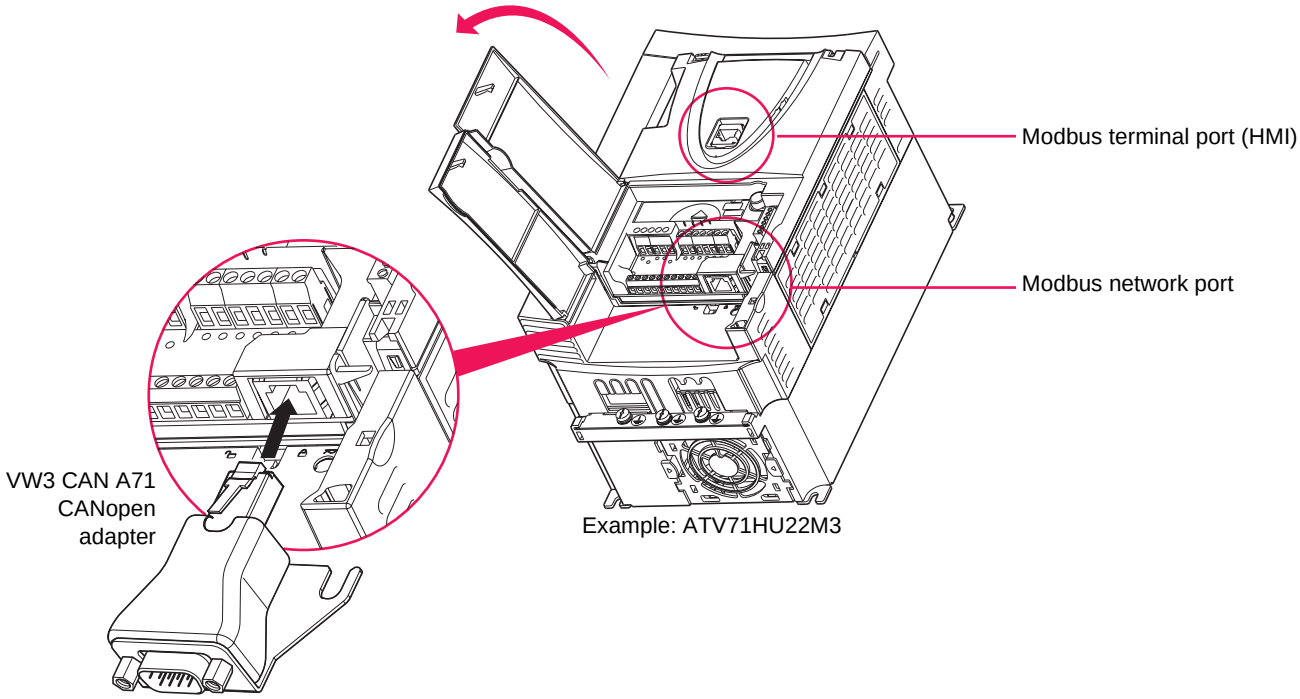
Formats

In this manual, hexadecimal values are written as follows: 16#.

Installing the CANopen adapter

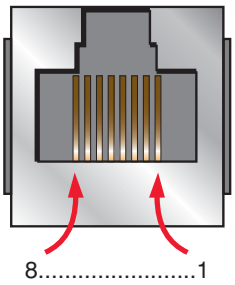
Install the VW3 CAN A71 CANopen adapter in the RJ45 port located on the drive's control terminals.

Note: This adapter MUST be screwed to the drive via its CANopen bus metal grounding plate.



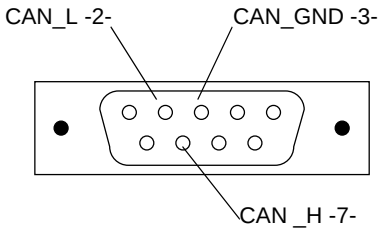
Modbus network port pinout

View from underneath



Pin	Signal
1	**CAN_H**
2	**CAN_L**
3	**CAN_GND**
4	D1 (1)
5	D0 (1)
6	Not connected
7	VP (2)
8	Common (1)

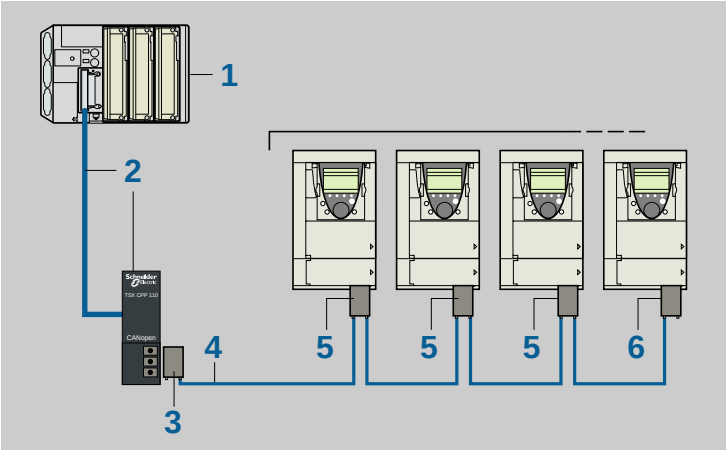
Pinout of 9-way male SUB-D connector on CANopen adapter



(1) Modbus signal
(2) Power supply for an RS232/RS485 converter (to PowerSuite)

Connecting to the CANopen bus

The diagram below illustrates an example configuration comprising four Altivar drives connected to a TSX Premium master PLC fitted with a TSX CPP 110 CANopen master PCMCIA card.
Connection accessories should be ordered separately (please consult our catalogs). See also below.



1. TSX Premium PLC + **TSX CPP 110** CANopen master PCMCIA card
2. Drop cable and cable connector supplied with **TSX CPP 110** card
3. **TSX CAN KCDF 180T** 9-way female SUB-D connector with active line terminator
4. **TSX CAN C•••** CANopen cable, available in lengths of 50, 100 or 300 m
5. **VW3 CAN A71** CANopen adapter + **VW3 CAN KCDF 180T** 9-way female SUB-D connector with deactivated line terminator
6. **VW3 CAN A71** CANopen adapter + **VW3 CAN KCDF 180T** 9-way female SUB-D connector with active line terminator

Description	Length (m)	Catalog number
CANopen adapter to be installed in the RJ45 port on the drive's control terminals. The adapter provides a 9-way male SUB-D connector conforming to the CANopen standard (CIA DRP 303-1).	–	**VW3 CAN A71**
CANopen connector (1) 9-way female SUB-D connector with line terminator (can be deactivated)	–	**VW3 CAN KCDF 180T**
CANopen cables LSZH CE certification. Low smoke, halogen free and flame retardant (IEC 60332-1).	50	**TSX CAN CA 50**
	100	**TSX CAN CA 100**
	300	**TSX CAN CA 300**
CANopen cables UL / IEC 60332-2 UL certification. Non flame propagating (IEC 60332-2).	50	**TSX CAN CB 50**
	100	**TSX CAN CB 100**
	300	**TSX CAN CB 300**
CANopen cables flexible LSZH HD For heavy duty or mobile applications. Low smoke, halogen free and flame retardant (IEC 60332-1). Oil resistant.	50	**TSX CAN CD 50**
	100	**TSX CAN CD 100**
	300	**TSX CAN CD 300**

(1) For ATV 71H•••M3, ATV 71HD11M3X, HD15M3X and ATV 71H075N4 to HD18N4 drives, this connector can be replaced by the TSX CAN KCDF 180T connector.

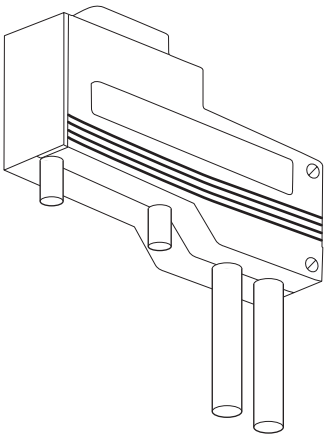
Length of CANopen bus

The maximum length of the bus depends on the communication speed:

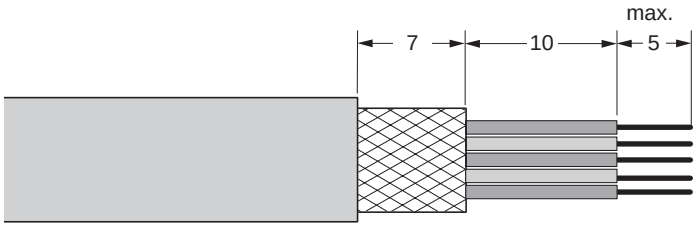
Communication speed	20 kbps	50 kbps	125 kbps	250 kbps	500 kbps	1000 kbps
Maximum length of bus	2500 m	1000 m	500 m	250 m	100 m	25 m

These lengths are for a CANopen bus and take into account actual dispersions on the components as well as when certain devices are optocoupled to the bus.
Telemecanique will not accept liability for longer lengths specified in other documents.

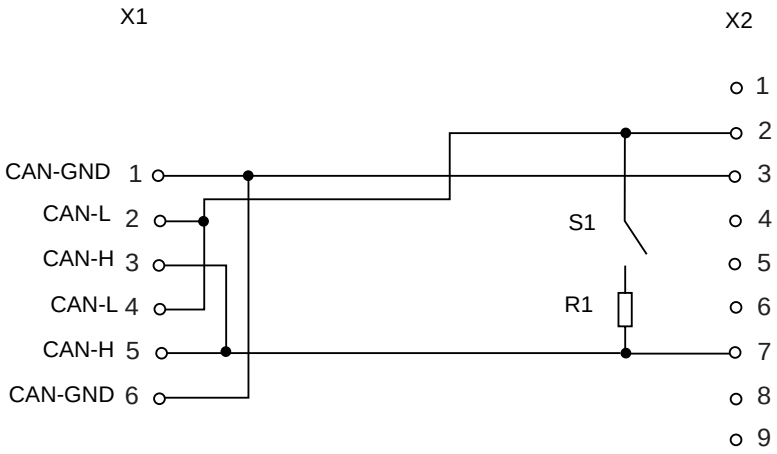
CANopen connector (VW3 CAN KCDF 180T)



How to strip the CANopen cable



Schematic



X1 = Internal screw terminal
X2 = 9-way female SUB-D

X1 = Internal screw terminal	X2 = 9-way female SUB-D	Signal
1, 6	3	CAN_GND
2, 4	2	CAN_L
3, 5	7	CAN_H

Configuring the communication parameters

The configuration of the CANopen communication functions on the Altivar is accessed via the **[1.9 - COMMUNICATION]** (COM-), menu (**[CANopen]** (CnO-) submenu) on the graphic display terminal or integrated display terminal.

Note:
The configuration can only be modified when the motor is stopped and the drive locked.
In order for modifications to take effect, the drive must be shut down then restarted.

Parameter	Possible values	Terminal display	Default value
[CANopen address] (AdCO)	CANopen deactivated 1 to 127	**[OFF]** (0FF) **[1]** (1)....**[127]** (127)	**[OFF]** (0FF)
[CANopen bit rate] (bdCO)	-	**[20 kbps]** (20) (1)	**[125 kbps]** (125)
	50 kbps	**[50 kbps]** (50)	
	125 kbps	**[125 kbps]** (125)	
	250 kbps	**[250 kbps]** (250)	
	500 kbps	**[500 kbps]** (500)	
	1 000 kbps	**[1000 kbps]** (1M)	

In this user's manual, the **[CANopen address]** (AdC0) parameter is referred to as the "NODE-ID".

Set this parameter to its default value, **(0FF)**, to deactivate CANopen communication on the Altivar.
To activate CANopen communication on the Altivar, write a value other than zero to the **[CANopen address]** (AdC0) parameter.

The value of the **[CANopen bit rate]** (bdC0) parameter must correspond to the communication speed of all other devices connected to the CANopen bus.

(1) Do not select **[20 kbps]** (20), possible malfunction.

Control-signal configuration

Numerous control-signal configurations are possible. Please consult the Programming Manual.
The following configurations are just some of the possibilities available.

Control via CANopen in I/O profile

The command and target come from CANopen.
The command is in I/O profile.

Configure the following parameters:

Parameter	Value	Comment
Profile	I/O profile	The run command is simply obtained by bit 0 of the control word.
Target 1 configuration	CANopen	The target comes from CANopen.
Command 1 configuration	CANopen	The command comes from CANopen.

Configuration via the graphic display terminal or the integrated display terminal:

Menu	Parameter	Value
[1.6 - COMMAND] (CtL-)	**[Profile]** (CHCF)	**[I/O profile]** (IO)
	[Ref. 1 channel] (Fr1)	**[CANopen]** (CAn)
	[Cmd channel 1] (Cd1)	**[CANopen]** (CAn)

Control via CANopen or the terminals in I/O profile

Both the command and target come from CANopen or the terminals. Input LI5 at the terminals is used to switch between CANopen and the terminals.
The command is in I/O profile.

Configure the following parameters:

Parameter	Value	Comment
Profile	I/O profile	The run command is simply obtained via bit 0 of the control word.
Target 1 configuration	CANopen	Target 1 comes from CANopen.
Target 1B configuration	Analog input 1 on the terminals	Target 1B comes from input AI1 on the terminals.
Target switching	Input LI5	Input LI5 switches the target (1 ↔ 1B).
Command 1 configuration	CANopen	Command 1 comes from CANopen.
Command 2 configuration	Terminals	Command 2 comes from the terminals.
Command switching	Input LI5	Input LI5 switches the command.

Target 1B is connected to the functions (summing, PID, etc.) that remain active, even after switching.

Configuration via the graphic display terminal or the integrated display terminal:

Menu	Parameter	Value
[1.6 - COMMAND] (CtL-)	**[Profile]** (CHCF)	**[I/O profile]** (IO)
	[Ref. 1 channel] (Fr1)	**[CANopen]** (CAn)
	[Cmd channel 1] (Cd1)	**[CANopen]** (CAn)
	[Cmd channel 2] (Cd2)	**[Terminals]** (tEr)
	[Cmd switching] (CCS)	**[LI5]** (LI5)
[1.7 APPLICATION FUNCT.] (FUn-) **[REFERENCE SWITCH.]**	**[Ref. 1B chan]** (Fr1b)	**[AI1]** (AI1)
	[Ref 1B switching] (rCb)	**[LI5]** (LI5)

Control via CANopen in DSP402 profile

The command and target come from CANopen.
The command is in DSP402 profile (not separate mode).

Configure the following parameters:

Parameter	Value	Comment
Profile	DSP402 profile not separate	The run commands are in DSP402 profile, the command and the target come from the same channel.
Target 1 configuration	CANopen	The command and the target come from CANopen.

Configuration via the graphic display terminal or the integrated display terminal:

Menu	Parameter	Value
[1.6 - COMMAND] (CtL-)	**[Profile]** (CHCF)	**[Not separ.]** (SIM) (factory setting)
	[Ref. 1 channel] (Fr1)	**[CANopen]** (CAn)

Control via CANopen or the terminals in DSP402 profile

Both the command and target come from CANopen or the terminals. Input LI5 at the terminals is used to switch between CANopen and the terminals.
The command is in DSP402 profile (not separate mode).

Configure the following parameters:

Parameter	Value	Comment
Profile	DSP402 profile not separate	The run commands are in DSP402 profile, the command and the target come from the same channel.
Target 1 configuration	CANopen	Target 1 comes from CANopen.
Target 2 configuration	Analog input 1 on the terminals	Target 2 comes from input AI1 on the terminals.
Target switching	Input LI5	Input LI5 switches the target (1 ↔ 2) and the command.

Warning: Target 2 is directly connected to the drive reference limit. If switching is performed, the functions that affect the reference (summing, PID, etc) are inhibited.

Configuration via the graphic display terminal or the integrated display terminal:

Menu	Parameter	Value
[1.6 - COMMAND] (CtL-)	**[Profile]** (CHCF)	**[Not separ.]** (SIM)
	[Ref. 1 channel] (Fr1)	**[CANopen]** (CAn)
	[Ref. 2 channel] (Fr2)	**[AI1]** (AI1)
	[Ref. 2 switching] (rFC)	**[LI5]** (LI5)

Command in DSP402 profile via CANopen and target switching at the terminals

The command comes from CANopen.
The command comes either from CANopen or from the terminals. Input LI5 at the terminals is used to switch the target between CANopen and the terminals.
The command is in DSP402 profile (separate mode).

Configure the following parameters:

Parameter	Value	Comment
Profile	Separate DSP402 profile	The run commands are in DSP402 profile, the command and the target can come from different channels.
Target 1 configuration	CANopen	Target 1 comes from CANopen.
Target 1B configuration	Analog input 1 on the terminals	Target 1B comes from input AI1 on the terminals.
Target switching	Input LI5	Input LI5 switches the target (1 ↔ 1B).
Command 1 configuration	CANopen	Command 1 comes from CANopen.
Command switching	Channel 1	Channel 1 is the command channel.

Target 1B is connected to the functions (summing, PID, etc.), which remain active, even after switching.

Configuration via the graphic display terminal or the integrated display terminal:

Menu	Parameter	Value
[1.6 - COMMAND] (CtL-)	**[Profile]** (CHCF)	**[Separate]** (SEP)
	[Ref. 1 channel] (Fr1)	**[CANopen]** (CAn)
	[Cmd channel 1] (Cd1)	**[CANopen]** (CAn)
	[Cmd switching] (CCS)	**[ch1 active]** (Cd1)
[1.7 APPLICATION FUNCT.] (FUn-) **[REFERENCE SWITCH.]**	**[Ref. 1B chan]** (Fr1b)	**[AI1]** (AI1)
	[Ref 1B switching] (rCb)	**[LI5]** (LI5)

Configuring monitored parameters

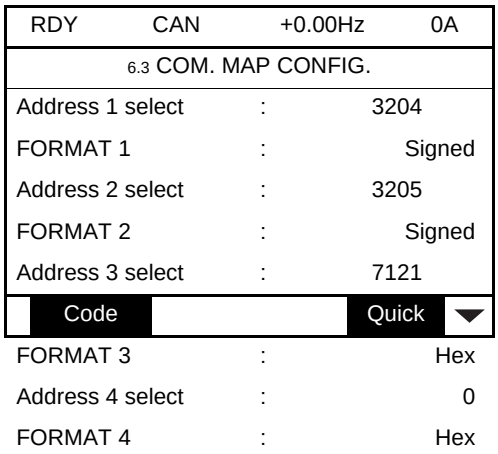
It is possible to select up to 4 parameters to display their values in the **[1.2 - MONITORING]** menu on the graphic display terminal.

The selection is made via the **[6 - MONITORING CONFIG.]** menu (**[6.3 - COM. MAP CONFIG.]** submenu).

Each parameter in the range **[Address 1 select] ... [Address 4 select]** can be used to select the parameter logic address. An address at zero is used to disable the function.

In the example given here, the monitored words are:

- Parameter 1 = Motor current (LCR): logic address 3204, signed decimal format.
- Parameter 2 = Motor torque (OTR): logic address 3205, signed decimal format.
- Parameter 3 = Last fault occurred (LFT): logic address 7121, hexadecimal format.
- Disabled parameter: 0; default format: Hexadecimal format



One of the three display formats below can be assigned to each monitored word:

Format	Range	Terminal display
Hexadecimal	0000 ... FFFF	**[Hex]**
Signed decimal	-32 767 ... 32 767	**[Signed]**
Unsigned decimal	0 ... 65 535	**[Unsigned]**

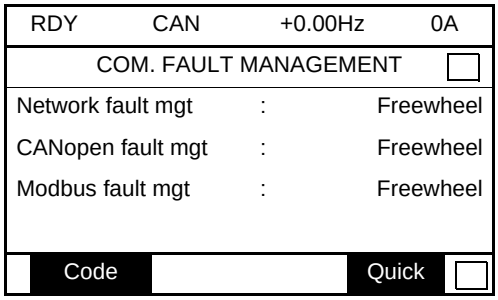
Note: If a monitored parameter:
- has been assigned to an unknown address
- has been assigned to a protected parameter
- has not been assigned

the value displayed in the **[COMMUNICATION MAP]** screen is: "**-----**" (see "Diagnostics" section).

Configuring communication fault management

The response of the drive in the event of a CANopen communication fault can be configured.

It can be configured via the graphic display terminal or the integrated display terminal, from the **[1.8 – FAULT MANAGEMENT]** (FLt-) menu, **[COM. FAULT MANAGEMENT]** (CLL-) submenu, via the **[CANopen fault mgt]** (COL) parameter.



The values of the **[CANopen fault mgt]** (COL) parameter, which trigger a drive fault **[CANopen com.]** (COF), are:

Value	Meaning
[Freewheel] (YES)	Freewheel stop (factory setting).
[Ramp stop] (rMP)	Stop on ramp.
[Fast stop] (FSt)	Fast stop.
[DC injection] (dCI)	DC injection stop.

The values of the **[CANopen fault mgt]** (COL) parameter, which do not trigger a drive fault, are:

Value	Meaning
[Ignore] (nO)	Fault ignored.
[Per STT] (Stt)	Stop according to configuration of **[Type of stop]** (Stt).
[fallback spd] (LFF)	Change to fallback speed, maintained as long as the fault persists and the run command has not been removed.
[Spd maint.] (rLS)	The drive maintains the speed at the time the fault occurred, as long as the fault persists and the run command has not been removed.

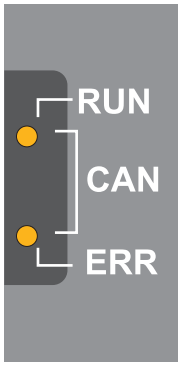
The fallback speed can be configured in the **[1.8 – FAULT MANAGEMENT]** (FLt-) menu using the **[Fallback speed]** (LFF) parameter.

Diagnostics

LEDs

The two LEDs located on the right-hand side of the 7-segment integrated display terminal indicate the CANopen communication status.

The two LEDs on the left-hand side are reserved for Modbus and are not described in this manual.



LED	LED state	Altivar / CANopen state
RUN		The drive CANopen controller is in the "OFF" state.
		The Altivar is in the CANopen "Stopped" state.
		The Altivar is in the CANopen "Pre-operational" state.
		The Altivar is in the CANopen "Operational" state.
ERR		No error signaled
		Warning output by the Altivar CANopen controller (e.g., too many error frames)
		ERROR due to the occurrence of a "Node Guarding" or "Heartbeat" event or an "overrun" on the CANopen bus (network overload)
		The CANopen controller is in the "OFF" state.

Key:

LED state	Visual description of the LED state	LED state	Visual description of the LED state
	The LED is **on**		The LED **flashes** at 2.5 Hz (on for 200 ms then off for 200 ms)
	The LED is in **single flash** mode (on for 200 ms then off for 1 second)		The LED is **off**
	The LED is in **double flash** mode (on for 200 ms then off for 200 ms, on for 200 ms then off for 1 second)		

Communication diagnostics

RUN	CAN	+50.00Hz	80A
COMMUNICATION MAP			
Command Channel	:		CANopen
Cmd value	:		000FHex
Active ref. channel	:		CANopen
Frequency ref.	:		500.0Hz
Status word	:		8627Hex
Code		Quick	
W3204	:		53
W3205	:		725
W7132	:		0000Hex
W0	:		-----
COM. SCANNER INPUT MAP			
COM SCAN OUTPUT MAP			
CMD. WORD IMAGE			
FREQ. REF. WORD MAP			
MODBUS NETWORK DIAG			
MODBUS HMI DIAG			
CANopen MAP			
PROG. CARD SCANNER			

On the display terminal, the **[1.2 - MONITORING]** (SUP-) menu (**[COMMUNICATION MAP]** (CMM-) submenu, **[CANopen MAP]** submenu) can be used to display the communication status on CANopen.

LED display

- **[RUN LED]** LED ("OFF", "Stopped", "Pre-operational" or "Operational" state of the CANopen controller)
- **[ERR LED]** LED (CANopen error)

These LEDS are equivalent to the "CAN RUN" and "CAN ERR" LEDs on the 7-segment integrated terminal (where supplied together with the drive).

The display on the screen opposite indicates that the CANopen controller is in the "Operational" state (**[RUN LED]** LED permanently lit) and that the controller has not detected any errors present (**[ERR LED]** not lit).

⊗ indicates an LED, which is not lit;

◒ indicates an LED, which is lit.

RUN	CAN	+50.00Hz	80A
CANopen MAP			
RUN LED	:		◒
ERR LED	:		⊗
PDO1 IMAGE	:		
PDO2 IMAGE	:		
PDO3 IMAGE	:		
Code		Quick	
Canopen NMT state	:		Operational
Number of TX PDO	:		2438
Number of RX PDO	:		2438
Error code			0
RX Error Counter			0
TX Error Counter			0

NMT chart display

The **[CANopen NMT state]** (NMTS) parameter (logic address 6057, CANopen index/subindex 16#201E/3A) indicates the NMT chart state. The various possible values are **[Boot]**, **[Stopped]**, **[Operational]** and **[Pre-Op]** (Pre-operational).

PDO counter display

[Number of RX PDO] and **[Number of TX PDO]** indicate the number of PDOs received and the number of PDOs transmitted by the drive (all PDO sets - PDO1, PDO2 and PDO3 - combined).
These counters are modulo 65 536 counters, i.e., the value is reset to zero once 65 535 is reached.

Last CANopen fault

The **[Error code]** (ErCO) parameter (index/subindex 16#201E/39) indicates the last active CANopen fault and maintains its value until the last fault has disappeared.

The possible values are listed below:

Display	Description
[0]	No errors detected since the start of CANopen communication
[1]	"Bus Off" requiring the drive to be restarted
[2]	"Life Guarding" fault requiring a return to the NMT "Initialization" state
[3]	"CAN overrun" error not requiring any specific action to be taken
[4]	"Heartbeat" fault requiring a return to the NMT "Initialization" state
[5]	NMT state chart fault (see section "CANopen NMT state chart")

Counters

The **[RX Error Counter]** (rEC1) parameter (logic address 6059, CANopen index/subindex 16#201E/3C) counts the number of frames received with errors for all types of frame (PDO, SDO, etc.).

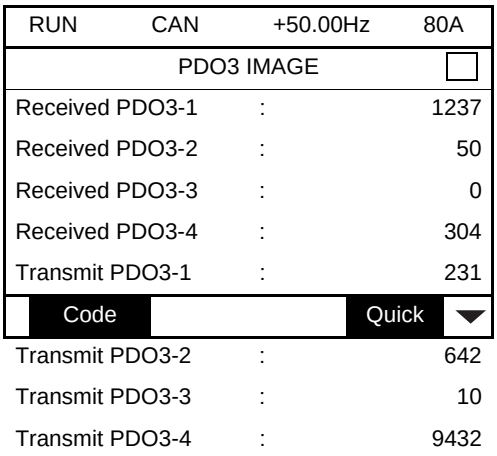
The **[TX Error Counter]** (tEC1) parameter (logic address 6058, CANopen index 16#201E/3B) counts the number of frames transmitted with errors for all types of frame (PDO, SDO, etc.).
These types of error can be caused, for example, by network load problems or the short-circuiting of electrical signals on the bus.

The maximum count value supported by these two counters is 65 535.

PDO value display

A second level of submenus can be accessed via the **[CANopen map]** submenu: **[PDO1 IMAGE]**, **[PDO2 IMAGE]** and **[PDO3 IMAGE].**

Each of these submenus can be used to access a screen displaying the values transmitted and received by each set respectively (PDO1, PDO2 and PDO3).



In each of these screens and for each PDO transmitted or received, only the **[Transmit PDO•-•]** or **[Received PDO•-•]** words actually transmitted and received on the CANopen bus are displayed.
This means, for example, that for a receive PDO2 containing only 4 data bytes (i.e., RP21 and RP22), the fields **[Received PDO2-3]** and **[Received PDO2-4]** will not be displayed.

Control-signal diagnostics

On the terminal, the **[1.2 - MONITORING]** menu (**[COMMUNICATION MAP]** submenu) can be used to display control-signal diagnostic information between the Altivar drive and the CANopen master:

- Active command channel
- Value of the control word (CMD) from the active command channel
- Active target channel
- Value of the target from the active target channel
- Value of the status word
- Values of the four parameters selected by the user
- The **[COM. SCANNER INPUT MAP]** submenu: is unnecessary for CANopen
- The **[COM SCAN OUTPUT MAP]** submenu: is unnecessary for CANopen
- In the **[CMD. WORD IMAGE]** submenu: control words from all channels
- In the **[FREQ. REF. WORD MAP]** submenu: frequency targets produced by all channels

Example of the display of communication diagnostic information

RUN	CAN	+50.00Hz	80A
COMMUNICATION MAP			
Command Channel	:		CANopen
Cmd value	:		000FHex
Active ref. channel	:		CANopen
Frequency ref.	:		500.0Hz
Status word	:		8627Hex
Code		Quick	▼
W3204	:		73
W3205	:		725
W7132	:		0000Hex
W0	:		-----
COM. SCANNER INPUT MAP			
COM SCAN OUTPUT MAP			
CMD. WORD IMAGE			
FREQ. REF. WORD MAP			
MODBUS NETWORK DIAG			
MODBUS HMI DIAG			
CANopen MAP			
PROG. CARD SCANNER			

Control word display

The **[Command Channel]** parameter indicates the active command channel.

The **[Cmd value]** parameter indicates the hexadecimal value of the control word (CMD) used to control the drive.

The **[CMD. WORD IMAGE]** submenu (**[CANopen cmd.]** parameter) is used to display the hexadecimal value of the control word sent by CANopen.

Frequency target display

The **[Active ref. channel]** parameter indicates the active target channel.

The **[Frequency ref]** parameter indicates the value (in 0.1 Hz units) of the frequency target (LFR) used to control the drive.

The **[FREQ. REF. WORD MAP]** submenu (**[CANopen ref.]** parameter) is used to display the value (in 0.1 Hz units) of the speed target sent by CANopen.

Status word display

The **[Status word]** parameter gives the value of the status word (ETA).

Display of the parameters selected by the user

The four **[W•••]** parameters give the value of the four monitored words selected by the user.

The address and display format of these parameters can be configured in the **[6 MONITORING CONFIG.]** menu (**[6.3 - COM. MAP CONFIG.]** submenu).

The value of a monitored word equals "-----" if:

- Monitoring has not been activated (address equals W0)
- The parameter is protected
- The parameter is not known (e.g., W3200)

Displaying communication scanner values

The communication scanner is unnecessary for CANopen.

Communication faults

CANopen communication faults are displayed by the ERR indicator of the integrated display terminal or graphic display terminal.

In factory settings, a CANopen communication fault triggers a resettable drive fault **[CANopen com.]** (**COF**) and a freewheel stop.

The response of the drive in the event of a CANopen communication fault can be changed (see "Configuring communication fault management"):
- Drive fault **[CANopen com.]** (**COF**) (freewheel stop, stop on ramp, fast stop or DC injection stop).
- No drive fault (stop, maintain, fallback).

The fault management is described in the user guide "Communication parameters", chapter "Communication monitoring" :
- After initialisation (power up), the drive checks that at least one of the command or target parameters has been written once via CANopen.
- Then, if a CANopen communication fault occurs, the drive reacts according to the configuration (stop, maintain, fallback ...).

The source of this fault is displayed on the terminal: **[1.2 - MONITORING]** (**SUP-**) menu, **[COMMUNICATION MAP]** (**CMM-**) submenu, **[CANopen MAP]** submenu, **[Error code]** (**ErCO**) parameter (described in the "Communication diagnostics" section).

In the event of a **[CANopen com.]** (**COF**) fault, the drive sends an EMCY message to the CANopen master.

Profiles

Communication profile

The Altivar communication profile is based on:

- CAN 2.A
- The CANopen specification (DS301 V4.02)

Simplified telegram structure (communication object):

Identifier (11 bits)	Data (maximum length 8 bytes)							
COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7

Identifier: The identifier of a CANopen telegram is coded on 11 bits. The identifier is also called the COB-ID (Communication OBject Identifier).

Bits 0 to 6: CANopen address (Node-ID) of the device sending/receiving the telegram

Bits 7 to 10: Telegram function code. The table below lists the function codes used by the Altivar:

Function code (bit 10 … bit 7)	Service	Range of possible identifiers	Index of parameterization objects (1)
2#0000	NMT	16#000	-
2#0001	SYNC	16#080	16#1005
	EMCY	16#081 to 16#0FF	-
2#0011	Transmit PDO1 (TPD01)	16#181 to 16#1FF	16#1800, 16#1A00
2#0100	Receive PDO1 (RPD01)	16#201 to 16#27F	16#1400, 16#1600
2#0101	Transmit PDO2 (TPD02)	16#281 to 16#2FF	16#1801, 16#1A01
2#0110	Receive PDO2 (RPD02)	16#301 to 16#37F	16#1401, 16#1601
2#0111	Transmit PDO3 (TPD03)	16#381 to 16#3FF	16#1802, 16#1A02
2#1000	Receive PDO3 (RPD03)	16#401 to 16#47F	16#1402, 16#1602
2#1011	Transmit SDO	16#581 to 16#5FF	16#1200
2#1100	Receive SDO	16#601 to 16#67F	16#1200
2#1110	Heartbeat	16#701 to 16#77F	16#1016, 16#1017

(1) These objects are described in the Object dictionary on page 56.

For more detailed information, visit the ***Can In Automation*** website at: http://www.can-cia.org.

Functional profile

See the Parameters Manual.

PDO (Process Data Objects)

PDO telegrams are used to exchange periodic I/O data between the PLC and the drive.

The Altivar has three predefined PDO sets.

- The first PDO set (PDO1) reserved for drive control-signaling. Active by default, it comprises:
  - RPD01 (receive), to control the drive in speed mode (2 words: "CMD" control word and "LFRD" speed target)
  - TPD01 (transmit), to monitor the drive in accordance with this same mode (2 words: status word "ETA" and output speed "RFRD")

  Each of these two PDOs can be reconfigured to include only the control word "CMD" or only the status word "ETA", thereby reducing its size to a single word.

  It can also be reconfigured to control the drive via its torque function. This changes its size to 3 words by means of the addition of the torque target "LTR" (receive PDO) and the motor torque "OTR" (transmit PDO).

  Finally, these PDOs can be reconfigured entirely (1 to 4 words of the user's choice).

- The second PDO set (PDO2) is deactivated by default and can be configured in full (1 to 4 words of the user's choice).

  It is reserved for adjustments and for additional control and monitoring functions.

  By default, TPD02 (transmit) and RPD02 (receive) are not configured.

- The third PDO set (PDO3) is reserved for the "Controller Inside" programmable card on the Altivar (catalog number VW3 A3 501). Deactivated by default, it cannot be configured and comprises:
  - RPDO3 (receive), containing 4 input words of the "Controller Inside" programmable card
  - TPDO3 (transmit), containing 4 output words of the "Controller Inside" programmable card

  PDO3 should only be activated and used on a drive supplied with the "Controller Inside" programmable card.

RPDO1, TPDO1, RPDO2, TPDO2, RPDO3, and TPDO3 can each be enabled or disabled independently.

Each PDO can be activated or deactivated using bit 31 of its COB-ID. Set this bit to 1 to deactivate the PDO. Reset it to zero to activate the PDO.

By default, these three PDO are asynchronous, although the transmission mode of each PDO can be reconfigured by the user in accordance with requirements:

- Asynchronous mode (255): The transmit PDO is only sent when the value of its data changes. In this mode, the "inhibit time" and "event timer" (e.g., objects 16#1800/03 and 16#1800/05 for TPD01) can be modified in order to adjust the PDO transmission frequency on the bus.
- Cyclic synchronous mode (1…240): The transmit PDO is sent each time a synchronization object (SYNC) is received or when a preconfigured number of synchronization objects (between 1 and 240) are received.
- Acyclic synchronous mode (0): The transmit PDO is sent each time the value of its data changes, but only during the synchronous "window" authorized by the next synchronization object (SYNC, not availble for the receive PDO).

The Altivar does not support the transmission of transmit PDOs on receipt of RTR frames (252, 253).

The drive optimizes the size of the TPDO frames (transmit); only useful data bytes are transmitted.
The length of PDO3 is always 8 data bytes.

PDO default configuration

Receive PDOs

No. of PDOs	Parameters configured by default		Comment
	Index	Name	
1	16#6040	Control word (CMD)	**Drive control in speed regulation mode** This PDO can be reconfigured. It is activated by default.
	16#6042	Speed target (LFRD)	
2		No parameter configured by default	**Additional settings or commands** This PDO can be reconfigured. It is deactivated by default.
3	16#2064/2	"Controller inside" output word 1 (RP31)	**"Controller inside" card control** This PDO cannot be reconfigured. It is deactivated by default.
	16#2064/3	"Controller inside" output word 2 (RP32)	
	16#2064/4	"Controller inside" output word 3 (RP33)	
	16#2064/5	"Controller inside" output word 4 (RP34)	

Transmit PDOs

No. of PDOs	Parameters configured by default		Comment
	Index	Name	
1	16#6041	Status word (ETA)	**Drive monitoring in speed regulation mode** This PDO can be reconfigured. It is activated by default.
	16#6044	Output speed (RFRD)	
2		No parameter configured by default.	**Additional monitoring** This PDO can be reconfigured. It is deactivated by default.
3	16#2064/C	"Controller inside" input word 1 (TP31)	**"Controller inside" card monitoring** This PDO cannot be reconfigured. It is deactivated by default.
	16#2064/D	"Controller inside" input word 2 (TP32)	
	16#2064/E	"Controller inside" input word 3 (TP33)	
	16#2064/F	"Controller inside" input word 4 (TP34)	

Optimizing the response time

The response time can be optimized for the PDO1 configurations described below. In these configurations:

- The taking into account of RPD01 (receive) is processed with the same priority as a logic input from the terminals.
- TPD01 (transmit) is updated with the same priority as an output from the terminals.

In all other configurations, a receive PDO is taken into account by the drive's background task.

Receive PDOs

No. of PDOs	Parameters configured		Comment
	Index	Name	
1	16#6040	Control word (CMD)	Drive control

No. of PDOs	Parameters configured		Comment
	Index	Name	
1	16#6040	Control word (CMD)	Drive control in speed regulation mode
	16#6042	Speed target (LFRD)	

No. of PDOs	Parameters configured		Comment
	Index	Name	
1	16#6040	Control word (CMD)	Drive control in torque control or speed regulation mode
	16#6042	Speed target (LFRD)	
	16#6071	Torque target (LTR)	

Transmit PDOs

No. of PDOs	Parameters configured		Comment
	Index	Name	
1	16#6041	Status word (ETA)	Drive state monitoring

No. of PDOs	Parameters configured		Comment
	Index	Name	
1	16#6041	Status word (ETA)	Drive monitoring in speed regulation mode
	16#6044	Output speed (RFRD)	

No. of PDOs	Parameters configured		Comment
	Index	Name	
1	16#6041	Status word (ETA)	Drive monitoring in torque control or speed regulation mode
	16#6044	Output speed (RFRD)	
	16#6077	Output torque (OTR)	

SDO (Service Data Objects)

SDO telegrams are used for configuration and adjustment.

The Altivar manages an SDO server, characterized by two identifiers:
- One for requests (telegrams sent from the PLC to the Altivar)
- One for responses (telegrams sent back to the PLC by the Altivar)

Although the Altivar supports segmented transfer, this is only required by the reading of object 16#1008 (device name).

Other available services

- Assignment by default of address-based identifiers
- NMT commands: Start_Remote_Node (16#01), Stop_Remote_Node (16#02), Enter_Pre_Operational (16#80), Reset_Node (16#81), Reset_Communication (16#82)
- Bootup
- Heartbeat producer and consumer
- Node Guarding
- Emergency (EMCY)
- SYNC, for all PDOs on the Altivar
- General broadcast support on identifier 0

Service not available
- Time-stamping object (TIME)

Description of identifiers taken into account

The identifiers will be referred to as COB-IDs (**C**ommunication **OBj**ect **ID**entifiers) in the rest of this user's manual.

Direction			Identifier (COB-ID)		Description
Master	➡	Drive	0 16#000		Service Network ManagemenT (NMT)
Master	➡	Drive	128 16#080		SYNChronization service (SYNC)
Master	➡	Drive	128 16#080	+ Node-ID	EMergenCY service (EMCY)
Master	⬅	Drive	384 16#180	+ Node-ID	Drive monitoring (TPDO1)
Master	➡	Drive	512 16#200	+ Node-ID	Drive control (RPDO1)
Master	⬅	Drive	640 16#280	+ Node-ID	Drive periodic input words (TPDO2)
Master	➡	Drive	768 16#300	+ Node-ID	Drive periodic output words (RPDO2)
Master	⬅	Drive	896 16#380	+ Node-ID	Periodic input words on the Altivar's "Controller Inside" programmable card (TPDO3)
Master	➡	Drive	1024 16#400	+ Node-ID	Periodic output words on the Altivar's "Controller Inside" programmable card (RPDO3)
Master	⬅	Drive	1408 16#580	+ Node-ID	Response to a drive setting (transmit SDO)
Master	➡	Drive	1536 16#600	+ Node-ID	Drive setting request (receive SDO)
Master	➡	Drive	1792 16#700	+ Node-ID	Network management (NMT, Node Guard, Heartbeat)
Master	⬅	Drive			Network management (Bootup)

The Altivar supports the automatic assignment of identifiers (COB-IDs), based on its CANopen address.

The term "master" designates a device transmitting a request to a variable speed drive (example: a PLC).

The terms "input" and "output" are understood from the point of view of the master.

Bit 31 of PDO COB-ID entry, coded on 32 bits, is equal to 1 for TPDO2, RPDO2, TPDO3, and RPDO3, as they are inactive by default.

Software setup with PL7 and SyCon

The following sections describe the settings to be made in **PL7 PRO** (**version ⩾ V4.3**) and **SyCon** (**version ⩾ V2.8**) in order for the Altivar to be recognized correctly by the CANopen master PLC. The following software versions are used in this context: **PL7 PRO V4.4** and **SyCon V2.8**.

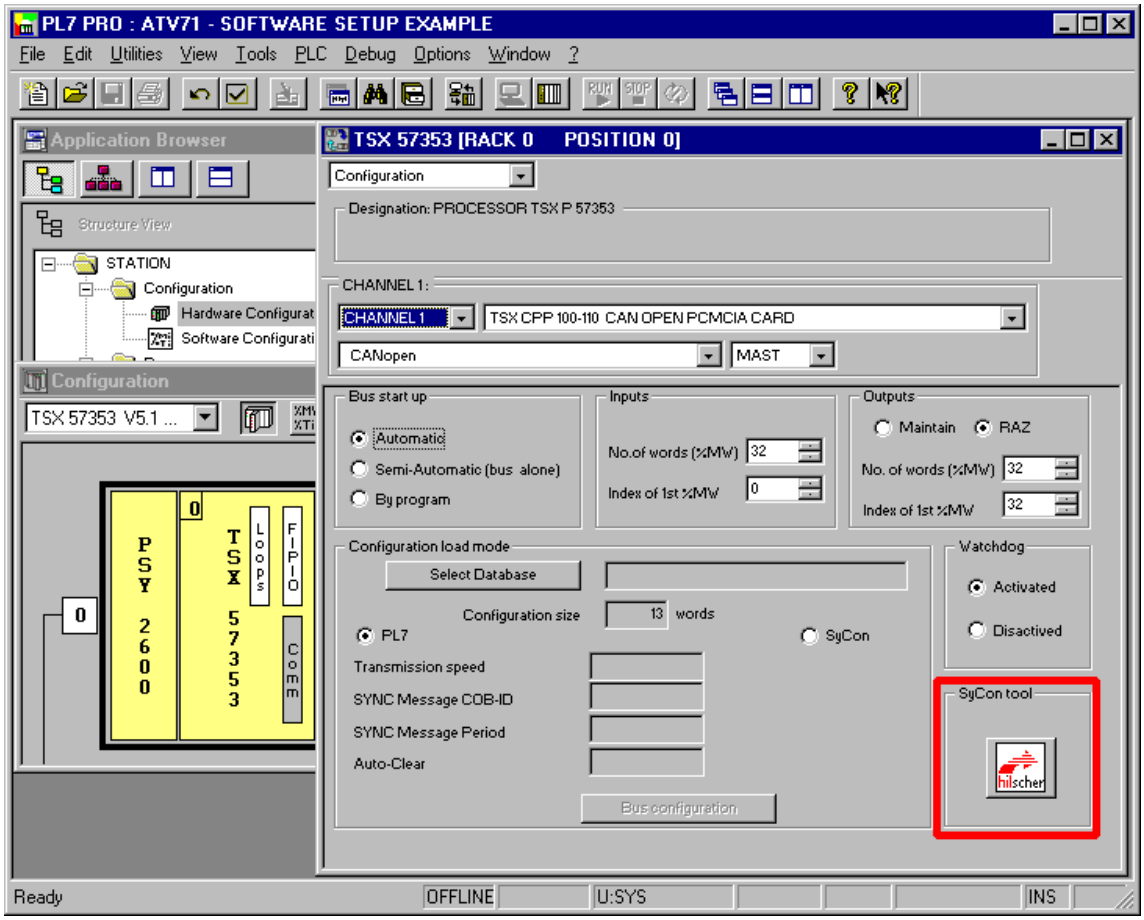
The CANopen bus, described in the following sections, includes just one CANopen master (Premium TSX 57353 V5.1 PLC + TSX CPP 110 CANopen master PCMCIA card) and one slave (Altivar).

Hardware configuration in PL7 PRO

In **PL7 PRO**, create a new application or open the application in which you wish to add a CANopen bus.

Edit the hardware configuration for this application by adding a "TSX CPP 110 PCMCIA CAN OPEN CARD" in "CHANNEL 1" of the "Comm" slot on the TSX 57353.

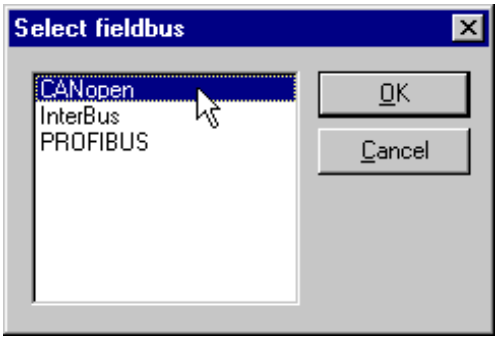
Note: TSX CPP 100 and TSX CPP 110 cards are configured in exactly the same way in PL7 PRO.



Click the "**hilscher**" button (in the red square above) to start up the **SyCon** configuration tool.

N.B. This button will only appear if **SyCon** is installed on the PC.

Creating a CANopen network in SyCon



Select "New" in the "File" menu to create a new configuration and select the "CANopen" bus type.

This command creates an empty network segment in the main **SyCon** window.

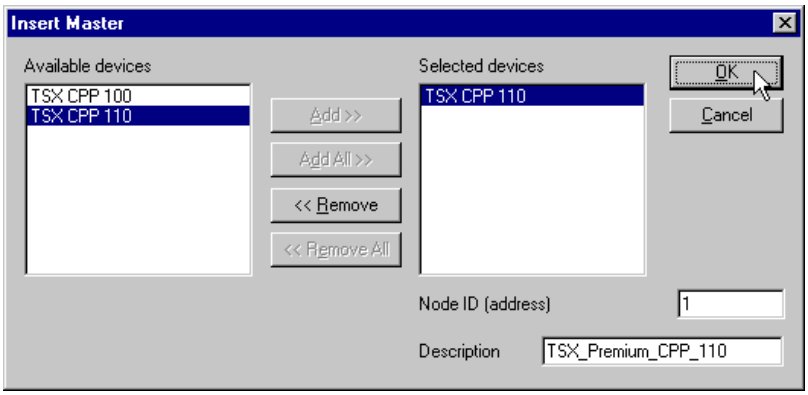
In our example, we save this configuration immediately under the name "`ATV71 – Example Software Setup.co`".

Selecting and adding a CANopen master PLC

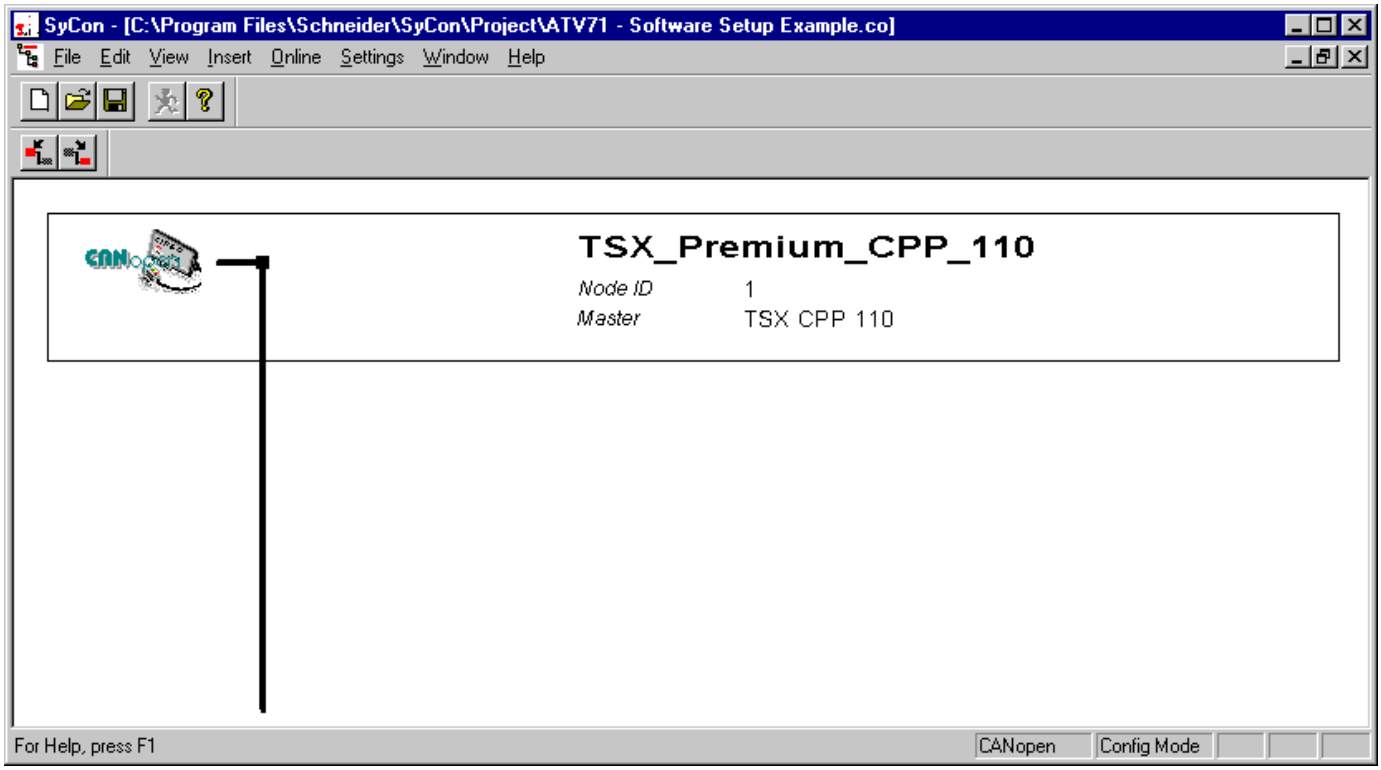
Select "Master…" in the "Insert" menu (or click the  button).

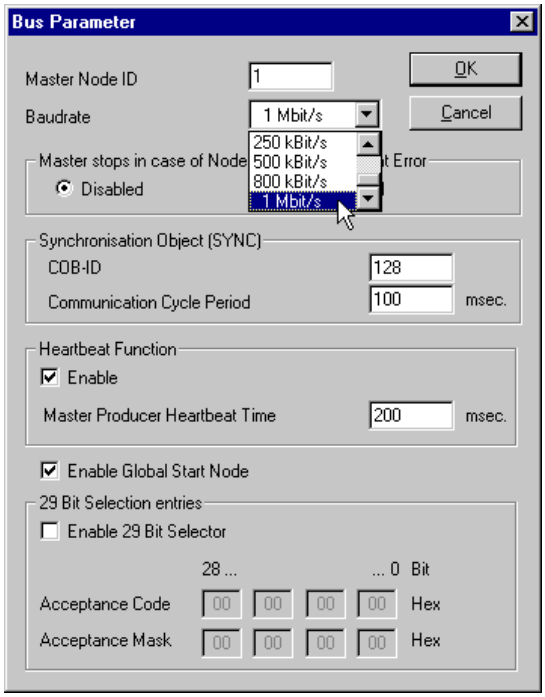
The "Insert Master" window appears. Select the "TSX CPP 110" (or "TSX CPP 100") master device and click "Add >>."

You can also modify the address (Node-ID) and the description of this master device in this window.



Click "OK" to return to the main **SyCon** window.
The selected master appears at the top:





Select the CANopen master and run the "Bus parameters…" command in the "Parameters" menu to set the transmission speed parameter on the CANopen network.

The other parameters in the "Bus Parameter" window are not described here. For more information about the functions of the bus parameters, please refer to the online help or the documentation for the **SyCon** tool.

N.B. If you are using a PDO set in "synchronous" mode (cyclic or acyclic), you may need to adjust the value of "Communication Cycle Period", the default value of which is 100 ms, as shown in the window opposite. The PDO is then actually synchronized on this "Communication Cycle".

Adding Altivar files to CANopen devices managed by SyCon

The EDS file describing the Altivar must be imported into **SyCon** so that it features in its device database. This file is called TEATV71xyE.eds.
x.y: Altivar software version:

- x: Major revision
- y: Minor revision

To import this file into **SyCon**, run the "Copy EDS" command in the "File" menu and select the EDS file indicated above. The user can choose whether or not to "import corresponding bitmap files". Click "Yes" to add the three Altivar state icons (shown below) to the **SyCon** bitmaps database.

If this command is executed correctly, a window will appear to inform the user that 1 EDS file and 3 bitmap files have been imported correctly.

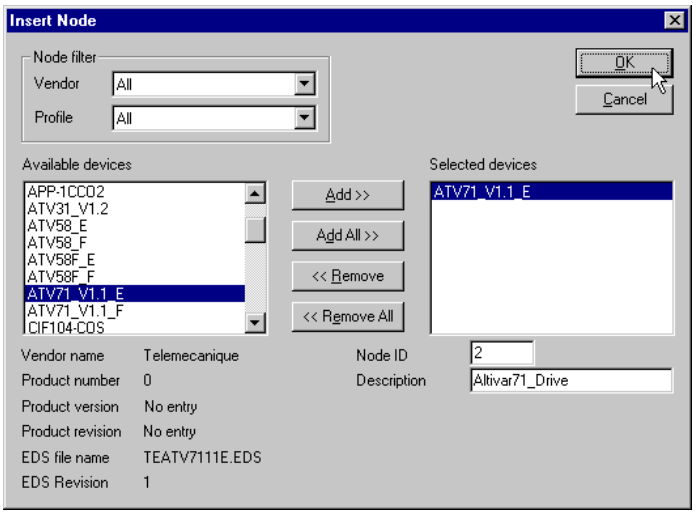
TEATV71xyE_s.dib

TEATV71xyE_r.dib

TEATV71xyE_d.dib

The EDS file and the icon files can be found on the PowerSuite CD-ROM or on the CD-ROM supplied with the Altivar drive.

Selecting and adding the Altivar to the CANopen bus



Select "Node..." in the "Insert" menu (or click the button).

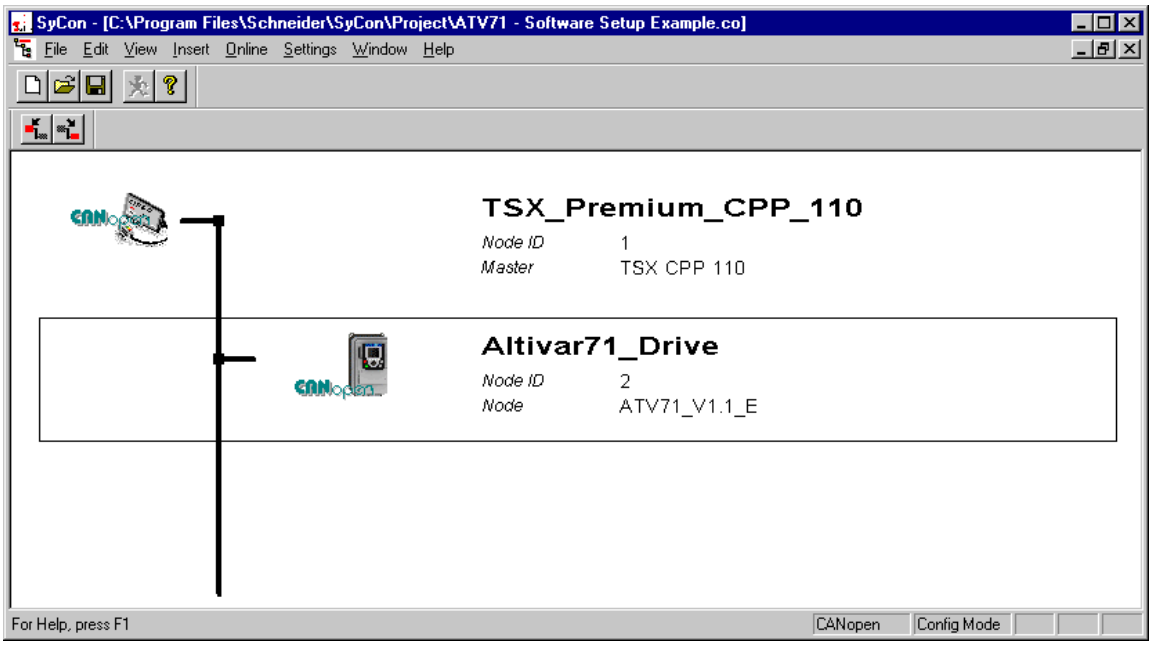
Move the mouse pointer (which now looks like this:) to the position where you wish to add the Altivar, then left-click.

In the "Insert Node" window that appears, select the "ATV71_Vx.y_F" device and click the "Add >>" button.

You can also modify the address (Node-ID) and the description of this node in this window.

Click "OK" to return to the main **SyCon** window.

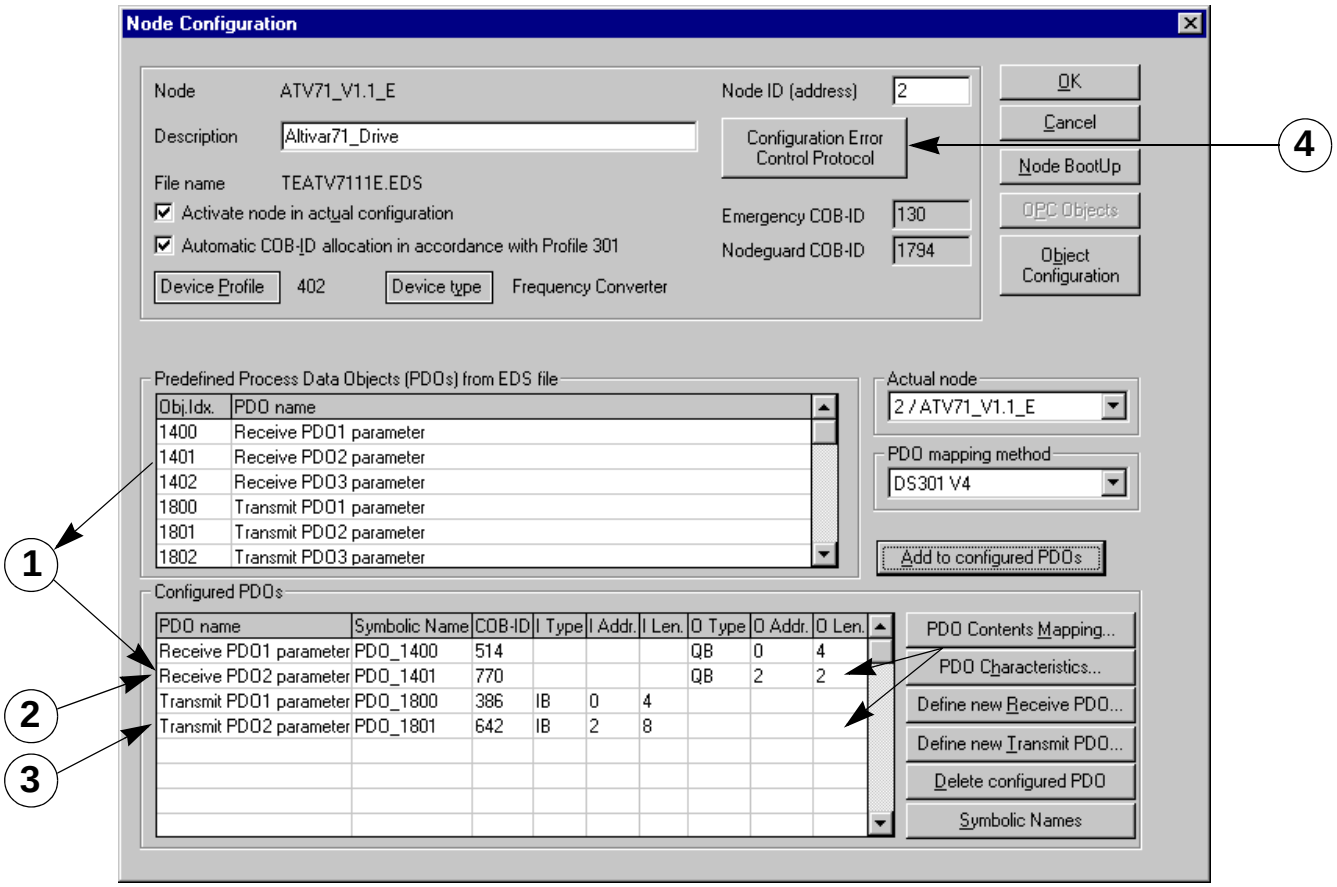
The selected device appears in the required position:



Example of how to edit and configure the Altivar

Double-click the line corresponding to the Altivar. The "Node Configuration" window appears.

The "Node Configuration" window shown below illustrates the small number of operations required to configure Receive PDO1, Transmit PDO1, Receive PDO2 and Transmit PDO2 at the same time:



These operations are summarized here:

① **Adding PDOs to configured PDOs:** In the "Predefined Process Data Objects (PDOs) from EDS file" section, select each of the PDOs to be added to the "Configured PDOs" section and click the "Add to configured PDOs" button or double-click the PDO to be added.
Before adding a PDO to the "Configured PDOs" section, **SyCon** will display a window in which you can configure a transmission mode (transmit or receive PDO).

A "specific CANopen transmission type" can be configured for each of the Altivar receive PDOs.

- Option 1 (value = 1 to 240): Cyclic synchronous mode
- Option 2 or 3 (value = 254 or 255): Asynchronous mode
- Acyclic synchronous mode not available.

A "specific CANopen transmission type" can be configured for each of the Altivar transmit PDOs.

- Option 1 (value = 0): Acyclic synchronous mode
- Option 2 (value = 1 to 240): Cyclic synchronous mode
- Option 3 or 4 (value = 252 or 253): Synchronous and asynchronous modes triggered by the receipt of RTR telegrams (remote frames); these modes are not supported by the Altivar.
- Option 5 or 6 (value = 254 or 255): Asynchronous mode

A given PDO can only be added once to the "Configured PDOs" section (duplication is forbidden).

Important note due to a special feature of Sycon (version V2.8):
If you are not using one (or both) of the PDO1 process data objects, it (or they) must be deactivated via the PL7 program, using the SDO service. Conversely, if you are using at least one of the four PDO2 and/or PDO3 process data objects, it must be activated using the same procedure. Please refer to the section "Activating/Deactivating PDOs via the PL7 program" on page 36.

② and ③ **Modifying the assignment of the "Receive" and "Transmit" parameters for configured PDOs.**
Please refer to the parameters manual, which describes all the drive parameters that can be configured in respect of PDO assignment on the Altivar.

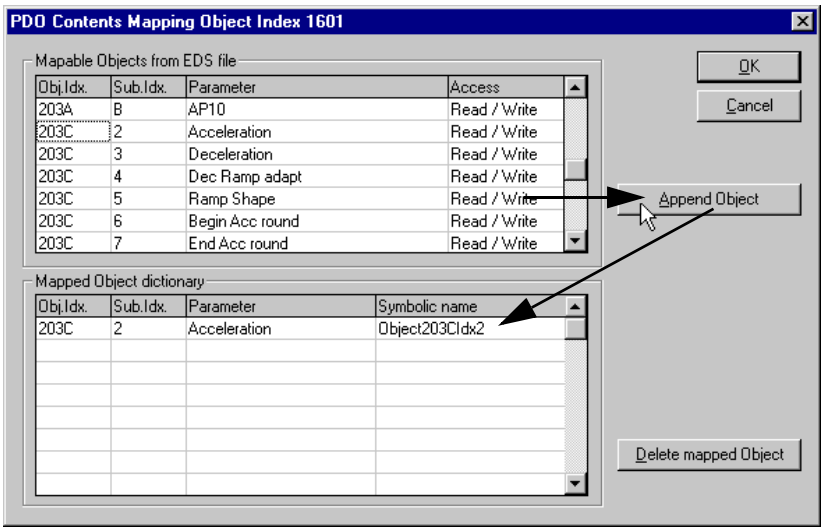
PDO1: In this example, we are not modifying the default assignment of the transmit PDO1 and the receive PDO1 in order to control and monitor the drive in accordance with DSP-402 profile "speed mode"; moreover, this enables the drive to react more quickly to this command via the CANopen bus (see page 44).

In **SyCon**, a new window listing these default assignments can be opened by double-clicking these PDOs or by selecting a PDO and clicking the "PDO Contents Mapping" button:

Receive PDO1 parameter			Transmit PDO1 parameter		
Index	**Subindex**	**Description**	**Index**	**Subindex**	**Description**
16#6040	16#00	Control word (CMD)	16#6041	16#00	Status word (ETA)
16#6042	16#00	Target velocity (LFRD)	16#6044	16#00	Control effort (RFRD)

If you wish to modify the content of these PDO, you can delete and add objects to the "Mapped Object dictionary" of the selected PDO (up to 4 objects for each PDO).
Each object in a PDO assignment uses 2 bytes.

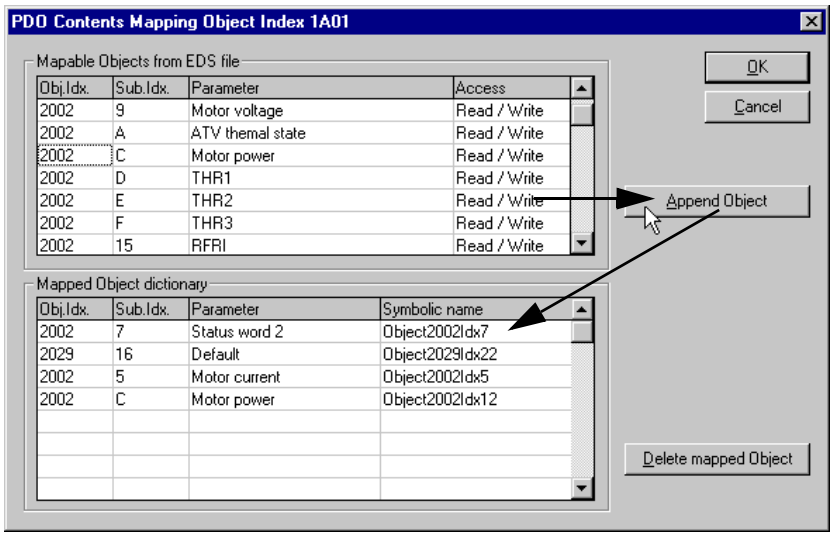
PDO2: In this example, we are using receive PDO2 to transmit the "Acceleration" (ACC) parameter to the drive and transmit PDO2 to receive the "Status word2" (ETI), "Default" (LFT), "Motor current" (LCR), and "Motor power" (oPR) parameters from the drive.



② **Receive PDO2:** To configure the assignment of receive PDO2, select the "Receive PDO2 parameter" PDO in the "Configured PDOs" section and click the "PDO Contents Mapping" button, or double-click this PDO.

Next, scroll down the "Mappable Objects from EDS file" list until you reach the "Acceleration" parameter (index/subindex = 16#203C/02).

Select this parameter and click the "Append Object" button or double-click the parameter to add the corresponding object to the PDO "Mapped Object dictionary", i.e., to the content of its assignment.



③ **Transmit PDO2:** To configure the assignment of transmit PDO2, select the "Transmit PDO2 parameter" PDO in the "Configured PDOs" section and click the "PDO Contents Mapping" button, or double-click this PDO.

Next, scroll down the "Mappable Objects from EDS file" list until you reach the "Status word 2" parameter (index/subindex = 16#2002/07).

Select this parameter and click the "Append Object" button or double-click this parameter to add the corresponding object to the PDO "Mapped Object dictionary" , i.e., to the content of its assignment.

Repeat these operations for the "Default" (index/subindex = 16#2029/16), "Motor current" (index/subindex = 16#2002/05) and "Motor power" (index/subindex = 16#2002/0C) parameters. These parameters correspond to drive parameters LFT, LCR and OPR.

PDO3: In this example, we are not using PDO3, as our drive does not have a "Controller Inside" programmable card.

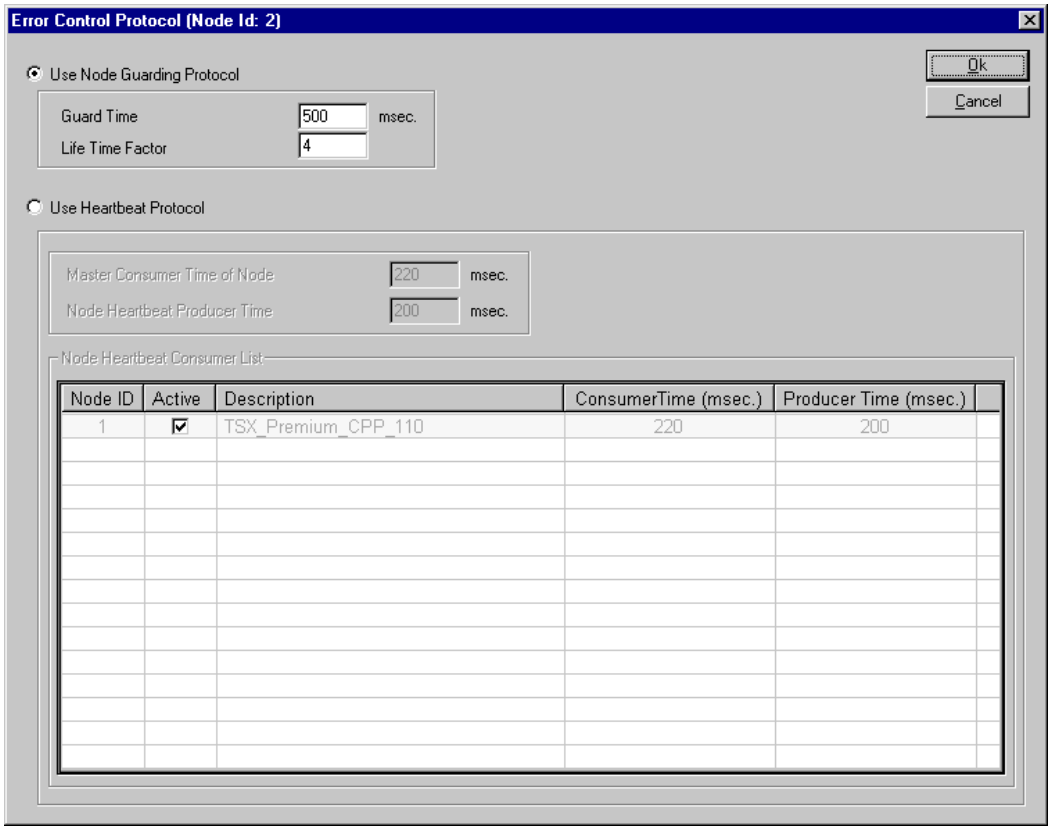
Regardless of its nature, the default assignment of PDO3 is static and cannot, therefore, be modified.

④ **Configuring the error control service on the Altivar**

Click the "Configuration Error Control Protocol" to display the following selection of error control services that can be used on the Altivar: "Node Guarding" service or "Heartbeat" service.

- Node Guarding service: If this service is selected, the two parameters "Guard Time" and "Life Time Factor" are used to generate a Life Time. For more information, please refer to the Node Guarding service section on page 53.
  - In our example, we are configuring a "Guard Time" of 500 ms and a "Life Time Factor" of 4, resulting in a Life Time of 4 x 500 ms = 2 seconds.

- Heartbeat service: If this service is selected, the "Master Consumer Time of Node" parameter (16#1016/01) must be greater than the "Node Heartbeat Producer Time" parameter (16#1017/00). If the "Master Consumer Time of Node" is 0 for the node currently configured, the master will not control the bus activity of this node.



N.B. To use the Heartbeat service with the CANopen master, you must first enable the "Heartbeat function" in the "Bus Parameter" window (see Creating a CANopen network in SyCon on page 18). The CANopen master must of course support this service, as the CANopen master used here (TSX CPP 110) does. The TSX CPP 100 CANopen master PCMCIA card is an example of a CANopen master that does not support this service.

The Altivar can also be configured to control another station (identified uniquely by its NODE-ID) on the bus in the "Node Heartbeat Consumer List" section. Each station appears in this list and its "Producer Time (ms)" is specified. Check a box in the "Active" column to have the corresponding "Node-ID" node checked at regular intervals by the Altivar (in this case, the "Consumer Time (msec.)" must be greater than the "Producer Time (msec.).").

Saving and opening the CANopen bus configuration in PL7 PRO

Save the CANopen configuration and give it a name ("Save" or "Save As…" in the "File" menu). This configuration is saved in a ".co" file.

In the **PL7 PRO** window shown in section "Hardware configuration in PL7 PRO" on page 28, click the "Select Database" button and select the file saved previously (e.g., "C:\Program Files\Schneider\SyCon\Project\ATV71 – Example Software Setup.co"). Once you have made your selection, the "Loading configuration mode" section is updated.

Activating/Deactivating PDOs via the PL7 program

Note:
By default, the Altivar transmit PDO1 and receive PDO1 are active. If you do not configure one (or both) of the two PDOs in the "Configured PDOs", **SyCon** will not deactivate it (or them) due to an internal anomaly. You must deactivate it (or them) via the PL7 program.

Transmit or receive PDOs (set PDO1) are deactivated in the PL7 application by using the SDO service to activate bit 31 of the "PDO COB-ID entry" parameter for the corresponding "Receive/transmit PDO1 parameter" object (see below). This bit activation will mark the PDO as "invalid", so that it will not be exchanged on the CANopen bus.
However, the 31 other bits in COB-ID entry PDO must not be modified.

Index	Subindex	Description	PDO activated	PDO deactivated
16#1400	16#01	COB-ID entry, receive PDO1	16#00000200 + Node-ID	16#80000200 + Node-ID
16#1800	16#01	COB-ID entry, transmit PDO1	16#00000180 + Node-ID	16#80000180 + Node-ID

Example: The PL7 example below deactivates the Altivar transmit PDO1 located in address 4:

```
%MD1000:=16#80000184;(* data to send = disabling of transmit PDO1 via node 4 *)
 %MW500:=16#1800;(* logic address - index in the LSB of %MD500 *)
 %MW501:=16#0001;(* logic address - subindex in the MSB of %MD500 *)
 %MW22:=50;(* time-out = 50 x 10 ms = 500 ms *)
 %MW23:=4;(* data length = 4 bytes *)
 (* SEND write command SDO WRITE *)
 WRITE_VAR(ADR#0.1.SYS,'SDO',%MD500,6,%MW1000:2,%MW20:4);
```

Note:
The Altivar transmit PDO2, receive PDO2, transmit PDO3, and receive PDO3 are inactive by default. If you configure one (or more) of these four PDOs in the "Configured PDOs", **SyCon** will not activate it (or them) due to an internal anomaly. You must activate it (or them) via the PL7 program.

Transmit or receive PDOs (set PDO2 or PDO3) are activated in the PL7 application by using the SDO service to deactivate bit 31 of the "PDO COB-ID entry" parameter for the corresponding "Receive/transmit PDO2/PDO3 parameter" object (see below). This bit deactivation will mark the PDO as "valid" so that it is exchanged on the CANopen bus.
However, the 31 other bits in COB-ID entry PDO must not be modified.

Index	Subindex	Description	PDO deactivated	PDO activated
16#1401	16#01	COB-ID entry, receive PDO2	16#80000300 + Node-ID	16#00000300 + Node-ID
16#1801	16#01	COB-ID entry, transmit PDO2	16#80000280 + Node-ID	16#00000280 + Node-ID
16#1402	16#01	COB-ID entry, receive PDO3	16#80000400 + Node-ID	16#00000400 + Node-ID
16#1802	16#01	COB-ID entry, transmit PDO3	16#80000380 + Node-ID	16#00000380 + Node-ID

Example: The PL7 example below deactivates the Altivar receive PDO2 located in address 6:

```
%MD1100:=16#00000306;(* data to send = activation of receive PDO2 by node 6 *)
%MW600:=16#1401;(* logic address - index in the LSB of %MD600 *)
%MW601:=16#0001;(* logic address - subindex of the MSB of %MD600 *)
%MW32:=50;(* time-out = 50 x 10 ms = 500 ms *)
%MW33:=4;(* data length = 4 bytes *)
(* SEND write command SDO WRITE *)
WRITE_VAR(ADR#0.1.SYS,'SDO',%MD600,6,%MW1100:2,%MW30:4);
```

The CANopen master supports the configuration of a variety of options:

Option	Default value	Possible values
(Task)	**MAST**	**MAST or FAST**
Used to select the type of system task, which will control the CANopen network. **N.B.** The PL7 PRO software application is subdivided into a "Mast Task" and a "Fast Task".		
Bus startup	**Automatic**	**Automatic, semi-automatic or via program**
Behavior of the bus when the CANopen master starts up.		
Inputs	**%MW0 to %MW31**	**(number of %MW) + (first %MW)**
Number of %MW words and index of the first %MW word on the master PLC to which the input data supplied by the TSX CPP 100 CANopen PCMCIA card will be assigned. Please refer to the CANopen master documentation and the master PLC documentation to ascertain the maximum number of words that can be assigned to inputs. It makes no sense to assign too many words. Neither do we recommend assigning the fewest words possible, as the bus configuration may need to be modified to meet future application requirements. The Altivar can use up to 24 bytes (12 words) of input data, but this can only be achieved by using the three "transmit PDOs" from the three PDO sets: up to 8 bytes for transmit PDO1 (by reconfiguring its default assignment), up to 8 bytes for transmit PDO2 (by reconfiguring its assignment) and 8 bytes for transmit PDO3 (static assignment of 4 objects). *Example*: In our previous example, we reduced the number of %MW input words to 6 (12 bytes), as the CANopen inputs corresponding to the Altivar are those of transmit PDO1 by default (ETA and RFRD parameters), plus those of transmit PDO2, which we configured to contain 4 objects (ETI, LFT, LCR, and OPR parameters). In **SyCon**, the "I Len." column in the "Configured PDOs" section indicates the size of the input data (type = IB = Input Bytes) of each transmit PDO configured for a given node. In our case, we have 4 IBs for transmit PDO1 and 8 IBs for transmit PDO2, for a total of 12 input bytes. The first input word remains %MW0; our CANopen input words are, therefore, %MW0 to %MW5. The order in which the transmit PDOs and the objects they contain are configured determines the content of the corresponding %MW input words. In our case, words %MW0 to %MW5 correspond to drive parameters ETA, RFRD, ETI, LFT, LCR, and OPR.		
Outputs	**%MW32 to %MW63**	**(number of %MW) + (first %MW)**
The "Inputs" description above applies equally to the outputs, although in this case, the outputs are master PLC output words and TSX CPP 110 CANopen PCMCIA card output data. The Altivar can use up to 24 bytes (12 words) of output data, but this can only be achieved by using the three "receive PDOs" from the three PDO sets: Up to 8 bytes for receive PDO1 (by reconfiguring its default assignment), up to 8 bytes for receive PDO2 (by reconfiguring its assignment) and 8 bytes for receive PDO3 (static assignment of 4 objects). *Example*: In our example, we reduce the number of %MW input words to 3 (6 bytes), as the CANopen outputs corresponding to the Altivar are those of receive PDO1 by default (CMD and LFRD parameters), plus those of receive PDO2, which we configured to contain just one object (CMI parameter). In **SyCon**, the "O Len." column in the "Configured PDOs" section indicates the size of the output data (type = OB = Output Bytes) of each receive PDO configured for a given node. In our case, we have 4 OBs for receive PDO1 and 2 OBs for receive PDO2, for a total of 6 output bytes. We assign the output words so that they are located just behind the input words: The first output word is %MW6; our CANopen words are, therefore, %MW6 to %MW8. The order in which the receive PDOs and the objects they contain are configured determines the content of the corresponding %MW output words. In our case, words MW6 to %MW8 correspond to drive parameters CMD, LFRD, and CMI.		
Outputs	**Reset**	**Maintain or reset**
Indicates if the CANopen output words are maintained or reset to zero when the associated task (see above) is stopped, as this type of stop does not stop the TSX CPP 110 card.		
Watchdog	**Activated**	**Activated or deactivated**
If this option is activated, the CANopen watchdog on the TSX CPP 110 CANopen PCMCIA card will be triggered as soon as the card experiences difficulty managing the CANopen bus. At the same time, all CANopen output words are reset to 0.		

Option	Default value	Possible values
Loading configuration mode	**PL7**	**PL7 or SyCon**
PL7: The CANopen bus configuration is downloaded to the destination master PLC using the PL7 software application. In the event of insufficient application memory for this configuration, PL7 will disable this mode. **SyCon:** The CANopen bus configuration is considered to have been loaded to the PCMCIA card, supposing, therefore, that it has been downloaded using **SyCon**. PL7 PRO simply checks that the configuration of the card is identical to the content of the "`.co`" file selected, thereby avoiding any configuration inconsistencies. Nevertheless, any bus parameter modifications must be performed in **SyCon**.		

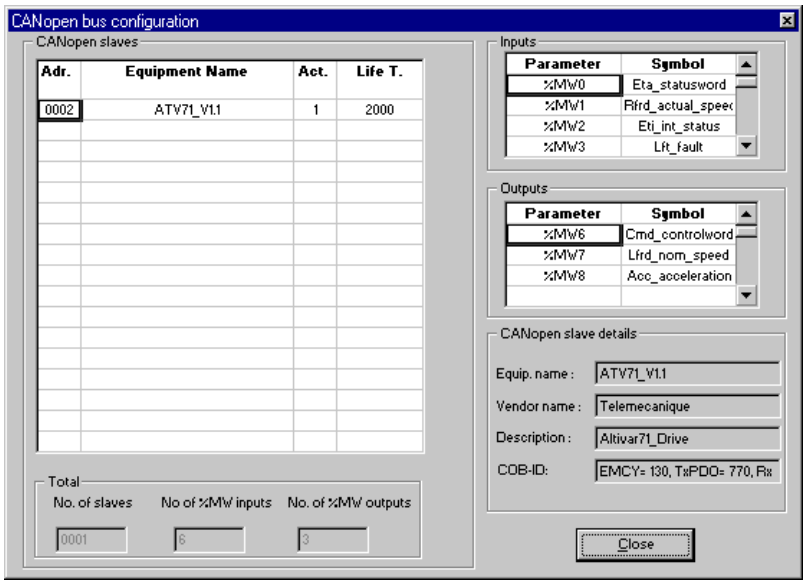
Viewing CANopen master inputs and outputs

PL7 PRO uses the information stored in the "`.co`" file selected to match up the data from each CANopen node directly with its equivalent %MW input and output words.

To view the Altivar inputs/outputs, click the "Bus Configuration" button. The "CANopen bus configuration" window appears (see opposite).

Select the "ATV71_V1.1" CANopen" slave (addr. 0002) to display the input and output words configured for this word only.

In this window, on the right-hand side, explicit symbols have been attributed in advance to words %MW0 to %MW8 in order to illustrate the link between these words and the PDOs configured in **SyCon**.



Note: These assignments are only valid if the Altiva is the only slave on the CANopen but and if the two PDOs in sets PDO1 and PDO2 are used as described in the previous example. Configuring other slaves on the same bus or modifying the configuration of the Altivar PDOs will change the assignment of the input and output words described above. In this case, **SyCon** features a command that can be used to view all inputs and outputs: Select "Table Addresses…" in the "View" menu.

Note:
Please note that **SyCon** displays the **addresses and byte sizes** ("IB" for inputs and "OB" for outputs). Do not forget that these bytes are **aligned to word addresses**. Therefore, an object with 1 byte assigned in an active PDO will in fact take an entire word: The object of 1 byte will be assigned to the MSB byte of this word and the LSB byte will become a "free" byte.

The table below indicates the correspondence between the Altivar PDOs configured and the PLC inputs and outputs:

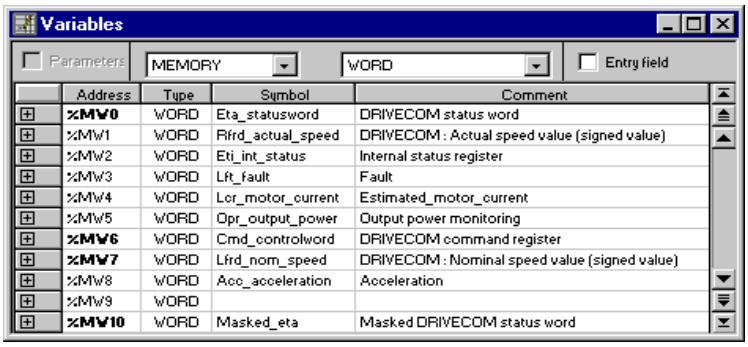
PDO	Type	SyCon I/O	PL7 PRO I/O	Description of the assigned object
Transmit PDO1	Inputs	IB0 - IB1	%MW0	"ETA" status word, default assignment
		IB2 - IB3	%MW1	"RFRD" output speed, default assignment
Transmit PDO2	Inputs	IB4 - IB5	%MW2	"ETI" extended status word configured in the example
		IB6 - IB7	%MW3	"LFT" fault configured in the example
		IB8 - IB9	%MW4	"LCR" estimated motor current configured in the example
		IB10 - IB11	%MW5	"OPR" output power configured in the example
Receive PDO1	Outputs	OB0 - OB1	%MW6	"CMD" control word, default assignment
		OB2 - OB3	%MW7	"LFRD" target speed, default assignment
Receive PDO2	Outputs	OB4 - OB5	%MW8	Acceleration "ACC" configured in the example

Example

This example is designed essentially to:

- Start up the Altivar in accordance with the DSP402 state chart
- Alternate between forward and reverse operation at 1500 rpm during time-out TM0

It uses the following memory objects:



Address	Type	Symbol	Comment
%MW0	WORD	Eta_statusword	DRIVECOM status word
%MW1	WORD	Rfrd_actual_speed	DRIVECOM : Actual speed value (signed value)
%MW2	WORD	Eti_int_status	Internal status register
%MW3	WORD	Lft_fault	Fault
%MW4	WORD	Lcr_motor_current	Estimated_motor_current
%MW5	WORD	Opr_output_power	Output power monitoring
%MW6	WORD	Cmd_controlword	DRIVECOM command register
%MW7	WORD	Lfrd_nom_speed	DRIVECOM : Nominal speed value (signed value)
%MW8	WORD	Acc_acceleration	Acceleration
%MW9	WORD		
%MW10	WORD	Masked_eta	Masked DRIVECOM status word

```
(* Masks the status word *)
%MW9:=%MW0 AND 16#00FF;

(* Status word "ETA" = 16#xx40 = ATV locked *)
IF(%MW9=16#0040)THEN
    %MW6:=16#0006;(* Control word "CMD" = 16#0006 = Shutdown *)
END_IF;

(* Status word "ETA" = 16#xx21/23 = ATV waiting/ATV ready *)
IF(%MW9=16#0021)OR(%MW9=16#0023)THEN
    %MW6:=16#000F;(* Control word "CMD" = 16#000F = Enable operation *)
    %MW7:=1500;(* Speed target "LFRD" = +1500 rpm *)
    START %TM0;(* Starts forward/reverse time-out *)
END_IF;

(* Status word "ETA" = 16#xx27 = ATV running *)
IF(%MW9=16#0027)THEN
    (* Forward/reverse time-out elapsed *)
    IF %TM0.Q THEN
       (* Inversion of direction of rotation by means of inversion of "LFRD" speed target sign*)
       (* Speed target "LFRD" = +/- 1500 rpm *)
       IF %MW7=1500 THEN %MW7:=-1500;
       ELSE %MW7:=1500;
       END_IF;
       (* Checks that the status word "ETA" has not changed --> "Enable operation" command *)
       %MW6:=%MW6 OR 16#000F;
       (* Restarts the forward/reverse time-out *)
       DOWN %TM0;START %TM0;
    END_IF;
END_IF;
```

NMT commands

Master ⇨ Drive

COB-ID	Byte 0	Byte 1
0 (16#000)	Command specifier (CS)	Node-ID (1)

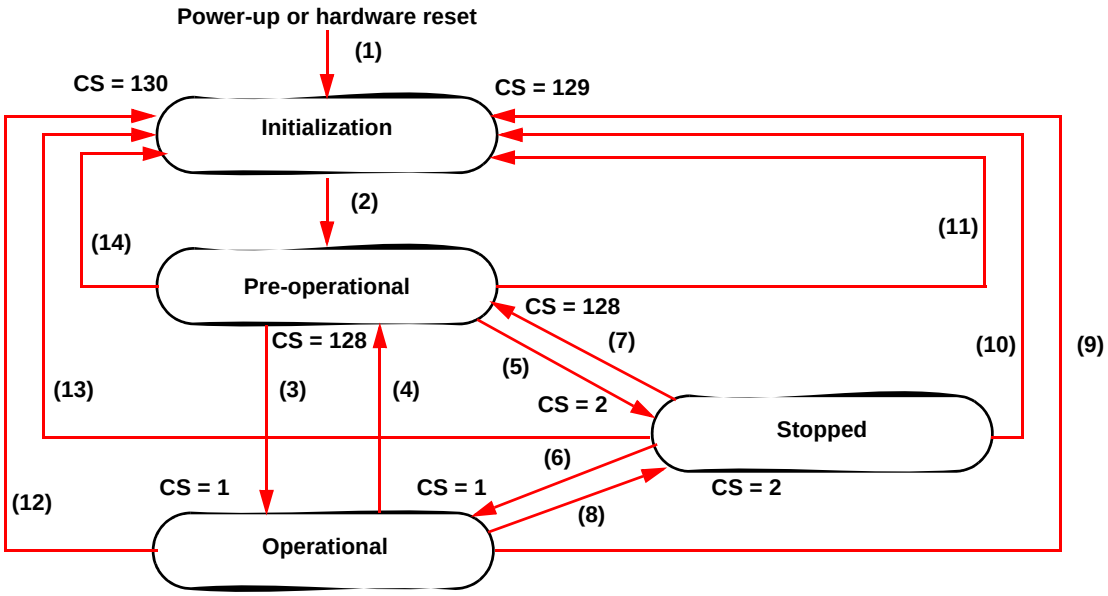
(1) If **Node-ID = 0**, the "command specifier" is broadcast to all CANopen slaves (including the Altivar). Each slave must then execute this NMT command, thereby completing the corresponding transition (see below). Check that your CANopen slaves support broadcast communication on COB-ID 0, which the Altivar does (see page 26).

Command specifier (CS)	Meaning
1 (16#01)	Start_Remote_Node
2 (16#02)	Stop_Remote_Node
128 (16#80)	Enter_Pre-Operational_State
129 (16#81)	Reset_Node
130 (16#82)	Reset_Communication

Example: Transition to pre-operational state (Enter_Pre-Operational_State = 16#80) of the Altivar located at CANopen address 4 (16#04).

16#000	16#80	16#04

CANopen NMT state chart



Transition	Description
(1)	On power-up, the node automatically changes to the initialization state.
(2)	Once initialization is complete, the pre-operational state is activated automatically.
(3), (6)	Start_Remote_Node
(4), (7)	Enter_Pre-Operational_State
(5), (8)	Stop_Remote_Node
(9), (10), (11)	Reset_Node
(12), (13), (14)	Reset_Communication

Depending on the communication status of the drive, the following services are available:

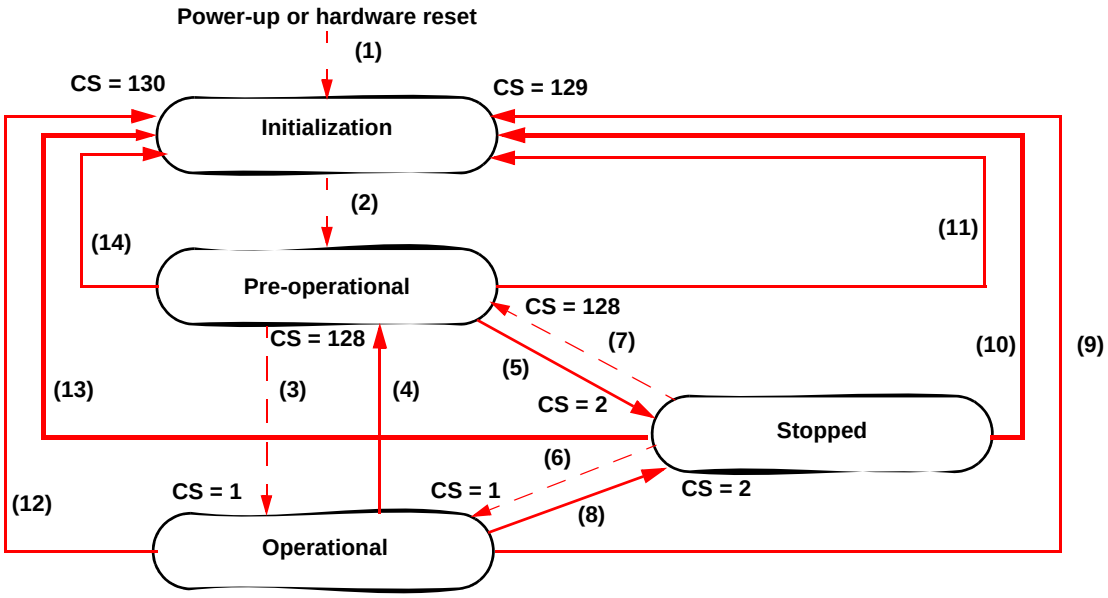
	Initialization	Pre-operational	Operational	Stopped
PDO			X	
SDO		X	X	
Synchronization (SYNC)		X	X	
Emergency (EMCY)		X	X	
Bootup service	X		X	
Network management (NMT)		X	X	X

In the event of a resettable fault, the drive must be in the "Operational" NMT state in order that the PDO carrying the control word CMD (receive PDO1, preferably) can reset the drive using the CMD "reset fault" bit. The CANopen "master" will then have to set the drive to the "Operational" state using an NMT telegram, with CS – 1 (Start_Remote_Node) and Node-ID = drive node address.

Note: The drive NMT state is transmitted by the drive if the Node Guarding protocol has been activated (see page 54). It is also available via the **[Canopen NMT state]** (**NMTS**) parameter (logic address 6057, CANopen index 16#201E/3A).

NMT state chart fault:

Some transitions in the NMT state chart will trigger a CANopen communication fault. These transitions are listed in the table and illustrated in the chart below (solid lines with arrows).



Transition	Description
(4)	Enter_Pre-Operational_State
(5), (8)	Stop_Remote_Node
(9), (10), (11)	Reset_Node
(12), (13), (14)	Reset_Communication

These transitions suppress a service, which can be used to control the drive. A communication fault must be triggered in order to avoid losing control of the drive.

Transition	Service lost
(4)	PDO
(5)	SDO
(8), (9), (10), (11), (12), (13), (14)	PDO and SDO

Detailed description of services

Bootup service

Master ⇦ Drive

COB-ID	Byte 0
1792 (16#700) +Node-ID	16#00

This service is used to indicate that the drive has changed to the "pre-operational" state following the "initialization" state (page 41).

The only data byte sent in a Bootup frame is always equal to 16#00.

Synchronization object - SYNC

Master ⇨ Drive

COB-ID
128 (16#080)

The SYNC object is sent cyclically by the CANopen master.
It does not contain data and its frame is, therefore, limited to its unique COB-ID identifier.

The purpose of this object is essentially to authorize synchronous communication modes for CANopen slaves. In the case of the Altivar, none of the PDOs used can be defined in cyclic or acyclic synchronous communication mode.

Emergency object - EMCY

Master ⇦ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
128 (16#080) +NODE-ID	Fault code (**Errd**)		Error register	0	0	0	0	0
	LSB	MSB	Bit 0 = 0 (no fault) or 1 (fault)					

An EMCY object is sent by the Altivar to other CANopen devices, with a high priority, every time a fault appears (byte 2/bit 0 = 1) or disappears (byte 2/bit 0 = 0). This is the case in particular for "Heartbeat" or "Life Guard" type faults. An EMCY object is never repeated.

The error code parameter (**Errd**) (logic address = 8606, CANopen index = 16#603F) is described in the parameters manual.

PDO1

Default assignment

Transmit TPDO1

Master ⇦ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3
384 (16#180) +Node-ID	Status word "ETA"		Output speed "RFRD"	
	LSB	MSB	LSB	MSB

Example: The Altivar located at CANopen address 4 (COB-ID = 16#180 + 4) is in the "Operation enabled" state and is not faulty (status word "ETA" = 16#xxx7). In our example, the status word "ETA" is equal to 16#0607.
Furthermore, the speed of the motor is equal to 1500 rpm (16#05DC).

16#184	16#07	16#06	16#DC	16#05

Receive RPDO1

Master ⇨ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3
512 16#200 +Node-ID	Control word "CMD"		Speed target "LFRD"	
	LSB	MSB	LSB	MSB

Example: The Altivar located at CANopen address 4 (COB-ID = 16#200 + 4) receives the command called "Enable operation" (control word "CMD" = 16#xxxF). In our example, the control word "CMD" is equal to 16#000F.
Furthermore, the speed of the motor is equal to 1200 rpm (16#04B0).

16#204	16#0F	16#00	16#B0	16#04

Extended default assignment

Transmit TPDO1

Master ⇦ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
384 (16#180) + Node-ID	Status word "ETA"		Output speed "RFRD"		Motor torque	
	LSB	MSB	LSB	MSB	LSB	MSB

Example: The Altivar located at CANopen address 4 (COB-ID = 16#180 + 4) is in the "Operation enabled" state and is not faulty (status word "ETA" = 16#xxx7). In our example, the status word "ETA" is equal to 16#0607, the speed of the motor is equal to 1500 rpm (16#05DC) and the motor torque is equal to 50% (500 x 0.1% = 16#01F4).

16#184	16#07	16#06	16#DC	16#05	16#F4	16#01

Receive RPDO1

Master ⇨ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
512 (16#200) + Node-ID	Control word "CMD"		Speed target "LFRD"		Torque target "LTR"	
	LSB	MSB	LSB	MSB	LSB	MSB

Example: The Altivar located at CANopen address 4 (COB-ID = 16#200 + 4) receives the command called "Enable operation" (control word "CMD" = 16#xxxF). In our example, the control word "CMD" is equal to 16#000F, the speed target of the motor is equal to 1200 rpm (16#04B0) and its torque target is equal to 50% (500 x 0.1% = 16#01F4).

16#204	16#0F	16#00	16#B0	16#04	16#F4	16#01

Restricted default assignment

Transmit TPDO1

Master ⇦ Drive

COB-ID	Byte 0	Byte 1
384 (16#180) + Node-ID	Status word "ETA"	
	LSB	MSB

Receive RPDO1

Master ⇨ Drive

COB-ID	Byte 0	Byte 1
512 (16#200) + Node-ID	Control word "CMD"	
	LSB	MSB

User assignment

Transmit TPDO1

Master ⇦ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
384 (16#180) +Node-ID	Altivar variable (by default: status word "ETA")		Altivar variable (by default: output speed "RFRD")		Altivar variable (by default: no parameter)		Altivar variable (by default: no parameter)	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

Note: Any bytes left unused at the end of this PDO will not be transmitted on the bus by the Altivar. For example, if no parameters have been assigned to bytes 6 and 7, the data length of transmit PDO1 will be 6 bytes.

Example: The two default assignment parameters are maintained (16#6041/00 and 16#6044/00) and are suffixed (i.e., in bytes 4 and 5) with the "Torque Actual Value" motor torque parameter (16#6077/00), thereby producing the *extended default assignment* of transmit PDO1.
This extended default assignment then becomes a user assignment by means of the suffixing (i.e., in bytes 6 and 7) of the parameters of the current in the motor object "LCR" (16#2002/05), thereby producing the following *user assignment*:

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
384 (16#180) +Node-ID	Status word "ETA"		Output speed "RFRD"		Motor torque "OTR"		Current in the motor "LCR"	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

Consider an Altivar located at CANopen address 4 (COB-ID = 16#180 + 4) and in the following state:

- Current state = "Operation enabled" and no faults (status word "ETA" = 16#xxx7). In our example, the status word "ETA" is equal to 16#0607.
- The output speed "RFRD" is equal to 1500 rpm (16#05DC).
- The motor torque is equal to 83% (830 x 0.1% = 16#033E).
- The current in the motor "LCR" is equal to 4.0 A (16#0028).

The corresponding telegram sent for this transmit PDO is therefore as follows (8 data bytes):

16#184	16#07	16#06	16#DC	16#05	16#3E	16#03	16#28	16#00

Receive RPDO1

Master ⇨ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
512 (16#200) +Node-ID	Altivar variable (by default: control word "CMD")		Altivar variable (by default: speed target "LFRD")		Altivar variable (by default: no parameter)		Altivar variable (by default: no parameter)	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

Example: The first default assignment object is maintained (16#6040/00) but the second is deleted (16#6042/00), thereby producing the *restricted default assignment* of receive PDO1.
This *restricted default assignment* then becomes a *user assignment* by means of the suffixing of the Acceleration ramp time "ACC"/ "Acceleration" (16#203C/02) and Deceleration ramp time "DEC"/"Deceleration" (16#203C/03) parameters in bytes 2 to 5, thereby producing the following *user assignment*:

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
512 (16#200) +Node-ID	Control word "CMD"		Acceleration ramp time "ACC"		Deceleration ramp time "DEC"	
	LSB	MSB	LSB	MSB	LSB	MSB

Consider an Altivar located at CANopen address 4 (COB-ID = 16#200 + 4) and controlled as follows:
- "Enable operation" command (control word "CMD" = 16#xxxF). In our example, the control word "CMD" is equal to 16#000F.
- The acceleration ramp time "ACC" is 1 s (10 = 16#000A).
- The deceleration ramp time "DEC" is 2 s (20 = 16#0014).

The corresponding telegram received for this receive PDO is therefore as follows (6 data bytes):

16#204

16#0F	16#00	16#0A	16#00	16#14	16#00

PDO2

Set PDO2 on the Altivar is disabled by default ("PDO COB-ID entry" parameter = 16#80000XXX).

To enable it, use the SDO service in write mode to set bit 31 of the "PDO COB-ID entry" parameter in transmit PDO2 (16#1801/01) and/or receive PDO2 (16#1401/01) to zero.

Unlike set PDO1, it is also possible to modify bits 0 to 6 of the COB-IDs in set PDO2 in order to authorize slave-to-slave communication.

Transmit TPDO2

Master ⇦ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
640 (16#280) +Node-ID	Altivar variable (by default: no parameter)		Altivar variable (by default: no parameter)		Altivar variable (by default: no parameter)		Altivar variable (by default: no parameter)	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

Note: Any bytes left unused at the end of this PDO will not be transmitted on the bus by the Altivar. For example, if no parameters have been assigned to bytes 6 and 7, the data length of transmit PDO2 will be 6 bytes.

Example: Assignment of 3 parameters: Last fault occurred "LFT" (16#2029/16), Current in the motor "LCR" (16#2002/05) and Motor power "OPR" (16#2002/0C). This produces the following user assignment:

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
640 (16#280) +Node-ID	Last fault occurred "LFT"		Current in the motor "LCR"		Motor power "OPR"	
	LSB	MSB	LSB	MSB	LSB	MSB

Consider an Altivar located at CANopen address 4 (COB-ID = 16#280 + 4) and in the following state:

- The last fault occurred "LFT" is "nOF"/no fault stored (16#0000).
- The current in the motor "LCR" is equal to 4.0 A (16#0028).
- The motor power "OPR" is equal to 50% (16#0032).

The corresponding telegram sent for this transmit PDO is therefore as follows (8 data bytes):

16#284	16#00	16#00	16#28	16#00	16#32	16#00

Receive RPDO2

Master ⇨ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
768 (16#300) +Node-ID	Altivar variable (by default: no parameter)		Altivar variable (by default: no parameter)		Altivar variable (by default: no parameter)		Altivar variable (by default: no parameter)	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

PDO3

This PDO set is reserved for the Altivar "Controller Inside" programmable card and must only be used on a drive equipped with this card.

Set PDO3 on the Altivar is disabled by default (bit 31 of PDO COB-ID entry = 1).

To enable it, use the SDO service in write mode to set bit 31 of the "PDO COB-ID entry" parameter in transmit PDO3 (16#1802/01) and/or receive PDO3 (16#1402/01) to zero.

Unlike set PDO2, it is also possible to modify bits 0 to 6 of the COB-IDs in set PDO3 in order to authorize slave-to-slave communication.

Unlike the assignment of PDO1 and PDO2 objects, the assignment of PDO3 objects cannot be modified.

Transmit TPDO3

Master ⇦ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
896 (16#380) +Node-ID	Output word TP31 on the "Controller Inside" programmable card		Output word TP32 on the "Controller Inside" programmable card		Output word TP33 on the "Controller Inside" programmable card		Output word TP34 on the "Controller Inside" programmable card	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

Receive RPDO3

Master ⇨ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
1024 (16#400) +Node-ID	Input word RP31 on the "Controller Inside" programmable card		Input word RP32 on the "Controller Inside" programmable card		Input word RP33 on the "Controller Inside" programmable card		Input word RP34 on the "Controller Inside" programmable card	
	LSB	MSB	LSB	MSB	LSB	MSB	LSB	MSB

SDO service

Request: Master ⇨ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
1536 (16#600) +Node-ID	Request code	Object index		Object subindex	Request data			
		LSB	MSB		Bits 7-0	Bits 15-8	Bits 23-16	Bits 31-24

Response: Master ⇦ Drive

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
1408 (16#580) +Node-ID	Response code	Object index		Object subindex	Request data			
		LSB	MSB		Bits 7-0	Bits 15-8	Bits 23-16	Bits 31-24

The content of the "request data" and "response data" will vary depending on the "request code" and the "response code". The two tables below indicate the various possible scenarios:

Request code	Description of the command	Byte 4	Byte 5	Byte 6	Byte 7
16#23	Write data 4 bytes in length (e.g., UNSIGNED 32)	Bits 7-0	Bits 15-8	Bits 23-16	Bits 31-24
16#2B	Write data 2 bytes in length (e.g., UNSIGNED 16)	Bits 7-0	Bits 15-8	16#00	16#00
16#2F	Write data 1 byte in length (e.g., UNSIGNED 8)	Bits 7-0	16#00	16#00	16#00
16#40	Read data 1/2/4 bytes in length	16#00	16#00	16#00	16#00
16#80	Cancel current SDO command (1)	16#00	16#00	16#00	16#00

Response code	Description of the response	Byte 4	Byte 5	Byte 6	Byte 7
16#43	Read data: 4 bytes of data: response (1)	Bits 7-0	Bits 15-8	Bits 23-16	Bits 31-24
16#4B	Read data: 2 bytes of data: response (1)	Bits 7-0	Bits 15-8	16#00	16#00
16#4F	Read data: 1 byte of data: response (1)	Bits 7-0	16#00	16#00	16#00
16#60	Write data 1/2/4 bytes in length: response	16#00	16#00	16#00	16#00
16#80	Error sending cancellation code (2)	Bits 7-0	Bits 15-8	Bits 23-16	Bits 31-24

(1) To use an SDO service reserved for reading multi-byte data, such as "manufacturer name" (parameter 16#1008: 16#00), *segmented transfer* is triggered between the master and the drive.
The "request code" 16#80 is used to stop this type of transfer.

(2) The response data (bytes 4 to 7) correspond to a 32-bit "cancellation code" (a complete list of all cancellation codes supported by the Altivar appears in the table on the next page).

Note: Segmented transfer can only be used for items of data larger than 4 bytes. It is only used for "Device name" (object 16#1008) in conjunction with the Altivar.

Cancellation code (1)	Description
16# 0503 0000	*Segmented transfer:* The toggle bit has not been modified.
16# 0504 0001	"Request code" invalid or unknown
16# 0601 0000	Parameter access error (e.g., write request on "read-only" parameter)
16# 0601 0002	Attempt to execute a write request on a "read-only" parameter
16# 0602 0000	The "index" sent in the request refers to an object, which does not exist in the object dictionary
16# 0604 0041	PDO object assignment: The parameter cannot be assigned to the PDO; this error occurs when writing to parameters 16#1600, 16#1601, 16#1602, 16#1A00, 16#1A01, and16#1A02 (assignments of PDO1, 2 and 3)
16# 0604 0042	PDO object assignment: The number and/or length of the parameters to be assigned exceeds the maximum PDO length.
16# 0609 0011	The "subindex" sent in the request does not exist
16# 0609 0030	Outside parameter value limits (for a write request only)
16# 0609 0031	Value of parameter written too high
16# 0800 0000	A general error has occurred

(1) Please note that the "cancellation codes" listed in this table have been written in accordance with general convention and must, therefore, be inverted in the case of byte-by-byte representation for "bytes 4 to 7" (e.g., 16# 0609 0030 becomes byte 4 = 16#30, byte 5 = 16#00, byte 6 = 16#09, byte 7 = 16#06).

Important notes about the SDO service

Do not try to use SDO write requests to a parameter that has been assigned in an RPDO (receive).

Example: If the speed target (rpm) is assigned in RPDO1, there is no point using an SDO to write this target.

Any parameter connected to one of the parameters configured in an RPDO (receive) must not be modified using an SDO write request.

Example: If the speed target in (rpm) is assigned in RPDO1, there is no point using an SDO to modify the frequency target (0.1 Hz).

Example of a read operation using the SDO service

This example explains how to read the "acceleration ramp time (ACC)" parameter on an Altivar located at CANopen address 4 (COB-ID = 16#580 + Node-ID *or* 16#600 + Node-ID). The "index/subindex" value of this parameter is equal to 16#203C/02.

Read request: Master ⇨ Drive

16#604	16#40	16#3C	16#20	16#02	16#00	16#00	16#00	16#00

Read response: Master ⇦ Drive

16#584	16#4B	16#3C	16#20	16#02	16#E8	16#03	16#00	16#00

The value of the parameter read is equal to 1000 (16#03E8), equivalent to an "acceleration ramp time (ACC)" of 100 s, as the unit of this parameter is "0.1 s".

Example of a write operation using the SDO service

This example explains how to write the value 100 s to the "acceleration ramp time (ACC)" parameter on an Altivar located at CANopen address 4 (COB-ID = 16#580 + Node-ID *or* 16#600 + Node-ID). The "index/subindex" value of this parameter is equal to 16#203C/02.

Write request: Master ⇨ Drive
The "request code" is 16#2B as we are trying to modify the value of an item of data 2 bytes in length.

16#604	16#2B	16#3C	16#20	16#02	16#E8	16#03	16#00	16#00

The "Request data" field indicates that the value we are trying to assign to the parameter is equal to 1000 (16#03E8), equivalent to an "acceleration ramp time (ACC)" of 100 s, as the unit of this parameter is "0.1 s".

Write response: Master ⇦ Drive

16#584	16#60	16#3C	16#20	16#02	16#00	16#00	16#00	16#00

Node Guarding service

Description

Either the Node Guarding service described here or the Heartbeat service described below can be used for communication monitoring. Only one of these two services can be active at any one time.

The Node Guarding service is deactivated by default on the Altivar.

Master ⇨ Drive
The master scans the drive at regular intervals ("Life Time") by sending "remote transmit requests" (RTR). The "Life Time" is calculated by multiplying the "Guard Time" by the "Life Time Factor". These two parameters are described on page 57.

If, once the "Life Time" has expired, the drive has not received the RTR:
- It triggers a "Life Guarding" fault (see section "Configuring communication fault management")
- and sends an emergency telegram (EMCY) (see page 43).

Master ⇦ Drive

COB-ID	Byte 0
1792 (16#700) +Node-ID	NMT information

The drive response indicates it NMT state via the "NMT information" field, described here:
Bit 7 = Toggle bit: The value of this bit must alternate from one drive response to the other. The value of the toggle bit for the first response following activation of the Node Guarding service is 0. This bit can only be reset to 0 by sending the "Reset_Communication" command to the drive (see section Detailed description of services, page 41). If a response is received with the same toggle bit value as the previous response, the new response is treated as if it had not been received.

Bits 6-0 = NMT state: Current NMT state of the Altivar: Initialization (16#00), Stopped (16#04), Operational (16#05) or Pre-operational (16#7F). The **[Canopen NMT state]** (**NMTS**) parameter (16#201E/3A) can be accessed via the drive display terminal (see section "CANopen communication diagnostics", page 17) and via the Power Suite software workshop.

If the drive does not send a response or if it sends an incorrect state, the master will trigger a "Node Guarding" event.

Example configuration for the Node Guarding service

The Altivar "Life Time" can be modified using the SDO service in order to write new values for the "Guard Time" and "Life Time Factor" parameters.

Parameter	Index	Subindex	Format	Unit
Guard Time	16# 100C	16# 00	16-bit unsigned integer	1 ms
Life Time Factor	16# 100D	16# 00	Unsigned byte	__

In our example, we are going to configure a Life Time of 2 seconds, with a Guard Time of 500 ms and a Life Time Factor of 4 (500 ms × 4 = 2 s).

1) Setting the "Guard Time" parameter to 500 ms
- COB-ID = 16#600 + Node-ID for the write request or 16#580 + Node-ID for the write response
- Request code (byte 0) = 16#2B to write data 2 bytes in length
- Response code (byte 0) = 16#60 if the write operation has been completed without errors
- Object index (bytes 1 and 2) = 16#100C
- Object subindex (byte 3) = 16#00
- Request data (bytes 4 and 5) = 16#01F4 (500)

Request: Master ⇨ Drive

16#604	16#2B	16#0C	16#10	16#00	16#F4	16#01	16#00	16#00

Response: Master ⇦ Drive

16#584	16#60	16#0C	16#10	16#00	16#00	16#00	16#00	16#00

2) Setting the "Life Time Factor" parameter to 4
- COB-ID = 16#600 + Node-ID for the write request or 16#580 + Node-ID for the write response
- Request code (byte 0) = 16#2F to write data 1 byte in length
- Response code (byte 0) = 16#60 if the write operation has been completed without errors
- Object index (bytes 1 and 2) = 16#100D
- Object subindex (byte 3) = 16#00
- Request data (byte 4) = 16#04 (4)

Request: Master ⇨ Drive

16#604	16#2F	16#0D	16#10	16#00	16#04	16#00	16#00	16#00

Response: Master ⇦ Drive

16#584	16#60	16#0D	16#10	16#00	16#00	16#00	16#00	16#00

Heartbeat service

Description

If you do not activate the Node Guarding service described in the previous section, you can use the Heartbeat service to monitoring communication with another node, provided that it supports this service.

The Heartbeat service is deactivated by default on the Altivar (the "Consumer Heartbeat Time" and "Producer Heartbeat Time" parameters are both set to 0).

Producer ⇨ Consumer

COB-ID	Byte 0
1792 (16#700) +Node-ID	Heartbeat Producer state

Each "Heartbeat Producer" sends Heartbeat messages at regular intervals (the "Producer Heartbeat Time" 16#1017/00).
All "Heartbeat Consumers" check that they receive these messages in a time less than the "Consumer Heartbeat Time" (16#1016/01).
The "Producer Heartbeat Time" must be less than the "Consumer Heartbeat Time".

If the drive has been configured as a consumer and a period of time equal to the "Consumer Heartbeat Time" elapses without a "Heartbeat message" being received, the drive will trigger a "Heartbeat" event and send an emergency telegram (EMCY).
If CANopen is the active channel, a CANopen fault (COF) will be triggered.

The "Heartbeat message" sent from the drive contains a "Heartbeat Producer state" field (byte 0), described here:
Bit 7 = Reserved: This bit is always equal to 0.
Bits 6-0 = Heartbeat Producer state: Current NMT state of the Altivar: Initialization (16#00), Stopped (16#04), Operational (16#05) or Pre-operational (16#7F).

Object dictionary

Index	Object
16#0000	Not used
16#0001 - 16#001F	Static data
16#0020 - 16#003F	Complex data
16#0040 - 16#005F	Not used (manufacturer-specific complex data)
16#0060 - 16#007F	Device-profile-specific static data
16#0080 - 16#009F	Device-profile-specific complex data
16#00A0 - 16#0FFF	Reserved
16#1000 - 16#1FFF	Communication profile zone
16#2000 - 16#5FFF	Altivar-specific profile zone
16#6000 - 16#9FFF	Standardized device profile zone (DSP402)
16#A000 - 16#FFFF	Reserved

Communication profile zone objects

Access: "R" designates an object that supports read access only via the SDO service, whereas "R/W" designates an object that supports both read and write access. Some "R/W" objects only support write access in certain states of the NMT CANopen chart.

Index	Sub-index	Access	Type	Default value	Description
16#1000	16#00	R	unsigned32	16#00010192	Type of device: Bits 24-31 not used (0) Bits 16-23 = Type of device (1) Bits 0-15 = Device profile number (402)
16#1001	16#00	R	unsigned8	16#00	**Error register:** Error (bit 0 = 1) or no error (bit 0 = 0)
16#1003	16#00	R	unsigned8	16#01	**Number of errors:** Only one possible error (1), located in object 16#1003/01
	16#01	R	unsigned32	16#00000000	Standard error field: Bits 16-31 = Additional information (always 0) Bits 0-15 = Error code parameter (**Errd**)
16#1005	16#00	R/W	unsigned32	16#00000080	COB-ID entry for SYNC message
16#1008	16#00	R	string	ATV71	**Device name:** The SDO service segmented transfer is used to read this object.
16#100B	16#00	R	unsigned32		**Node-ID:** This object receives the value of the CANopen address configured on the Altivar.
16#100C	16#00	R/W	unsigned16	16#0000	**Guard Time:** By default, the Node Guarding protocol is deactivated (0); the unit for this object is 1 ms. If you use this protocol (Guard Time > 0), make sure that the Heartbeat protocol is deactivated on the Altivar (Producer Heartbeat Time = 0).
16#100D	16#00	R/W	unsigned8	16#00	**Life Time Factor:** Multiplier coefficient applied to the Guard Time to obtain the Life Time. The value 0 deactivates the Node Guarding service in respect of the drive.
16#100E	16#00	R	unsigned32	16#00000700 +Node-ID	**Node Guarding identifier:** COB-ID entry used for the Node Guarding protocol and managed by the configurator (**SyCon**)
16#100F	16#00	R	unsigned32	16#00000001	**Number of SDOs supported**
16#1014	16#00	R	unsigned32	16#00000080 +Node-ID	**COB-ID entry for emergency message** (**EMCY**)
16#1016	16#00	R	unsigned8	16#01	**Consumer Heartbeat Time:** Number of objects
	16#01	R/W	unsigned32	16#00000000	**Consumer Heartbeat Time:** Bits 24-31 not used (0) Bits 16-23 = Node-ID of Heartbeat Producer Bits 0-15 = Max. duration of Consumer Heartbeat (unit = 1 ms) **Note:** A single Heartbeat Producer can be configured here. By default, no producers are monitored (value = 0).
16#1017	16#00	R/W	unsigned16	16#0000	**Producer Heartbeat Time:** By default, the Altivar does not produce Heartbeat messages; the unit of this object is 1 ms. If you use this protocol (Producer Heartbeat Time > 0), make sure that the Node Guarding protocol is deactivated on the Altivar (Guard Time = 0).

Object dictionary

Index	Sub-index	Access	Type	Default value	Description
16#1018	16#00	R	unsigned8	16#03	**ID object: Number of objects**
	16#01	R	unsigned32	16#0200005A	**ID object: Supplier ID** This value is manufacturer-specific.
	16#02	R	unsigned32	71	**ID object: Product identification code**
	16#03	R	unsigned32	Example: 16#00010002 for "v1.2"	**ID object: Product version** Bits 16-31 = Primary version ID Bits 0-15 = Secondary version ID
16#1400	16#00	R	unsigned8	16#02	**Receive PDO1: Number of objects**
	16#01	R/W	unsigned32	16#00000200 +Node-ID	**Receive PDO1: COB-ID entry** Only bit 31 can be accessed in write mode: PDO activated (0) or PDO disabled (1).
	16#02	R/W	unsigned8	16#FF	**Receive PDO1: Transmission method** "asynchronous" (254 or 255), "cyclic synchronous" (1-240).
16#1401	16#00	R	unsigned8	16#02	**Receive PDO2: Number of objects**
	16#01	R/W	unsigned32	16#80000300 +Node-ID	**Receive PDO2: COB-ID entry** Bit 31 can be accessed in write mode: PDO disabled (1) or PDO active (0). Bits 0-10 can be accessed in write mode to enable slave-to-slave communication.
	16#02	R/W	unsigned8	16#FF	**Receive PDO2: Transmission speed** "asynchronous" (254 or 255), "cyclic synchronous" (1-240).
16#1402	16#00	R	unsigned8	16#02	**Receive PDO3: Number of objects**
	16#01	R/W	unsigned32	16#80000400 +Node-ID	**Receive PDO3: COB-ID entry** Bit 31 can be accessed in write mode: PDO disabled (1) or PDO active (0). Bits 0-10 can be accessed in write mode to enable slave-to-slave communication.
	16#02	R/W	unsigned8	16#FF	**Receive PDO3: Transmission speed** "asynchronous" (254 or 255), "cyclic synchronous" (1-240).
16#1600	16#00	R/W	unsigned8	16#02	**Receive PDO1 assignment: Number of objects assigned** Two objects are assigned by default in receive PDO1, although between 0 and 4 objects can be assigned.
	16#01	R/W	unsigned32	16#60400010	**Receive PDO1 assignment: 1st object assigned** Control word "CMD" (16#6040/00)
	16#02	R/W	unsigned32	16#60420010	**Receive PDO1 assignment: 2nd object assigned** Speed target "LFRD" (16#6042/00)
	16#03	R/W	unsigned32	16#00000000	**Receive PDO1 assignment: 3rd object assigned** No 3rd object assigned
	16#04	R/W	unsigned32	16#00000000	**Receive PDO1 assignment: 4th object assigned** No 4th object assigned

Object dictionary

Index	Sub-index	Access	Type	Default value	Description
16#1601	16#00	R/W	unsigned8	16#00	**Receive PDO2 assignment: Number of objects assigned** No objects are assigned by default in receive PDO2, although between 0 and 4 objects can be assigned.
	16#01	R/W	unsigned32	16#00000000	**Receive PDO2 assignment: 1st object assigned** No 1st object assigned
	16#02	R/W	unsigned32	16#00000000	**Receive PDO2 assignment: 2nd object assigned** No 2nd object assigned
	16#03	R/W	unsigned32	16#00000000	**Receive PDO2 assignment: 3rd object assigned** No 3rd object assigned
	16#04	R/W	unsigned32	16#00000000	**Receive PDO2 assignment: 4th object assigned** No 4th object assigned
16#1602	16#00	R/W	unsigned8	16#04	**Receive PDO3 assignment: Number of objects assigned** 4 objects are assigned in receive PDO3 by default, although this number can be reduced to 0.
	16#01	R	unsigned32	16#20640210	**Receive PDO3 assignment: 1st object assigned** Input word RP31 on the "Controller Inside" card
	16#02	R	unsigned32	16#20640310	**Receive PDO3 assignment: 2nd object assigned** Input word RP32 on the "Controller Inside" card
	16#03	R	unsigned32	16#20640410	**Receive PDO3 assignment: 3rd object assigned** Input word RP33 on the "Controller Inside" card
	16#04	R	unsigned32	16#20640510	**Receive PDO3 assignment: 4th object assigned** Input word RP34 on the "Controller Inside" card
16#1800	16#00	R	unsigned8	16#05	**Transmit PDO1: Number of objects**
	16#01	R/W	unsigned32	16#00000180 +Node-ID	**Transmit PDO1: COB-ID entry** Only bit 31 can be accessed in write mode: PDO active (0) or PDO disabled (1)
	16#02	R/W	unsigned8	16#FF	**Transmit PDO1: Transmission speed** Choice of 3 modes for this PDO: "asynchronous" (254 or 255), "cyclic synchronous" (1-240) and "acyclic synchronous" (0). The values 252 and 253 (modes on receipt of RTR frames) are not supported by the Altivar.
	16#03	R/W	unsigned16	300	**Transmit PDO1: Inhibit Time** Minimum time between two transmissions; unit = 100 µs; minimum value = 100 (10 ms).
	16#05	R/W	unsigned16	1000	**Transmit PDO1: Event Timer** In "asynchronous" mode, this object defines a minimum transmission frequency for this PDO; unit = 1 ms; minimum value = 10 (10 ms). The duration of this event timer must be greater than that of the inhibit timer (subindex: 16#03). Asynchronous PDO transmission will therefore take place when the data to be transmitted change, with these two periods as temporal limits.

Index	Sub-index	Access	Type	Default value	Description
16#1801	16#00	R	unsigned8	16#05	**Transmit PDO2: Number of objects**
	16#01	R/W	unsigned32	16#80000280 +Node-ID	**Transmit PDO2: COB-ID entry** Bit 31 can be accessed in write mode: PDO disabled (1) or PDO active (0). Bits 0-10 can be accessed in write mode to enable slave-to-slave communication.
	16#02	R/W	unsigned8	16#FF	**Transmit PDO2: Transmission speed** Choice of 3 modes for this PDO: "asynchronous" (254 or 255), "cyclic synchronous" (1-240) and "acyclic synchronous" (0). The values 252 and 253 (modes on receipt of RTR frames) are not supported by the Altivar.
	16#03	R/W	unsigned16	300	**Transmit PDO2: Inhibit Time** Minimum time between two transmissions; unit = 100 µs; minimum value = 100 (10 ms).
	16#05	R/W	unsigned16	1000	**Transmit PDO2: Event Timer** In "asynchronous" mode, this object defines a minimum transmission frequency for this PDO; unit = 1 ms; minimum value = 10 (10 ms). The duration of this event timer must be greater than that of the inhibit timer (subindex: 16#03). Asynchronous PDO transmission will therefore take place when the data to be transmitted change, with these two periods as temporal limits.
16#1802	16#00	R	unsigned8	16#05	**Transmit PDO3: Number of objects**
	16#01	R/W	unsigned32	16#80000380 +Node-ID	**Transmit PDO3: COB-ID entry** Bit 31 can be accessed in write mode: PDO disabled (1) or PDO active (0). Bits 0-10 can be accessed in write mode to enable slave-to-slave communication.
	16#02	R/W	unsigned8	16#FF	**Transmit PDO3: Transmission speed** Choice of 3 modes for this PDO: "asynchronous" (254 or 255), "cyclic synchronous" (1-240) and "acyclic synchronous" (0). The values 252 and 253 (modes on receipt of RTR frames) are not supported by the Altivar.
	16#03	R/W	unsigned16	300	**Transmit PDO3: Inhibit Time** Minimum time between two transmissions; unit = 100 µs; minimum value = 100 (10 ms).
	16#05	R/W	unsigned16	1000	**Transmit PDO3: Event Timer** In "asynchronous" mode, this object defines a minimum transmission frequency for this PDO; unit = 1 ms; minimum value = 10 (10 ms). The duration of this event timer must be greater than that of the inhibit timer (subindex: 16#03). Asynchronous PDO transmission will therefore take place when the data to be transmitted change, with these two periods as temporal limits.
16#1A00	16#00	R/W	unsigned8	16#02	**Transmit PDO1 assignment: Number of objects assigned** Two objects are assigned by default in transmit PDO1, although between 0 and 4 objects can be assigned.
	16#01	R/W	unsigned32	16#60410010	**Transmit PDO1 assignment: 1st object assigned** Default status word "ETA" (16#6041/00)
	16#02	R/W	unsigned32	16#60440010	**Transmit PDO1 assignment: 2nd object assigned** Default speed output "RFRD" (16#6044/00)
	16#03	R/W	unsigned32	16#00000000	**Transmit PDO1 assignment: 3rd object assigned** No 3rd object assigned by default
	16#04	R/W	unsigned32	16#00000000	**Transmit PDO1 assignment: 4th object assigned** No 4th object assigned by default

Object dictionary

Index	Sub-index	Access	Type	Default value	Description
16#1A01	16#00	R/W	unsigned8	16#00	**Transmit PDO2 assignment: Number of objects assigned** No objects are assigned by default in transmit PDO2, although between 0 and 4 objects can be assigned.
	16#01	R/W	unsigned32	16#00000000	**Transmit PDO2 assignment: 1st object assigned** No 1st object assigned by default
	16#02	R/W	unsigned32	16#00000000	**Transmit PDO2 assignment: 2nd object assigned** No 2nd object assigned by default
	16#03	R/W	unsigned32	16#00000000	**Transmit PDO2 assignment: 3rd object assigned** No 3rd object assigned by default
	16#04	R/W	unsigned32	16#00000000	**Transmit PDO2 assignment: 4th object assigned** No 4th object assigned by default
16#1A02	16#00	R	unsigned8	16#04	**Transmit PDO3 assignment: Number of objects assigned** 4 objects are assigned in transmit PDO3 by default, although this number can be reduced to 0.
	16#01	R	unsigned32	16#20640C10	**Transmit PDO3 assignment: 1st object assigned** Output word TP31 on the "Controller Inside" programmable card
	16#02	R	unsigned32	16#20640D10	**Transmit PDO3 assignment: 2nd object assigned** Output word TP32 on the "Controller Inside" programmable card
	16#03	R	unsigned32	16#20640E10	**Transmit PDO3 assignment: 3rd object assigned** Output word TP33 on the "Controller Inside" programmable card
	16#04	R	unsigned32	16#20640F10	**Transmit PDO3 assignment: 4th object assigned** Output word TP34 on the "Controller Inside" programmable card