This document is based on US standards and is not valid for use outside the U.S.A.

Compact / CANopen / Drive Controller / ATV-IMC + Optimized Hoisting

System User Guide

8000DB1014.02

MAR 2012

Schneider Electric

Contents

Important Information

NOTICE Read these instructions carefully, and look at the equipment to become familiar with the device before trying to install, operate, or maintain it. The following special messages may appear throughout this documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of this symbol to a Danger safety label indicates that an electrical hazard exists, which will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates an imminently hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a potentially hazardous situation which, if not avoided, **can result in** death or serious injury.

⚠ CAUTION

CAUTION indicates a potentially hazardous situation which, if not avoided, **can result in** minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

PLEASE NOTE Electrical equipment should be installed, operated, serviced, and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved

Before You Begin

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to the operator of that machine.

⚠ WARNING
UNGUARDED MACHINERY CAN CAUSE SERIOUS INJURY • Do not use this software and related automation products on equipment which does not have point-of-operation protection. • Do not reach into machine during operation. **Failure to follow these instructions can result in death, serious injury or equipment damage.**

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions, government regulations, etc. In some applications, more than one processor may be required, as when backup redundancy is needed.

Only the user can be aware of all the conditions and factors present during setup, operation and maintenance of the machine; therefore, only the user can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, the user should refer to the applicable local and national standards and regulations. A “National Safety Council’s” Accident Prevention Manual (nationally recognized in the United States of America) also provides much useful information.

In some applications, such as packaging machinery, additional operator protection such as point-of-operation guarding must be provided. This is necessary if the operator’s hands and other parts of the body are free to enter the pinch points or other hazardous areas and serious injury can occur. Software products alone cannot protect an operator from injury. For this reason the software cannot be substituted for or take the place of point-of-operation protection.

Ensure that appropriate safeties and mechanical/electrical interlocks related to point-of-operation protection have been installed and are operational before placing the equipment into service. All interlocks and safeties related to point-of-operation protection must be coordinated with the related automation equipment and software programming.

NOTE: Coordination of safeties and mechanical/electrical interlocks for point-of-operation protection is outside the scope of the Function Block Library, System User Guide, or other implementation referenced in this documentation.

START-UP AND TEST

Before using electrical control and automation equipment for regular operation after installation, the system should be given a start-up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such a check be made and that enough time is allowed to perform complete and satisfactory testing.

⚠ CAUTION

EQUIPMENT OPERATION HAZARD

- Verify that all installation and set up procedures have been completed.
- Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices.
- Remove tools, meters and debris from equipment.

Failure to follow these instructions can result in injury or equipment damage.

Follow all start up tests recommended in the equipment documentation. Store all equipment documentation for future reference.

Software testing must be done in both simulated and real environments.

Verify that the completed system is free from all short circuits and grounds, except those grounds installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to prevent accidental equipment damage.

Before energizing equipment:

- Remove tools, meters, and debris from equipment.
- Close the equipment enclosure door.
- Remove ground from incoming power lines.
- Perform all startup tests recommended by the manufacturer.

OPERATION AND ADJUSTMENTS

The following precautions are from the NEMA Standards Publication ICS 7.1-1995 (English version prevails):

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and ratings of components, there are hazards that can be encountered if such equipment is improperly operated.
- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer's instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer's instructions and the machinery used with the electrical equipment.
- Only those operational adjustments actually required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Only use software tools approved by Schneider Electric for use with this equipment.
- Update your application program every time you change the physical hardware configuration.

Failure to follow these instructions can cause death, serious injury or equipment damage.

Introduction

Introduction

This document is intended to provide a quick introduction to the described system. It is not intended to replace any specific product documentation, nor any of your own design documentation. On the contrary, it offers additional information to the product documentation for installing, configuring and implementing the system.

The architecture described in this document is not a specific product in the normal commercial sense. It describes an example of how Schneider Electric and third-party components may be integrated to fulfill an industrial application.

A detailed functional description or the specification for a specific user application is not part of this document. Nevertheless, the document outlines some typical applications where the system might be implemented.

The architecture described in this document has been fully tested in our laboratories using all the specific references you will find in the component list near the end of this document. Of course, your specific application requirements may be different and will require additional and/or different components. In this case, you will have to adapt the information provided in this document to your particular needs. To do so, you will need to consult the specific product documentation of the components that you are substituting in this architecture. Pay particular attention in conforming to any safety information, different electrical requirements and normative standards that would apply to your adaptation.

It should be noted that there are some major components in the architecture described in this document that cannot be substituted without completely invalidating the architecture, descriptions, instructions, wiring diagrams and compatibility between the various software and hardware components specified herein. You must be aware of the consequences of component substitution in the architecture described in this document as substitutions may impair the compatibility and interoperability of software and hardware.

⚠ CAUTION

EQUIPMENT INCOMPATIBILITY OR INOPERABLE EQUIPMENT

Read and thoroughly understand all hardware and software documentation before attempting any component substitutions.

Failure to follow these instructions can result in injury or equipment damage.

This document describes a generic architecture based on Altivar ATV-IMC Drive controller G-Type and hoisting architecture based on Altivar ATV-IMC Drive controller S-Type.

Abbreviations

Abbreviation	Signification
AC	Alternating Current
CB	Circuit Breaker
CFC	Continuous Function Chart - a programming language based on function chart
DI	Digital Input
DO	Digital Output
DC	Direct Current
DFB	Derived Function Blocks
EDS	Electronic Data Sheet
E-STOP	Emergency Stop
FBD	Function Block Diagram - an IEC-61131 programming language
HMI	Human Machine Interface
I/O	Input/Output
IL	Instruction List - a textual IEC-61131 programming language
IMC	Integrated Machine Controller
LD	Ladder Diagram - a graphic IEC-61131 programming language
PC	Personal Computer
POU	Programmable Object Unit, Program Section in SoMachine
PDO	Process Data Object (CANopen)
PS	Power Supply
RMS	Root Mean Square
RPM	Revolutions Per Minute
RPDO	Receive Process Data Object (CANopen)
SE	Schneider Electric
SFC	Sequential Function Chart - an IEC-61131 programming language
SDO	Service Data Object
ST	Structured Text - an IEC-61131 programming language
TPDO	Transmit Process Data Object (CANopen)
TVDA	Tested, Validated and Documented Architecture
VSD	Variable Speed Drive
WxHxD	Dimensions: Width, Height and Depth

Glossary

Expression	Signification
Altivar (ATV)	SE product name for a family of VSDs
CANopen	Name for a communications machine bus system
FDT/DTM	Field Device Tool / Device Type Manager, technology
Harmony	SE product name for a family of switches and indicators
Lexium (LXM)	SE product name for a family of servo drives
Magelis	SE product name for a family of HMI devices
ATV-IMC	SE product name for drive controller
Modbus	A Communications protocol
OsiSense	SE product name for a family of sensors
Phaseo	SE product name for a family of power supplies
PLCopen	An international standard for industrial controller programming.
Preventa	SE product name for a family of safety devices
SoMachine	SE product name for an integrated software tool
TeSys	SE product name for a family for motor protection devices and load contactors
Vijeo Designer	SE product name for Magelis HMI devices configuration software

Application Source Code

Introduction

The example source code is in the form of configuration, application and import files. Use the appropriate software tool to either open or import the files.

Extension	File Type	Software Tool Required
AIW	Configuration file	Advantys Configuration Software
CSV	Comma Separated Values, Spreadsheet	MS Excel
DCF	Device Configuration File	Advantys Configuration Software
DOC	Document file	Microsoft Word
DOP	Project File	Vijeo Designer Lite
EDS	Electronic Data Sheet - Device Definition	Industrial standard
ISL	Island file, project file	Advantys Configuration Software
PDF	Portable Document Format - document	Adobe Acrobat
PROJECT	Project file	SoMachine
VDZ	Project file	Vijeo Designer
ZW1	Project archive file	EPLAN P8

Typical Applications

Introduction

Here you will find a list of the typical applications, and market segments, where this system or subsystem can be applied:

Hoisting

- Self-erecting
- Tower crane
- Gantry crane
- Overhead traveling crane

Textile

- Opening and closing machines
- Circular knitting machines
- Plucking machines
- Blending machines
- Carding machines
- Drawing frame machines
- Combing machines
- Ring spinning machines

Pumping

- Booster stations
- Compressors
- Vacuum pumps
- Irrigation

Others

- Winding / Unwinding machines
- Wood working machines
- Cutting machines
- Sanders
- Sawing machines

SPECIAL NOTE
The products specified in this document have been tested under actual service conditions. Of course, your specific application requirements may be different from those assumed for this and any related examples described herein. In that case, you will have to adapt the information provided in this and other related documents to your particular needs. To do so, you will need to consult the specific product documentation of the hardware and/or software components that you may add or substitute for any examples specified in this documentation. Pay particular attention and conform to any safety information, different electrical requirements and normative standards that would apply to your adaptation.

The application examples and descriptions contained in this document have been developed based on products and standards available and defined for Europe. Some or all of the application examples may contain recommendations of products that are not available in your country or locality, or may recommend wiring, products, procedures or functions that are in conflict with your local, regional or national electrical or safety codes and/or normative standards.

⚠ WARNING

REGULATORY INCOMPATIBILITY

Be sure that all equipment applied and systems designed comply with all applicable local, regional and national regulations and standards.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE:
The use and application of the information contained herein require expertise in the design and programming of automated control systems. Only the user or integrator can be aware of all the conditions and factors present during installation and setup, operation, and maintenance of the machine or process, and can therefore determine the automation and associated equipment and the related safety provisions and interlocks which can be effectively and properly used. When selecting automation and control equipment, and any other related equipment or software, for a particular application, the user or integrator must also consider any applicable local, regional or national standards and/or regulations.

System

Introduction

The system chapter describes the architecture, the dimensions, the quantities and different types of components used within this system.

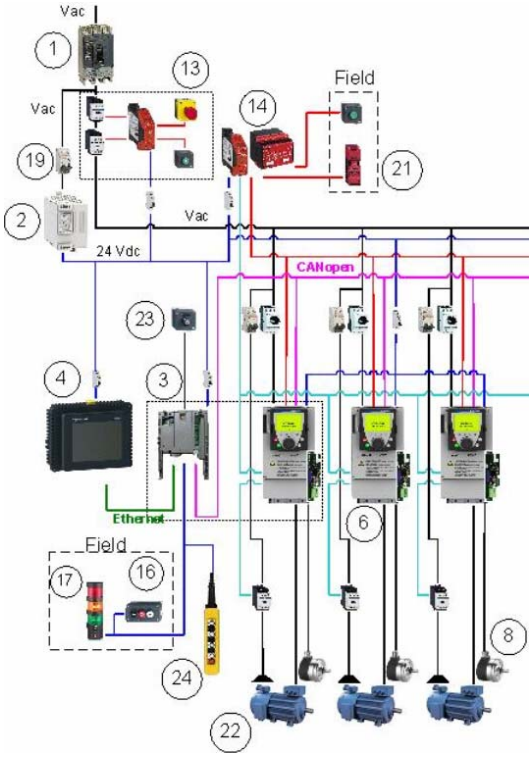
Architecture

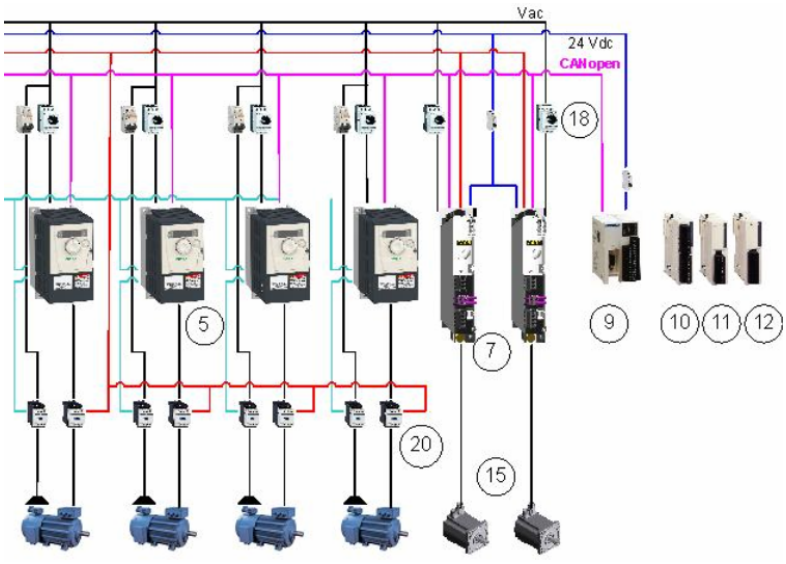
General

The controller in this application is an Altivar ATV-IMC Drive controller. The user can control the application using the Magelis HMI device. The VSDs, servo drives and Advantys OTBs are connected to the ATV-IMC via a CANopen fieldbus. The example application includes two functional safety options according to EN ISO 13849-1 standards: an Emergency Stop function supervised by a Preventa Safety Module (see the appropriate hardware manual), and a second Preventa Safety Module to evaluate protective door sensors.

Layout

1. Schneider Electric main switch
2. Phaseo power supply ABL8
3. Altivar ATV-IMC drive controller
4. Magelis graphic HMI STU655
5. Altivar drive ATV312
6. Altivar drive ATV71 + encoder card
7. Lexium 32A servo drive
8. OsiSense (Osicoder) XCC
9. Advantys OTB I/O island
10. Modicon TM2 I/O extension (digital In)
11. Modicon TM2 I/O extension (digital Out)
12. Modicon TM2 I/O extension (analog In/Out)
13. Harmony E-Stop enclosure XALK
14. Preventa safety module XPS
15. Lexium servo motor BMH
16. Harmony pushbutton enclosure XAL-D
17. Harmony tower light XVBC
18. TeSys motor protection GV2
19. Multi 9 circuit breaker
20. TeSysD contactor LC1D
21. Preventa safety switch XCS
22. AC motor
23. operator switch (direct I/O vs. CANopen)
24. Harmony pendant control station XAC





Components

Hardware:

- Schneider Electric main switch
- TeSys motor circuit breaker GV2
- Phaseo ABL8 power supply
- TeSysD load contactors
- Altivar ATV-IMC Drive controller
- Altivar 71 variable speed drive with OsiSense (Osicoder) encoder
- Altivar 312 variable speed drive
- Lexium 32A servo drive
- Lexium BMH servo motor
- Preventa XPS safety module
- Harmony XALK Emergency Stop switch with rotation release
- Harmony pushbuttons
- Magelis HMISTU655 Graphic display terminal
- Advantys OTB distributed I/O island

Software:

- SoMachine
- Advantys Configuration Tool

Quantities of Components

For a complete and detailed list of components, the quantities required and the order numbers, please refer to the components list at the rear of this document.

Degree of Protection

Not all the components in this configuration are designed to withstand the same environmental conditions. Some components may need additional protection, such as housings, depending on the environment in which you intend to use them. For environmental details of the individual components please refer to the list in the appendix of this document and the corresponding user manual.

Functional Safety Notice

(EN ISO 13849-1 EN IEC 62061)

The standard and level of functional safety you apply to your application is determined by your system design and the overall extent to which your system may be a hazard to people and machinery.

Whether or not a specific functional safety category should be applied to your system should be ascertained with a proper risk analysis.

This document is not comprehensive for any systems using the given architecture and does not absolve users of their duty to uphold the functional safety requirements with respect to the equipment used in their systems or of compliance with either national or international safety laws or regulations.

Emergency Stop

Emergency Stop/Emergency Disconnection function

This function for stopping in an emergency is a protective measure which compliments the safety functions for the safeguarding of hazardous zones according to EN ISO 12100-2.

Safety Function

Door guarding

up to Performance Level (PL) = b, Safety Integrity Level (SIL) = 1

Installation

Introduction

This chapter describes the steps necessary to set up the hardware and configure the software required to fulfill the described function of the application.

Assembly

Front side

Main rack interior

Notes

The components designed for installation in a cabinet, i.e. safety modules, circuit breakers, contactors, motor starters/protectors, power supply and Advantys OTB I/O modules can be mounted on a 35 mm DIN rail.

Main switch, Lexium 32A servo drives and Altivar 312, Altivar 71 variable speed drives are screwed directly onto the mounting plate. Alternatively if an adapter is used, the Altivar 312 and Altivar 71 can be mounted on a DIN rail.

The Emergency Stop button, the door guard switches and the pushbutton housing for the display and acknowledgement indicators are designed for on-wall mounting in the field. All switches (except the door guard switch) can also be installed directly in a control cabinet (e.g., in a cabinet door) without special housings.

There are two options for installing XB5 pushbuttons or indicator lamps. These can be installed either in a 22 mm hole, e.g., drilled into the front door of the control cabinet, or in an XALD-type housing suitable for up to 5 pushbuttons or indicator lamps. The XALD pushbutton housing is designed for backplane assembly or direct wall mounting (IP54 protection degree).

The individual components must be interconnected in accordance with the detailed circuit diagram in order to ensure that it works correctly.

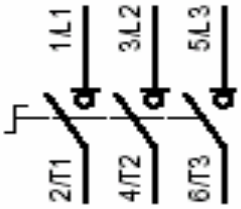
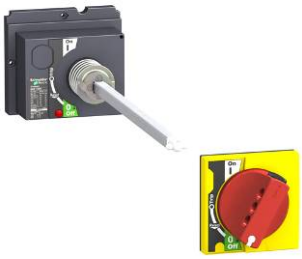
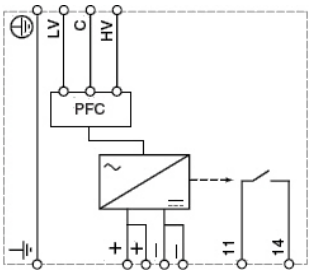
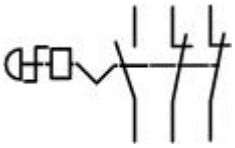
CANopen cables are installed for the communications link between the controller and the ATV312, LXM32A, ATV71 and Advantys OTB I/O modules.

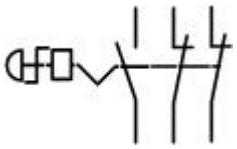
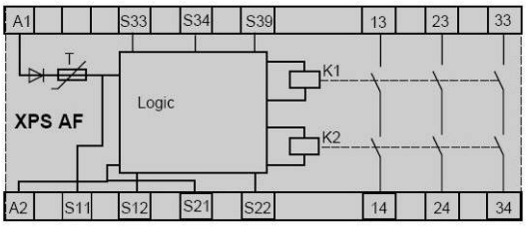
The Ethernet cable is installed for the communications link between the controller and the HMI.

Hardware

General

General description of the hardware and assembly instructions.

Main Switch **by Schneider Electric Regionally dependent component*** ***consult local catalog**		
Main Switch **Rotary handle** **Terminal shield** ***consult local catalog**	 Rotary handle with red handle on yellow front	 Terminal shield short
Power supply **Phaseo** **ABL8RPS24100** Primary 200…500 Vac, Secondary 24 Vdc, 240 W, 10 A		
Emergency Stop **Switch** (trigger action) **Harmony** **XALK178G**		

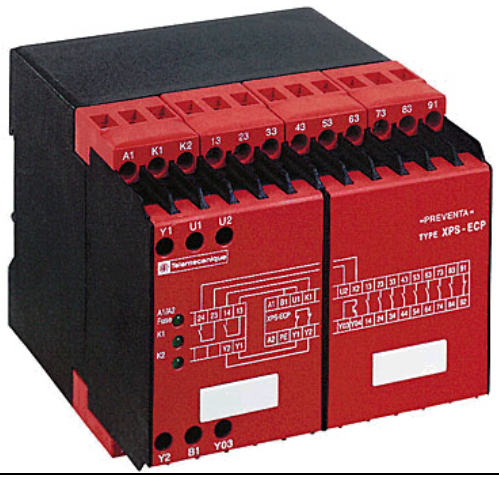
Emergency Stop **Harmony** **XB5AS844 + B5AZ141** Including Label **ZBY8330**	 	
Safety Module **Preventa** **XPSAF5130**	 	

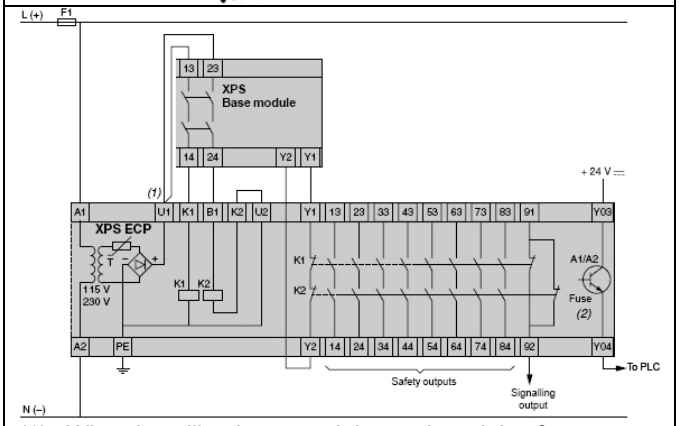
Expansion Module

Preventa

XPSECP5131

to increase the number of safety output contacts of the base module.





(1) When installing base modules and modules for increasing the number of safety contacts into different electrical enclosures, run separate cables for terminals U1-13 and U1-23.

(2) Operating status of internal electronic fuse.

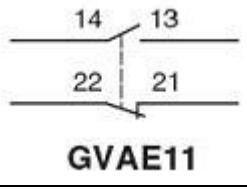
Motor Circuit Breaker

GV2*

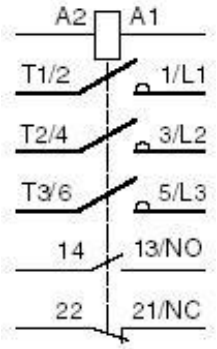
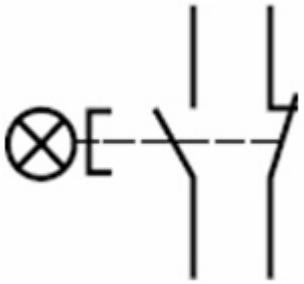
Used together with auxiliary contact

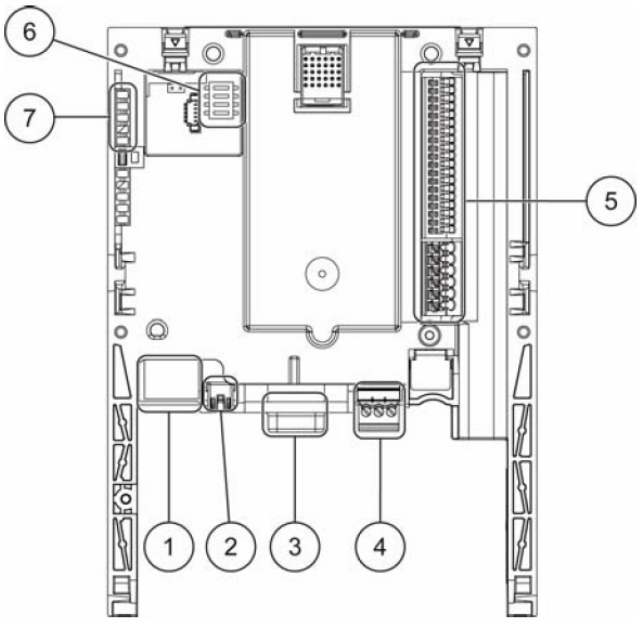
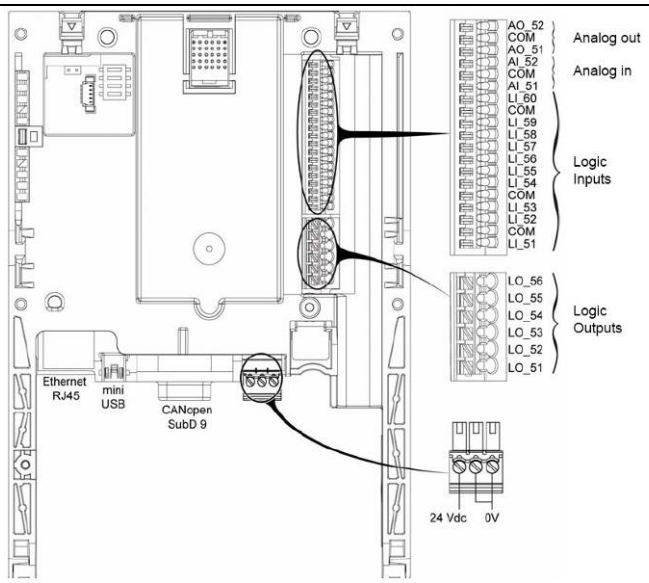
GVAE11

***consult local catalog**



GVAE11

Load Contactor **TeSysD** **LC1D18BL**		
Signal Lamps and Illuminated Pushbutton **Harmony Style 5** **XB5AVB1** **XB5AW36B5**		
Circuit Breaker **Multi 9*** ***consult local catalog**	 	 
Altivar ATV-IMC **Drive Controller** **VW3A3521** **(G-Type)** **VW3A3521S0** **(S-Type)** Is mounted on Altivar 71 drive as option card.		

Connectors and ports	 1 Ethernet port used for programming with SoMachine and for Modbus TCP communication. 2 mini-USB B port used for programming with SoMachine. 3 9-pin male SUB-D connector for connection to the CANopen® bus. 4 Connector with removable screw terminals, 3 contacts intervals of 3.81 mm (0.15 in.) for the 24 Vdc power supply. 5 10 logic inputs, 6 logic outputs, 2 analog inputs, 2 analog outputs and 5 commons. 6 Block of 4 configuration switches. 7 5 LEDs, comprising: - 1 LED G/Y **ETH** (EtherNet activity) - 1 LED G/R **NS** (Network Status) - 1 LED G/R **MS** (Module Status) - 1 LED G/R **CAN** (CANopen®) - 1 LED G/R **USER** programmable from the customer
I/Os and power supply	

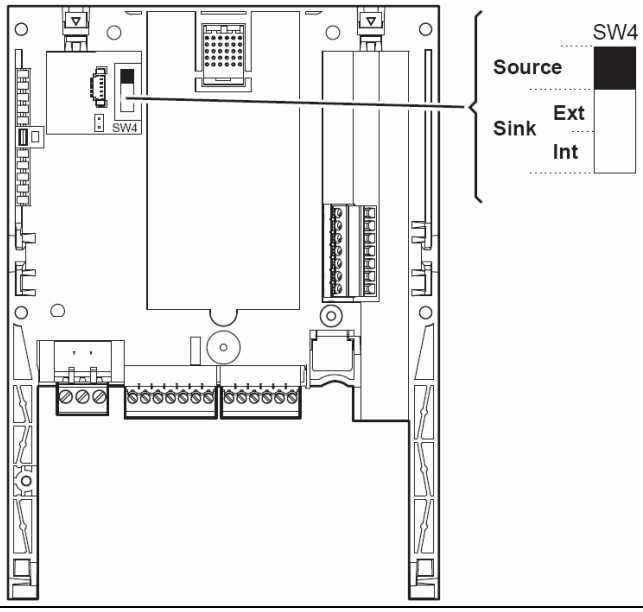
Altivar extension card

VW3A3202

Is mounted on Altivar 71 drive as option card.

the picture shows the default setting of switch 4

(SW4 = Source)

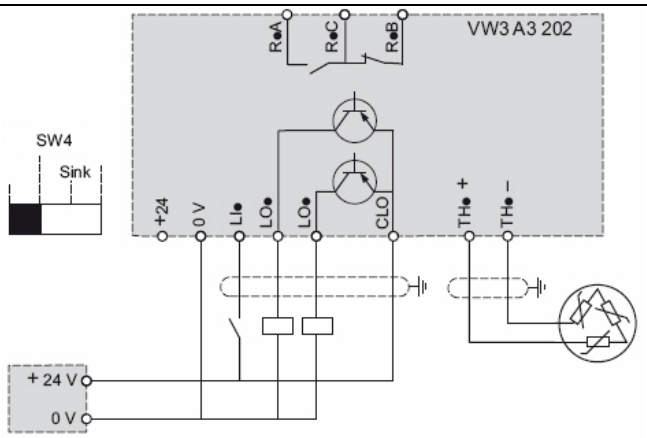


Altivar extension card

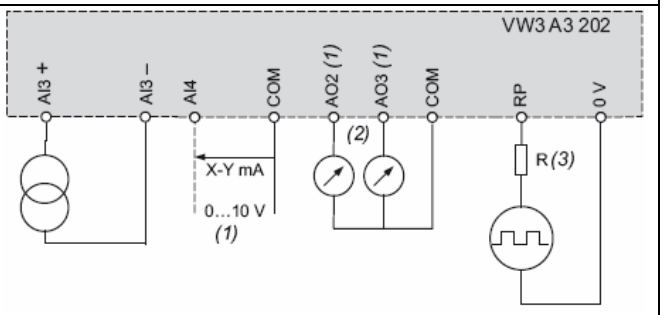
VW3A3202

Digital I/Os

Connection with with external power supply



Analog I/Os



(1) Analog input: can be configured as current input (0…20mA) or voltage input (0..10V)
(2) Analog output: can be configured as current output (0...20mA) or voltage output (±10V or 0…10V)
(3) Add a resistor if pulse sequence exceeds 5V

Variable Speed Drive

Altivar 71

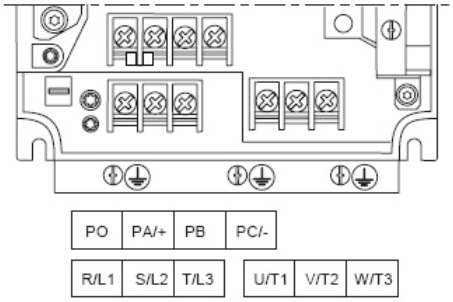
ATV71H075*

3-phase

1 HP (0.75 kW)

***consult local catalog**

Terminal	Function
⏚	Protective ground connection terminal
R/L1 S/L2 T/L3	Power supply
PO	DC bus + polarity
PA/+	Output to braking resistor (+ polarity)
PB	Output to braking resistor
PC/-	DC bus - polarity
U/T1 V/T2 W/T3	Outputs to the motor



Encoder Interface card for ATV71

VW3A3401

Zero speed torque

Accurate speed Regulation

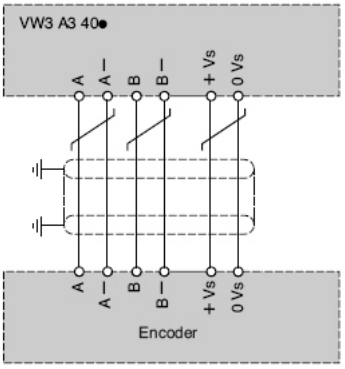
Torque accuracy

Shorter response times on a torque surge

Improved dynamic performance in transient state

Overspeed detection

Load slipping detection



Incremental Encoder

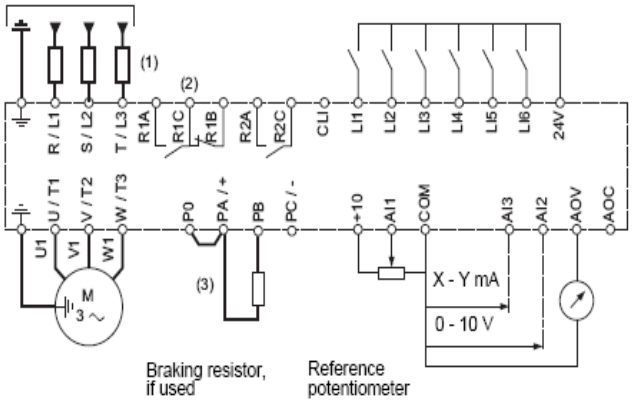
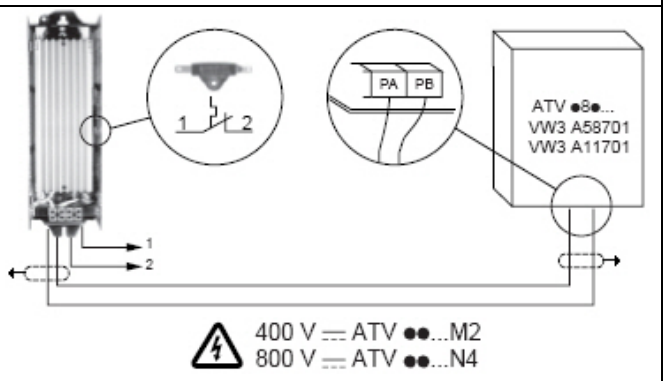
XCC1510PS11X

The encoder is used on the hoisting motor shaft and is connected to the encoder interface card on the ATV71.

Incremental Encoder

10-wire Encoder Cable

XCCPM23121L5

Variable Speed Drive **Altivar 312** **ATV312H075*** 3-phase 1 HP (0.75 kW) ***consult local catalog**	 
Braking Resistor for Altivar 312 **VW3A58732** Protected Braking Resistor, 100 Ω 28 W at 122 °F (50 °C)	
Braking Resistor for Altivar 71 **VW3A7801** Hoist Resistor, 100 Ω 1.6 kW at 122 °F (50 °C)	

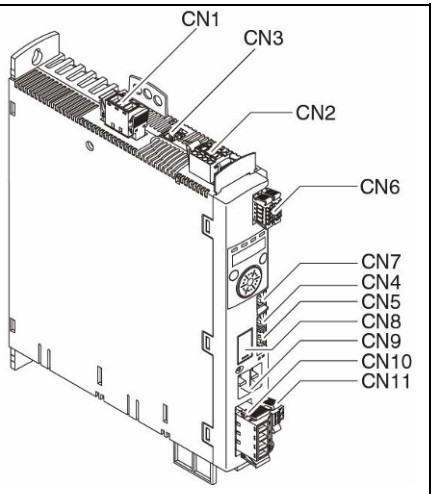
Servo Drive

Lexium 32A

LXM32AD18M2

1-phase

continuous output current:
6 A RMS at 6000 RPM



Connection	Assignment
CN1	Power stage supply voltage
CN2	24 controller supply voltage and STO safety function
CN3	Motor encoder (encoder 1)
CN4	PTO (encoder simulation ESIM)
CN5	PTI (pulse/direction, A/B, CW/CCW)
CN6	Analog inputs and digital inputs/outputs
CN7	Modbus (commissioning interface)
CN8	External braking resistor
CN9	DC bus connection for parallel operation
CN10	Motor phases
CN11	Holding brake

Servo Drive

Lexium 32A

LXM32AD18M2

Embedded Human Machine Interface

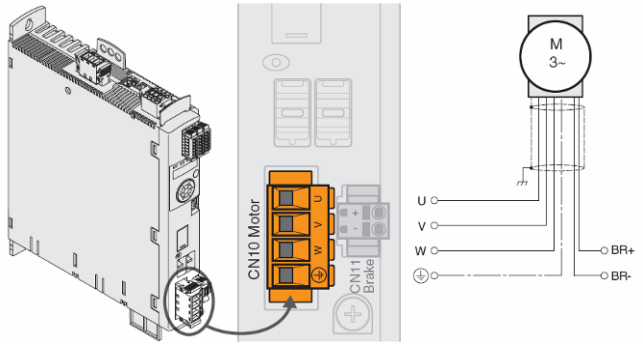
Servo Drive

Lexium 32A

1-phase

LXM32AD18M2

Wiring diagram
Power cable connection to motor (Length 3 m)



Connection	Meaning	Color
U	Motor phase	Black L1 (BK)
V	Motor phase	Black L2 (BK)
W	Motor phase	Black L3 (BK)
PE	Protective ground conductor	Green/yellow (GN/YE)
BR+	Holding brake +	White (WH)
BR-	Holding brake -	Gray (GR)

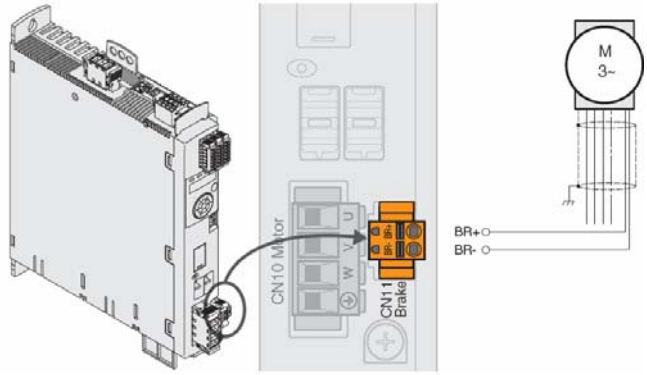
Servo Drive

Lexium 32A

1-phase

LXM32AD18M2

Wiring diagram holding brake



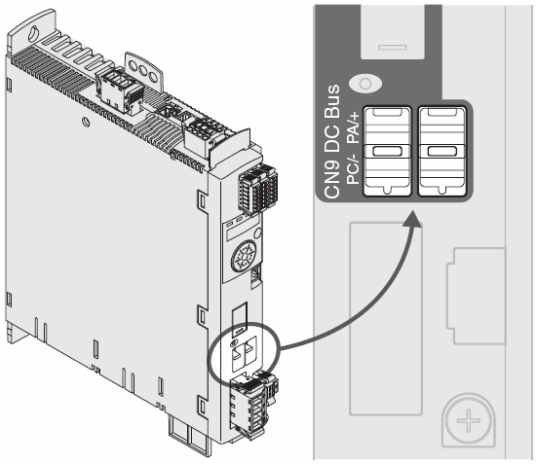
Servo Drive

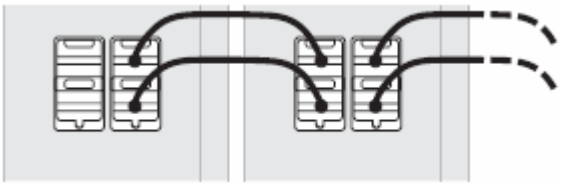
Lexium 32A

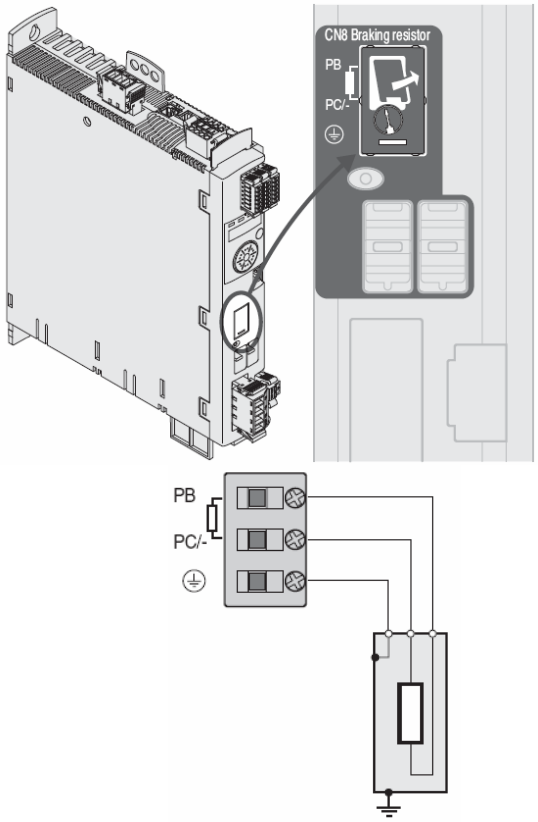
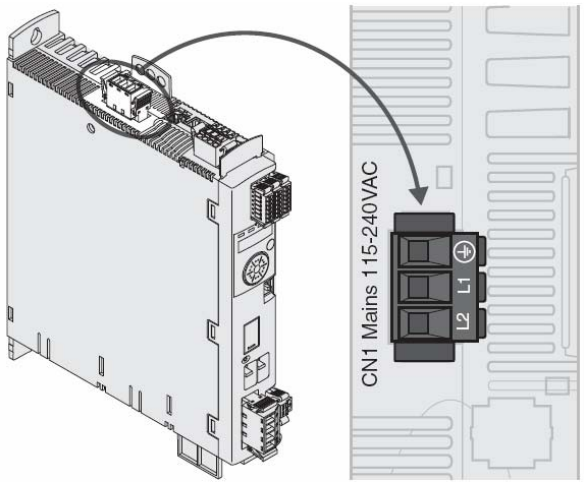
1-phase

LXM32AD18M2

Parallel connection DC bus





Servo Drive **Lexium 32A** 1-phase **LXM32AD18M2** Connecting the external braking resistor	
Servo Drive **Lexium 32A** 1-phase **LXM32AD18M2** Wiring diagram power stage supply voltage for 1-phase device	

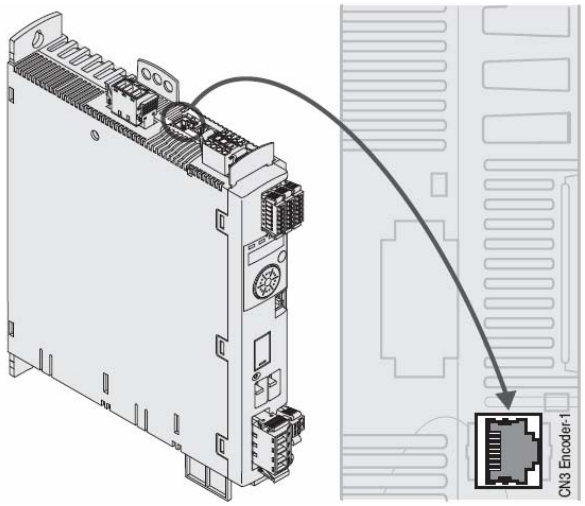
Servo Drive

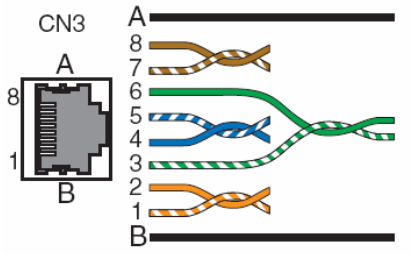
Lexium 32A

1-phase

LXM32AD18M2

Wiring diagram motor encoder





Pin	Signal	Motor, pin	Pair	Meaning	I/O
1	COS+	9	2	Cosine signal	I
2	REFCOS	5	2	Reference for cosine signal	I
3	SIN+	8	3	Sine signal	I
6	REFSIN	4	3	Reference for sine signal	I
4	Data	6	1	Receive data, transmit data	I/O
5	$\overline{\text{Data}}$	7	1	Receive data and transmit data, inverted	I/O
7	reserved		4	Not assigned	
8	reserved		4	Not assigned	
A	ENC+10V_OUT	10	5	Encoder supply	O
B	ENC_0V	11	5	Reference potential for encoder supply	
	SHLD			Shield	

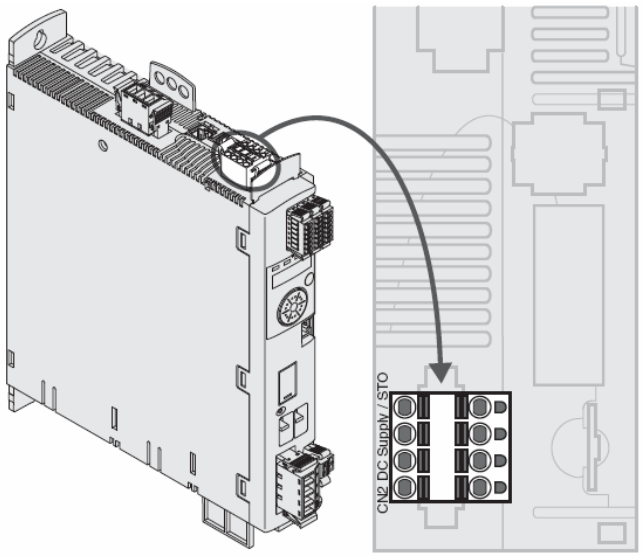
Servo Drive

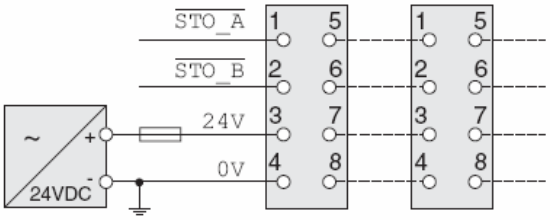
Lexium 32A

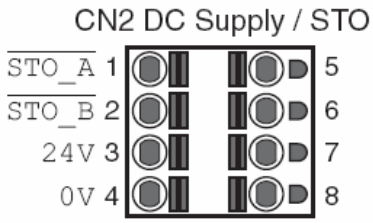
1-phase

LXM32AD18M2

Wiring diagram controller supply voltage







Pin	Signal	Meaning
1, 5	STO_A	Safety function STO: Dual-channel connection, connection A
2, 6	STO_B	Safety function STO: Dual-channel connection, connection B
3, 7	+24VDC	24 V controller supply voltage
4, 8	0VDC	Reference potential for 24V controller supply voltage; Reference potential for STO

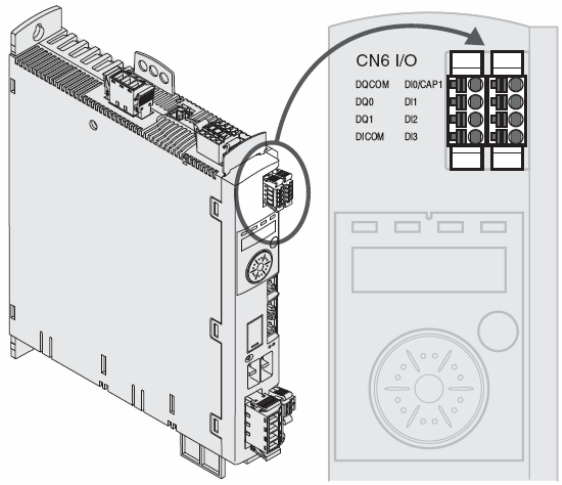
Servo Drive

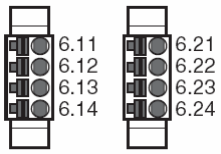
Lexium 32A

1-phase

LXM32AD18M2

Wiring diagram, digital inputs/outputs





Pin	Signal	1)	Meaning	I/O
CN6.11	DQ_COM		Reference potential digital outputs	
CN6.12	DQ0	X	Digital output 0	O (24 V)
CN6.13	DQ1	X	Digital output 1	O (24 V)
CN6.14	DI_COM		Reference potential digital inputs	

1) Connector coding, X=coding

Pin	Signal	1)	Meaning	I/O
CN6.21	DI0 / CAP1	X	Digital input 0 / touch probe 0	I (24 V)
CN6.22	DI1		Digital input 1	I (24 V)
CN6.23	DI2		Digital input 2	I (24 V)
CN6.24	DI3	X	Digital input 3	I (24 V)
	DI_COM		Reference potential digital inputs: CN6.14	

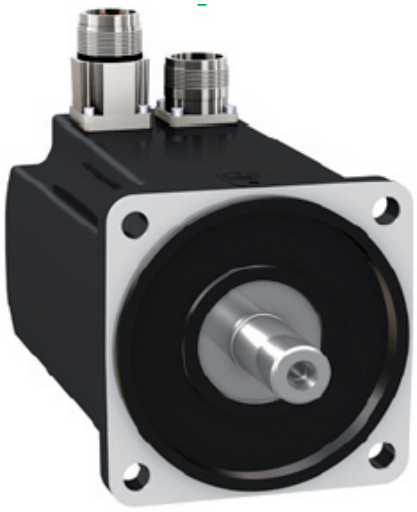
1) Connector coding, X=coding

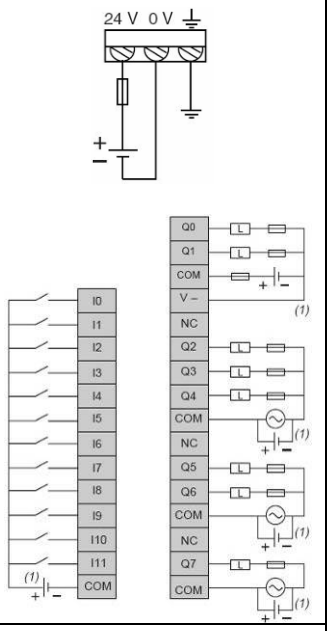
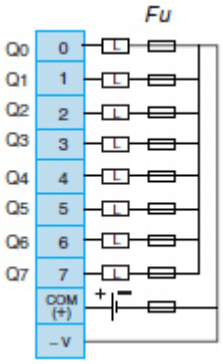
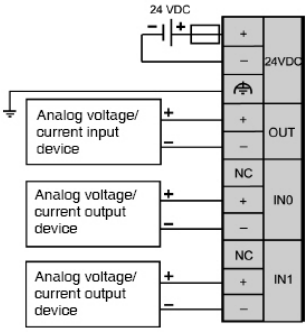
Servo Motor

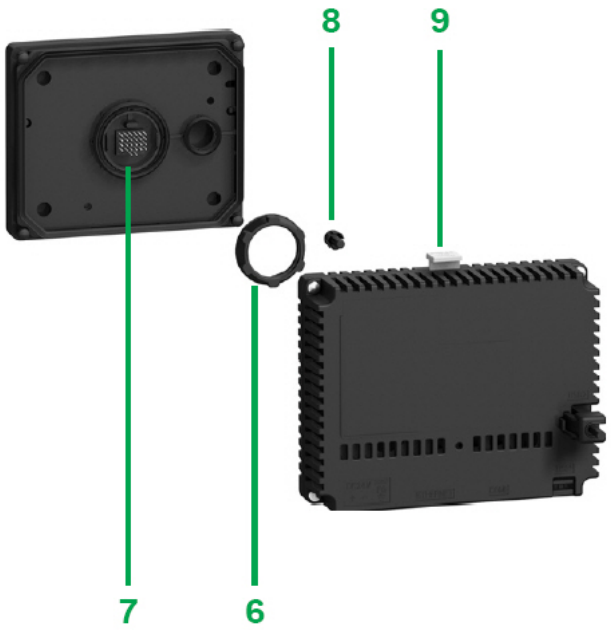
BMH0702T02F2A

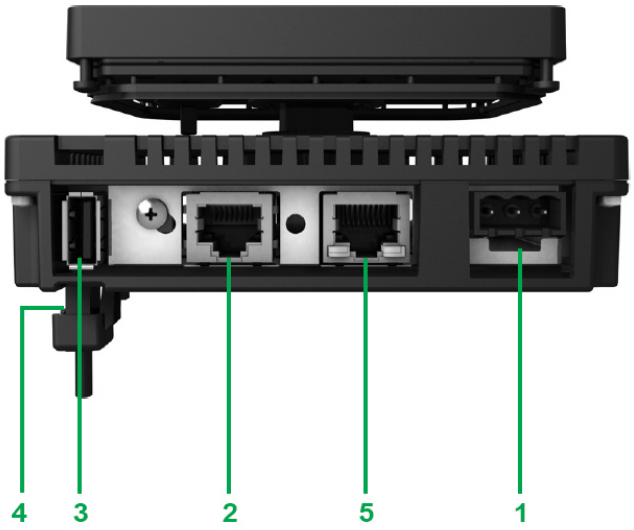
with brake

Continuous stall torque: 2.5 Nm at 6000 RPM



Distributed I/O **Advantys OTB** **OTB1C0DM9LP** CANopen interface module, 14 Digital Inputs, 2 Solid-State Outputs, 6 Relay Outputs, Removable Screw Terminal Blocks		
I/O Module **Advantys** **TM2DDO8TT** 8 Digital Source Outputs, 24 Vdc, Removable Screw Terminal Block		 Fu: 0.3 A quick-blow fuse
I/O Module **Advantys** **TM2AMM3HT** 2 Analog Inputs (0…10 Vdc / 4…20 mA) 1 Analog Output (0..10 Vdc / 4..20 mA)		
Magelis HMI **STU655**		

Magelis HMI **STU655**	 *Exploded view of Magelis STU Small Panel: simple installation by means of a 22 mm diameter hole*
	 A Magelis STU Small Panel is made up of a front module (comprising the screen) and a rear module (comprising the CPU plus terminals and connectors). The two modules are fixed together by means of a hole measuring 22 mm in diameter. The fixing system contains the following elements: **6** An adjusting nut **7** A seal **8** An anti-rotation tee (can be used as an option) **9** A release mechanism: simply press to separate the two modules once they have been fixed together

Magelis HMI **STU655** **Power and communication connectors**	 1 A removable screw terminal block for the 24 V ⎓ power supply 2 An RJ45 connector for RS 232C or RS 485 serial link connection to PLCs (COM1) 3 A USB type A host connector for: □ Connection of a peripheral device □ Connection of a USB memory stick □ Application transfer 4 A USB mini-B device connector for application transfer (on the left-hand side) 5 An RJ45 connector for the Ethernet TCP/IP 10BASE-T/100BASE-TX link

Software

General

The main programming work lies in programming the Altivar ATV-IMC Drive controller, the configuration of the CANopen fieldbus and creating the screens for the HMI display.

Programming the Altivar ATV-IMC Drive controller is done using SoMachine software.

Programming of the Magelis HMISTU655 is done by using Vijeo Designer which is integrated into SoMachine software.

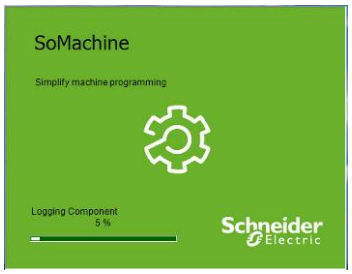
The basic configuration of the drives ATV71, ATV312 and LXM32A is done using the control panel. The configuration of the Advantys OTB Island is done using the Advantys Configuration Software.

To use the software packages, your PC must have the appropriate Microsoft Windows operating system installed:

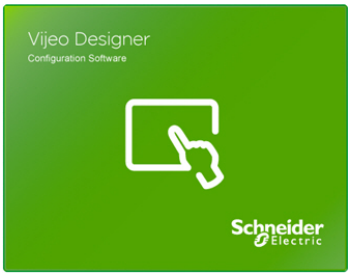
- Windows XP Professional

The software tools have the following default install paths:

- SoMachine
  C:\Program Files\Schneider Electric\SoMachine



- Vijeo Designer (installed with SoMachine)
  C:\Program Files\Schneider Electric\Vijeo Designer



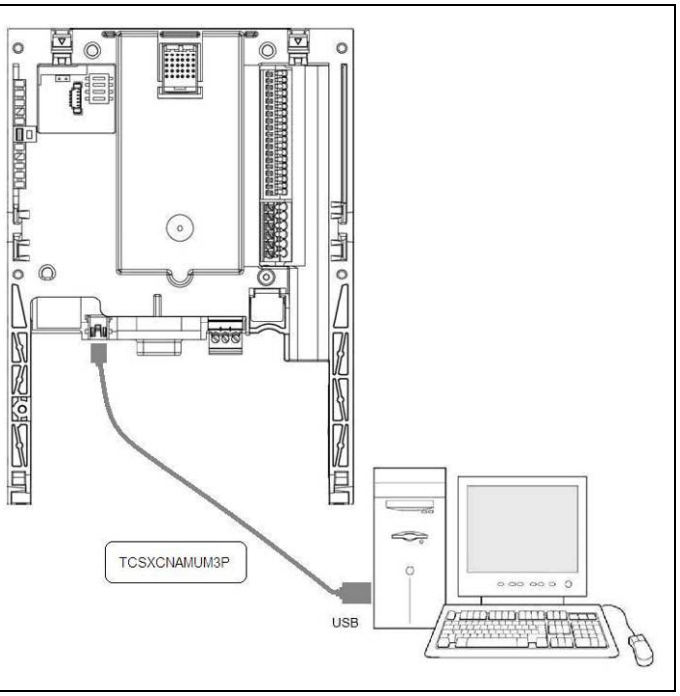
- Advantys Configuration Software
  C:\Program Files\Schneider Electric\Advantys

Communication

General

This TVDA architecture includes two different communication networks or fieldbusses. The CANopen fieldbus connects the ATV-IMC as CANopen Master and the Altivar drives, Advantys OTB and Lexium 32A servo drives as CANopen nodes.
All the drives and the I/O-Island are connected via CANopen TAPs. The CANopen transmission rate is 500 kbit/s.

The Altivar ATV-IMC Drive controller and the Magelis HMISTU655 communicate via Ethernet network. The download from the PC to the ATV-IMC and the HMI is done using a single cable connection. The PC has to be connected to the ATV-IMC via USB and using this connection the data is sent across to the HMI.

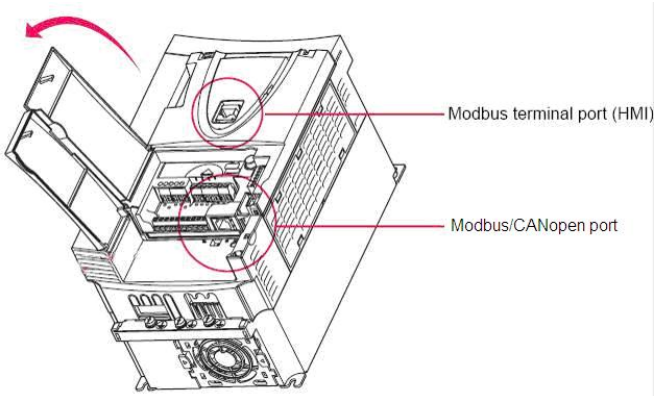
Programming-cable for **ATV-IMC** **TCSXCNAMUM3P**	

CANopen connection

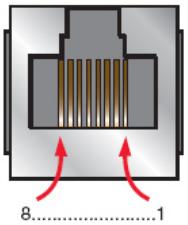
Altivar 71

Node ID: 1..3

Note:
In case of CANopen, the CANopen Tap **TSXCANTDM4** is used to connect the drive to the CANopen fieldbus via RJ45 socket.



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)

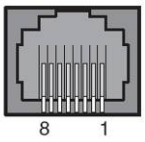
(1) Modbus signal
(2) Power supply for an RS232/RS485 converter (to PowerSuite)

CANopen connection

Altivar 312

Node ID: 4..7

Note:
In case of CANopen, the CANopen Tap **TSXCANTDM4** is used to connect the drive to the CANopen fieldbus via RJ45 socket.



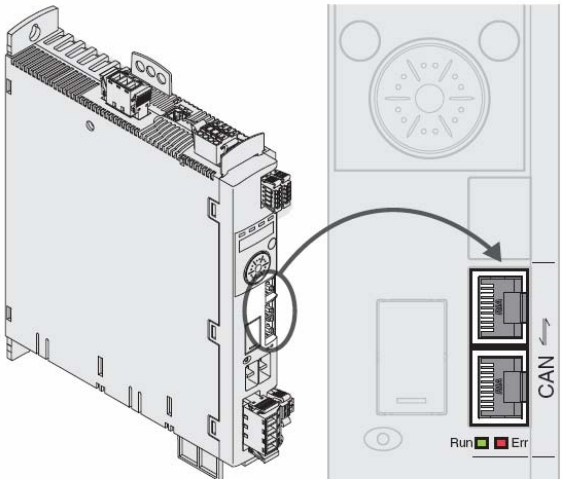
Pin	Signal	Description
1	CAN_H	data wire
2	CAN_L	data wire, inverted
7	MOD+10V_OUT	10V power supply
8	MOD_0V	Reference potential for MOD+10V_OUT

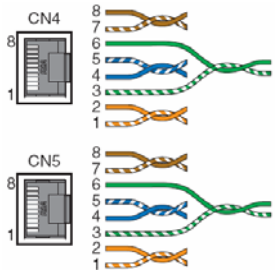
Pin	Signal	Description	
4	MOD_D1	Bidirectional transmit/receive signal	RS485 level
5	MOD_D0	Bidirectional transmit/receive signal, inverted	RS485 level
7	MOD+10V_OUT	10 V power supply, max. 150 mA	Output
8	MOD_0V	Reference potential for MOD+10V_OUT	Output

CANopen connection

Lexium 32

Node ID: 8..9





Pin Signal Meaning I/O
1. CAN_H CAN interface CAN level
2. CAN_L CAN interface CAN level
3. CAN_0V Reference potential CAN -
4. nc not used -
5. nc not used -
6. nc not used -
7. nc not used -
8. nc not used -

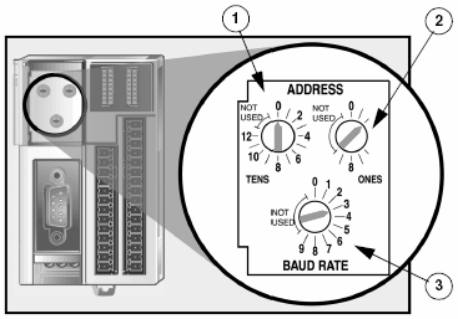
Advantys OTB

CANopen

OTB1CODM9LP

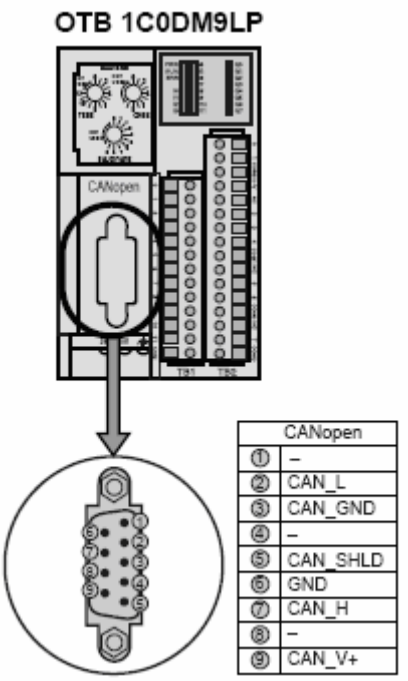
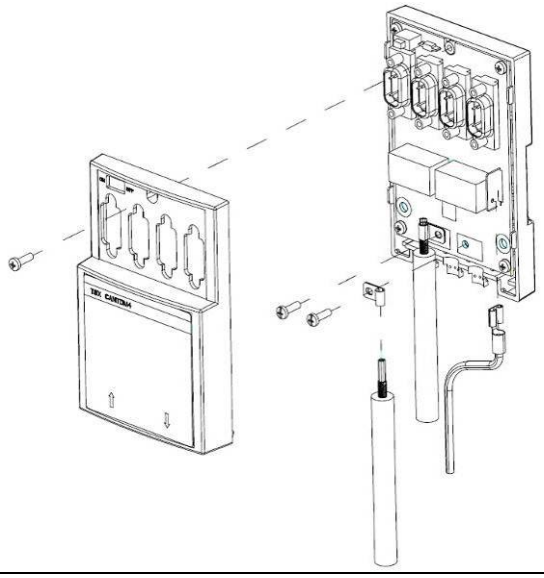
Node ID: 10

(at 500 kbit/s)



1. Network address (Node-ID x10) encoder wheel
2. Network address (Node-ID x1) encoder wheel
3. Transmission speed encoder wheel

Position (lower encoder rate)	Baud Rate
0	10 kbits/s
1	20 kbits/s
2	50 kbits/s
3	125 kbits/s
4	250 kbits/s
5	500 kbits/s
6	800 kbits/s
7	1 Mbits/s
8	Automatic
9	Default (250 kbits/s

Advantys OTB **OTB1C0DM9LP** CANopen port	
CANopen TAP **TSXCANTDM4** 4 port CANopen junction box For the purpose of this application, the sliding switch should be set to **OFF** if it is not at the end of the CANopen line.	 

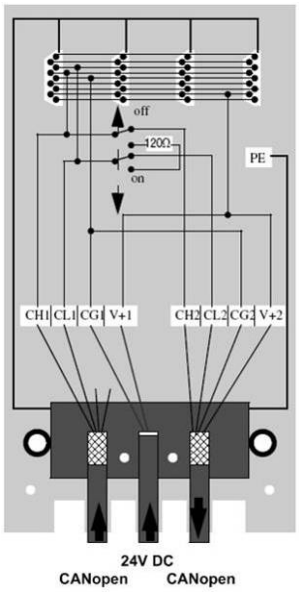
CANopen TAP

TSXCANTDM4

Note:
When using devices which require a 24 Vdc power supply on CANopen line (such as TeSysU) the 24 Vdc power must be wired.

Power supply:

V+1 24 Vdc
CG1 0 Vdc



Signal	Terminal block 1	Terminal block 2	Wire color
CAN_H	CH1	CH2	white
CAN_L	CL1	CL2	blue
CAN_GND	CG1	CG2	black
CAN_V+	V+1	V+2	red

CANopen connector

VW3CANKCDF90T, VW3CANKCDF90TP

These connectors are used for the link to the CANopen node.

VW3CANKCDF90T, VW3CANKCDF90TP

CANopen connector

VW3CANKCDF180T

This connector should be used with ATV-IMC board

VW3CANKCDF180T

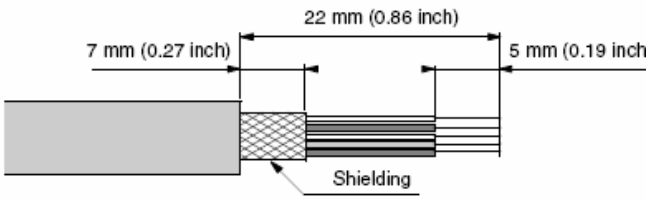
<table>
<tr>
<td>CANopen pre-assembled connection cable</td>
<td> 
TCSCCN4F3M1T
(length: 1 m)

Used for connection between ATV312/71, LXM32A and TSXCANTDM4.</td>
<td> 
TSXCANCADD1
(length: 1 m)

Used for connection between Advantys OTB and TSXCANTDM4.</td>
</tr>
<tr>
<td>CANopen cable

TSXCANCxy

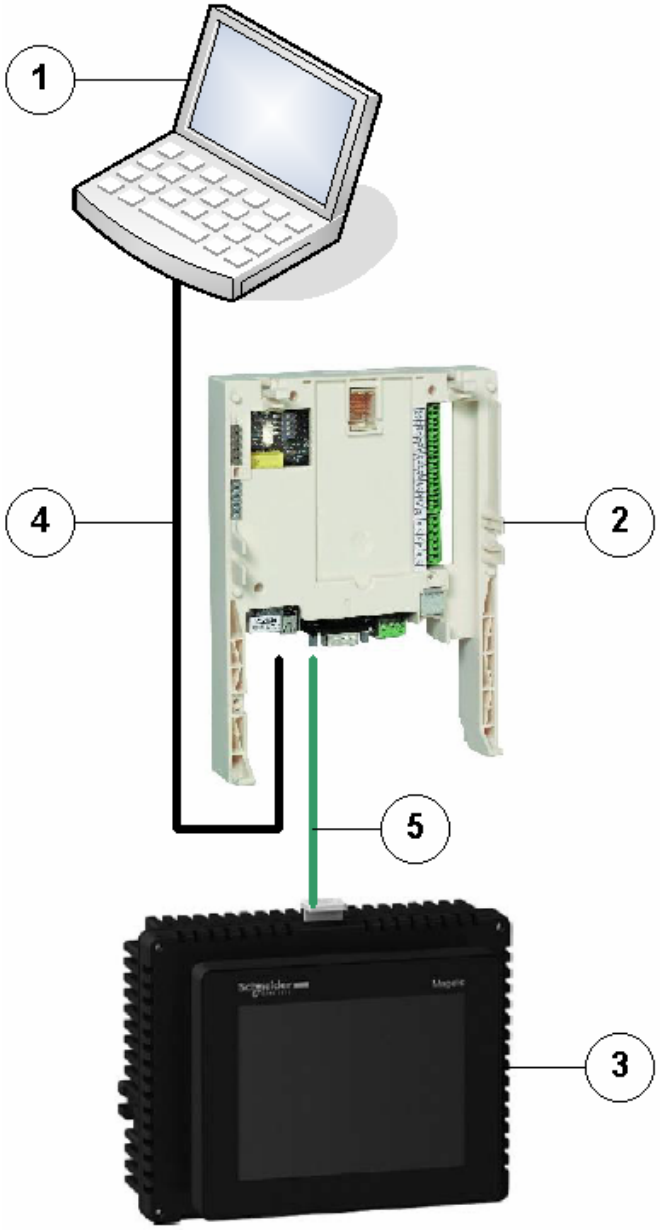
The cable is available in various versions (x):
A - Standard
B - No Flame
D - Heavy Duty

and various lengths (y):
50 - for 50 m
100 - for 100 m,
300 - for 300 m.</td>
<td colspan="2"> 
 
</td>
</tr>
</table>

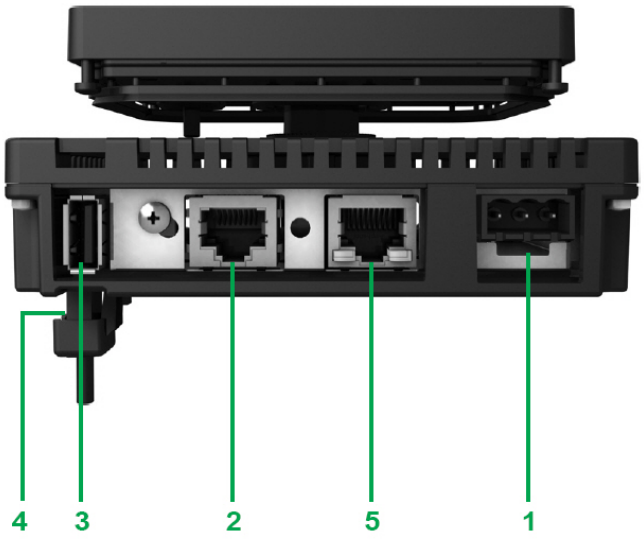
PC ↔ ATV-IMC ↔ STU655

The download direction is from the PC to the ATV-IMC drive controller and via the ATV-IMC drive controller to the HMI STU655

Note:
For a direct connection from the PC to the controller the **TCSXCNAMUM3P** cable can be used.



(1) PC
(2) Altivar ATV-IMC Drive controller
(3) Magelis HMISTU655
(4) USB to USB cable TCSXCNAMUM3P
(5) RJ45 (HMI) to RJ45 (ATV-IMC) cable 490NTW00002

Magelis **HMISTU655** communication ports	 1 A removable screw terminal block for the 24 V ⎓ power supply 2 An RJ45 connector for RS 232C or RS 485 serial link connection to PLCs (COM1) 3 A USB type A host connector for: □ Connection of a peripheral device □ Connection of a USB memory stick □ Application transfer 4 A USB mini-B device connector for application transfer (on the left-hand side) 5 An RJ45 connector for the Ethernet TCP/IP 10BASE-T/100BASE-TX link
PC ↔ HMI connection cable **XBTZG935** Cable for the connection between a SoMachine-equipped PC and HMISTU655	
Controller ↔ HMI connection cable **490NTW00002 (2m)** Cable for connecting an HMISTU655 and an ATV-IMC	

Implementation

Introduction

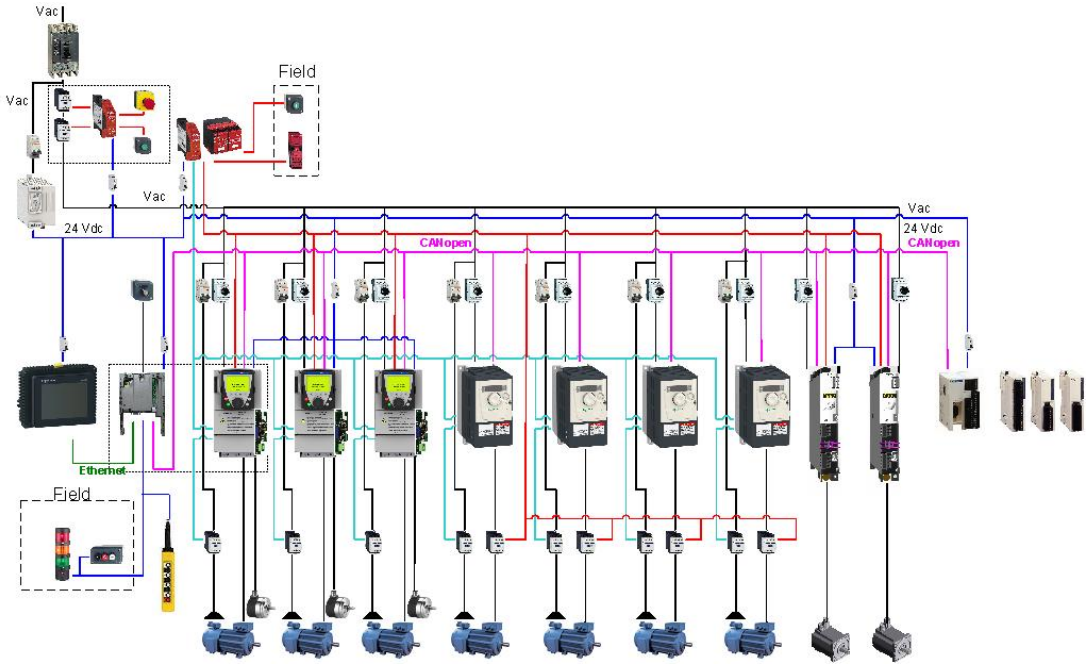
The implementation chapter describes all the steps necessary to initialize, to configure, to program and startup the system to achieve the application functions as listed below.

Function

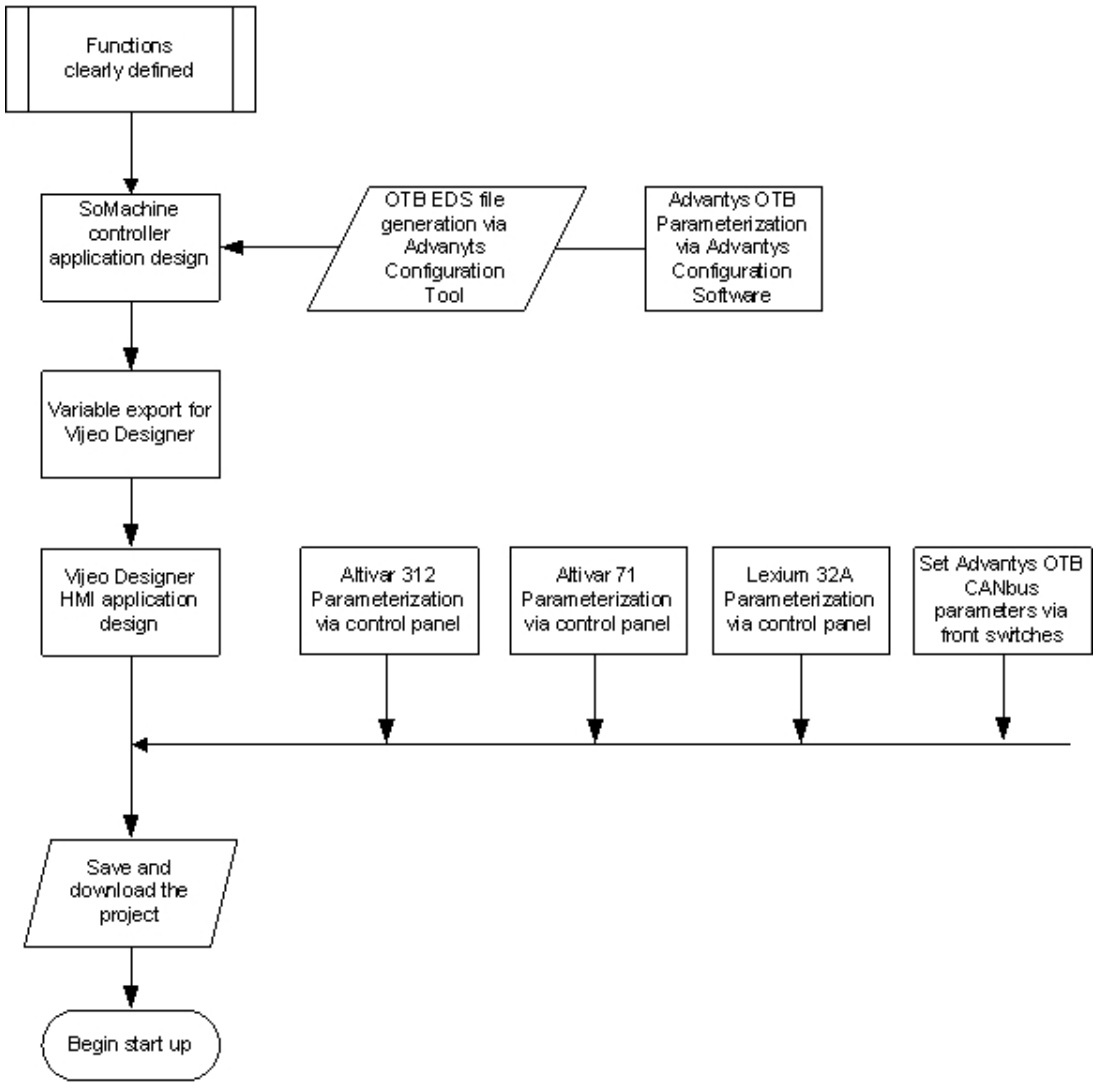
Start up and functional description

1. Ensure all motor starters/protectors and Multi9 circuit breakers are in the ON position.
2. Ensure that the main switch is in the ON position.
3. Press the "ACKN E-STOP" blue illuminated pushbutton on the main cabinet door to acknowledge the system is energized. The blue illuminated pushbutton will turn OFF if the system is energized.
4. Ensure that all machine interlocks are engaged (i.e. the door guard switches)
5. Press the "ACKN DOOR-READY" blue illuminated pushbutton on the motor rack to acknowledge the system is ready for operation. The blue illuminated pushbutton will turn OFF if the system is ready for operation.
6. Use Magelis STU HMI to control/monitor the system.
    a. The “BUS” and “SAFETY” screens can be used to monitor the network, system status and alarm messages.
    b. The “ATV71” screen can be used to control/monitor Altivar 71 variable speed drives.
    c. The “ATV312” screen can be used to control/monitor Altivar 312 variable speed drives.
    d. The “LXM32A” screen can be used to control/monitor Lexium 32A servo drives.
    e. The “OTB” screen can be used to observe the status of the Advantys OTB I/O.

Functional Layout



Course of Action



Communication Setup

Introduction

This chapter describes the data passed via the communication networks or fieldbusses (e.g. CANopen or Ethernet) that are not bound directly with digital or analog hardware.

The list contains:

- The device links
- Direction of data flow
- Symbolic name and
- Bus address of the device concerned.

Device Links

This application uses CANopen fieldbus and Ethernet networks.

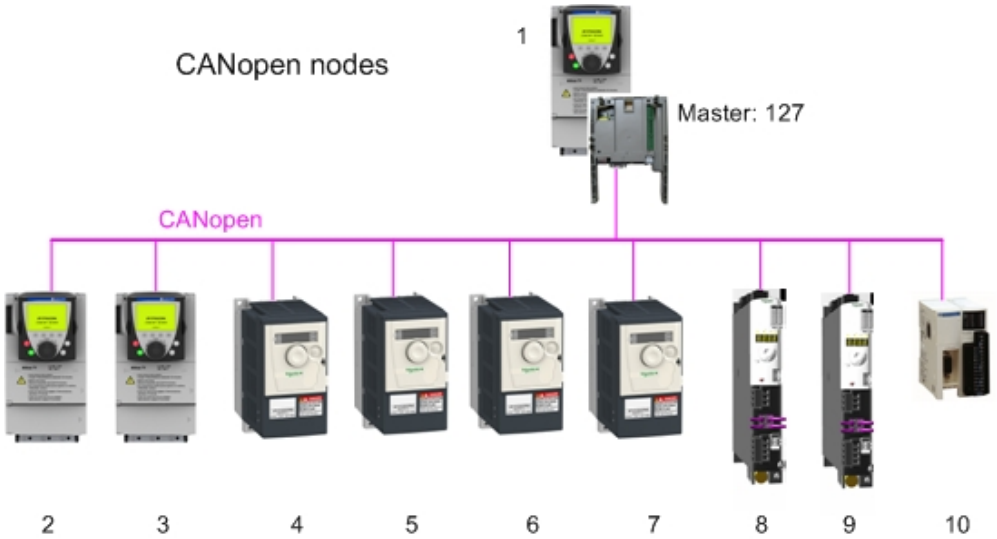
The Altivar ATV-IMC Drive controller and Magelis STU HMI are connected via Ethernet using SoMachine protocol.

The SoMachine protocol over Ethernet connects:
Magelis STU HMI (IP 192.168.100.30)
Altivar ATV-IMC Drive controller (IP 192.168.100.20)

CANopen connects the following devices:

1 ATV-IMC bus master, address 127
3 Altivar 71 variable speed drives, bus addresses 1..3
4 Altivar 312 variable speed drives, bus addresses 4..7
2 Lexium 32A servo drives, bus addresses 8..9
1 Advantys OTB I/O islands, bus address 10

The used CANopen baudrate is 500kbit/s.



NOTE

For the data exchange between the ATV-IMC and the Altivar 71, the Altivar 312 and the Lexium 32A, PLCopen function blocks are used. It is not necessary to configure the data exchange manually.

Datalink	**ATV-IMC** *(CANopen-Master, #127)*		**Advantys OTB** *(CANopen-Slave #10)*	
	Data Direction Advantys OTB → ATV-IMC			
Advantys OTB → ATV-IMC	**Channel**	**Variable**	Address	**Designation**
	Read input 0 to 7 module 0	usiIputOTB_10_1	%IB114	First input byte (OTB1CODM9LP)
	Read input 8 to 11 module 0	usiIputOTB_10_2	%IB115	Second input byte (OTB1CODM9LP)
	Read input 0 to 7 module 1	usiIputOTB_10_3	%IB116	First input byte (TM2DDI16DT)
	Read input 8 to 11 module 1	usiIputOTB_10_4	%IB117	Second input byte (TM2DDI16DT)
	Read anlog input 0 module 3	iIputOTB_10_5	%IW59	First input word (TM2ALM3RT)
	Read anlog input 0 module 4	iIputOTB_10_6	%IW60	Second input word (TM2ALM3RT)
ATV-IMC → Advantys OTB	***Data Direction ATV-IMC → Advantys OTB***			
	Channel	**Variable**	Address	**Designation**
	write output 0 to 7 module 0	usiOputOTB_10_1	%QB176	First output byte (OTB1CODM9LP)
	write output 0 to 7 module 2	usiOputOTB_10_2	%QB177	First output byte (TM2DRA16RT)
	write output 8 to 15 module 2	usiOputOTB_10_3	%QB178	Second output byte (TM2DRA16RT)
	write analog output module 3	iOputOTB_10_4	%QW90	First output word (TM2ALM3RT)

Controller

Introduction

The controller chapter describes the steps required for the initialization and configuration and the source program required to fulfill the functions.

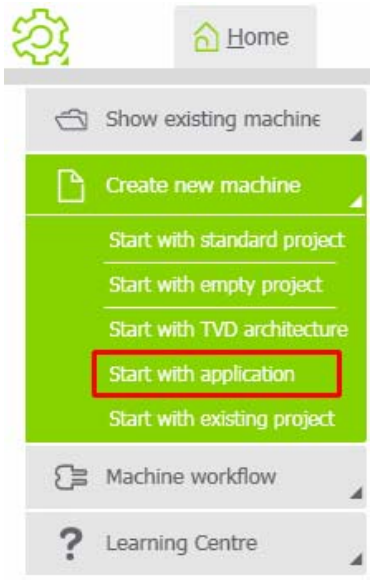
Requirements

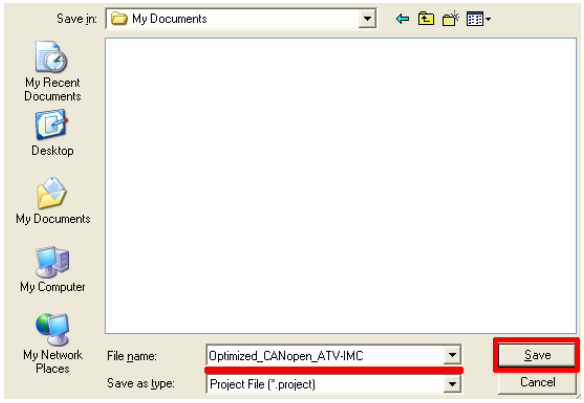
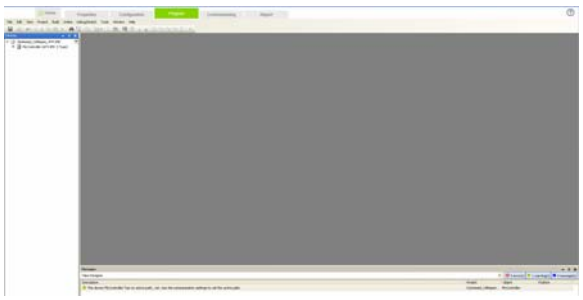
- SoMachine is installed on your PC
- The Altivar ATV-IMC Drive controller is switched on and running
- The ATV-IMC is connected to the HMI with the Ethernet cable **490NTW00002**
- The ATV-IMC is connected to the PC via the USB cable **TCSXCNAMUM3P**

Setting up the ATV-IMC is done as follows:

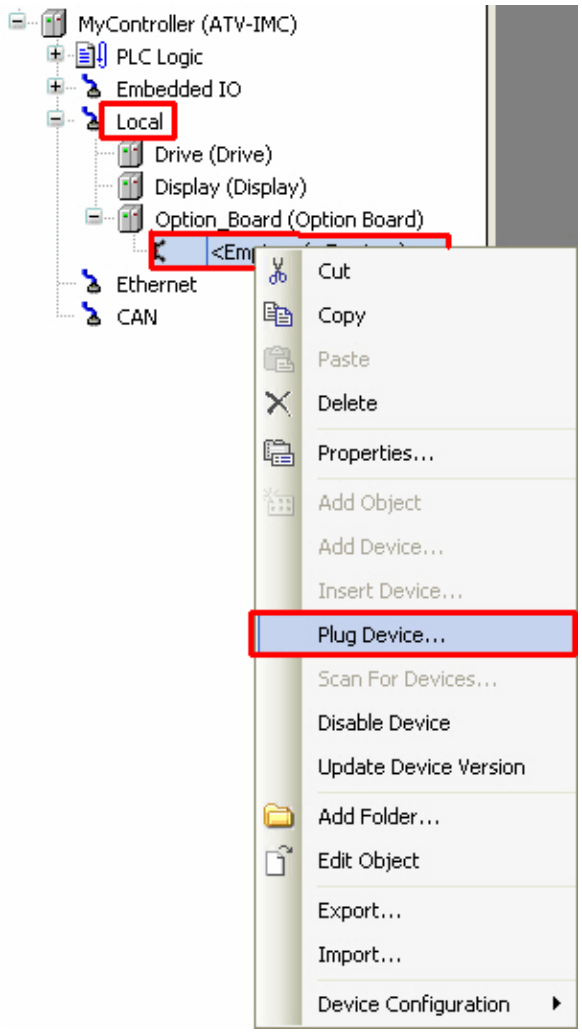
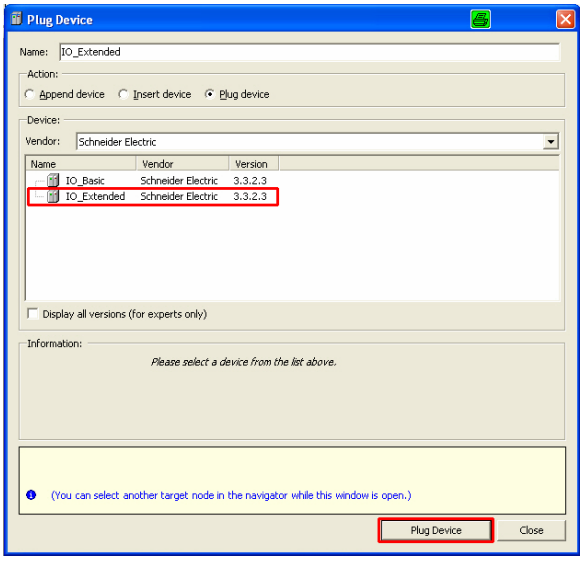
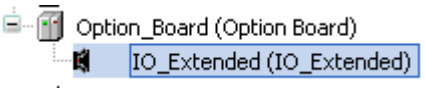
- Create a new project
- Add an Option Board
- Add the CANopen fieldbus
- Import of the Advantys OTB EDS file
- Add CANopen devices
- ATV CANopen configuration
- LXM32A CANopen configuration
- Advantys OTB CANopen configuration
- Add Library
- Add Folder
- Add a POU
- Task configuration
- Configure controller ↔ HMI data exchange
- Add Vijeo Designer HMI
- Ethernet settings
- Communication setting controller ↔ PC
  - Communication setting controller ↔ HMI
- Save the Project
- Build Application
- Download the controller and HMI program
- Login to the controller
- Application overview

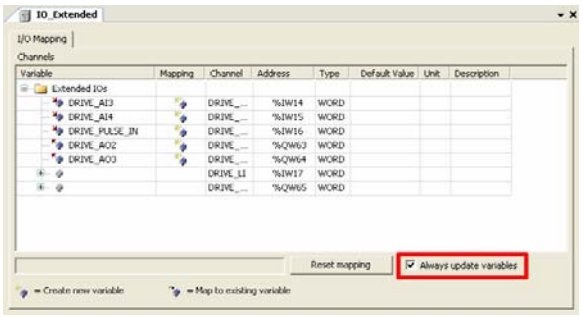
Create a new project

1	To create a new project select **Create new Machine.** **→ Start with application**	

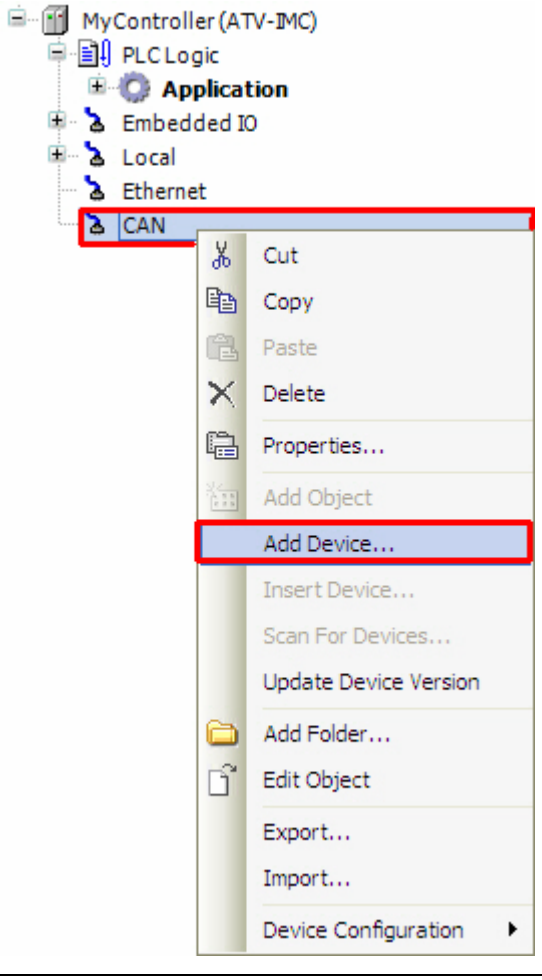
2	Under **Controller Templates** select: **ATV_IMC_Template.project** **Note:** If the SoMachine Solution Extension is not installed on the PC, the templates related to ATV-IMC S-Type will not appear.	
3	Save the project at the desired location and enter a **File name**. In this case the **File name** is **Optimized_CANopen_ATV-IMC.** Click on **Save.**	
4	Click on **Program.**	
5	The Program window appears.	

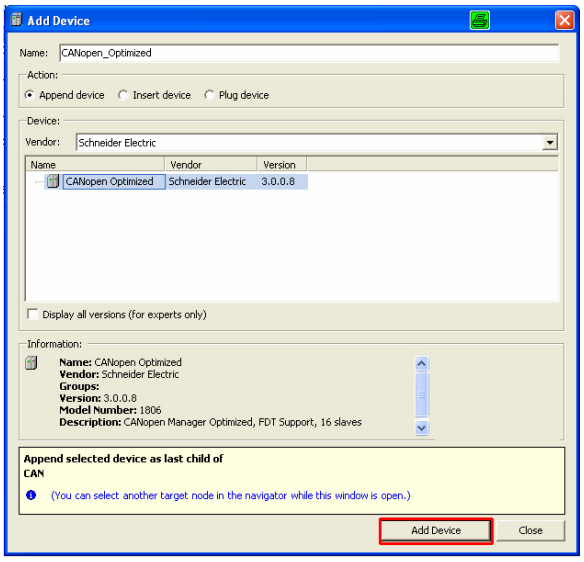
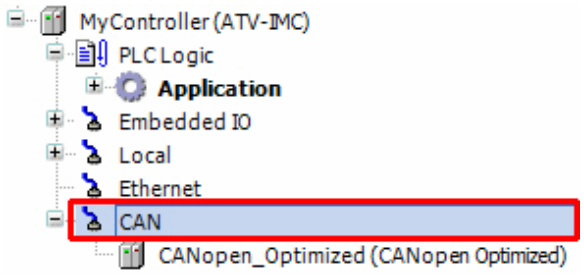
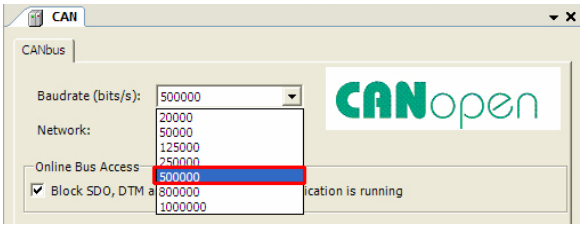
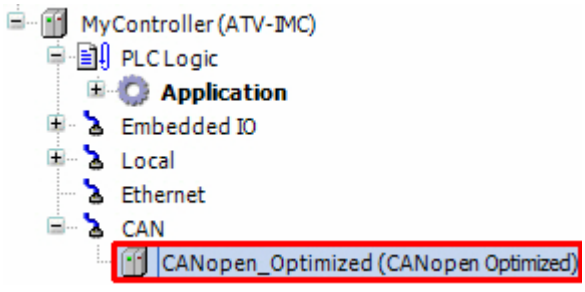
Add an Option Board

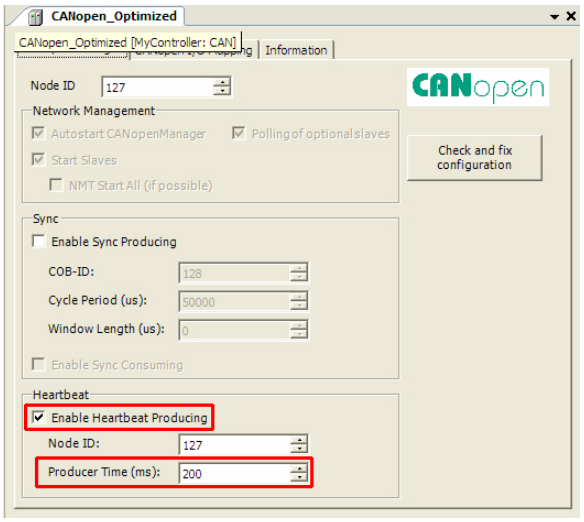
1	To add an option board, in the Device Browser right click on **MyController → Local → Option_Board → <Empty>** and click on **Plug Device**	
2	Select the **IO_Extended** and click on **Plug Device.** **Close** the dialog.	
3	The added option board is shown in the Devices navigator. Double-click on **IO_Extended** to open the Option_Board parameter	

4	The **I/O Mapping** dialog box opens. While the variable names in this dialog box can be changed by the user, the addresses and data types are automatically assigned by SoMachine. However, for your application program use only the variables and not the assigned addresses. To update the variables with the newest I/O data, check **Always update variables.**	

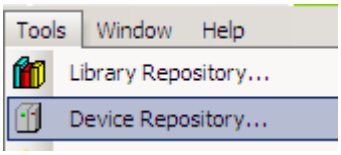
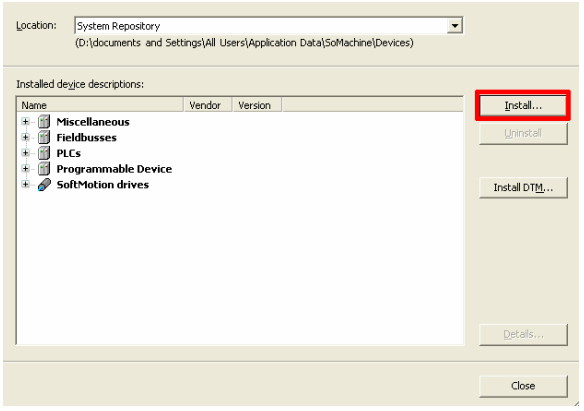
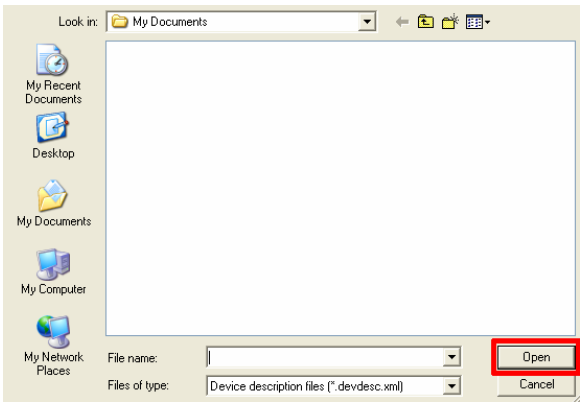
Add the CANopen fieldbus

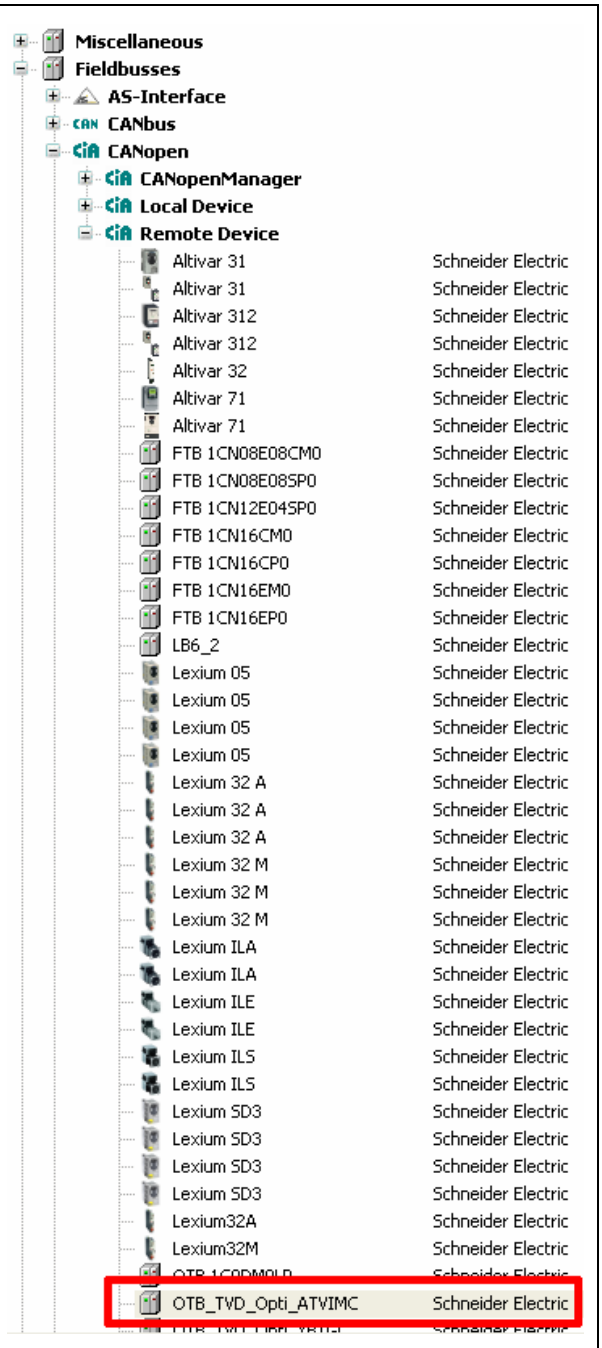
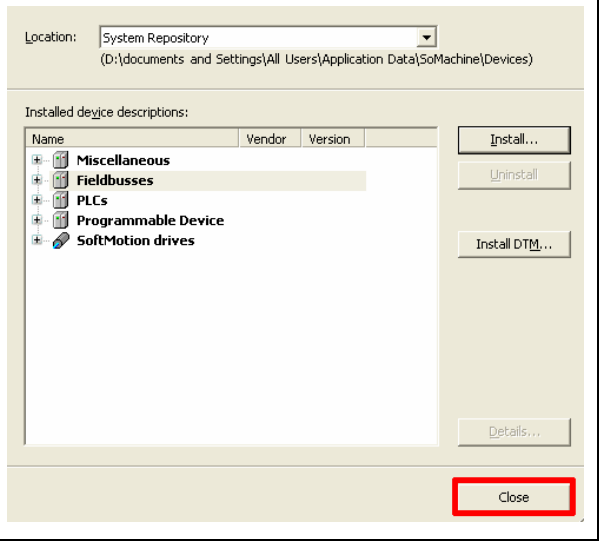
1	In the Devices Browser right click on **CAN** and select: **Add Device...** Note: The assemby of the bus is shown in the chapter on communication.	

2	Select the CANopen master: **CANopen Optimized** Click on **Add Device.**	
3	Double-click on **CAN** in the Devices browser.	
4	On the CANbus Tab, set the **Baudrate** of the CANopen bus, select **500000** as a **Baudrate.**	
5	Double-click on **CANopen_Optimized** in the Devices navigator.	

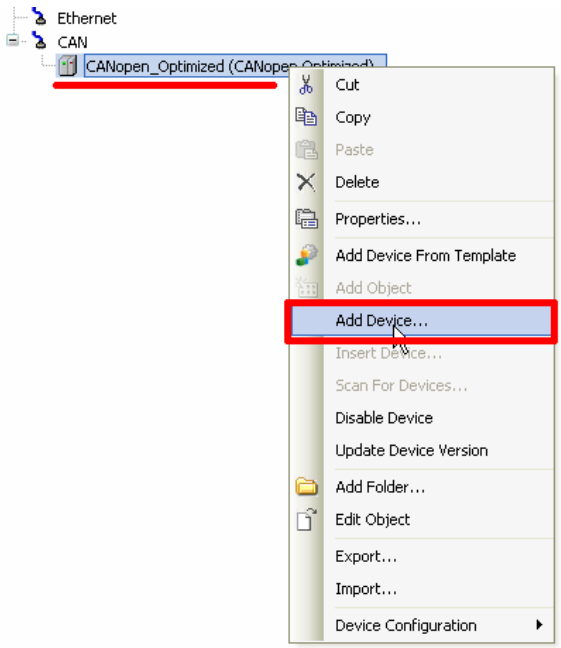
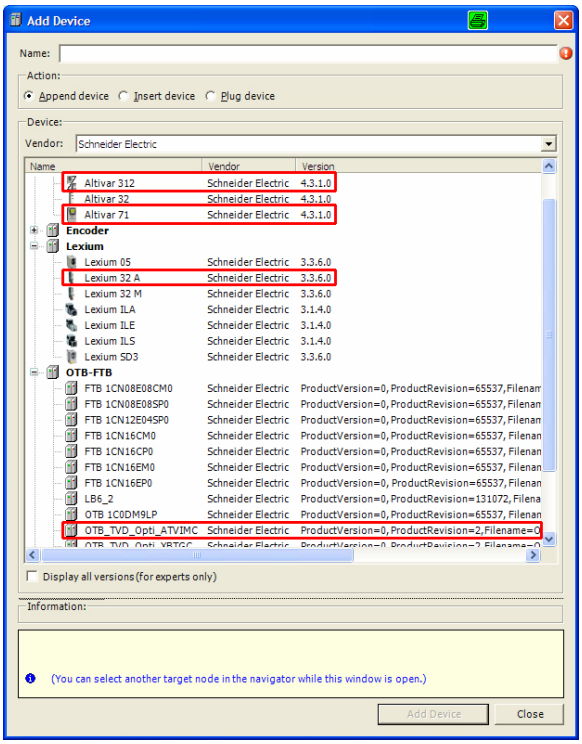
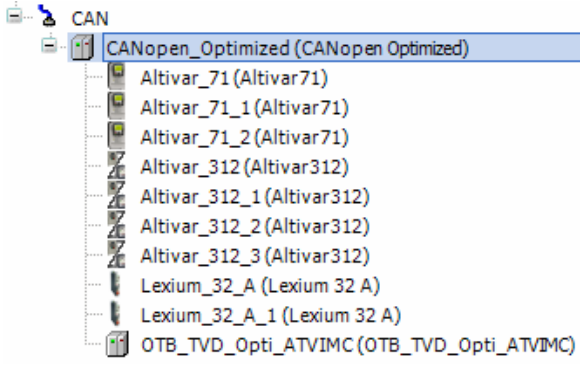
6	To activate the **Heartbeat** of the CANopen fieldbus double-click the **CANopen_Optimized** and check **Enable heartbeat Producing.** The **Node ID** for the CANopen master is **127**. The **Heartbeat Time** is **200** ms.	

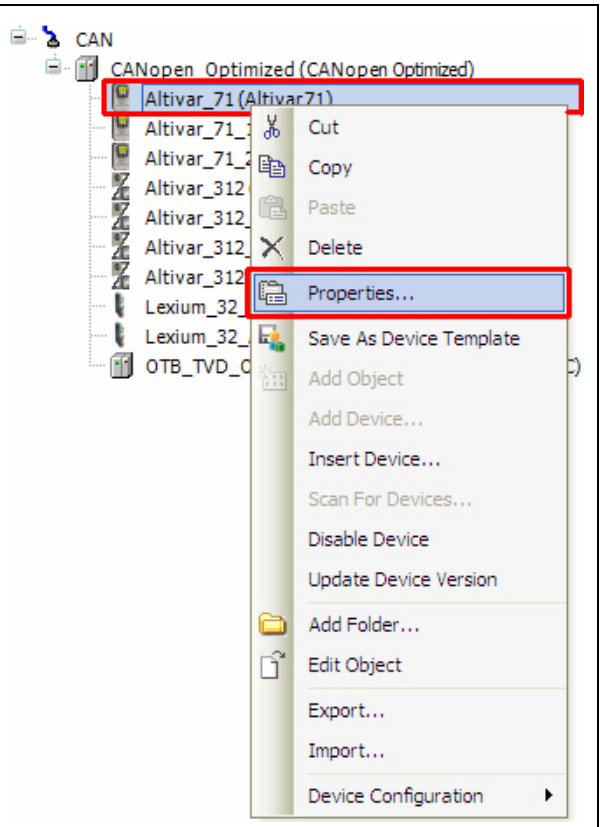
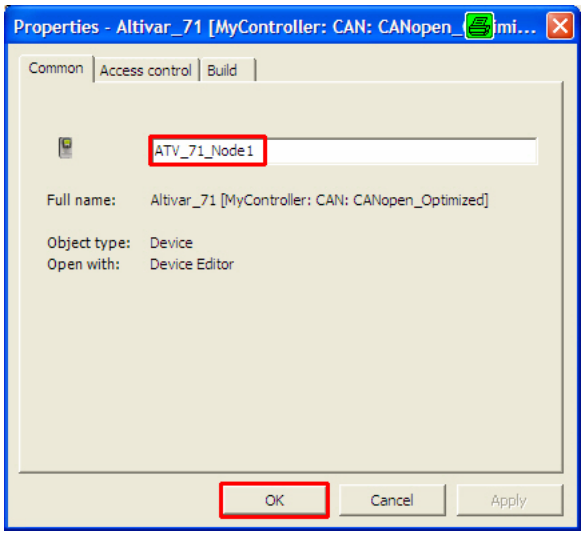
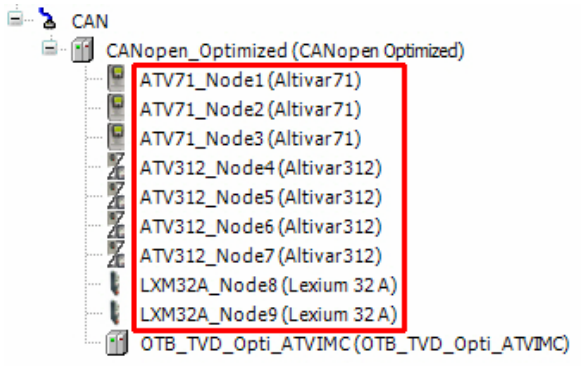
Import the Advantys OTB EDS file

1	To use the extended Advantys OTB island (configured by Advantys Configuration Software) you have to import the Advantys OTB eds file. Select from the main menu **Tools→ Device Repository**…	
2	In the **Device Repository** select **Install…**	
3	Select the Advantys OTB EDS file. The EDS file could be created from the Advantys Configuration Software, described in the Chapter Advantys OTB. Change to the location where the file is stored. In this project the Advantys OTB EDS file is named **OTB_TVD_Opti_ATVIMC.eds** Click on **Open**.	

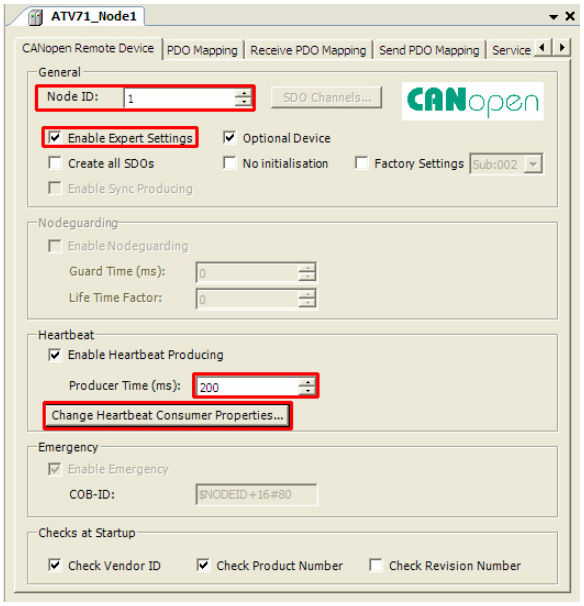
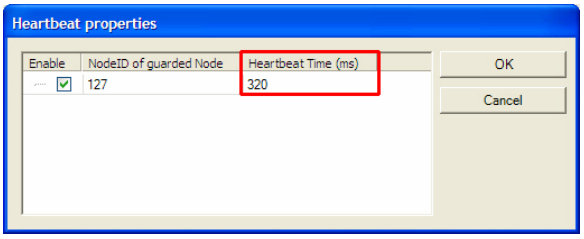
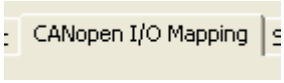
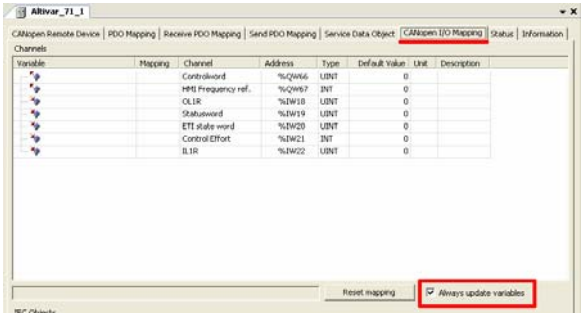
4	**Note:** The used eds file for this TVDA project can be found under **Fieldbusses → CANopen → Remote Device → OTB_TVD_Opti_ATVIMC**	
5	Click on **Close**.	

Add CANopen devices

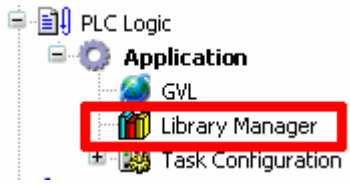
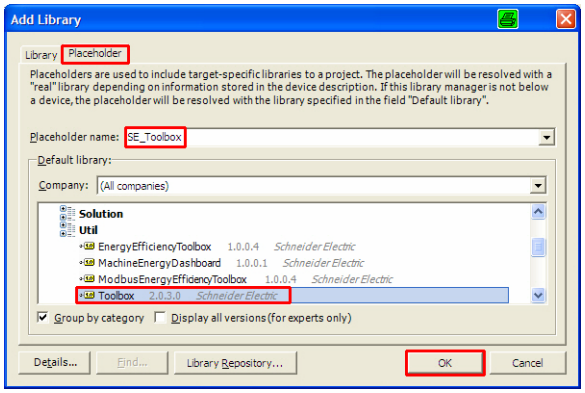
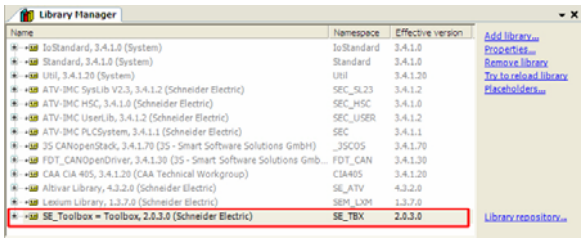
1	In the Devices browser right click on the **CANopen_Optimized** and select **Add Device…** in the pop-up menu.	
2	In this project the following devices are connected to the CANopen bus: **3x Altivar 71** **4x Altivar 312** **2x Lexium 32 A** **1x OTB_TVD_Opti_ATVIMC** Add each device by choosing it and clicking on **Add Device**. Once you have added all the devices click on **Close.**	
3	The new devices are now listed under **CANopen_Optimized** In the devices browser.	

4	To change the name of a device, right click on the device and select **Properties…** from the pop-up menu.	
5	Enter the new name without any blanks and confirm the name change by **OK**. **Note**: The name of the device is also the AXIS REF name for the PLCopen functions used in the application program.	
6	The new names of the devices are now listed under **CANopen_Optimized**. To configure the devices, double click on the specific item.	

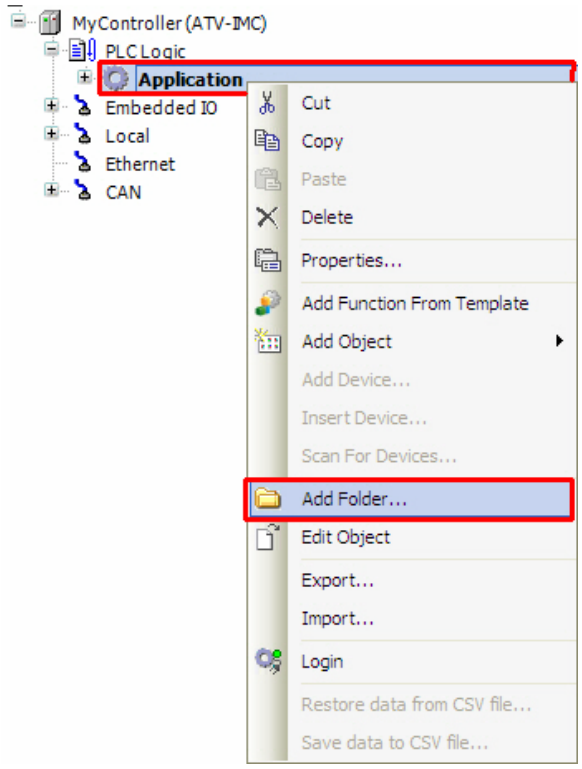
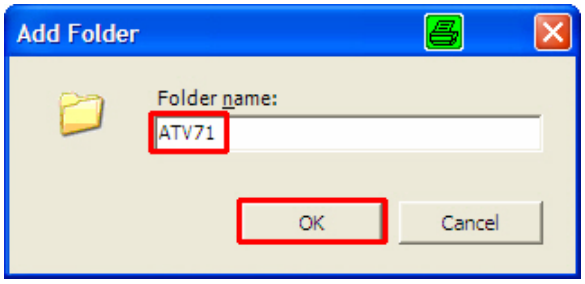
CANopen configuration

1	Double click on the individual device in the devices browser to do the CANopen configuration. Set the **Node ID** for the chosen device. 3x ATV 71, **Node ID: 1..3** 4x ATV 312, **Node ID: 4..7** 2x Lexium32, **Node ID: 8..9** 1x OTB, **Node ID: 10** Check **Enable Expert Setting** Set the Heartbeat **Producer Time** to **200 ms**. Click on **Change Heartbeat Consumer Properties…** and check the time.	
2	Minimum value allowed for **Heartbeat Time** is: **Producer Time** multiplied by 1.5. **Heartbeat Time** and **Producer Time** have to be multiples of the cycle time value. With a cycle time of 40 ms and a producer time of 200 ms this results in: 200 ms x 1.5 = **<u>300 ms</u>** As 300 ms is not a multiple of 40 ms, the **Heartbeat time** needs to be set to **320 ms**. **Note:** These values depend on the application.	
3	Change to Tab **CANopen I/O Mapping.**	
4	Check **Always update variables.** **Note:** All Variables can have a **Global Name** in the field **Variable** and can be used in the Application program.	

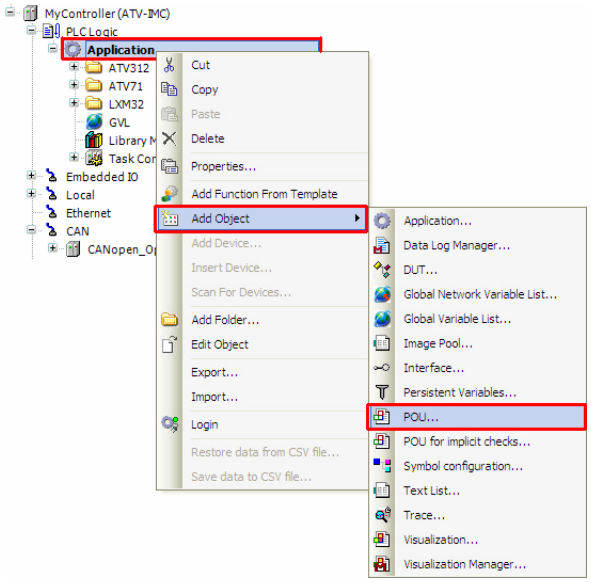
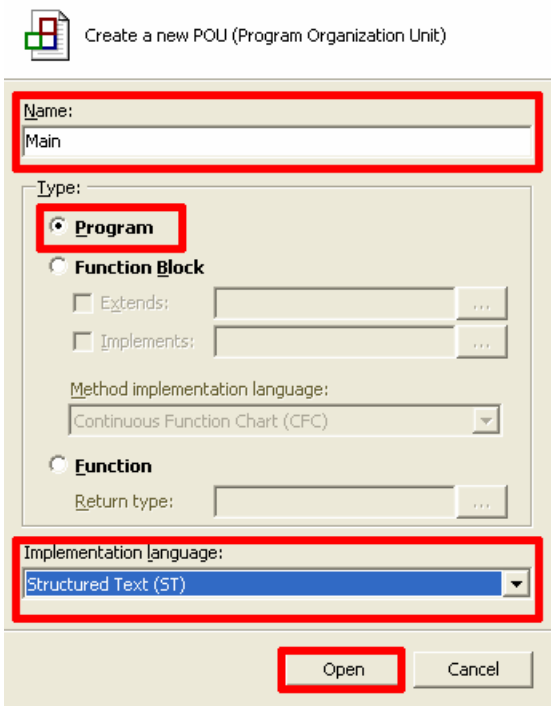
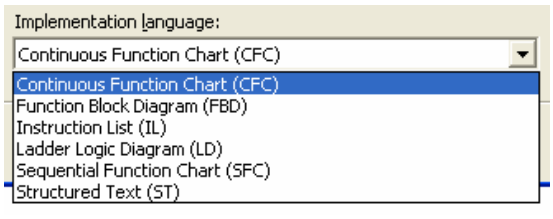
Add Library

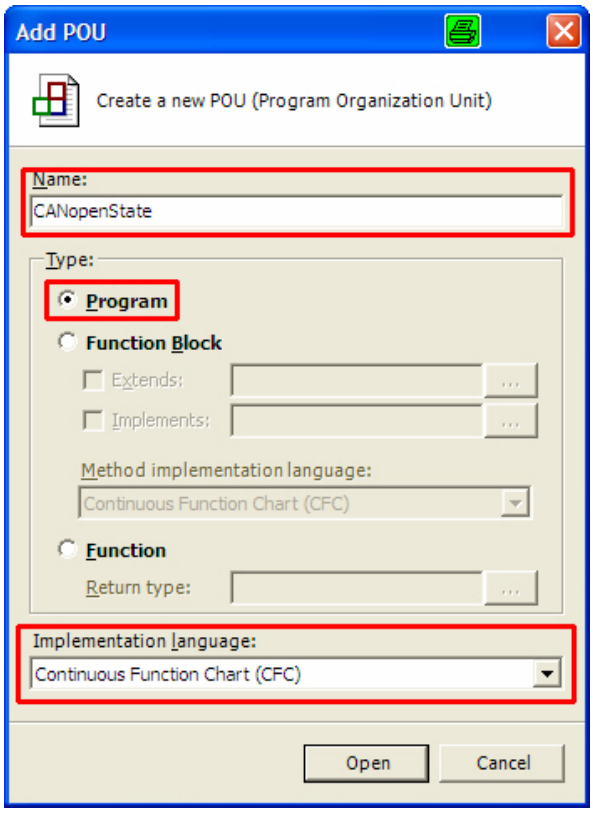
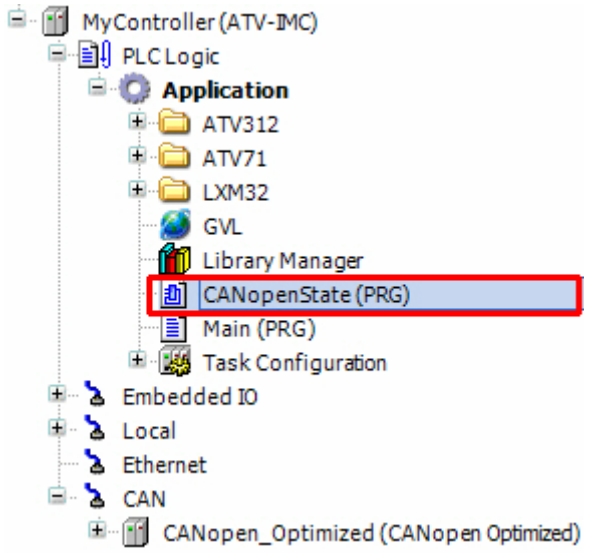
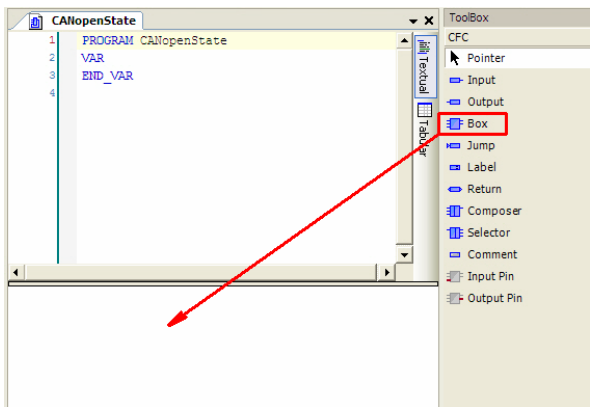
1	To use special functions you need special libraries. These can be inserted by double clicking on **Library Manager** in the devices browser.	
2	In the **Library Manager** click on **Add library…**	
3	In the **Add library** dialog on the **Placeholder** tab select: **Placeholder name→ SE_Toolbox** and as **Default Library** **Util → Toolbox** for the **Toolbox** lib. In each case, click on **OK** to add the library.	
4	The new library is now listed in the **Library Manager**.	
5	If you wish to add more libraries repeat steps 1 to 3. In case you want to add the libraries Conveying, Hoisting or Packaging make sure you have installed the SoMachine Solution Extension on your PC and an S-Type controller is used in your project configuration.	

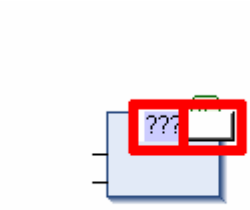
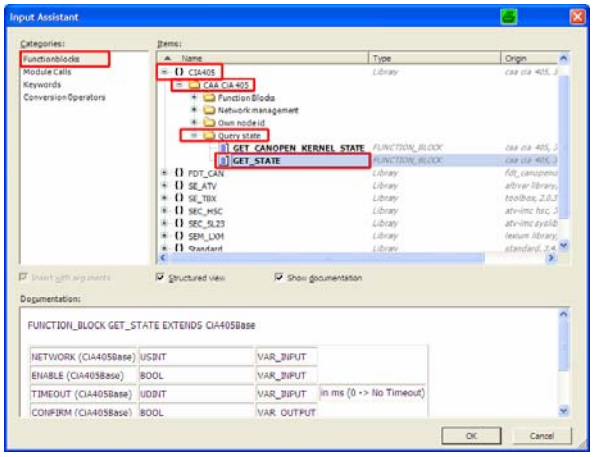
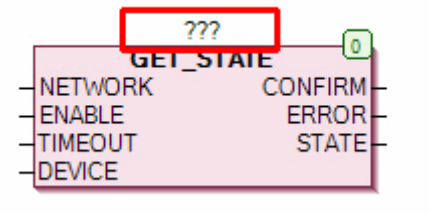
Add Folder

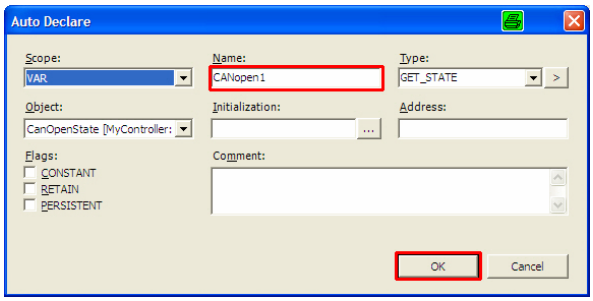
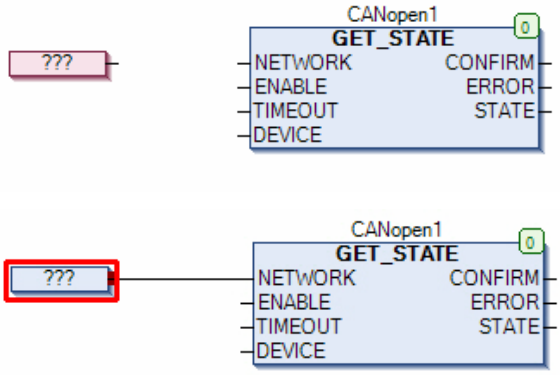
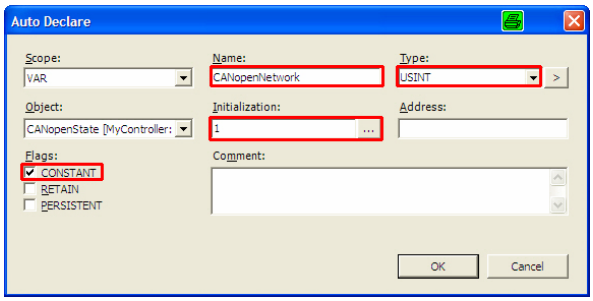
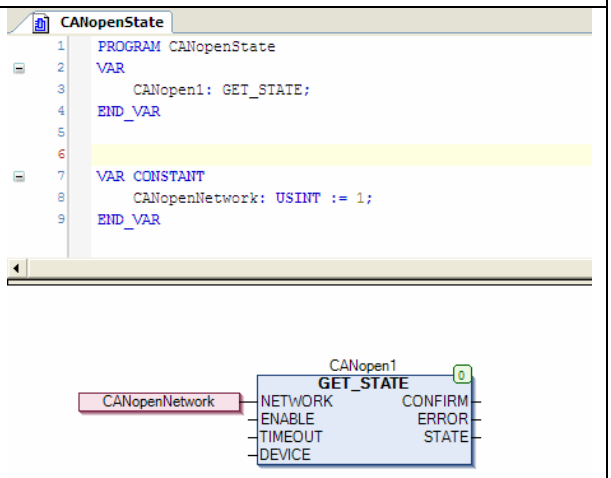
1	In the devices browser right click on **Application→ Add Folder…**	
2	Type in the Folder name: e.g. **ATV71** Click on **OK**.	
3	To include additional folders, repeat steps 1 through 2	

Add POU

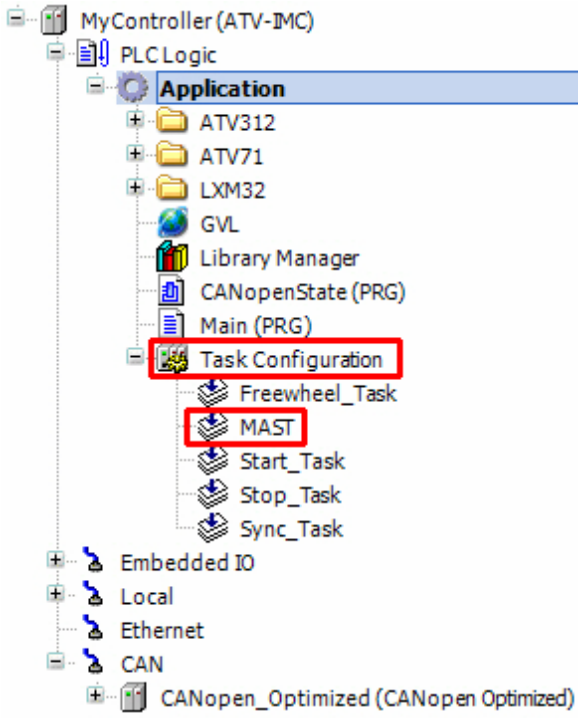
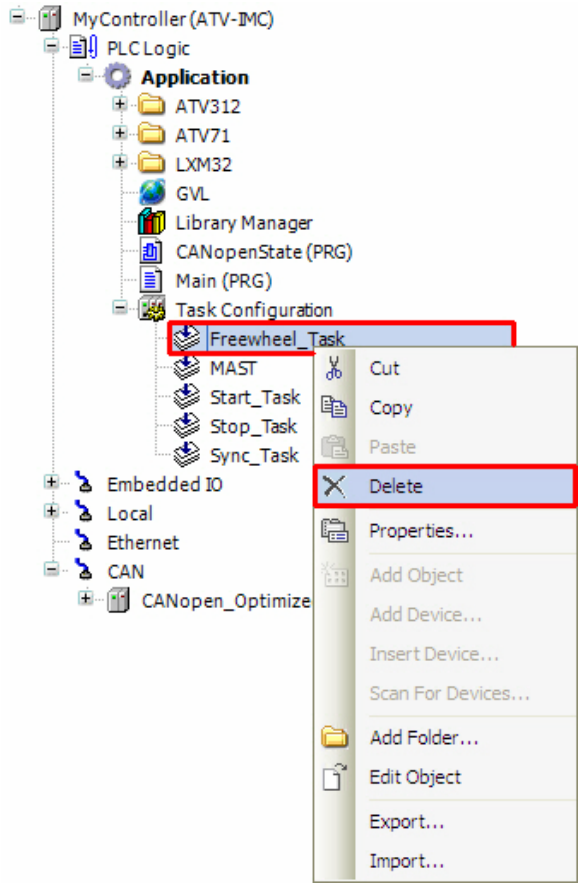
1	In the devices browser right click on: **Application → Add Object → POU…**	
2	Enter the **Name: Main** **Type: Program** **Implementation language:** **Structured Text (ST)** Click on **Open**	
3	**Note**: All the IEC languages can be used for programs, function blocks and functions.	

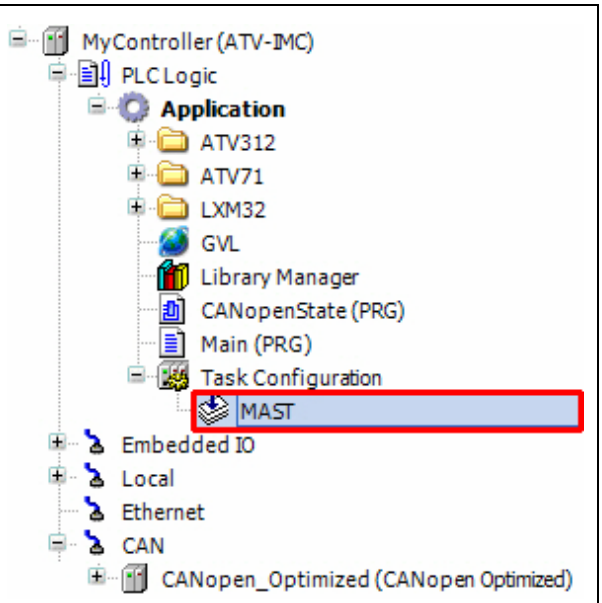
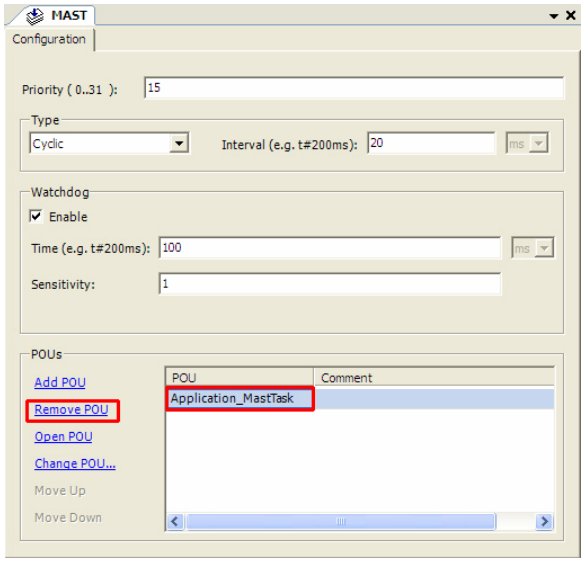
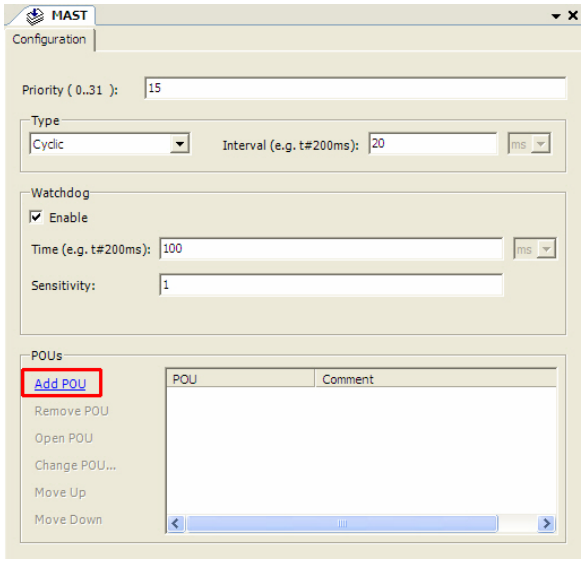
4	Repeat step 1 to add another POU called: **Name: CANopenState** **Type: Program** **Implementation language:** **Continuous Function Chart (CFC)** Click on **Open**.	
5	The new **CANopenState** POU is now visible under **Application**. Double click on **CANopenState** to open it.	
6	The upper frame displays the declaration section. The lower frame is for programming. On the right side is the **ToolBox**. Use drag and drop with the **ToolBox** to place example templates in the programming section. E.g. **Box** for function or function blocks.	

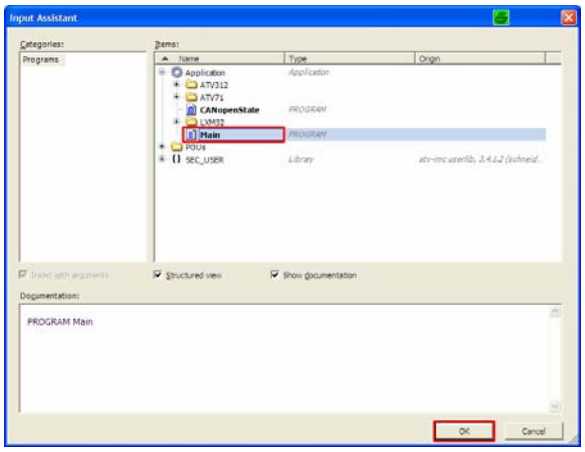
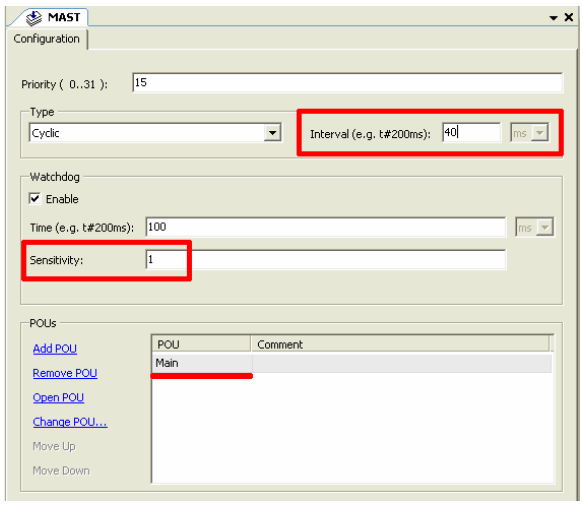
7	Once you have placed a template in the programming section click on the Function block name (**???**). A white box is displayed next to the ??? Click on the white box.	
8	This will invoke the **Input Assistant** window Select **Categories:** **Functionblocks** **Items:** **CIA405 →** **CAA CIA 405 →** **Query state →** **GET_STATE** Click on **OK.**	
9	The same, **GET_STATE**, will appear as the FB. The **???** indicates that this function block is still not instantiated. Click on **???**. Type in a user defined variable name for the function block. Finalize the input by pressing Enter. In this case the variable name is **CANopen1**.	

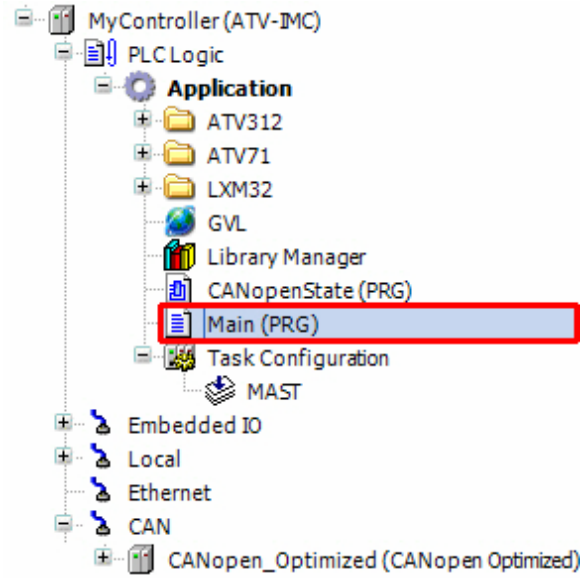
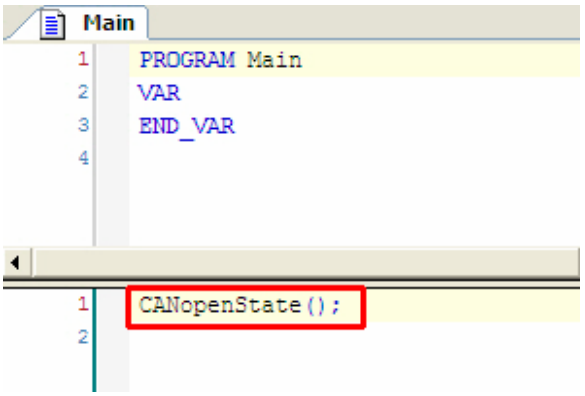
10	The Auto Declare window opens. **CANopen1** is visible under **Name**. Click on **OK**.	
11	Drag and drop an **Input** from the ToolBox and place it at the first input of the function block. The connection between the two is done with the mouse. Click on **???** and type in a user defined variable name. **Name: CANopenNetwork**	
12	The Auto Declare window opens. **CANopenNetwork** is visible under **Name:**. As **CANopenNetwork** will be used as constant, mark the checkbox **CONSTANT** under **Flags:**. For the value of this new constant **CANopenNetwork** type in **1** in the **Initialization** filed. Click on **OK**.	
13	Based on the steps before, complete your application.	

Task Configuration

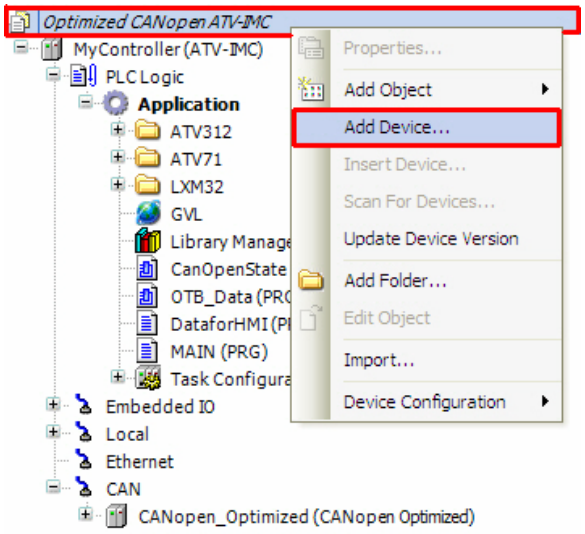
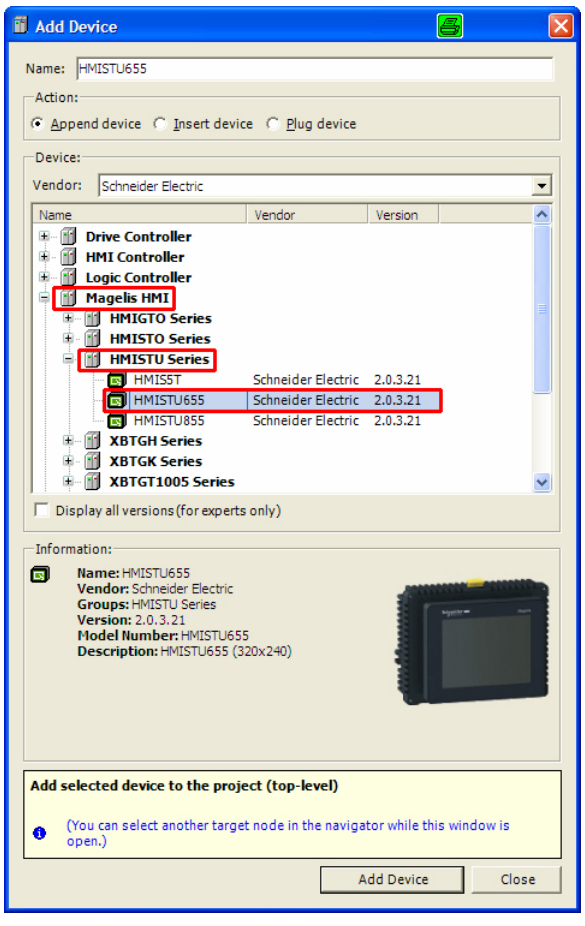
1	In the **Mast** task of the **Task Configuration** there must be at least one POU, otherwise no program code will be invoked cyclically. Other tasks than the MAST task are not needed and can be deleted from here.	
2	To do so, right click on **Freewheel** task you like to delete and select from the pop-up menu **Delete**. Delete also the following tasks in the same way: • **Start_Task** • **Stop_Task** • **Sync_Task**	

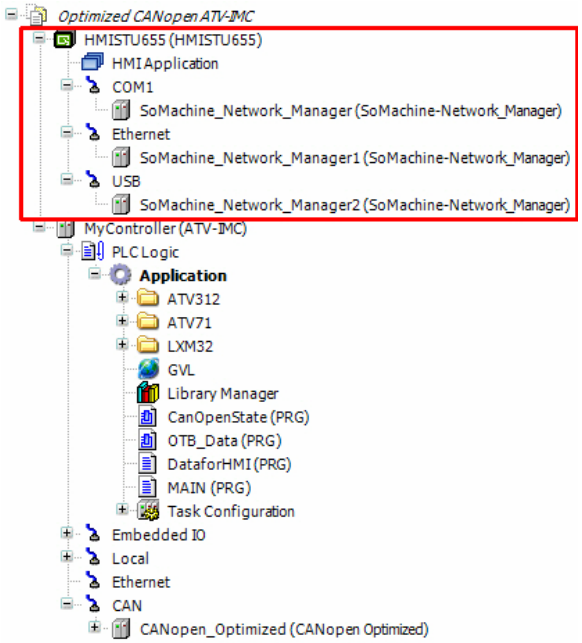
3	Double click on **MAST** in the devices browser.	
4	In the **MAST** task is already one POU called **Application_MastTask** predefined. To remove this item, click on **Application_MastTask** and click on **Remove POU**.	
5	**Click on Add POU**.	

6	In the Input Assistant select **Categories: Programs** **Items: Application → Main** Click on **OK**.	
7	Now the **Main** POU is in the **MAST** task. Change the following parameters: **Interval: 40** **Sensitivity: 1**	
8	**Note:** The **Interval** time for one cycle is connected to the Heartbeat check of the CANopen. The Heartbeat must be always a multiple of the **Interval** time. E.g.: **Interval** = 40ms => Heartbeat **Producer Time** = 200ms and Heartbeat **Consumer Time** = 320ms (>Producer and x*Interval) (See CANopen configuration ATV71)	

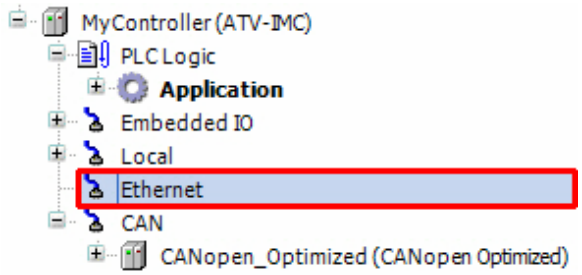
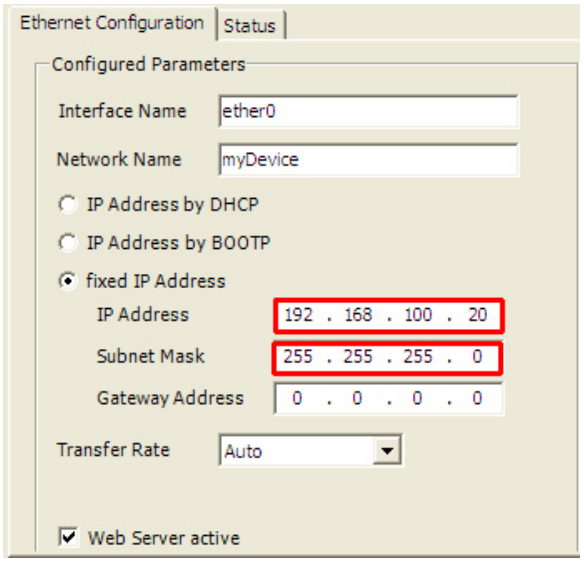
9	The other POUs will be invoked by the Main POU. Double click on **Main (PRG)** in the devices browser.	
10	Enter in body of the POU a function call: e.g.: **CANopenState ( );** Now the Main program calls the CANopenState program. Proceed in the same way with any additional program or function you would like to call.	

Add Vijeo Designer HMI

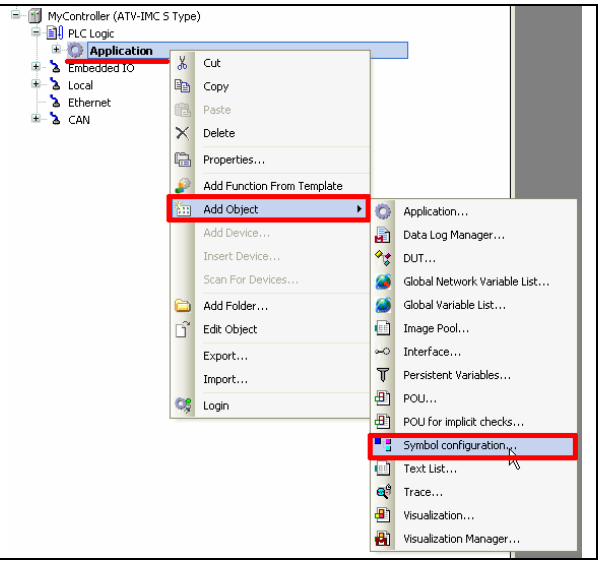
1	To add an HMI unit to the project right click on **Optimized_CANopen_ATV-IMC** in the devices browser and select: **Add Device…** in the drop down menu.	
2	Select: **Magelis HMI → HMISTU Series → HMISTU655** and click on **Add Device**	

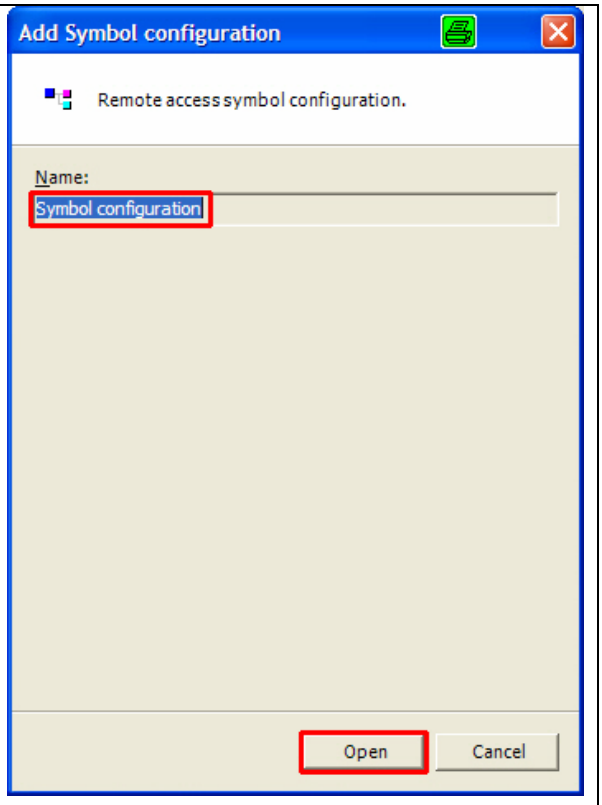
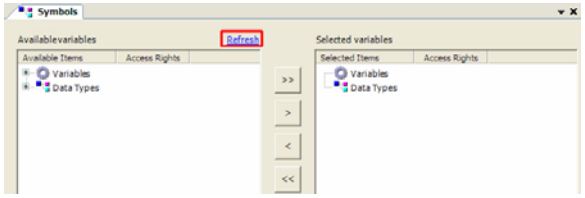
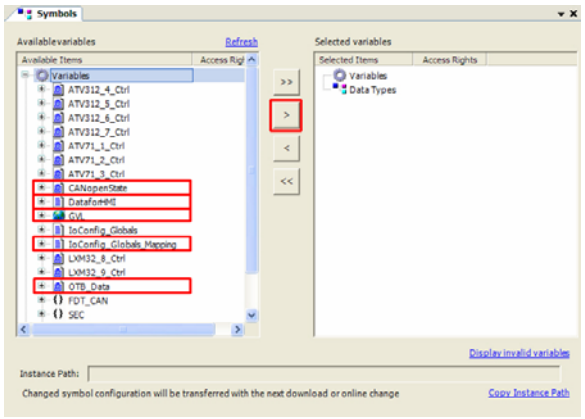
3	The new HMI **HMISTU655** is now listed in the devices browser. And the program Vijeo Designer opens for you to start programming. (See chapter HMI)	

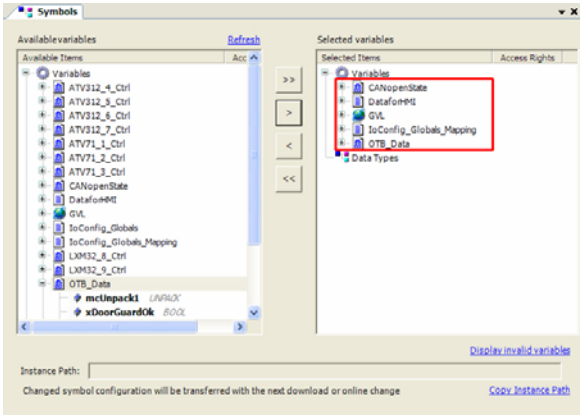
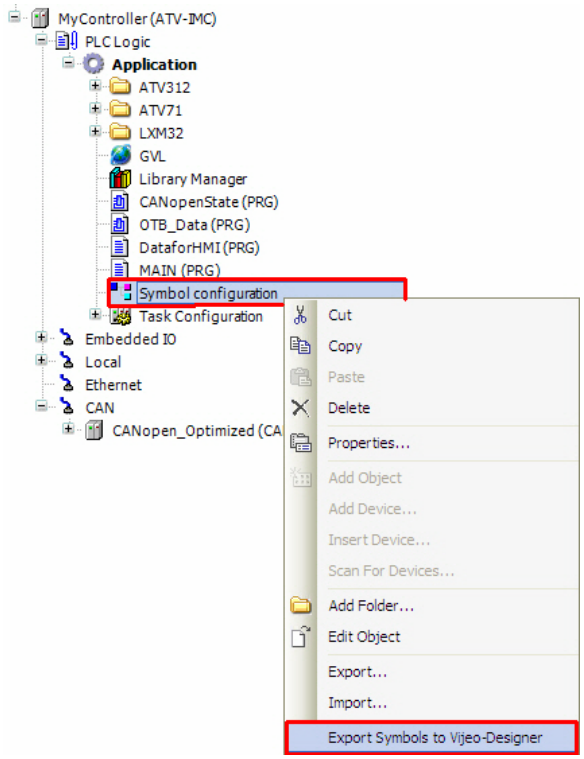
Ethernet settings

1	To change the Ethernet settings double click on **Ethernet** in the devices browser.	
2	Chose the **fixed IP Address** radio button and set an **IP Address** (in this project **192.168.100.20**) and a **Subnet Mask** (in this project **255.255.255.0**) **Note:** The USB cable **TCSXCNAMUM3P** must be used for the initial project download. For subsequent downloads, the Ethernet connection can be used.	

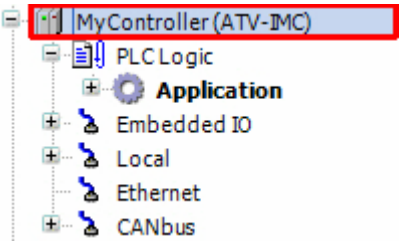
Configure Controller ↔ HMI Data Exchange

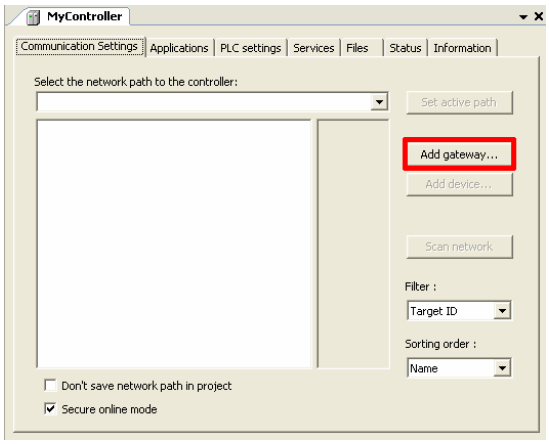
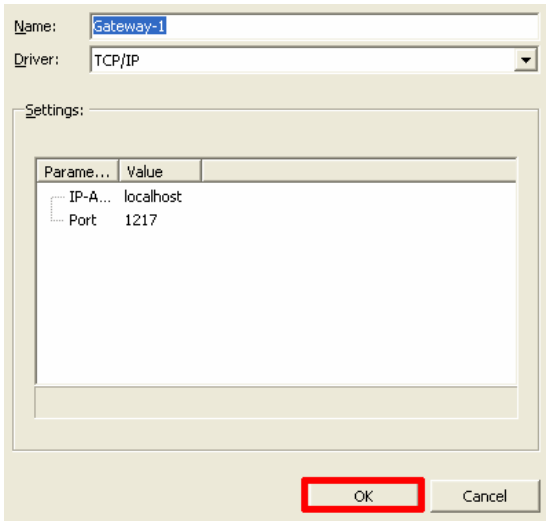
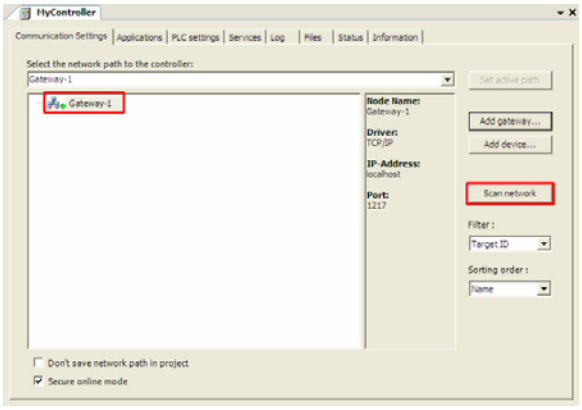
1	In the devices browser right click on: **Application →Add Object → Symbol configuration**	

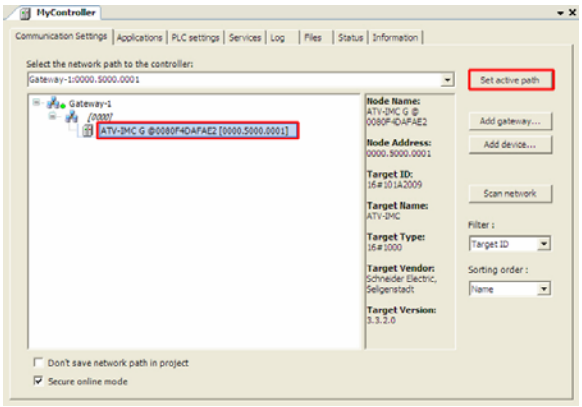
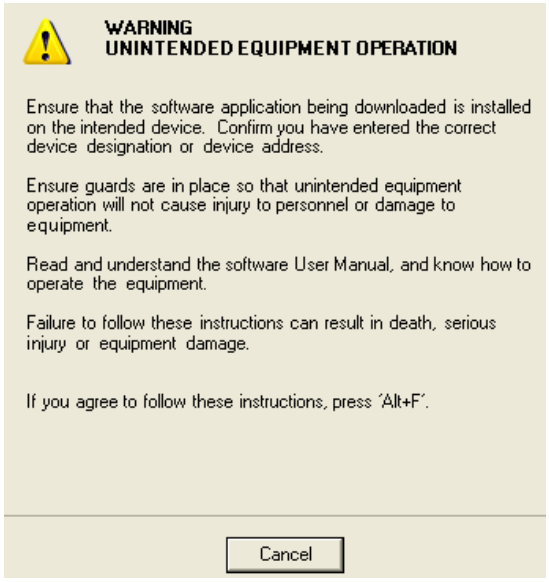
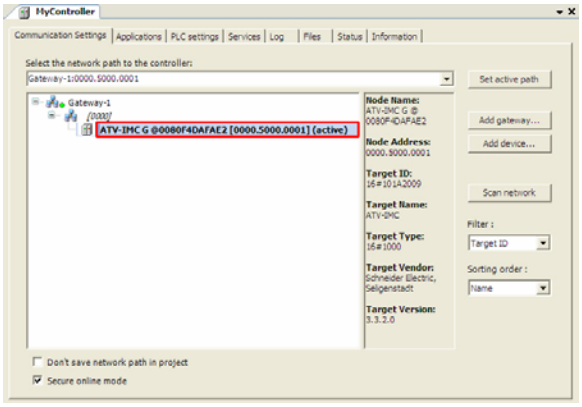
2	The name of the **Symbol configuration** is predefined and cannot be changed. Click on **Open** to proceed.	
3	Click on **Refresh** in the now open symbol configuration.	
4	All the variables created in the user program are shown in the **Available variables** list. The global variables are located in the **GVL** folder. To export variables to the HMI, select them and click on **>**. In this example following variables groups are selected: • **CANopenState** • **DataforHMI** • **Ioconfig_Globals_Mapping** • **OTB_Data**	

5	The right frame lists the **Selected variables** which are to be used in the HMI.	
6	Right click on **Symbol configuration** and select **Export Symbols to Vijeo-Designer** in the drop down menu to export the variable-list.	

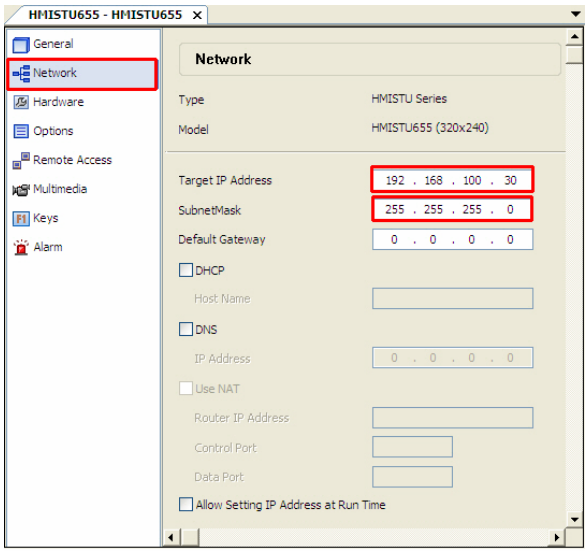
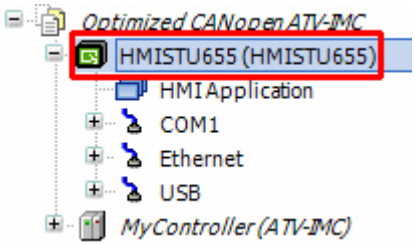
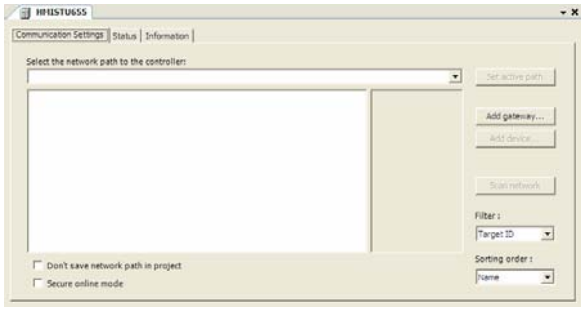
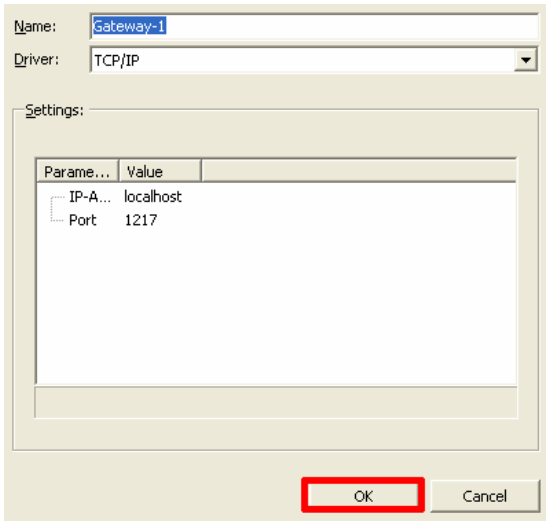
Communication setting controller ↔ PC

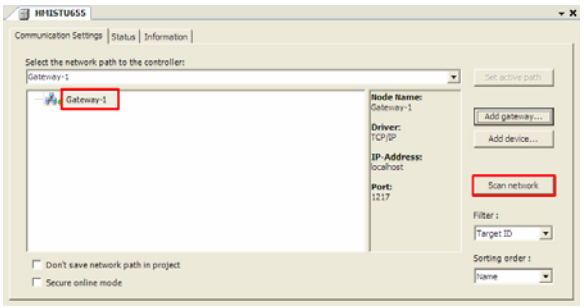
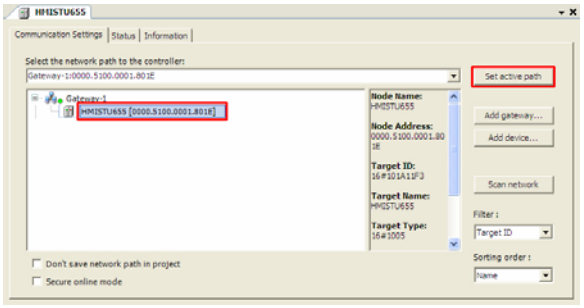
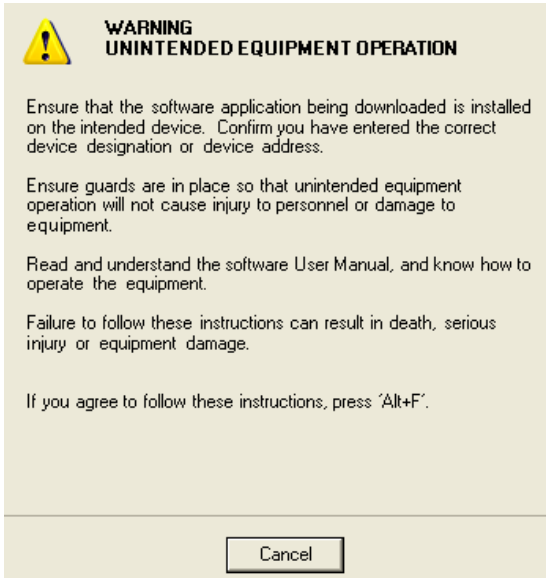
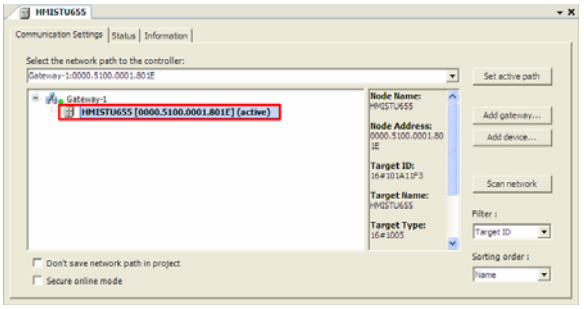
1	In SoMachine double click on **MyController** in the devices browser.	

2	On the **Communication settings** tab click on **Add gateway…**	
3	Retain the factory settings and and click on **OK**	
4	Select **Gateway-1.** Click on **Scan network.**	

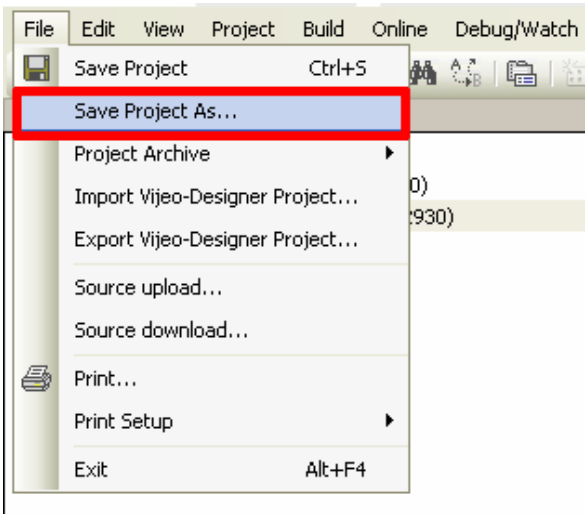
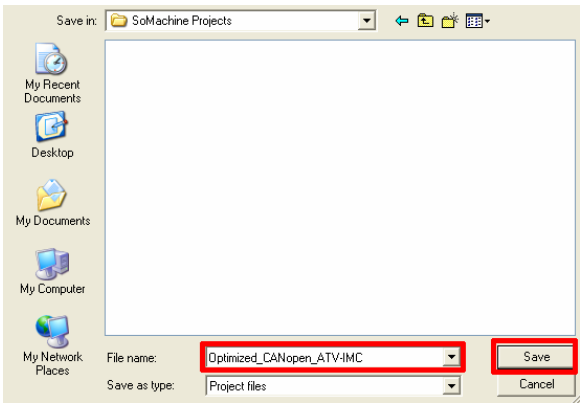
5	During the scan, the **Scan network** button becomes inactive. When the scan is finished, the **Scan network** button becomes active again and the devices that have been detected are listed under **Gateway-1.** Select the controller that is being used and click on **Set active path**.	
6	A hazard message appears. Read the message and confirm to continue**.**	
7	The controller used is now marked as **active.** This means that the online functions are available (e.g. program download, monitoring, online change)	

Communication setting controller ↔HMI

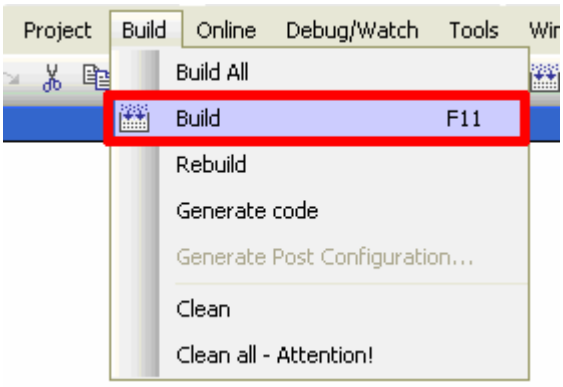
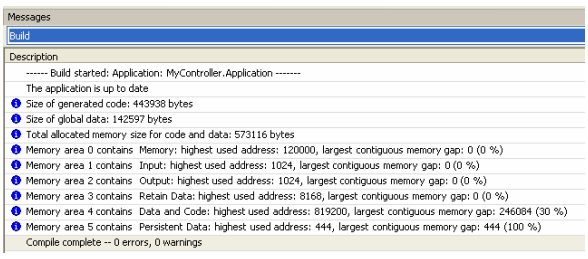
1	In **Vijeo Designer** select the HMI in the Project browser. The HMI is connected with the controller via Ethernet. In the **Network** area enter the **Target IP Address** (in this project 192.168.100.30) and the **SubnetMask** (in this project 255.255.255.0) of the HMI.	
2	In **SoMachine** double click on **HMISTU655** in the devices browser.	
3	On the **Communication settings** tab click on **Add gateway…**	
4	Retain the factory settings and and click on **OK**.	

5	Select **Gateway-1**. Click on **Scan network**. **Note:** In order to be able to connect to the HMI via ATV-IMC by using Ethernet, the project needs to be downloaded the first time by a direct connection between PC and HMI by using the USB **TCSXCNAMUM3P** download cable.	
6	During the scan, the **Scan network** button becomes inactive. When the scan is finished, the **Scan network** button becomes active again and the devices that have been detected are listed under **Gateway-1.** Select the HMI that is being used and click on **Set active path**.	
7	A hazard message appears. Read the message and confirm to continue**.**	
8	The HMI used is now marked as **active.** **Note:** Whenever the connection will be changed (e.g., from USB connection to Ethernet link) the steps 5 to 8 need to be repeated.	

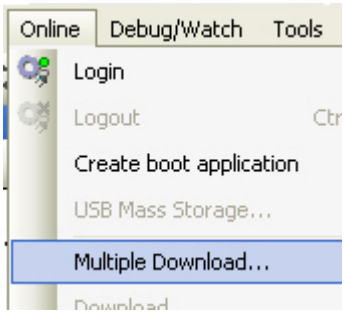
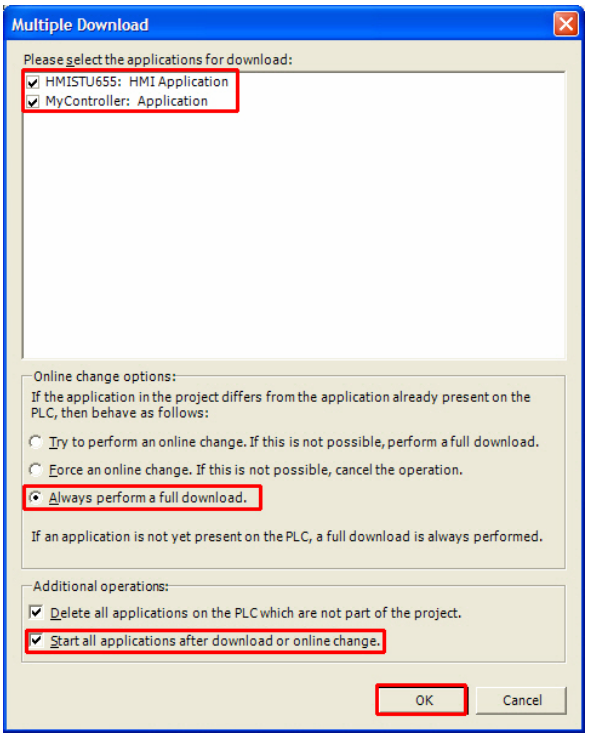
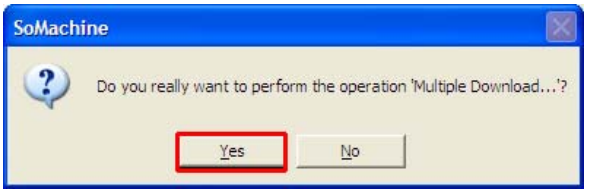
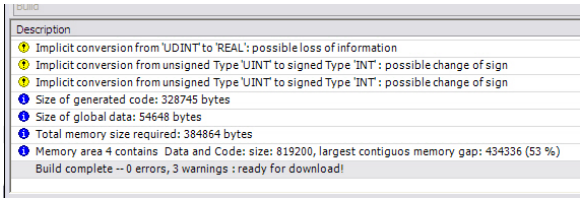
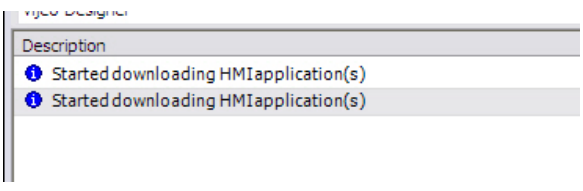
Save the project

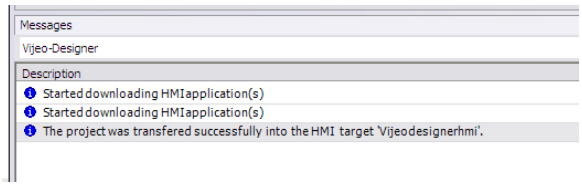
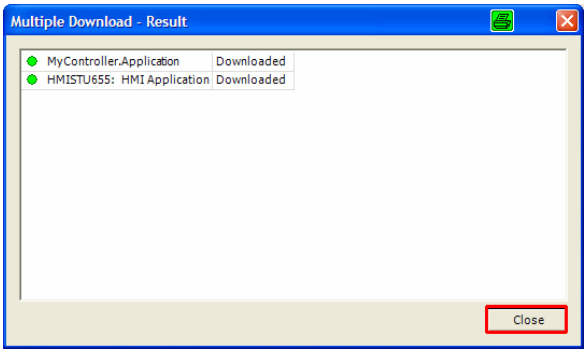
1	To save the project and change the name select: **File → Save Project As…**	
2	Select the desired location and input a file name. In this case the file name is **Optimized_CANopen_ATV-IMC** Click on **Save.**	

Build Application

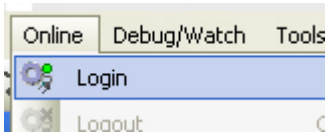
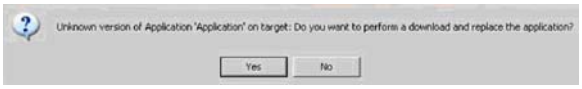
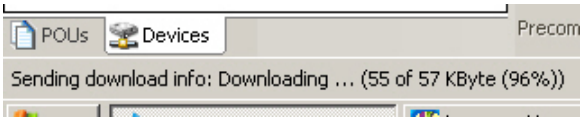
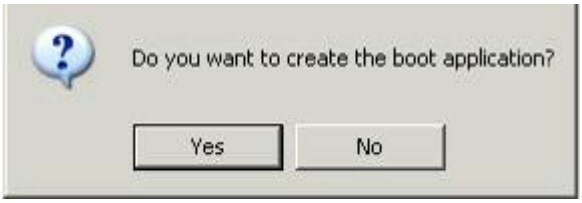
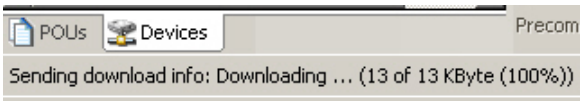
1	To build the application select: **Build → Build** **Note:** If you wish to build the whole project (HMI and Controller) select **Build → Build All**	
2	After the build you are notified in the **Messages** field as to whether the build was successful or not. If the build was not successful there will be a list of detected warnings or errors in the **Messages** field.	

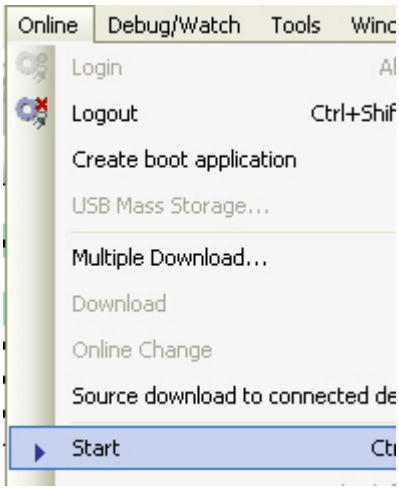
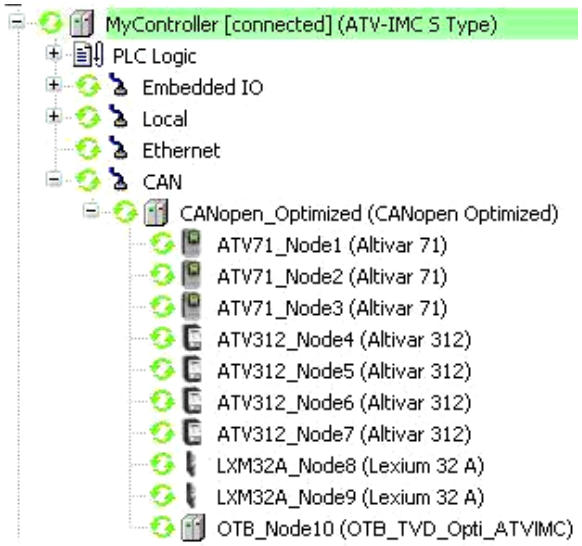
Download the controller and HMI program

1	To download the application to the controller and the HMI in SoMachine select on: **Online → Multiple Download…**	
2	Check the **HMI** and the **controller application**. For the first download it is recommencded to activate the option **Always perform a full download**. To start the application after download, check the option **Start all applications after downloading or online change**. Click on **OK**.	
3	Another dialog box pops up. Read the question and click on **Yes** to perform the Multiple Download operation.	
4	Before the download starts a build of the complete project is done. The result of the build is displayed in the **message** window.	
5	Once the download to the controller is finished, the HMI download starts.	

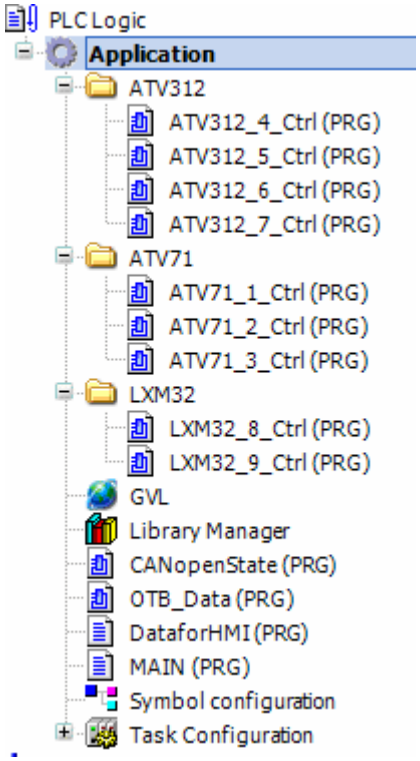
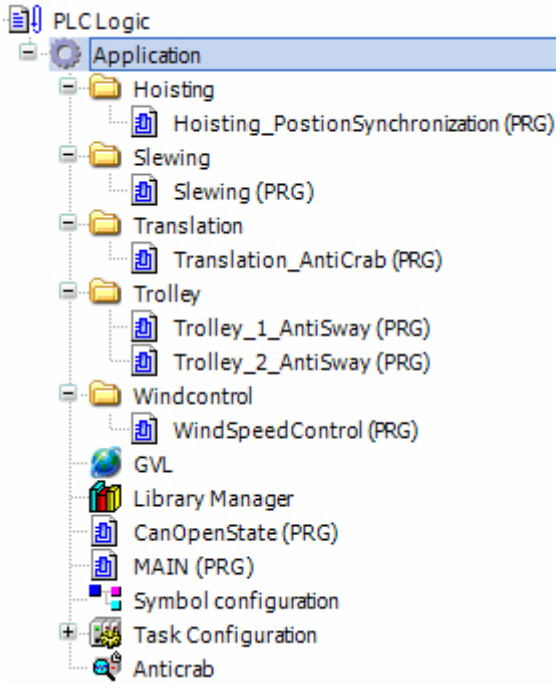
6	The result of the HMI download is displayed in the **Messages** window.	
7	The results of the download are displayed in the **Multiple Download - Result** window. Click on **Close** to close the results window**.**	

Login to controller

1	To log in to the controller select: **Online → Login**	
2	If the controller program is different from the program on the PC a message asks you if you wish to replace the old controller program. If you do not wish to replace the controller program continue with step 6, otherwise click **Yes** to confirm the download**.**	
3	The actual download status is displayed at the bottom of the main window.	
4	You are offered to option to create a boot project. A boot project is stored in FLASH memory so that a power loss does not mean you have to repeat the download on re-start. Select **Yes** to create a boot application.	
5	The actual creation status is displayed at the bottom of the main window.	

6	To start the new Application select **Online → Start**	
7	If everything is running properly the devices and folders are marked in green otherwise they will be marked in red.	

Application overview

1	With the SUG two example application codes are available. The first is generic, the second is based on the hoisting AFB Library.	
2	The picture on the right shows the structure of the Generic Application example code. The program is split into function-groups with folders.	
3	The picture on the right shows the structure of the Hoisting Application example code. The program is split into function-groups with folders.	

HMI

Introduction

This application uses a Magelis HMISTU655. This HMI device communicates via the SoMachine protocol on Ethernet with the controller. The HMI is programmed using the software tool Vijeo Designer (delivered with SoMachine), described briefly in the following pages.
The PC with SoMachine is connected via **USB cable TCSXCNAMUM3P** to the ATV-IMC, and from there via Ethernet to the HMI panel using the **Ethernet cable 490NTW00002**.

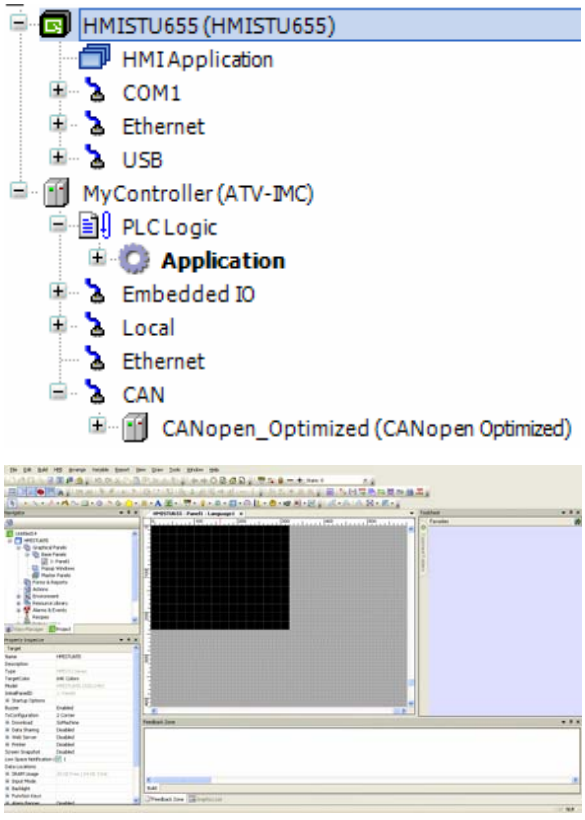
Note:
The Vijeo Designer tool is opened and closed via the SoMachine software. For more information see the chapter:

Controller: Add Vijeo Designer HMI

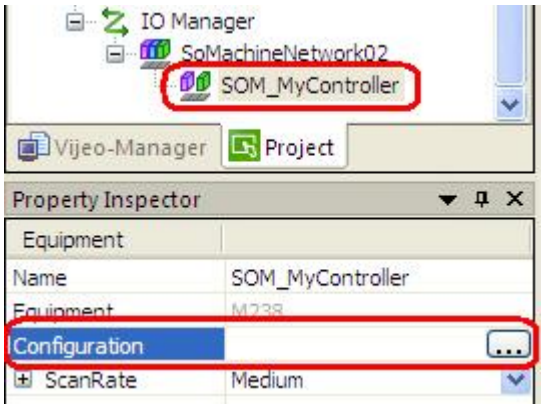
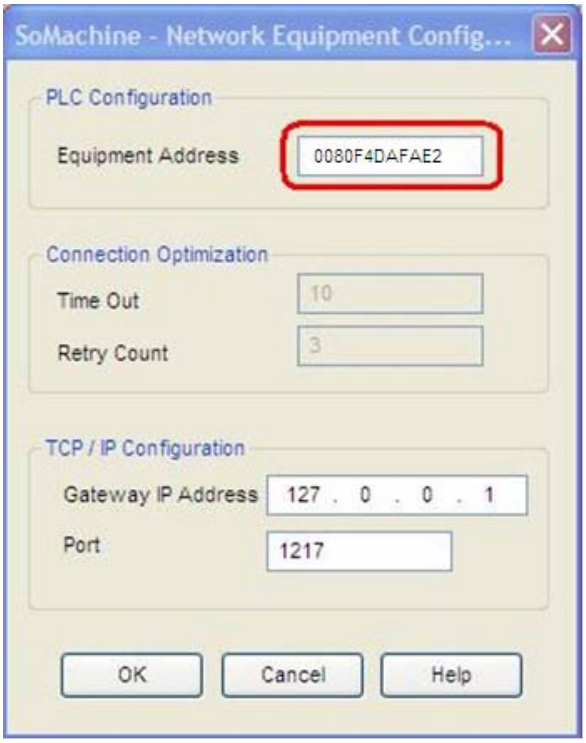
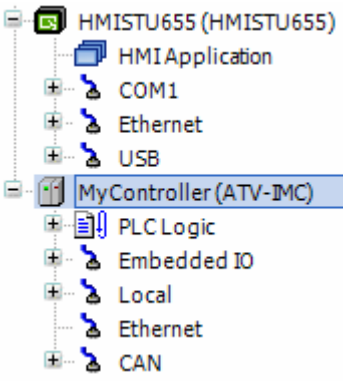
Setting up the HMI is done as follows:

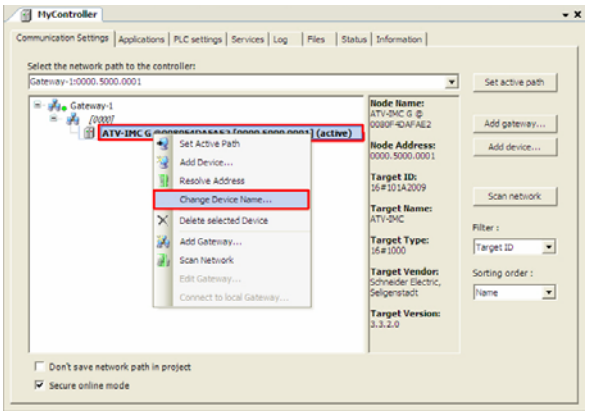
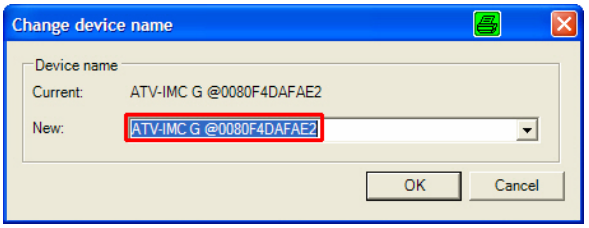
- Main Window
- Communication settings
- Create a switch
- Create a numeric display
- Example screens Generic
- Example screens Hoisting

Main Window

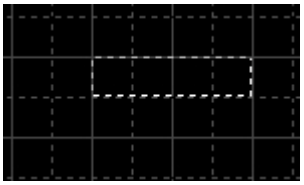
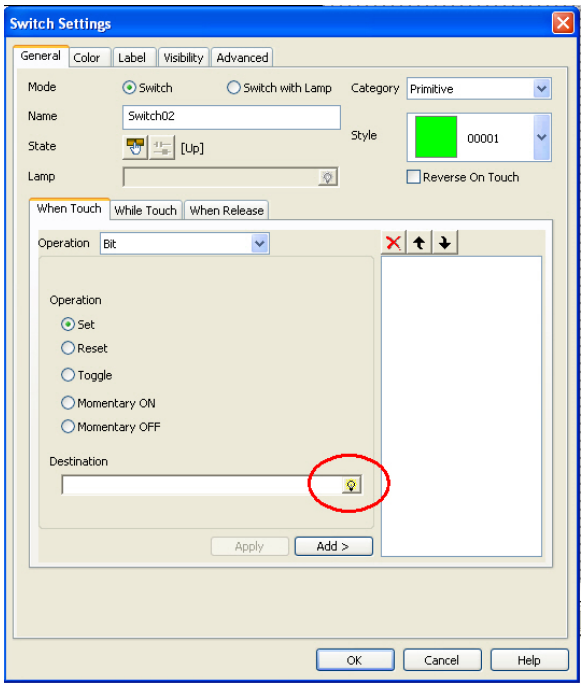
1	After double clicking on the **HMISTU655** in the SoMachine devices browser, Vijeo Designer opens the HMI main window.	

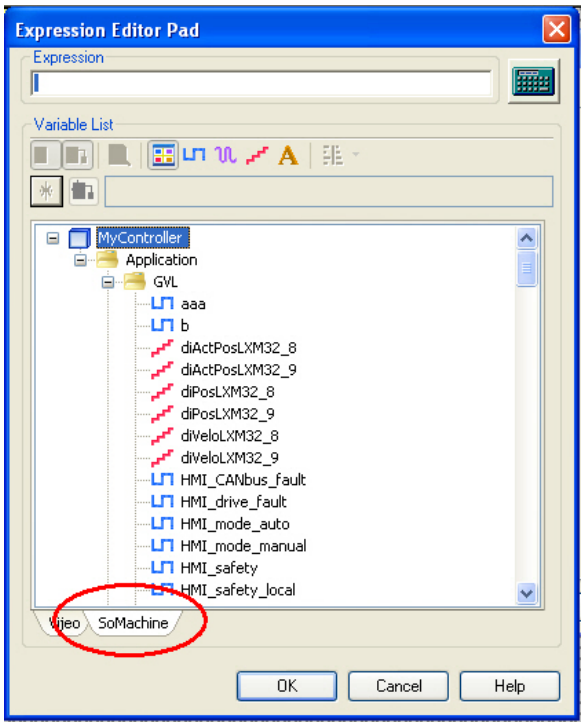
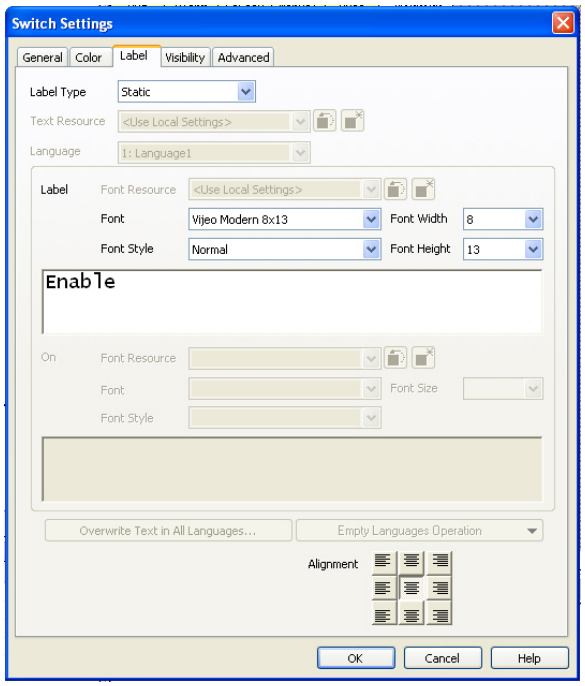
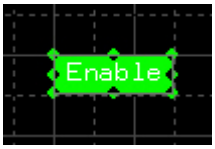
Communi-cation settings

1	To set the communication parameters, in the Navigator select **IO Manager → SoMachineNetwork02 → SOM_MyController** and click on the browser button for the configuration in the Property Inspector	
2	In the dialog set the controller **Equipment Address.** You will find this address in SoMachine… (see next step)	
3	… by double clicking the **MyController** entry in the devices browser.	

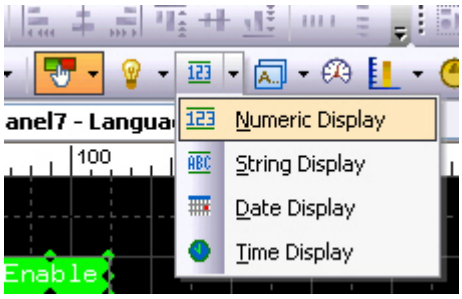
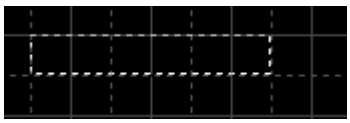
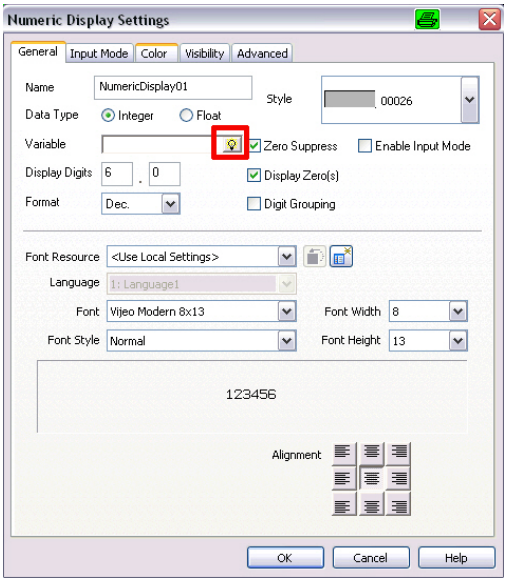
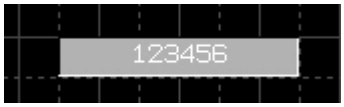
4	In the **Communication Settings** tab right click on the controller and select **Change Device Name…**	
5	The Equipment address of the controller is displayed under **Device Name → Current:** and can be changed under → **New:**	

Create a switch

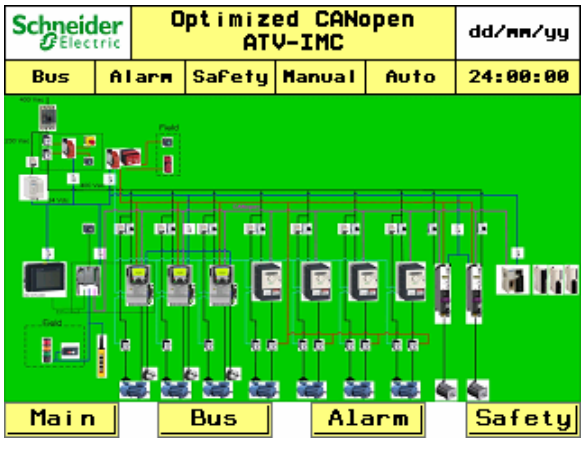
1	Select the **Switch** icon in the **Tool bar.**	
2	Select the position and dimension where you wish to place the button by opening a rectangle on the display and pressing enter.	
3	In the **Switch Settings** dialog, select the variable that should be linked use the bulb icon to browse for a variable to the button.	

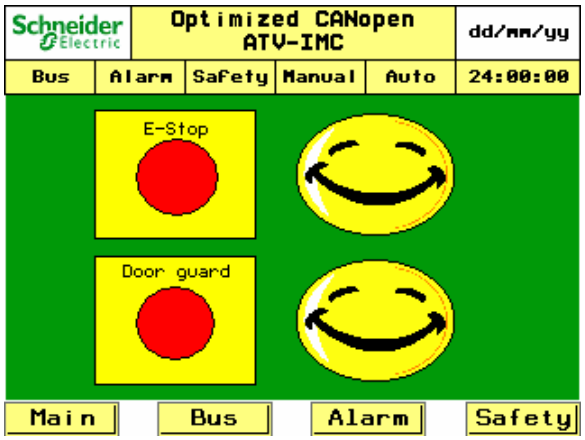
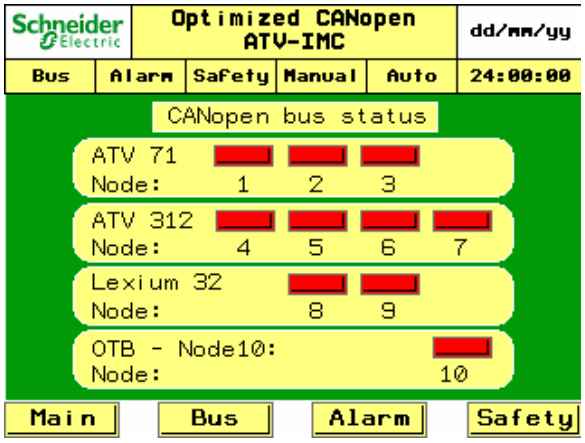
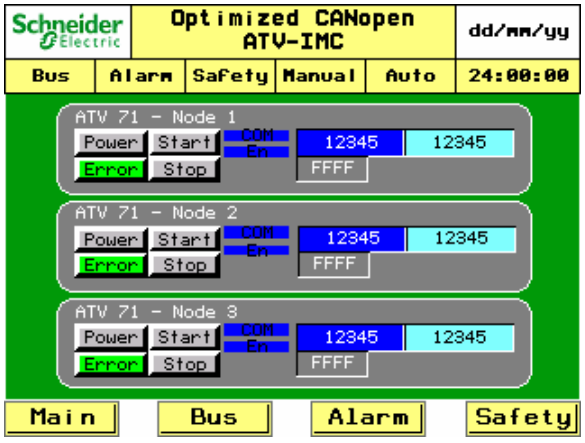
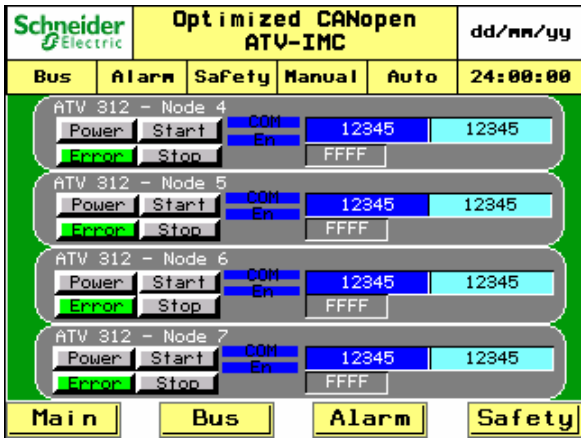
4	Click on the bulb icon (as indicated in the image above) to open the **Variables List**.. Use the tab **SoMachine**. Select the required variable and click **OK.**	
5	Go to the **Label** tab**.** Here select **Label Type: Static** and enter a name for the button, e.g. **Enable.** Once you have finished your settings click on **OK**.	
6	The display now shows the new button.	

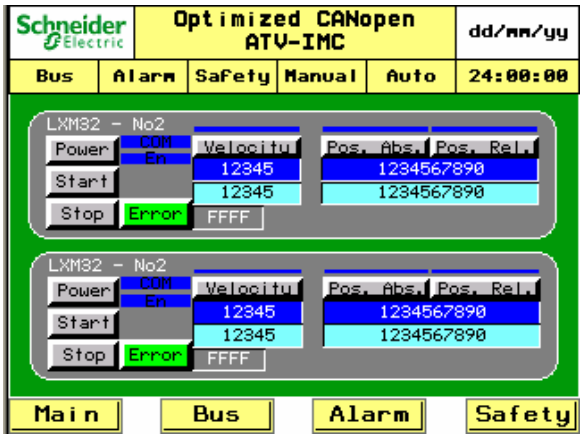
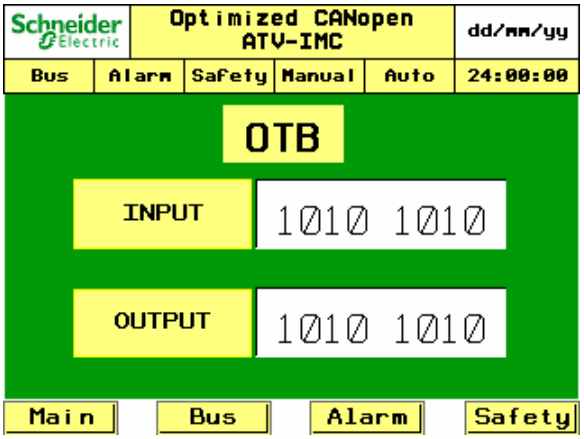
Create a Numeric Display

1	Click on the **Numeric Display** icon in the tool bar.	
2	Select the spot where you want to position the display by opening the rectangle and pressing enter.	
3	In the **Numeric Display Settings** dialog go to the **General** tab. In **Display Digits** you can set the maximum number of the digits to be displayed for both integral and fractional part of the value. To link a **Variable** to the display click on the bulb icon to browse for a variable**.** Click on **OK.**	
4	The display shows the new numeric display.	

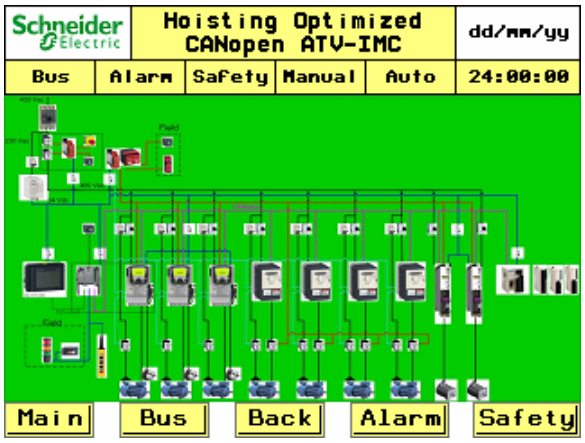
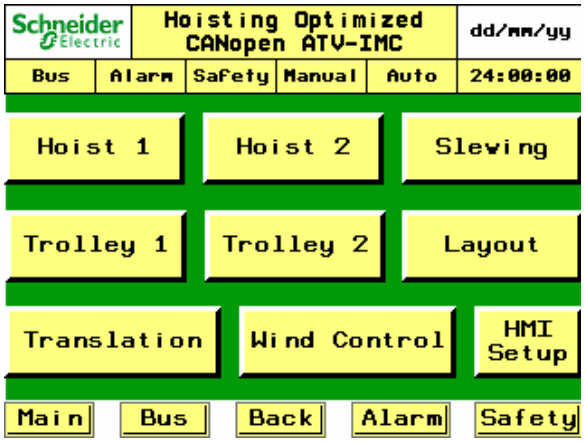
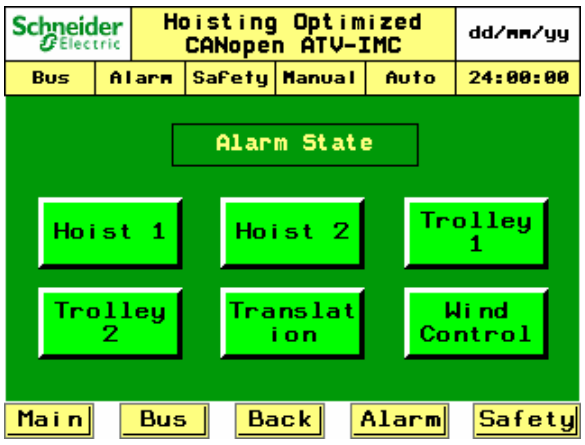
Example Screens Generic

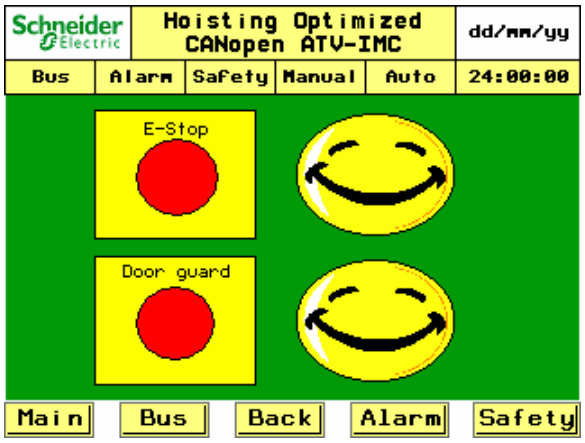
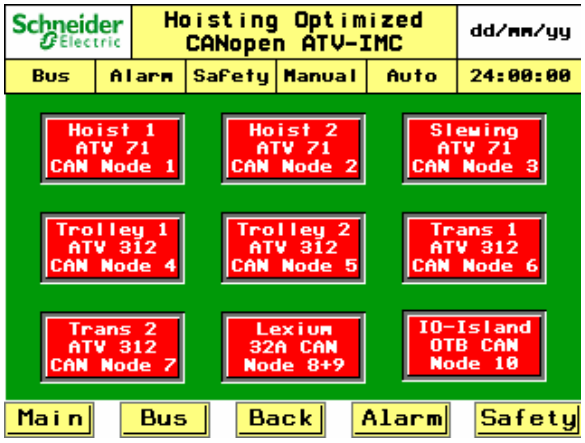
1	The **Home** page shows the Optimized CANopen ATV IMC architecture.	

2	The **Safety** page shows the status of the Emergency Stop relay.	
3	The **Bus** page shows the state of all CANopen nodes.	
4	Via the **ATV71** page it is possible to control Altivar 71 variable speed drives.	
5	Via the **ATV312** page it is possible to control Altivar 312 variable speed drives.	

6	Via the **LXM32** page it is possible to control both Lexium 32A servo drives.	
7	The **OTB** page shows the status of the input and output bits of the Advantys OTB device.	

Example Screens Hoisting

1	The **Home** page shows the Hoisting Optimized CANopen ATV IMC architecture.	
2	The **Main** page allows to navigate to the desired function: • **Hoist 1+2** • **Slewing** • **Trolley 1+2** • **Translation** **Windcontrol**	
3	The **Alarm State** page provides an overview in case an alarm is active on one of the listed functions here. If for instance the background colour of the **Trolley 2** turns red, more details on the current alarm state can be requested by pressing on the button **Trolley 2**. In this case a new screen will appear and provide more information on the alarm.	

4	The **Safety** page shows the status of the Emergency Stop relay.	
5	The **Bus** page shows the state of all CANopen Nodes.	

Devices

Introduction

This chapter describes the steps required to initialize and configure the different devices required to attain the described system function.

General

Altivar 312, Altivar 71 and Lexium 32A drives must be configured using the local control panel on the device itself.

Note

If this is not a new drive you should re-establish the factory settings. If you need instructions on how to do this, please read the drive documentation.

Be sure that the controller is in STOP state before configuring the drives.

Altivar 312

Introduction

The ATV312 parameters can be entered or modified via the local control panel on the front of the device itself.

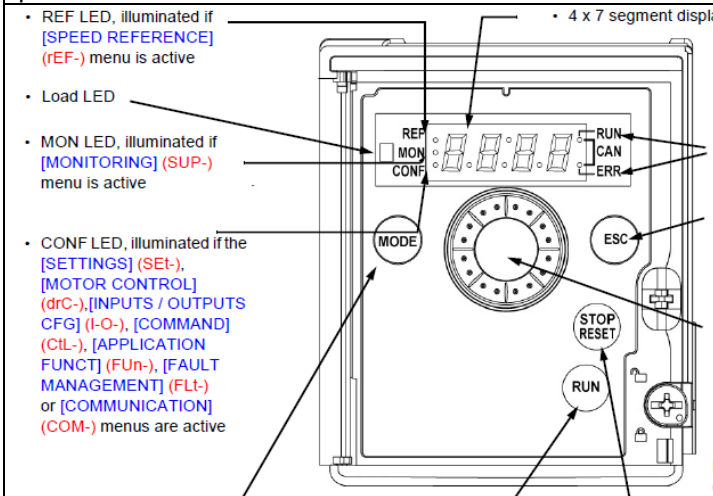
Note

If this is not a new drive you should re-establish the factory settings. If you need instructions on how to do this, please read the drive documentation.

The jog dial is part of the local control panel and can be used for navigation by turning it clockwise or counter-clockwise. Pressing the jog dial enables the user to make a selection or confirm information.

Control panel

1	The drive parameters can be input using the buttons and the jog dial on the control panel of the Altivar.

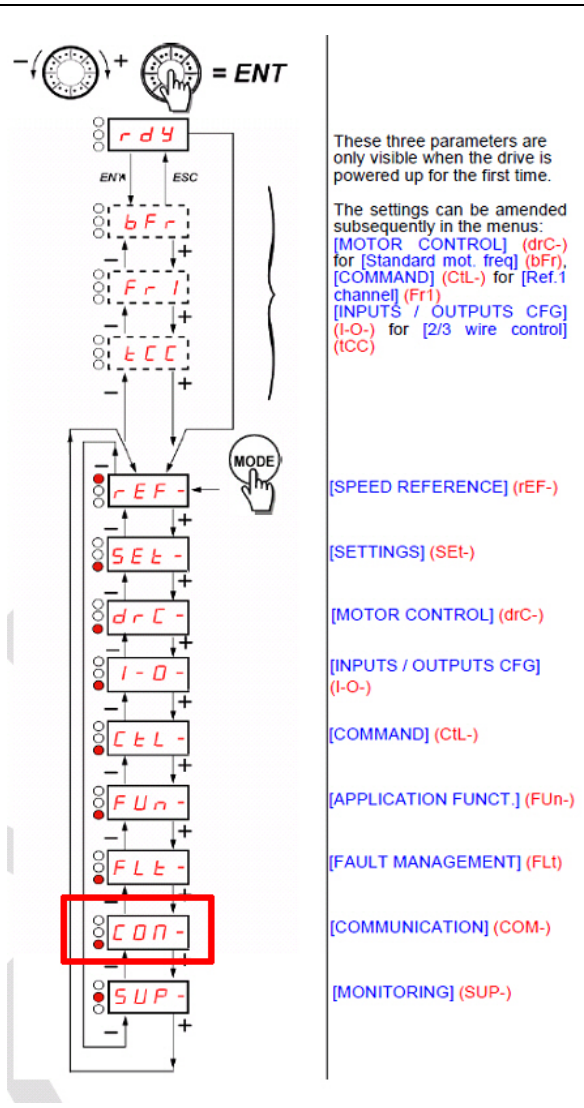


- REF LED, illuminated if [SPEED REFERENCE] (rEF-) menu is active
- Load LED
- MON LED, illuminated if [MONITORING] (SUP-) menu is active
- CONF LED, illuminated if the [SETTINGS] (SEt-), [MOTOR CONTROL] (drC-),[INPUTS / OUTPUTS CFG] (I-O-), [COMMAND] (CtL-), [APPLICATION FUNCT] (FUn-), [FAULT MANAGEMENT] (FLt-) or [COMMUNICATION] (COM-) menus are active
- MODE button (1): If [SPEED REFERENCE] (rEF-) is displayed, this will take you to the [SETTINGS] (SEt-) menu. If not, it will take you to the [SPEED REFERENCE] (rEF-) menu.
- 4 x 7 segment display
- RUN button: Controls powering up of the motor for forward running if the [2/3 wire control] (tCC) parameter in the [INPUTS / OUTPUTS CFG] (I-O-) menu is set to [Local] (LOC), page 45
- 2 CANopen status LEDs
- Used to quit a menu or parameter or to clear the value displayed in order to revert to the value in the memory
- Jog dial - can be used for navigation by turning it clockwise or counter-clockwise - pressing the jog dial enables the user to make a selection or confirm information. = *ENT*

  Functions as a potentiometer if [Ref.1 channel] (Fr1-) in the [COMMAND] (CtL-) menu is set to [Image input AIV1] (AIV1)
- STOP/RESET button
  - Enables detected fault to be reset
  - Can be used to control motor stopping
    - If [2/3 wire control] (tCC) is not set to [Local] (LOC), freewheel stop
    - If [2/3 wire control] (tCC) is set to [Local] (LOC), stop on ramp or freewheel stop during DC injection braking

CANopen settings

1	Using the buttons on the front panel, select the sub-menu **Communication (COM).**
2	In the **Communication (COM)** sub-menu input the CANopen address in the parameter **AdC0.** In the example application the addresses for the four drives are **4** to **7**
3	Also in the **Communication (COM-)** sub-menu, in the parameter **BdC0,** set the Baudrate to **500.0** (kBits).
4	For the drive to operate with the new parameters, a power cycle (on, off, on) is required.



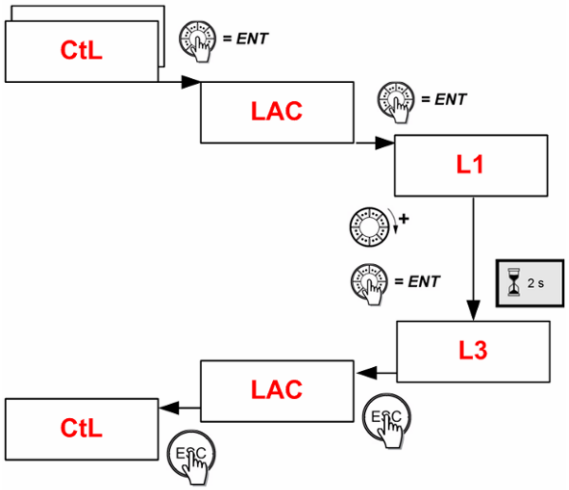
⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

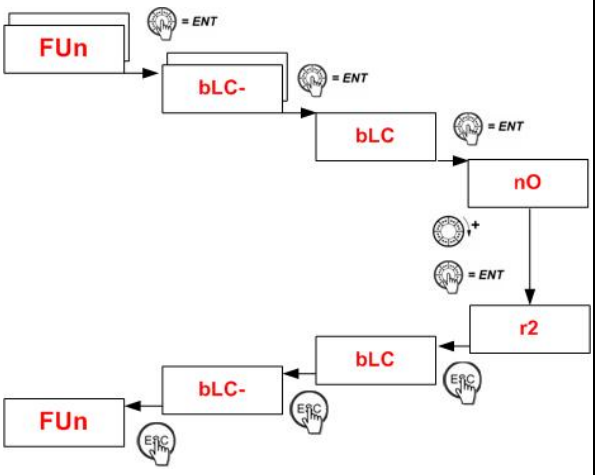
After making any configuration changes or adjustments, be sure to cycle power (remove and reapply power) on the drive.

Failure to follow these instructions can cause death, serious injury or equipment damage.

Changing the Access Level LAC

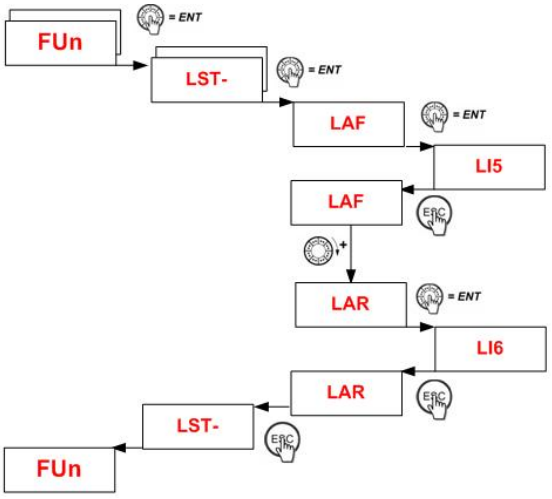
1	To set the parameters for the brake function a higher access level (L3) is required.	
2	To go to expert mode L3: ➔ Select **CtL-** [COMMAND] and press enter ➔ Select **LAC** [ACCESS LEVEL] and press enter ➔ **L1** (Level 1) is displayed ➔ Select **L3** (Level 3) and press enter for 2 seconds to set the new level. Return to the **LAC** with **ESC.** Return to the **CtL-** with **ESC.**	

Brake settings

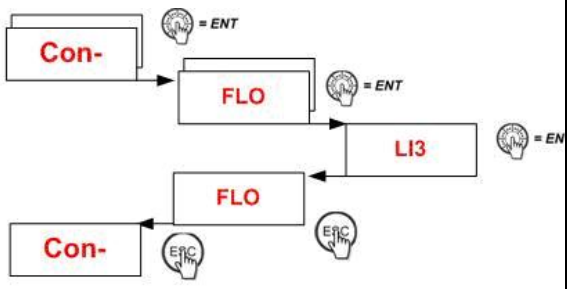
1	The R2 relay output is used for brake control.	
2	To assign the R2 relay output: ➔ Select **FUn-** [APPLICATION FUNCT.] and press enter ➔ Select **bLC-** [BRAKE LOGIC CONTROL] and press enter ➔ Select **bLC** [BRAKE LOGIC CONTROL] and press enter ➔ Select **r2** and press enter. Set the parameters to the values shown here on the right. **Note:** These parameters are for the machine described in this example only. In all likelihood, you will need to adapt these parameters for your specific machine. After all parameters are set return to the **bLC** with **ESC.** Return to the **bLC-** with **ESC.** Return to the **Fun-** with **ESC.**	

Brake Logic parameters	
Parameter	**Value**
bLC [Brake assignment]	R2
brL [Brake release freq]	0 Hz
Ibr [Brake release I FW]	0 A
brt [Brake release time]	0 s
LSP [Low speed]	0 Hz
bEn [Brake engage freq]	0 Hz
bEt [Brake engage time]	0.5 s
bIP [Brake impulse]	No

Limit switches configuration

1	The input LI5 and LI6 are assigned to limit switches	
2	To assign the inputs LI5 and LI6 for the forward and reverse limit switches: ➔ Select **FUn-** [APPLICATION FUNCT.] and press enter ➔ Select **LST-** [Limit switches.] and press enter ➔ Select **LAF** [Forward limit switch.] and press enter. ➔ Select **LI5** and press enter. ➔ Press **ESC** ➔ Select **LAR** [Reverse limit switch.] and press enter. ➔ Select **LI6** and press enter. Once the limit switches are configured return to the **LST-** with **ESC.** Return to the **FUn-** with **ESC.**	

Local mode configuration

1	The input LI3 is assigned to local mode	
2	To assign LI3 for the local mode configuration ➔ Select **Con-** [Communication] and press enter ➔ Select **FLO-** [Forced local mode.] and press enter ➔ Select LI3 and press enter. Once the the local mode is configured return to the **FLO-** with **ESC.** Return to the **Con-** with **ESC.**	
	Local mode: Local mode is required to manage the axis movement when the wiring is directly connected to the Altivar and not to the controller. Local mode is used to test the axis during: • Commissioning of the crane for the first time; in this case the operator is able to check that the movement is correct. • Maintenance, when it is required to move an axis without the help of the controller.	

Altivar 71

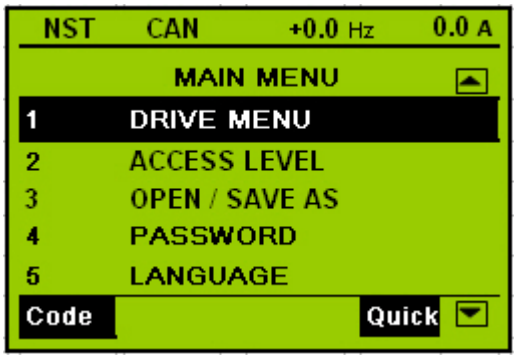
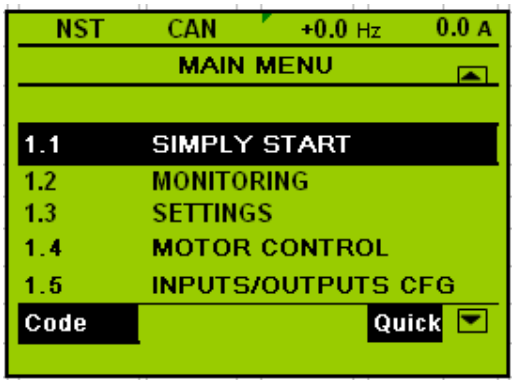
Introduction

The ATV71 parameters can be entered or modified using the graphic keypad panel. For the generic application example the first part, From **Configuration setting** (**1**) up to **Enable Motor Tuning** (**19**) usage of configuration is enough. For the hoisting application example the second MultiMotor configuration is needed for the usage of CANopen or Hardwired modus.

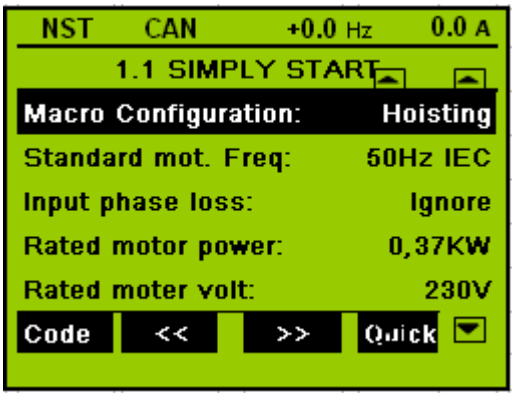
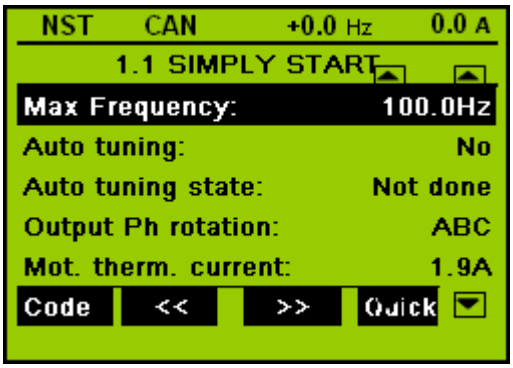
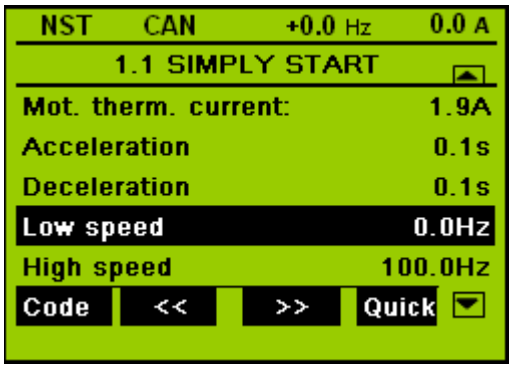
Note

If this is not a new drive you should re-establish the factory settings. If you need instructions on how to do this, please read the drive documentation.

Configuration settings

1	The drive configuration can be done using the jog dial on the front panel of the Altivar. To select push **ENT** (ENT is the enter of the Jog dial) To leave a menu push **ESC.**	
2	To change settings go to: **1 DRIVE MENU** and press **ENT.**	
3	Go to **1.1 SIMPLY START.** and press **ENT**	
4	**Note:** These example parameters are for the machine described in this example only. In all likelihood, you will need to adapt these parameters for your specific machine. **You have to check with the actual equipment you have on your local crane.**	

Simply Start

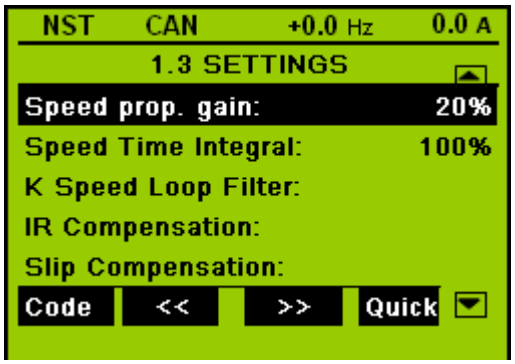
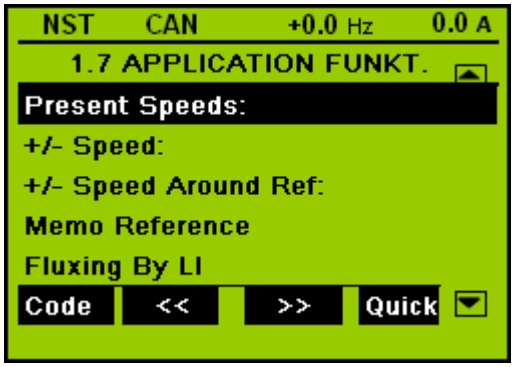
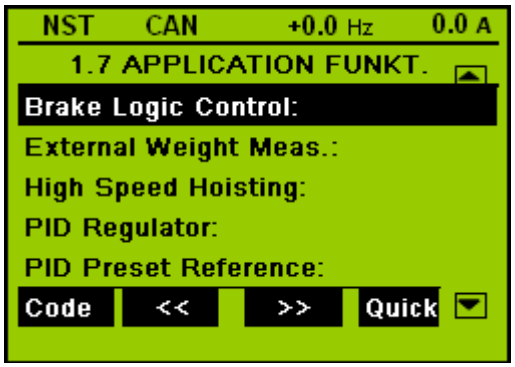
5	Configuring a hoist drive, the **Macro configuration** needs to be set to **Hoisting**. Configure all **Rated motor** data as power, voltage, speed current etc.	
6	For using highspeed functions as **Speed optimization and rope slack AFB**, the **Max frequency** must be changed from 60.0Hz to the maximum possible of the crane. 50.0Hz motors should be able to have 100.0Hz max frequency.	
7	**Acceleration** and **Deceleration** must be set to the fastest reaction time **0.1s** for **Anti-sway axis drive (i.e. Trolley).** For other axis drives using AFBs from Segment Library useful values need to be found for the application during commissioning. **Low speed** must be set to **0.0Hz** on closed loop drives (FVC) and **High speed** to the **same as Max frequency** (100.0Hz)	
8	To make sure that the internal PID is set to the right values for the application used it is suggested that you create a trace of the actual speed and do a test run of the drive. ▪ Verify that the drive reaches the target speed. If the drive does not reach this speed, the time integral of the speed loop is not steep enough and needs to be increased ▪ If the speed profile oscilates around the target speed, stop the drive and reduce the speed proportional gain and repeat the test	

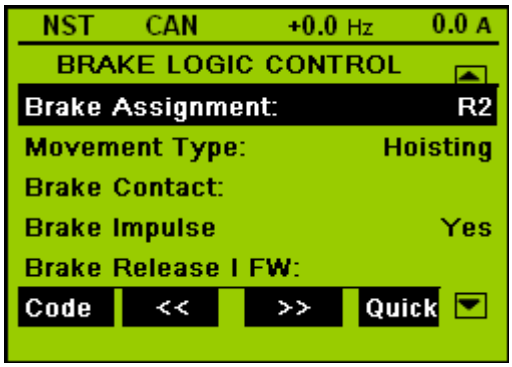
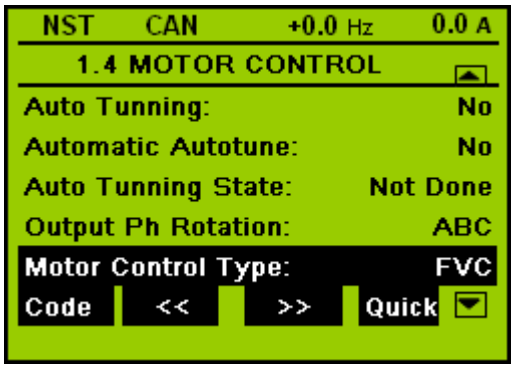
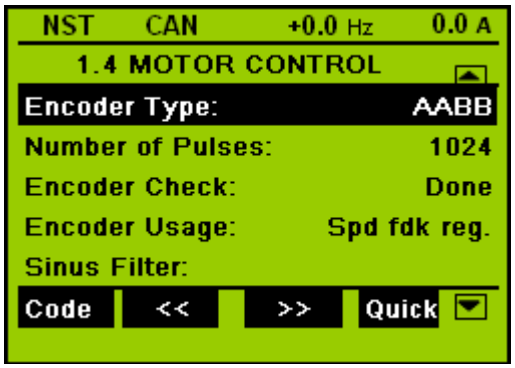
⚠ WARNING

UNGUARDED MACHINERY CAN CAUSE SERIOUS INJURY

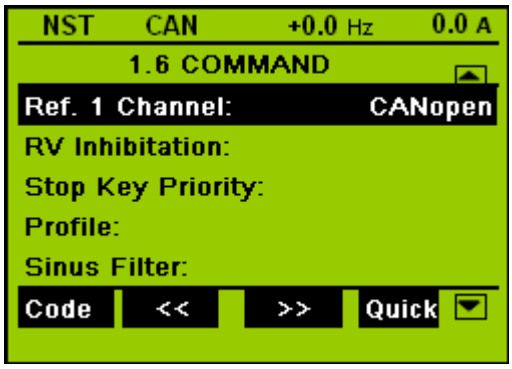
- Do not use this software and related automation products on equipment which does not have point-of-operation protection.
- Do not reach into machine during operation.

Failure to follow these instructions can cause death, serious injury or equipment damage.

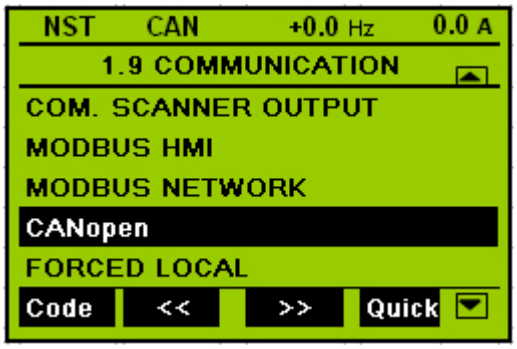
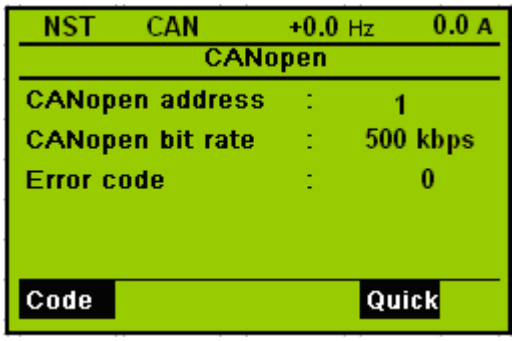
ATV PID	**9**	Change to **1.3 Settings** **Speed prop. gain** and **Speed Time Integral** need to be set on application values.	
Application Function	**10**	Change to **1.7 Application Function** Menu. All **Present Speeds** must be set to **No.**	
Brake Logic	**11**	Change to **Brake Logic Control**.	

	12	In the **Brake Logic Control** menu the following must be configured. **Brake Assignment: R2** **Movement Type: Hoisting** **Brake Impulse: Yes** Depending on the used brake and crane equipment. **Brake Release Time: 0.50s** **Brake Engage Delay: 0.00s** **Brake Engage Time: 0.50s** These values need to be adapted to the hardware that is used. The shown here values are only used for our example. For **Hoisting position synchronization** and **Anti-crab AFB** axis drives the **Brake Engage Delay** must be set to **0.50s**.	 | Parameter name	Value
---	---
Brake Assignment	R2
Movement Type	Hoisting
Brake Impulse	Yes
Brake Release Time	0.50s
Brake Engage Delay	0.00s
Brake Engage Time	0.50s	 Overview of Parameter
Motor Control Type	13	Change to **1.4 Motor Control** menu. The **Motor control Type** must be set to **FVC.**	
Encoder Settings	14	For the Encoder configuration the following settings need to be done: **Encoder Type: AABB** **Number of Pulses: 1024** **Encoder Usage: Spd fbk reg.** **Encoder Check: Done**	

Commands via CANopen

15	Change to **1.6 Command** And set **Ref. 1 Channel** to **CANopen**	

CANopen Settings

16	Change to **1.9 COMMUNICATION** and go to the **CANopen** menu.	
17	Set the **CANopen address** to **1** for the first ATV71, and **2** for the second ATV71, etc. Set the **CANopen bit rate** to **500 kbps.**	
18	After changing the configuration it is necessary to perform a power cycle on the drive (on-off-on). **Note:** For high power drives [more than 121 HP (90 kW)] you should do an automatic reboot with the graphic keypad panel (refer to drive user's manual for details)	

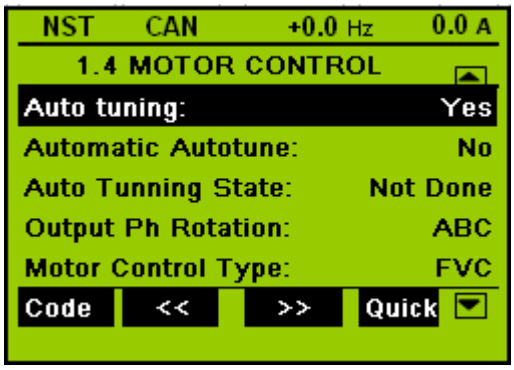
⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

After making any configuration changes or adjustments, be sure to cycle power (remove and reapply power) on the drive.

Failure to follow these instructions can cause death, serious injury or equipment damage.

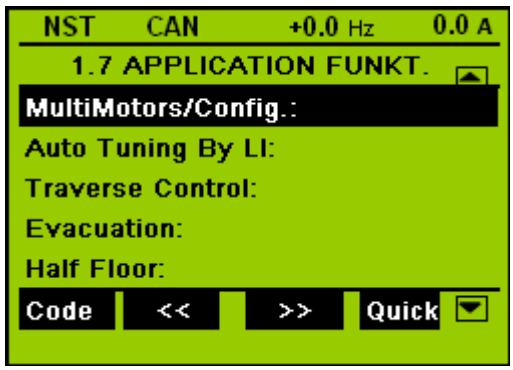
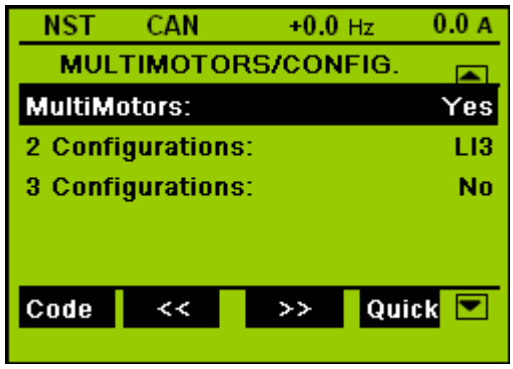
Enable Auto tuning

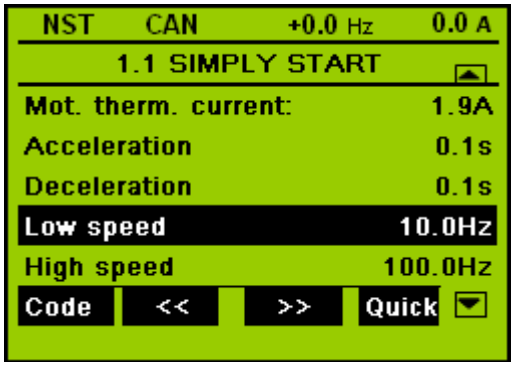
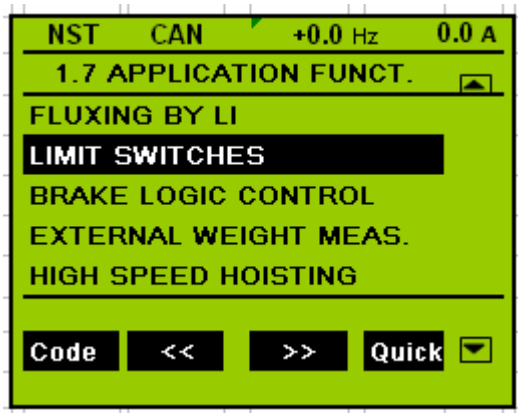
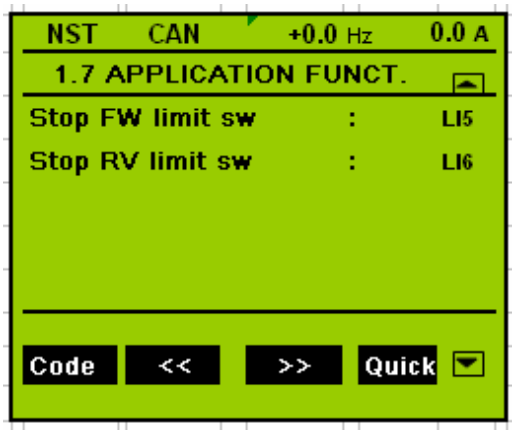
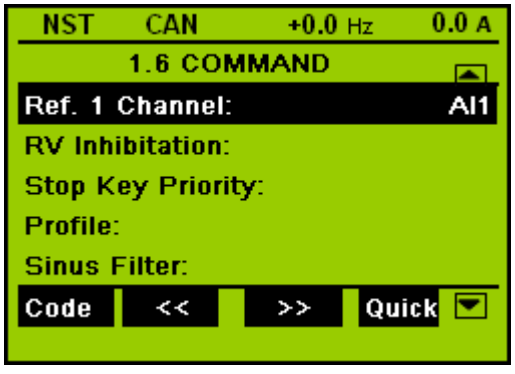
19	To enable the auto tuning go to **1.4 MOTOR CONTROL** and set **Auto tuning: Yes**	

Second Motor Configuration

1	The Hoisting application shown in the example, includes a fallback function to continue running the crane, even when the CANopen bus or the controller have become inoperational. For this case a key switch is mounted inside the TVDA to switch between CANopen commands and hardwired commands. To implement this, a second configuration needs to be done on the drives. Using **Application Function → Multimotor Configuration** the function can be activated. In CANopen command mode the key switch sends all drives a false signal on LI3 and the controller gets the signals from the drive inputs over CANopen. The controller does all the stop and start commands and all coded functions are included. In hardwired command mode the key switch sends all drives a true signal and they work hardwired with limited function, but can still be moved and used. **Limitations** for the Hardwired command mode are: • **No AFB function available** • **Only Stop Limit switch** • **Running with slow speed**	

Multimotor

2	Go to **1.7 Application Function** And select **MultiMotors/Configuration**	
3	Set **MultiMotors: Yes** **2 Configurations: LI3**	

	4	The first configuration for CANopen control mode must be done as seen before in CANopen configuration. To change to the **second configuration for hardwired** control mode the **LI3 must be activated with 24V** Input (set to High) .	
	5	Repeat the Steps 1 to 14 from the CANopen control mode configuration. Only in Step 7 for Low Speed use another Value (see next point).	
	6	With the second Hardwired control mode configuration **only one speed (LSP) is possible**. Therefore it must be set to a different value other than 0Hz. Set **Low speed: 10.0Hz**	
Limit switch configuration	**7**	Go to **1.7 Application Function** and enter the **Limit Switches** configuration	
	8	Configure the Forward (**FW**) and the Reverse (**RV**) limit switches. **Stop FW limit sw: LI5** **Stop RV limit sw: LI6**	
	9	Change to **1.6 Command** and set **Ref. 1 Channel: AI1**	

10	After changing the configuration it is necessary to perform a power cycle on the drive (on-off-on). **Note:** For high power drives [more than 121 HP (90 kW)] you should do an automatic reboot with the graphic keypad panel (refer to drive user's manual for details)

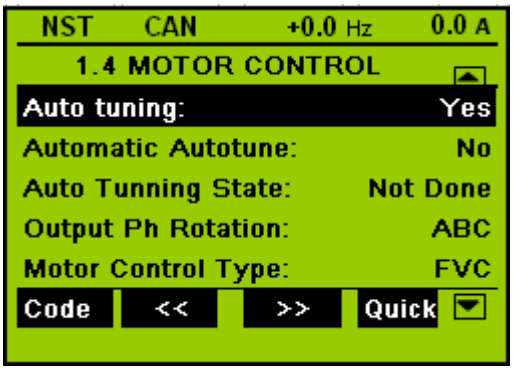
⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

After making any configuration changes or adjustments, be sure to cycle power (remove and reapply power) on the drive.

Failure to follow these instructions can cause death, serious injury or equipment damage.

Enable Auto tuning

11	To enable the auto tuning go to **1.4 MOTOR CONTROL** and set **Auto tuning: Yes**	

Lexium 32A

Introduction

The LXM 32A parameters can be entered or modified using the local control panel on the front of the device itself. Before you can start the drive configuration with SoMachine you must set the CANopen address and the baudrate.

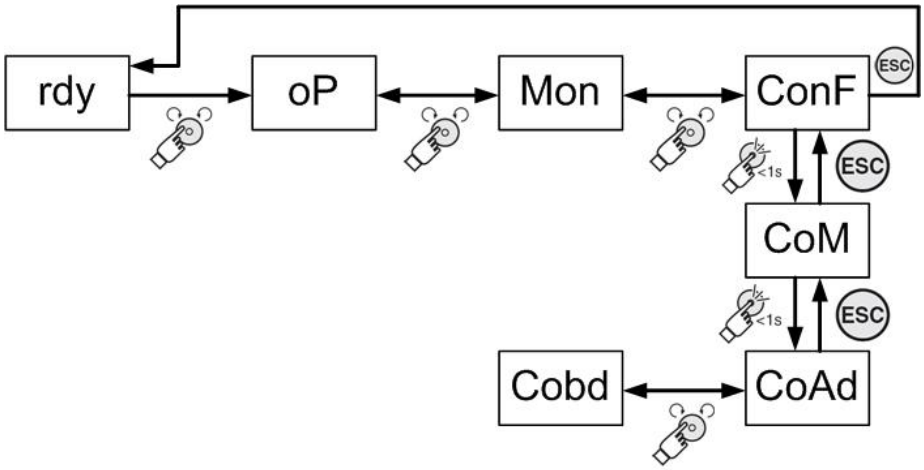
Note:
If this is not a brand new drive you should re-establish the factory settings. If you need instructions on how to do this, please refer to the drive documentation.

CANopen settings

1 If the drive is being started for the first time, the **FSu** (First Setup) is invoked. Only the CANopen address (**CoAd**) and the baudrate (**Cobd**) are initially needed.

If the drive has never been started before, follow the steps below to change the address or the baudrate.

In this project the **CANopen addresses** for the drives are **8 + 9**. The **Baudrate** for the drives is **500 kBaud**.



Parameter name HMI menu HMI name	Description
CANaddress ConF → CoN- ConF → FSu- CoAd	CANopen address (node number) Changed settings become active the next time the product is switched on.
CANbaud ConF → CoN- ConF → FSu- Cobd	CANopen baud rate **50 / 50 kBaud /** 50: 50 kBaud **125 / 125 kBaud /** 125: 125 kBaud **250 / 250 kBaud /** 250: 250 kBaud **500 / 500 kBaud /** 500: 500 kBaud **1000 / 1 MBaud /** 1000: 1 MBaud Changed settings become active the next time the product is switched on.

2	For the drive to operate with the new parameters, a power cycle (on, off, on) is required.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

After making any configuration changes or adjustments, be sure to cycle power (remove and reapply power) on the drive.

Failure to follow these instructions can cause death, serious injury or equipment damage.

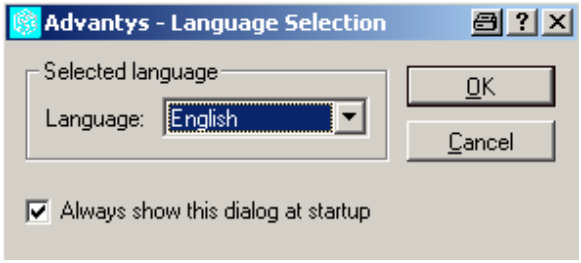
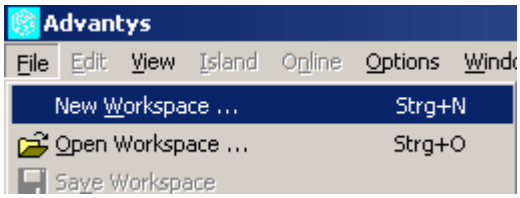
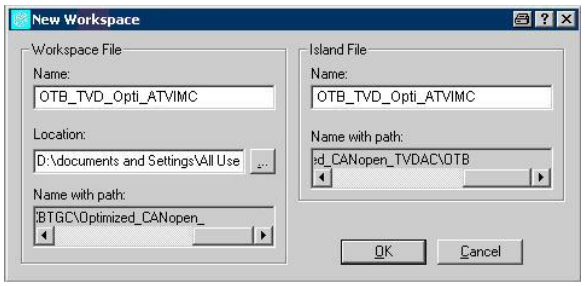
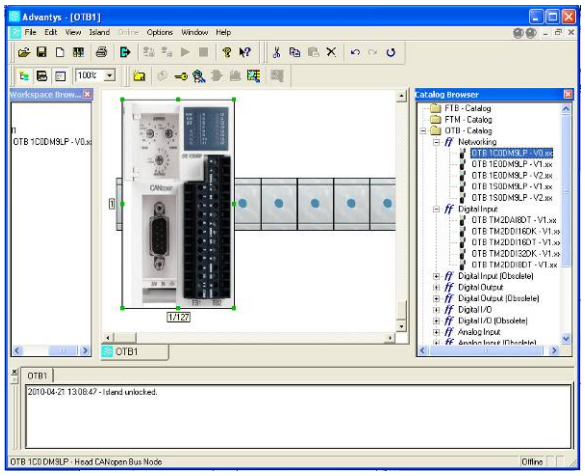
Advantys OTB

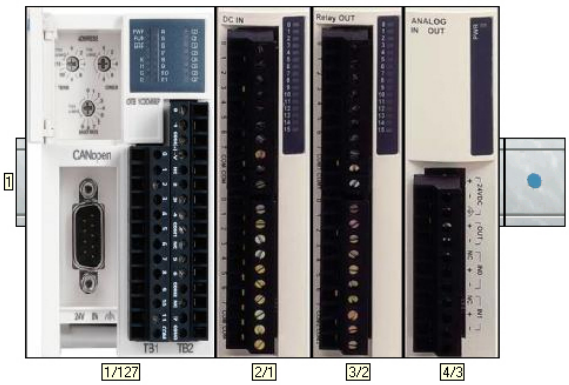
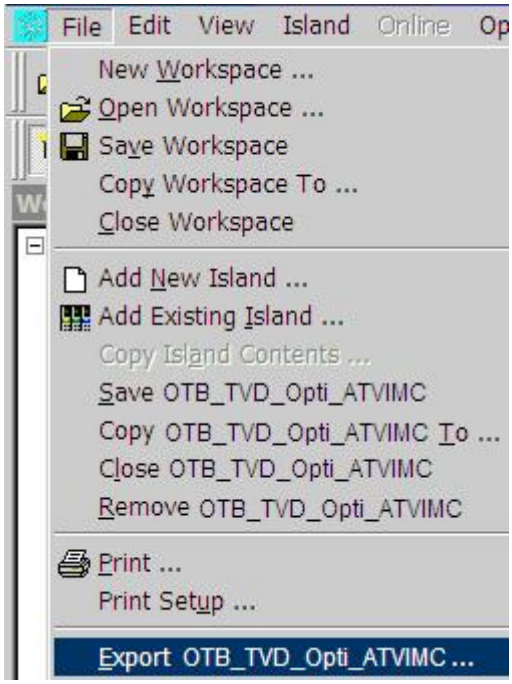
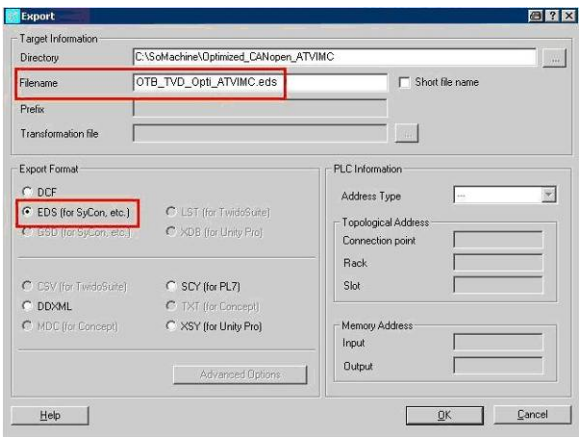
General

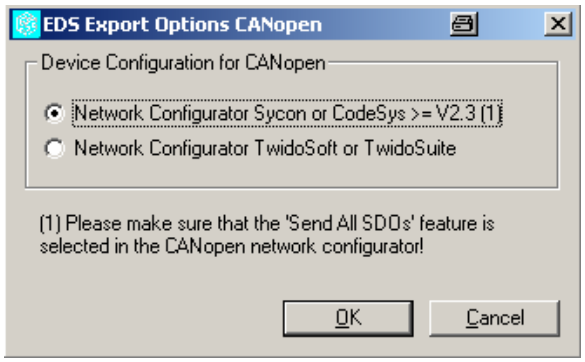
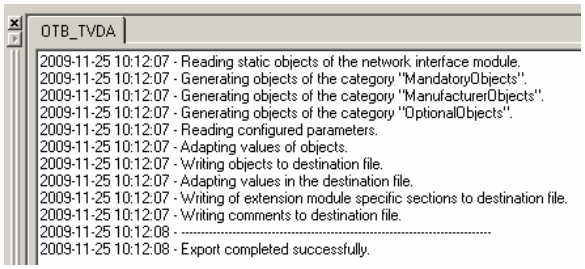
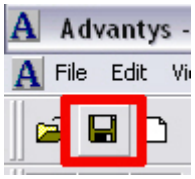
The Advantys OTB EDS (electronic data sheet) file is generated using the Advantys Configuration Software. This section describes how to generate an EDS file, that can be imported into SoMachine Device Repository (see chapter ***Controller***).

Note:
If the user is using only the basic Advantys OTB module; the OTB1CODM9LP device can be used, this is already installed in the SoMachine Device Repository.

Advantys OTB Parameter setting

1	On start-up of Advantys Software select your **Language** and click on **OK**.	
2	Select: **File → New Workspace…**	
3	Type in the **Workspace File Name** and the **Island File Name**. Click on **OK**.	
4	The empty workspace opens. On the right side of the workspace is the **Catalog Browser.** Here you can select the devices you need for your island. Example: **1x OTB1CODM9LP** **1x TM2DDDI16DT** **1x TM2DDRA16RT** **1x TM2AML3LT**	

5	The image on the right shows the finished configured rack.	
6	To generate the **EDS File** select **File → Export OTB_TVD_Opti_ATVIMC …**	
7	Enter the **Filename** **OTB_TVD_Opti_ATVIMC.eds** and select **EDS as Export Format**. Continue the export with **OK**.	

8	Select **Network Configuration SyCon or CoDeSys** and click **OK**.	
9	The successful export is indicated at the bottom of the main window.	
10	To save the island click on the save icon in the toolbar.	
11	**Note:** Refer to the Communication chapter for how to set up the Advantys OTB CANopen baudrate and bus address.	

Appendix

The Hoisting Application

Introduction

Different machines and processes share the same initial requirements that can be implemented with a generic architecture employing the current Schneider Electric product offer. These generic architectures include power supply, controller, motion, visual indication, communication and safety aspects. The use of these generic architectures to implement customer solutions covers not only a large section of customer automation requirements but allows the implementation of a tested and validated software and hardware solution.

This chapter describes the Schneider Electric hoisting application function blocks used in this architecture. This document does not provide a functional description for the application solution. The functions listed here are not comprehensive and form only a foundation for real life applications. It is not intended to provide an application that fulfills a real life situation in all aspects. For a comprehensive description of the function blocks please refer to the library and function block documentation.

The information given here is intended to support the user in the assembly, configuration and implementation of the described solution. The information provided here is additional information to the product documentation, with a focus on the specific components used in the solution provided here.

It is expected that the reader has at least a basic knowledge of the industrial application for which this solution is provided and understands the professional jargon normally used in that type of application. This document is not an introduction into the specific type of industrial application for which this solution is provided.

Application Basics

The solution described in this document is provided for hoisting applications which may consist of the following machine types:

- Building cranes
  - Self-erecting crane
  - Tower crane
- Industry cranes
  - Gantry crane
  - Overhead traveling crane

Application Specifics

Application Dedicated Hardware

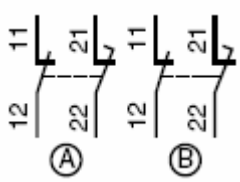
This chapter describes the hardware used in this architecture that is specifically dedicated to the application it is intended for.

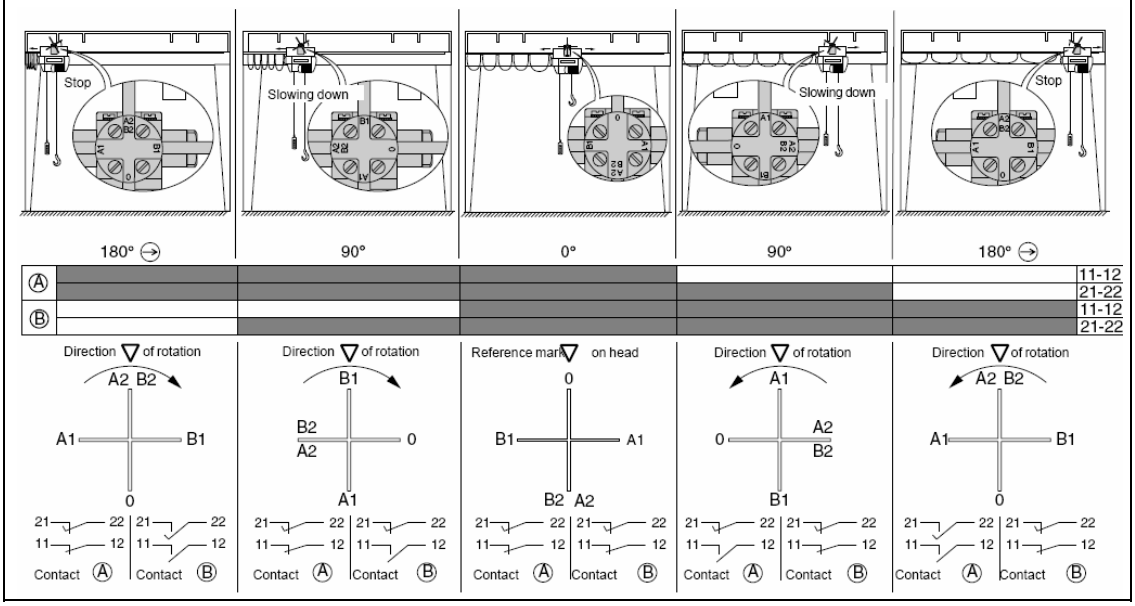
Limit Switch

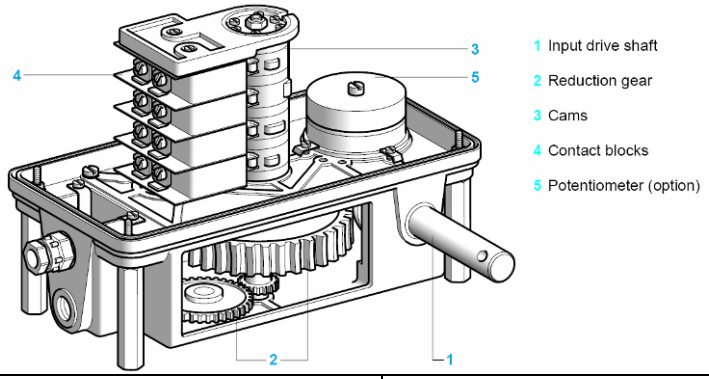
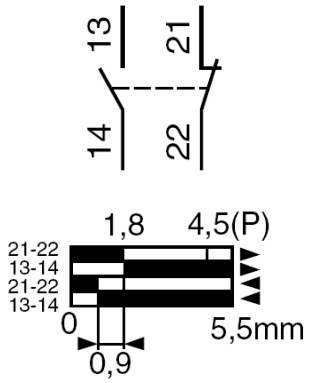
OsiSense

XCKMR54D1H29

2 x 2-pole NC + NC





Screw Limit Switch (3rd Party Component) **TER International** Reference depends on the Scaling rates		
		
Overload Limit Switch **OsiSense** **XCKM110**		

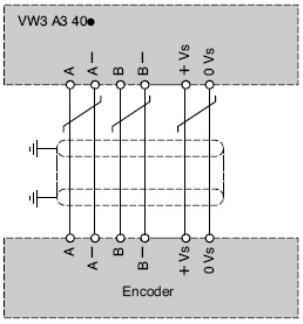
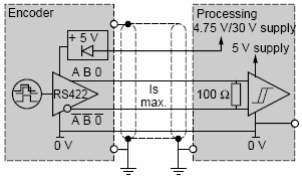
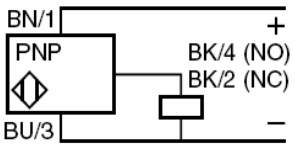
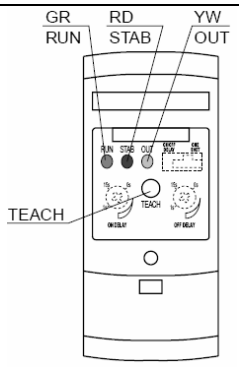
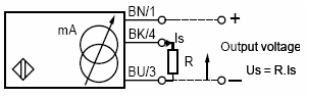
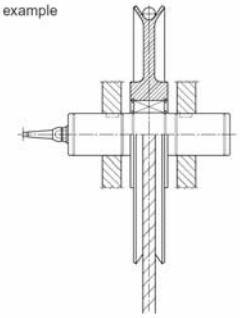
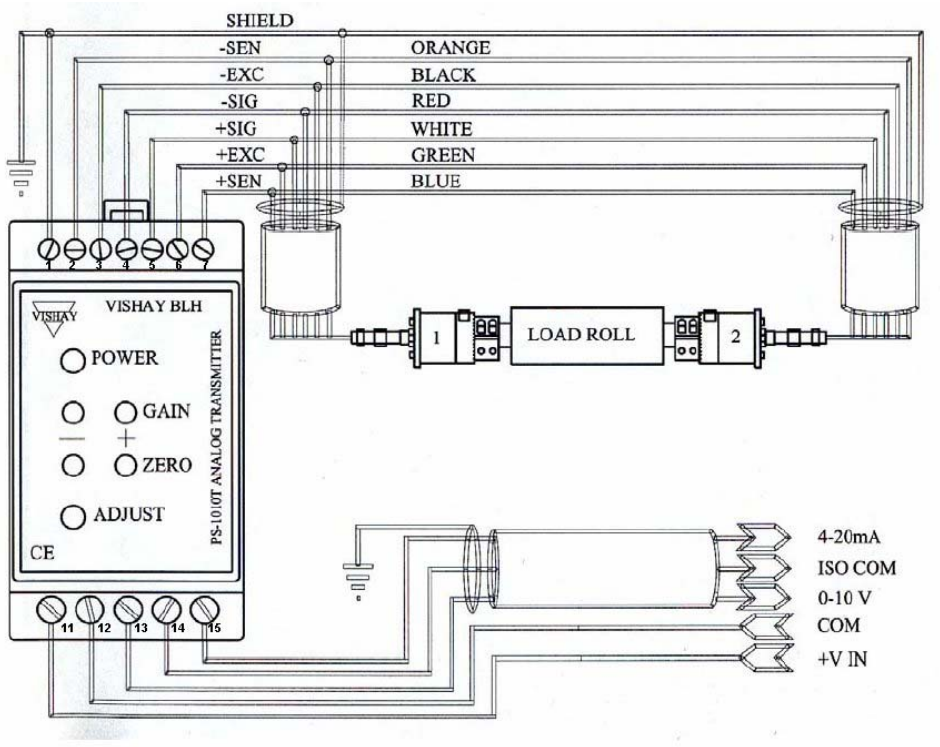
Encoder Interface card for ATV71 **VW3A3401** Zero speed torque Accurate speed Regulation Torque accuracy Shorter response times on a torque surge Improved dynamic performance in transient state Overspeed detection Load slipping detection		
Incremental Encoder **OsiSense** **XCC1510PS11X** Type R (N): 5 Vdc output driver, RS 422, 4.5…5.5 Vdc. Spring coupling **XCCRAR1010**		
Incremental Encoder **10-wire Encoder Cable** **XCCPM23121L5**		
Inductive Proximity Sensor **OsiSense** **XS618B1PBL2**		 **BU : Blue** **BN : Brown** **BK : Black**

Photo-electric Sensor **OsiSense** **XUX1ARCNT16** **REFLECTOR** **XUZC80**	 	 **Relay output :** 1 ⊘ ≂ 2 ⊘ ≂ 3 ⊘ NO 4 ⊘ Relay common 5 ⊘ NC **Transmitter:** 1 ⊘ ≂ 2 ⊘ ≂
Inductive Proximity Sensor **OsiSense** **XS9C111A1M12**		  **Output voltage / Load impedance value** 0…10 V — R = 1000 Ω
Load Cell **(3rd Party Component)** **Vishay** **KISD-6** **Web Tension Transmitter** **PS-1010T**	 Installation example 	



Amplifier Inputs:

1. Shield
2. –SEN (Orange)
3. –EXC (Black)
4. –SIG (Red)
5. +SIG (White)
6. +EXC (Green)
7. +SEN (Blue)

Amplifier Outputs:

8. +VIN
9. COM
10. 0-10V
11. ISOCOM
12. 4-20 mA

Note:

1. For this application that uses a four-wire load cell cable, two jumpers must be installed between +SEN and +EXC and between –SEN and –EXC.

2. The 4 to 20 mA outputs are used for this architecture.

Application Functions

Application Functions

Schneider Electric provides an **A**pplication **F**unction **B**lock Library (AFB) for Hoisting. This is a set of function blocks, tested & validated for functionality and has been designed to help OEMs develop their crane applications.

For the hardware, Schneider Electric offers a set of function blocks (FBs) that provide various application and device functions.

In this example application:

Hoisting Movement:
- Limit switch management
- Overload control
- Load overspeed control
- Speed optimization and rope slack
- Hoisting position synchronization

Trolley Movement:
- Anti-sway

Travel Movement:
- Limit switch management
- Anti-crab

General:
- Monitoring data storage
- Speed select
- Scale input
- Electronic potentiometer

Application Function Blocks

To facilitate the software engineering tasks associated with the application described, Schneider Electric has developed an Application Function Block Library that has been tested and validated.

The following pages show the application function blocks that are running on the architecture described here.

The **Hoisting** library needs to be included in the application program (See the chapter *Controller: Include new library file*)

For more detailed information concerning the Hositing AFBs, see *SoMachine Hoisting Application Functions Hoisting Library Guide*, also available as online help in SoMachine.

Note:
Application function blocks within Hoisting library can only be used with S-type controllers. With G-type controllers, a message *"Use of <Name of FB> is not authorized with the current type of device*" appears during the build phase

The Hoisting Library and the S-Type option will be available after installation of the solution extension of SoMachine.

Limit switch management

Limit switch management

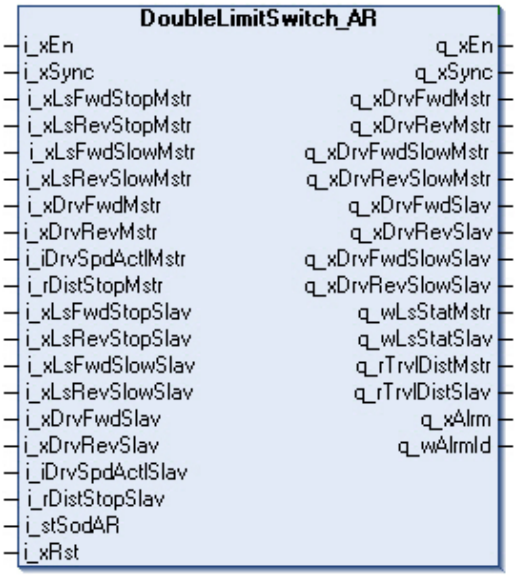
Limit switch management is used to help to prevent the trolley, bridge or hoist from moving to positions that are undesirable for the functioning or functional safety of the crane, load, other equipment and personnel. The function block can monitor up to four limit positions / switches.

This function block helps to prevent moving parts of a crane from reaching areas out of their operational range. It can be used on the Trolley, Bridge, Hoist or Slewing movement of any crane equipped with limit switch sensors.

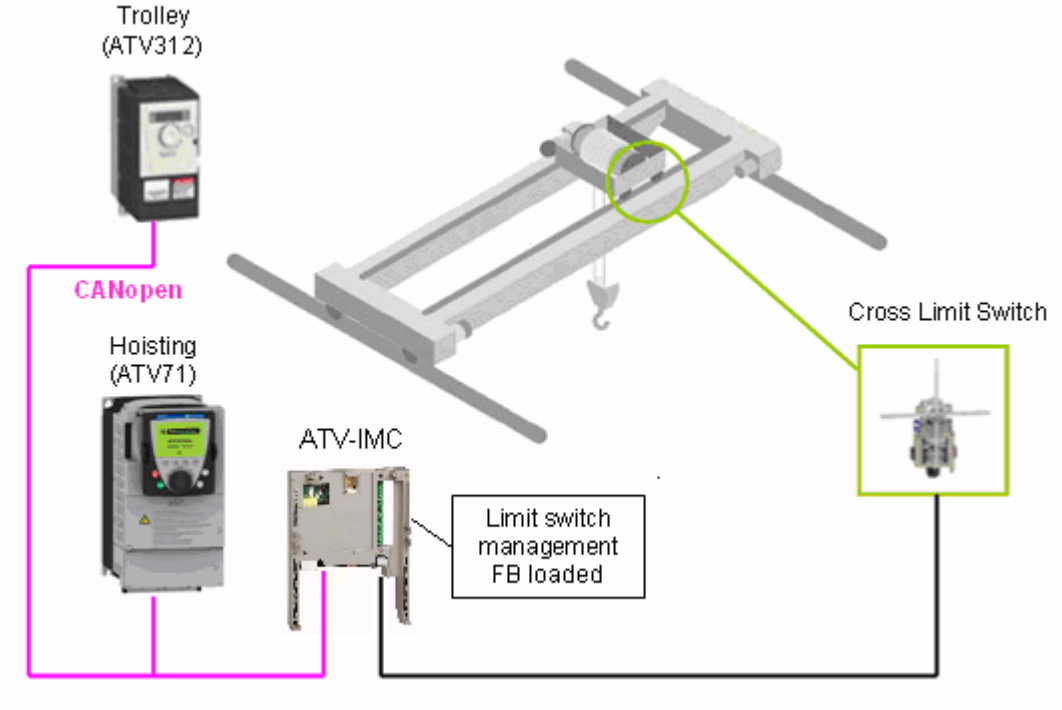
The adaptive ramp feature allows higher performance of a crane while moving in the slow-down area by calculating the maximum allowed velocity at the actual position rather than simply slowing down to a pre-defined slow speed after a slow-down limit switch is activated.

Unlike the single limit switch management function block (**LimitSwitch_AR**), the double limit switch management function block administers two devices in synchronous mode which are intended to be used with the **HoistPositionSync** function block.

NOTE:
The function block is constructed using 2 instances of the **LimitSwitch_AR** function block. The behavior of the FB in the non-sync mode is similar to the **LimitSwitch_AR** function block.



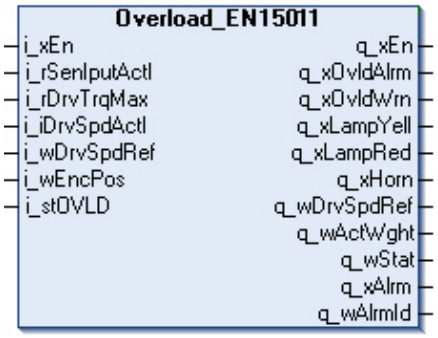
The Limit switch management function block limits the movement of the Trolley/Bridge/Hoist along a rail. This function block controls the Trolley/Bridge/Hoist by stopping/slowing it down according to the status of limit switch/sensors.



Overload according to EN15011

This function reads the torque inputs from the hoist drive. It detects an overload situation according to the calibrated torque threshold and gives an alarm indication by lights and horn as well as stopping the upwards movement of the hoist.

Additionally a warning indication by lights and horn at 90% of the nominal calibrated load is generated.



The function also shows the actual mass of the load with an accuracy of +/- 5%, depending on the setting of the speed loop of the drive.

The FB is fully compatible with the requirements of EN15011. Furthermore it is possible to connect a load cell instead of the drive torque value.

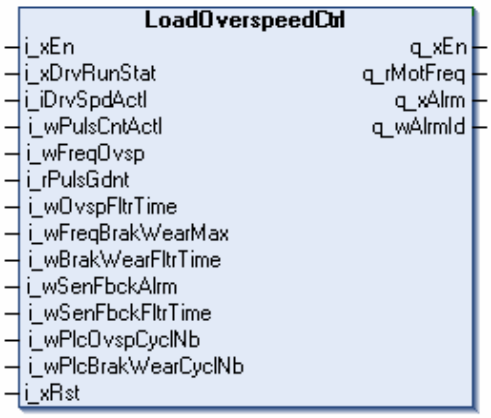
NOTE:
The function block **Overload_EN15011** replaces the former function blocks

- **OverloadCtrlTrq**
- **OverloadCtrlDist**
- **OverloadCtrlEnc**

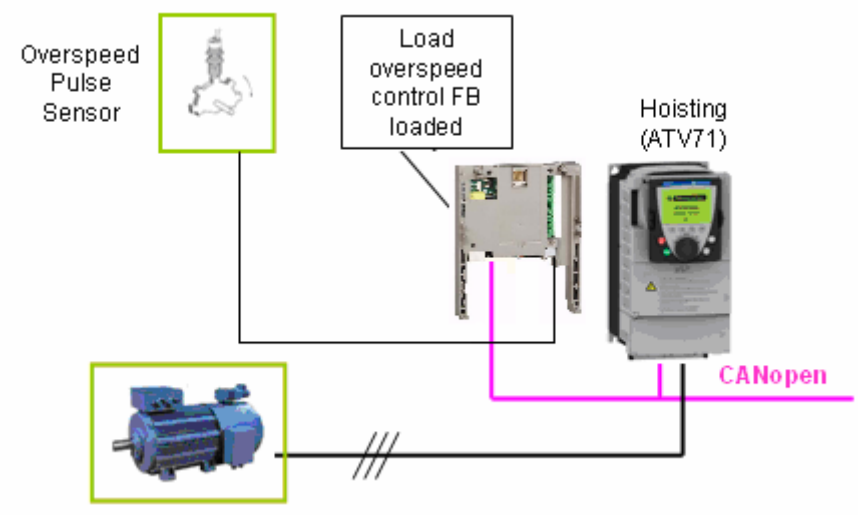
For consistency and maintenance reasons these three FBs listed above are still kept in the latest Hoisting library. However, it is recommended to start new projects by using **Overload_EN15011** FB only.

Load overspeed control

The **Load overspeed** control detects if the drive exceeds the maximum allowed speed during movement and also detects any movement once the brakes have been fully applied. Detection is done by monitoring the feedback from a pulse sensor.



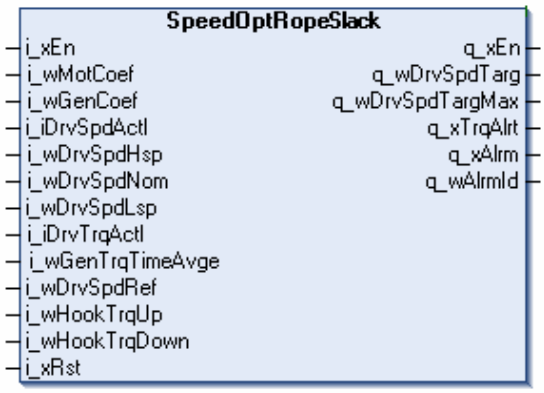
The Load overspeed control function block has two functions. It detects if the hoist drive can hold the load during the hoist movement. This is done by checking the actual speed against the maximum speed via the pulse sensor. While the drive is stopped and the brake is on, it detects load movement due to brake failure.



Speed optimization & rope slack

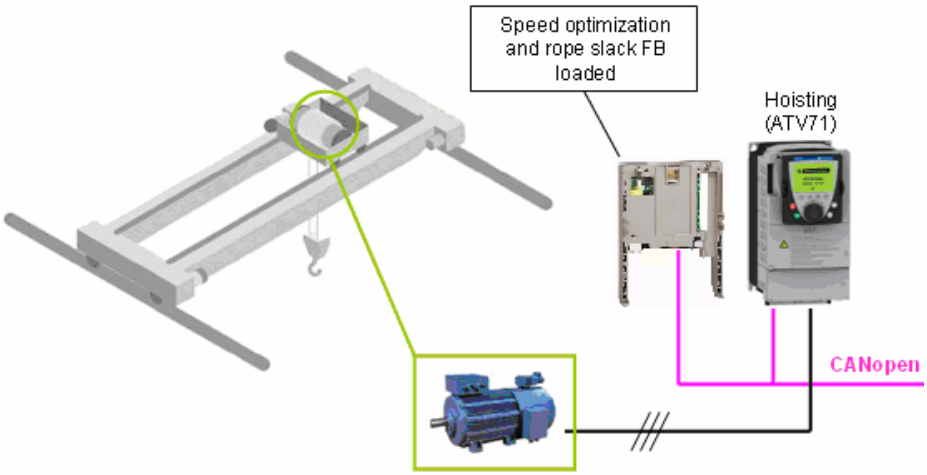
The **Speed optimization** allows operation at constant power consumption in order to reach a speed higher than the rated speed without exceeding the rated motor current. Constant power operation increases the system efficiency.

The **Rope slack** function is used to help avoid extra rope being let out after the hook has already touched the ground.

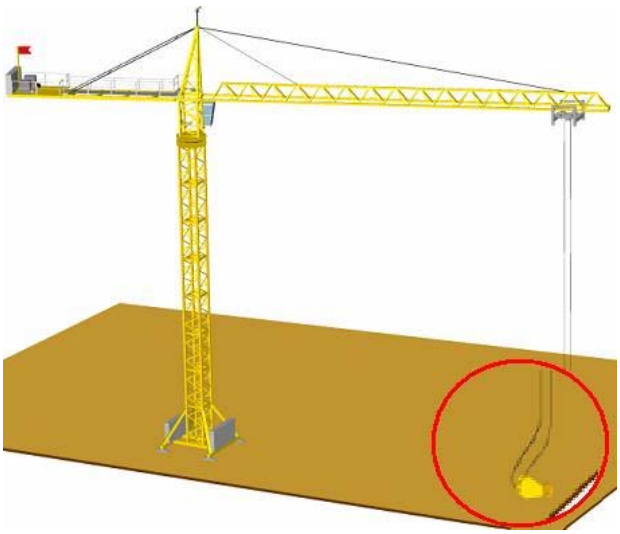


The Speed optimization and Rope slack function block includes two functions in one block.

The Speed optimization function gives the operator the possibility to use the second speed value as the optimum speed for the hoisting movement with the actual load. The actual speed is compared with the actual and maximum allowable torque to obtain the optimum speed for the current load.



The Rope slack function detects when the load drops below a minimum torque (the weight of the hook) and reduces the speed to the minimum (maintenance) speed. This protects the crane against unwinding the whole cable from the drum. Only when the torque returns to a value above the minimum is normal movement allowed again.

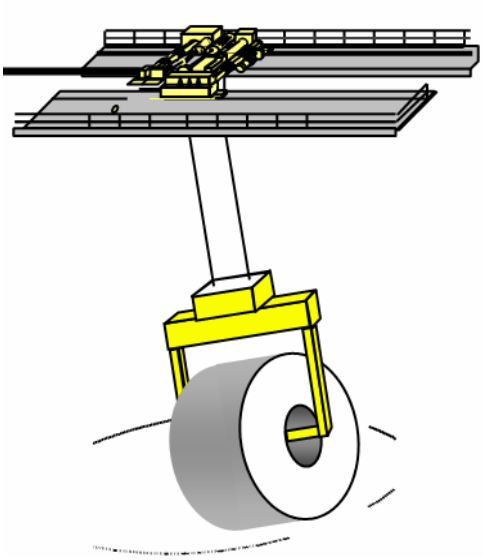


Anti-sway

Anti-sway calculates the optimum speed profile to avoid sway during sideways motion (Trolley / Bridge travel). The calculation is based on the speed, the load height, and the rope length.
The Schneider Electric Anti-sway solution does not require sensors.

The Anti-sway function interacts with one axis (trolley or travel) and, based on the operational input, calculates a movement profile so that the drive compensates for the normal sway effect of the load.

The Anti-sway solution from Schneider Electric works without using sensors to measure the amount of sway.

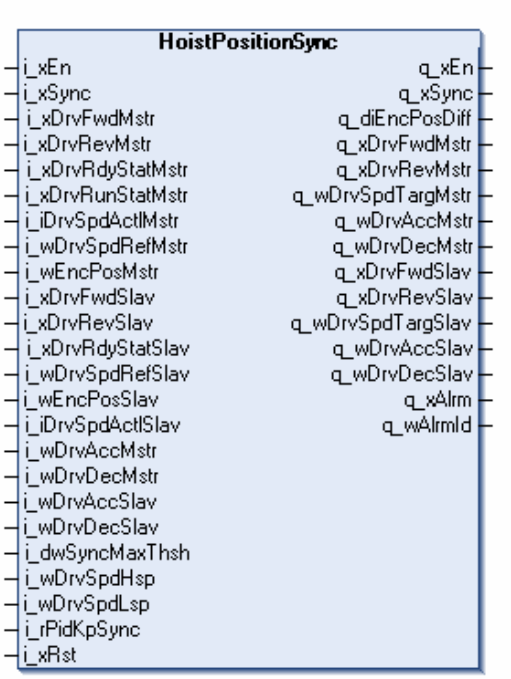


NOTE:
The function block **AntiSwayOpenLoop_2** is the successor of the function block **AntiSwayOpenLoop**.

For consistency and maintenance reasons this FB is still kept in the latest Hoisting library. However. it is recommended to start new projects by using **AntiSwayOpenLoop_2** FB only.

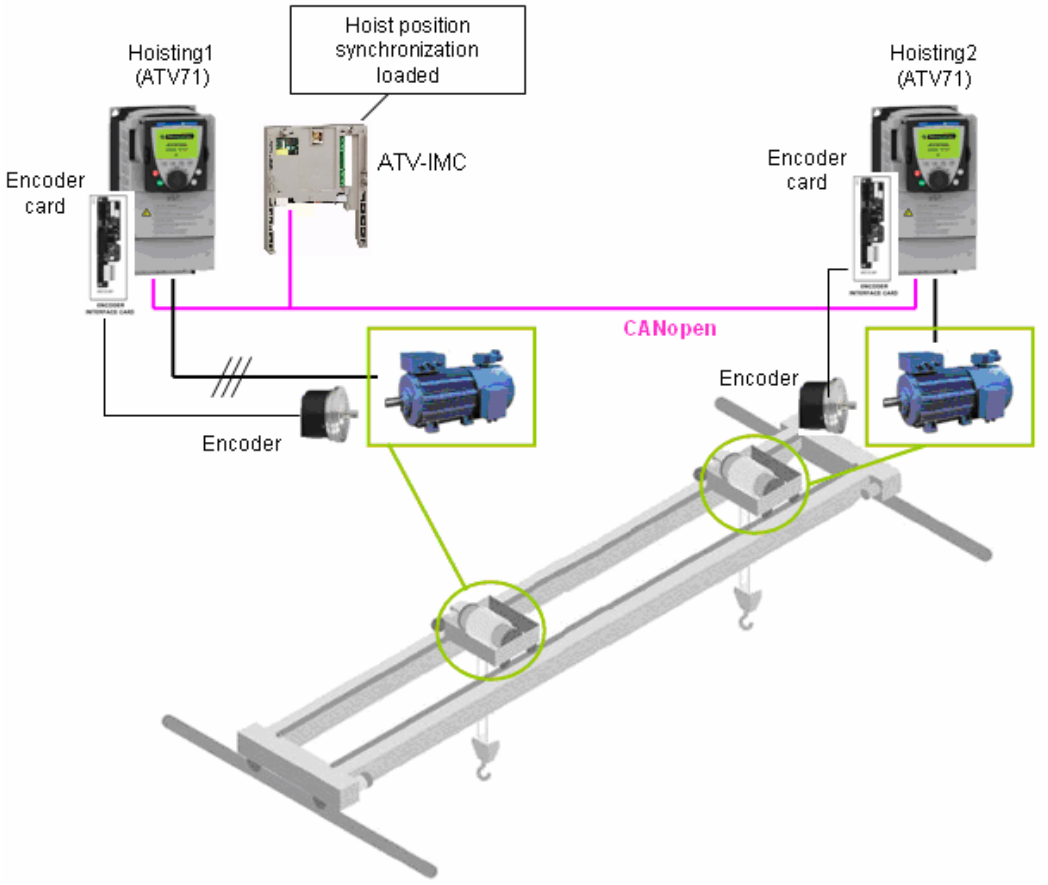
Hoisting position synchronization

Hoisting position synchronization is designed to synchronize the hoisting movement when simultaneous handling with two Hoists is required for large objects. It controls the hoisting of both trolleys, so they act as a single unit to handle the load.



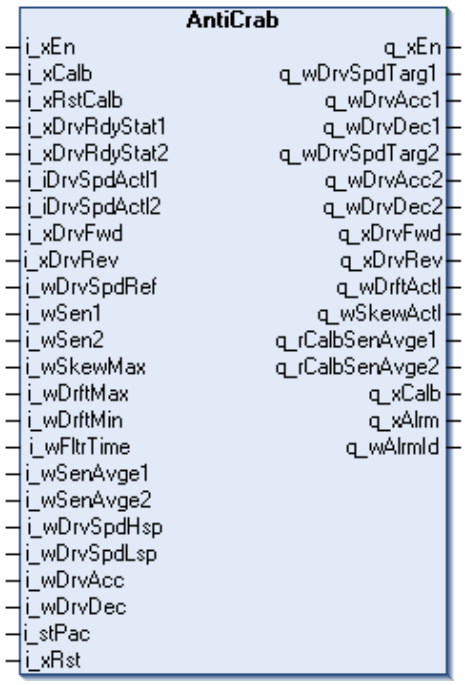
The Hoisting position synchronization function for industrial cranes is designed to synchronize the movement of two axes for simultaneous handling of large singular objects. These axes are controlled by the hoist drives.

The image below shows an over-head crane with a trolley synchronization function: For the industry crane it is used for the hoist movement.

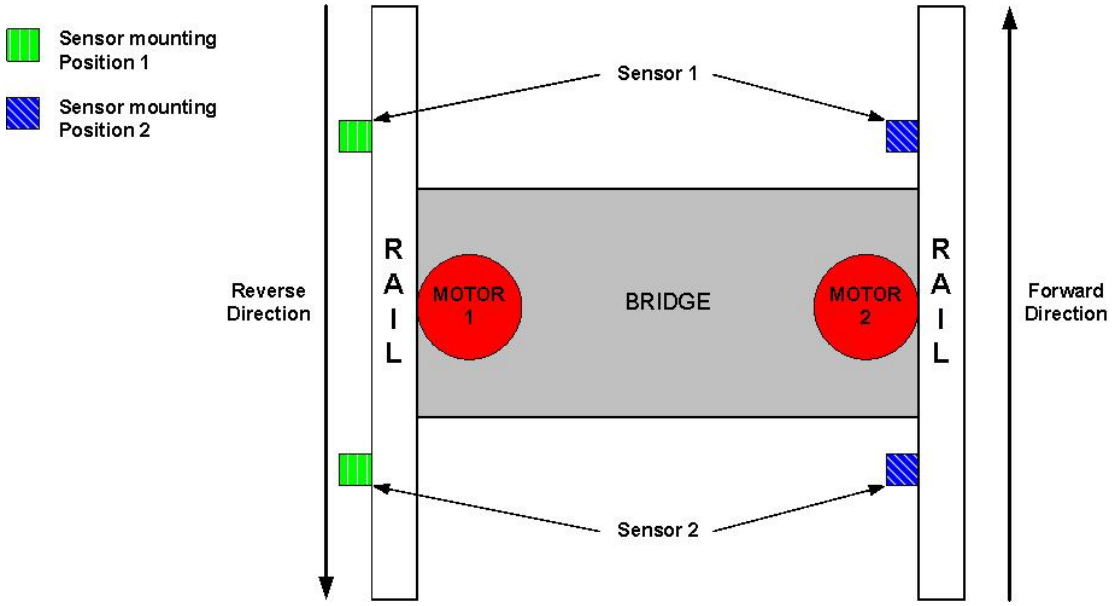


Anti-crab

During travel, both sides of the bridge are moved by different drives. The **Anti-crab** function detects a drift and skew of the bridge using 2 inductive sensors. If the function detects non-parallel movement, it calculates a new drive speed for the variable speed drives in order to correct the situation.

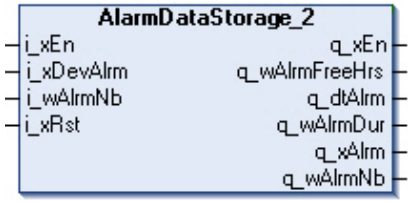


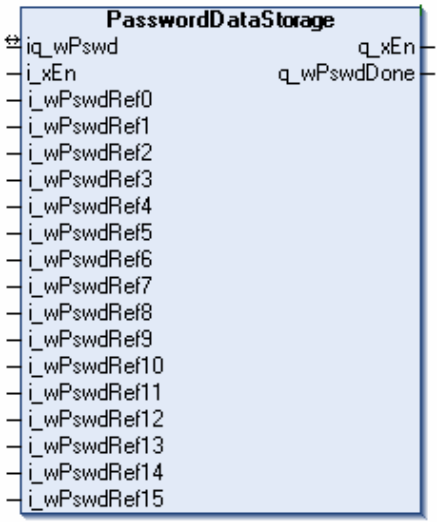
The Anti-crab function detects a skew or drifts in the bridge against the rail and calculates a corrected speed for both of the drives in order to maintain the bridge parallel to the rail.

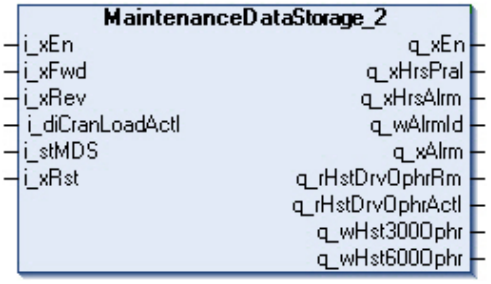


Monitoring data storage

There are several function blocks which collect and monitor the data associated with the crane's operation for mandatory statistics, maintenance and troubleshooting.







NOTE:
Besides the function blocks **AlarmDataStorage_2**, **MaintenanceDataStorage_2** and **StatisticDataSorage_2** the library contains as well their preceding function blocks:

- **MaintenanceDataStorage**
- **AlarmDataStorage**
- **StatisticDataStorage**

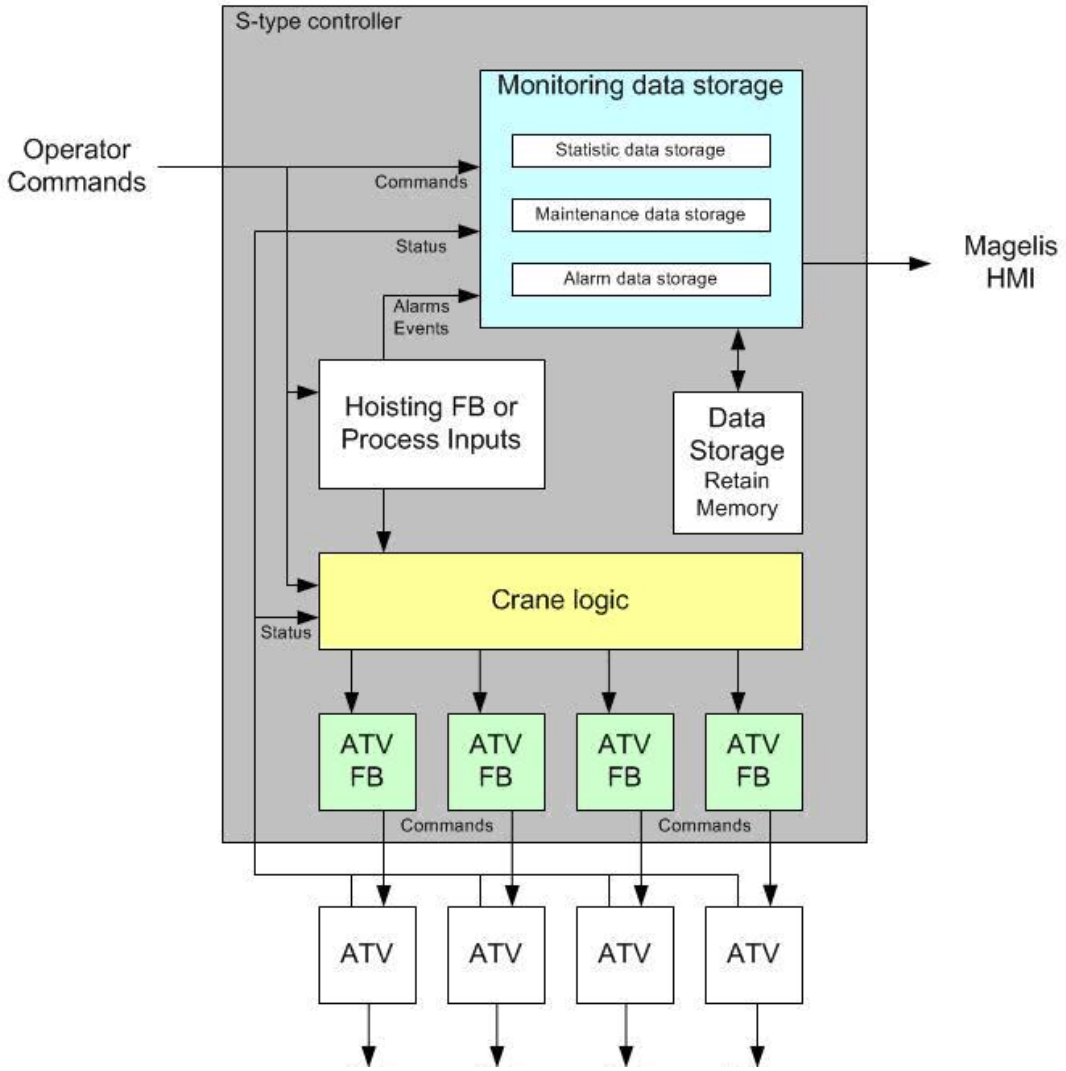
For consistency and maintenance reasons these three FBs listed above are still kept in the latest Hoisting library. However, it is recommended to start new projects by using the most recent version of the function blocks (**AlarmDataStorage_2**, **MaintenanceDataStorage_2**, **StatisticDataSorage_2**) only .

Monitoring data storage

The Monitoring data storage function consists of three parts.

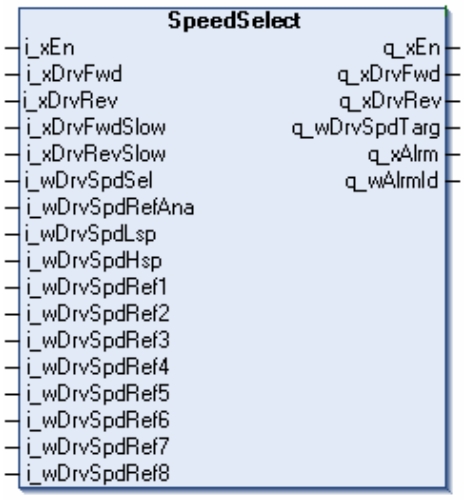
- Alarm data storage function block series
- Statistic data storage function block
- Maintenance data storage function block

The Alarm data storage function block series are pre-configured for the following events: Overload, Over-speed, Encoder Alarm, Over-torque and Load Slippage. All event signals are archived with their respective date, time and duration.
The Statistic data storage function block records all events that are part of the movement. The function block records four separate movements. Hoisting, Trolley, Bridge travel and Slewing. Depending on the crane, the inputs can also be used for other movements. For each type of movement the function block counts the events and calculates an overall runtime for each axis.
The maintenance block uses actual loading information to calculate the amount of used theoretical lifetime for each movement and generates an alarm if maintenance is needed.

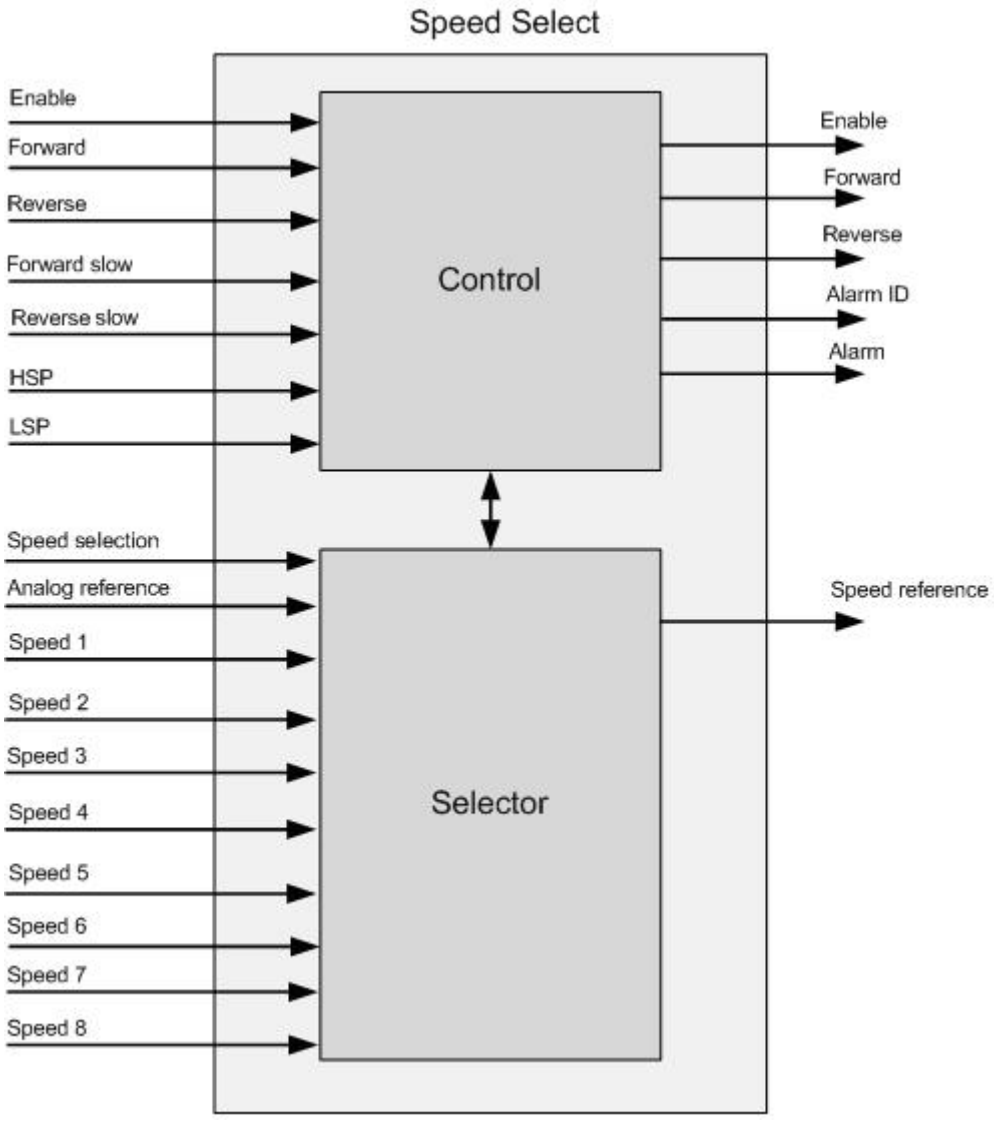


Speed select

The **Speed select** allows the user to select the target speed. The user can select the speed reference by using either an analog input or a selector switch.

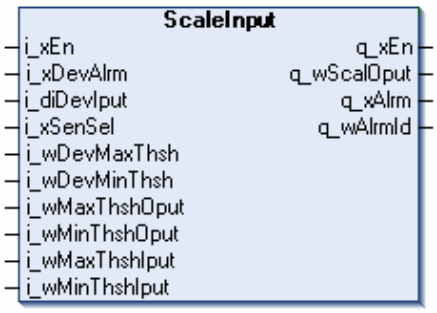


The Speed select function block is used to select a speed value from different sources. The required speed value can be either a fixed or an analog value.



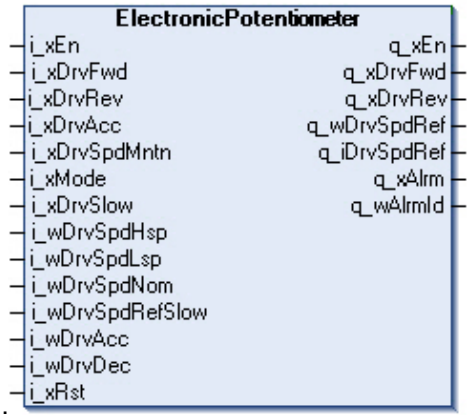
Scale input

Scale input is a scaling function which converts an input value to a specified range. Its range is linear and it generates an error signal in the case of invalid parameters.



Electronic Potentiometer

The **ElectronicPotentiometer** FB offers an effective approach of setting an arbitrary analog speed reference and direction of movement of a variable speed drive using 3 or 4 digital inputs. Although this FB does not require specific hardware to work, it is recommended to use it together with a joystick control.



Detailed Component List

The following is a list of components for the main components of the Optimized CANopen ATV-IMC architecture.

Sarel cabinet

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
1.1	1	Cabinet with mounting plate	NSY*	
1.2	2	Side wall 1800 x 600 mm	NSY2SP186	
1.3	1	Filter fan, 300 m³/h, IP54	NSYCVF300M***PF	
1.4	1	Cabinet filter	NSYCAG223LPF	
1.5	1	Wiring diagram pocket	NSYSDP6M	
1.6	1	Cabinet light incl. socket, magnetic fixing	NSYLAM75	
1.7	1	Thermostat with Fahrenheit scale, 1 NO, 32…140 °F (0…60 °C)	NSYCCOTHOF	
1.8	1	Cable Gland Plate 1 Part 1200x600	NSYEC126	
1.9	2	Set of 4 Enclosure Casters	NSYSW	

Main switch

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
2.1	1	Main switch	*consult local catalog	
2.2	1	Contact block	*consult local catalog	
2.3	1	Terminal cover	*consult local catalog	
2.4	1	Rotary drive with door interface	*consult local catalog	

Power supply

Optional

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
3.1	1	Power supply, input 100…120 or 200…500 Vac, output 24 Vdc, 10 A	ABL8RPS24100	
3.2	3	Circuit Breaker C60N 1P, C, 2 A	23726 *consult local catalog	
3.3	1	Circuit Breaker C60N 2P, C, 2 A	23747 *consult local catalog	
3.4	1	Circuit Breaker C60N 2P, C, 10 A	23756 *consult local catalog	
3.5	5	Circuit Breaker C60N 1P, C, 3 A	24427 *consult local catalog	
3.6	8	Circuit Breaker C60N 2P, C, 3 A	24444 *consult local catalog	
3.7	7	Auxiliary contacts for C60N	26924 *consult local catalog	
3.8	1	Power supply, input 120 or 230 Vac, output 24 Vdc, 10 A	ABL4RSM24100	

ATV-IMC, Advantys OTB

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
4.1a	1	Altivar ATV-IMC Drive controller For use in generic applications without solution libraries (Packaging, Conveying, Hoisting)	VW3A3521	V1.1IE21
4.1b	1	Altivar ATV-IMC Drive Controller For applications with use of solution libraries (Packaging, Conveying, Hoisting)	VW3A3521S0	V1.1IE21
4.2	1	Advantys OTB I/O island	OTB1CODM9LP	
4.3	1	Analog card	TM2ALM3LT	
4.4	1	Advantys OTB I/O card 24 Vdc	TM2DDI16DT	
4.5	1	Relay output card	TM2DRA16RT	

Drives

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
5.1	4	ATV312 variable speed drive 1 HP (0.75 kW)	ATV312H075* *consult product catalog	V5.1
5.2	3	ATV71 variable speed drive 1 HP (0.75 kW)	ATV71H075* *consult product catalog	V3.3
5.3	3	Brake resistor for ATV71H075*	VW3A7801	
5.4	1	Extended I/O card ATV 71	VW3A3202	
5.5	2	Logic I/O card ATV 71	VW3A3201	
5.6	3	ATV71 encoder card	VW3A3401	
5.7	2	Lexium 32A servo drives continuous output current: 6 A RMS at 6000 RPM	LXM32AD18M2	V01.03.17
5.8	2	BMH Servo motor with brake	BMH0702T02F2A	
5.9	4	Power cable for Lexium 32, 5 m	VW3M5101R50	
5.10	4	Encoder cable for Lexium 32, 5 m	VW3M8102R50	
5.11	7	Magnetic circuit breaker 4 A	GV2* *consult local catalog	
5.12	2	Magnetic circuit breaker 6.3 A	GV2* *consult local catalog	
5.13	9	Auxiliary contacts for circuit breaker 1 NO 1 NC	GVAE11	

Sensors

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
6.1	2	Inductive Proximity Sensor OsiSense	XS618B1PBL2	
6.2	2	Photo-electric Sensor OsiSense	XUX1ARCNT16	
6.3	2	Reflector for Photo-electric Sensor OsiSense	XUZC80	
6.4	2	Inductive Proximity Sensor OsiSense	XS9C111A1M12	
6.5	6	Limit Switch OsiSense	XCKMR54D1H29	
6.6	2	Overload Limit Switch OsiSense	XCKM110	
6.7	2	Screw Limit Switch TER International (Third party)	depends on the Scaling rates	
6.8	2	Load Cell Vishay (Third party)	KISD-6	
6.9	2	Web Tension Transmitter for Load Cell	PS-1010T	

Encoder

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
7.1	3	OsiSense incremental encoder 58 mm	XCC1510PS11X	
7.2	3	Encoder cable 10 pin	XCCPM23121L5	
7.3	3	Shaft coupling with spring	XCCRAR1010	
7.4	3	Fixing brackets for 58 mm encoder	XCCRE5RN	

HMI

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
8.1	1	Magelis HMISTU655 touch display	HMISTU655	V6.1
8.2	1	Ethernet cable, 2m	490NTW00002	

Preventa

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
9.1	2	Safety module	XPSAF5130	
9.2	1	Safety extension module	XPSECP5131	
9.3	1	Door guarding safety module XPS AC	XPSAC5121	
9.4	1	E-Stop pushbutton for field	XALK178G	
9.5	1	Emergency Stop pushbutton for cabinet door	XB5AS844	
9.6	1	Auxiliary contacts for cabinet E-Stop	ZB5AZ141	
9.7	3	Illuminated push button, 1 NC, blue	XB5AW36B5	
9.8	1	Assembly housing	XALD01	
9.9	1	Door guard switch	XCSPA792	
9.10	1	Actuator for door guard switch	XCSZ13	
9.11	2	Contactors 7.5 kW	LC1D18BL	
9.12	2	Auxiliary contact 2 NO+2 NC	LADN22	
9.13	11	Door guarding contactors 2.4 kW	LC1D09BL	

Harmony

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
10.1	1	Key lock selector switch	XB5AG41	
10.2	1	Box for 3 button	XALD03	
10.3	1	Signal lamp LED white	XB5AVB1	
10.4	2	Pushbutton with LED green	XB5AW33B5	
10.5	1	Pushbutton with LED red	XB5AW34B5	
10.6	3	Pushbutton with LED yellow	XB5AW35B5	
10.7	1	Connection element	XVBC21	
10.8	1	Signal element green	XVBC2B3	
10.9	1	Signal element red	XVBC2B4	
10.10	1	Signal element blue	XVBC2B6	
10.11	1	Signal element white	XVBC2B7	
10.12	1	Signal element orange	XVBC2B5	
10.13	1	Tube with connection	XVBZ02	

CANopen

Hardware-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
11.1	3	CANopen tap with 4x SUBD9	TSXCANTDM4	
11.2	1	CANopen cord set SUBD9 SUBD9 1m	TSXCANCADD1	
11.3	8	CANopen cord set SUBD9 RJ45, 1 m	TCSCCN4F3M1T	
11.4	1	M12 female socket for CANopen 5 m	1525704	

Software Tools

Software-Components				
Pos.	**Qty**	**Description**	**Part Number**	**Rev./ Vers.**
12.1	1	SoMachine (includes Vijeo Designer) on DVD, trial version	MSDCHNSFNV31	V3.1
12.2	1	Single user license for SoMachine	MSDCHNL●UA	
12.3	1	SoMachine (includes Vijeo Designer) with Solution Extension on DVD, single user license (instead position 12.1 and 12.2)	MSDCHLLMUV31S0	V3.1
12.4	1	Advantys Configuration Tool	STBSPU1000	V5.6
12.5	1	PC→ATV-IMC Programming cable, USB to USB	TCSXCNAMUM3P	

Component Protection Classes

Positioning Protection Class	In Field, On Site			Cabinet		
				Front		Inside
Component	IP54	IP65	IP67	IP55	IP65	IP20
Main switch						**X**
Emergency Stop switch housing XALK		**X**				
Preventa relays XPS						**X**
Single/Double switch housing, complete		**X**				
Control switch, 3 positions					**X**	
Indicator buttons		**X**				
Buttons with LED + 1 switch						**X**
Labels 30x40		**X**				
Position switch Universal		**X**				
Contactor						**X**
Phaseo Power Supply						**X**
ATV-IMC drive controller						**X**
Altivar 312 and Altivar 71						**X**
Lexium 32 servo drive						**X**
BMH Servo motor		**X** shaft end IP40				
Advantys OTB						**X**
Magelis HMISTU655					**X**	**X**

Environmental Characteristics

NOTE: The equipment represented in the architecture(s) of this document has been rigorously tested to meet the individually specified environmental characteristics for operation and storage, and that information is available in the product catalogs. If your application requirements are extreme or otherwise do not appear to correspond to the catalog information, your local Schneider Electric Support will be eager to assist you in determining what is appropriate for your particular application needs.

Component Features

Components

Schneider Electric Main Switch/Circuit Breaker
*consult local country catalog

Power Supply Phaseo: ABL8RPS24100

- 1 or 2-phase connection
- 100...120 Vac and 200...500 Vac input
- 24 Vdc output
- 10 A output
- Diagnostic relay
- Protected against overload and short circuits

Preventa safety module: XPSAC5121

Main technical characteristics:

For monitoring	Emergency Stop
Max. Category accord. EN 13849-1	3
No. of safety circuits	3 N/O
No. of additional circuits	1 Solid-State
Indicators	2 LED
Power supply AC/DC	24 V
Response time on input opening	< 100 ms
AC-15 breaking capacity	C300
DC-13 breaking capacity	24 Vdc / 2 A - L/R 50ms
Minimum voltage and current	17 V / 10 mA
Dimensions (mm)	114 x 22.5 x 99
Connection	Captive screw-clamp terminals
Degree of protection	IP20 (terminals) IP40 (casing)

Safety modules XPS AC are used for monitoring Emergency Stop circuits conforming to standards EN ISO 13850 and EN 60204-1 and also meet the safety requirements for the electrical monitoring of switches in protection devices conforming to standard EN 1088 / ISO 14119. They provide protection for both the machine operator and the machine by immediately stopping the dangerous movement on receipt of a stop instruction from the operator, or on detection of a fault in the safety circuit itself.

Altivar ATV-IMC Drive controller:
VW3A3521 and VW3A3521S0

- 10 logic inputs (2 inputs can be used for 2 counters or 2 inputs can be used for 2 incremental encoders)
- 2 analog inputs
- 6 logic outputs
- 2 analog outputs
- A master port for the CANopen fieldbus
- A mini-USB B port for programming with SoMachine software
- An Ethernet port to be used for programming with SoMachine software or Modbus TCP communication.

Magelis Display Terminal: HMISTU655

- High-definition TFT QVGA with 24 Vdc power supply
- 64kK colors
- Serial port RJ45 RS485/232 multiprotocol interface
- 10/100 BaseT RJ45 Ethernet port interface
- 1 mini USB Device + 1 USB Host 2.0 interface
- Temperature range: 32…122 °F (0…50 °C)
- Certificates: UL, CE, cULus, C-Tick

Altivar 71 Variable Speed Drive

- 200 Vac to 240 Vac 1-phase,
  0.5 HP to 10 HP (0.37 kW to 7.5 kW)
- 200 Vac to 240 Vac 3-phase,
  0.5 HP to 100 HP (0.37 kW to 75 kW)
- 380 Vac to 480 Vac 3-phase,
  1 HP to 670 HP (0.75 kW to 500 kW)
- 500 Vac to 690 Vac 3-phase,
  3 HP to 845 HP (2.2 kW to 630 kW)

- Integrated EMC filter
- Temperature range: 14…122 °F (-10…50 °C)
- Speed range 0 to 1000 Hz
- Graphical display for control and parameterization
- Operation via Modbus, CANopen or other buses possible
- 2 analog inputs plus 1 analog output
- Digital inputs, 2 digital status outputs
- 1 shutdown output (Power removal function)
- Option cards for communication buses, Extended I/O and encoder
- Protections of drive and motor
- Compact design, side-by-side installation possible

Altivar 312 Variable Speed Drive

The Altivar 312 drive is a variable speed drive for 3-phase squirrel cage asynchronous motors. The Altivar 312 is robust, compact, easy to use and conforms to EN 50190, IEC/EN 61800-2, IEC/EN 61800-3 standards UL/CSA certification and to CE marking.

Altivar 312 drives communicate on Modbus and CANopen industrial buses. These two protocols are integrated as standard.

Multiple units can be mounted side by side to save space.

Drives are available for motor ratings between 0.25 and 20 HP (0.18 kW and 15 kW), with four types of power supply:

- 200 Vac to 240 Vac 1-phase,
  0.25 to 3 HP (0.18 kW to 2.2 kW)
- 200 Vac to 240 Vac 3-phase,
  0.25 to 20 HP (0.18 kW to 15 kW)
- 380 Vac to 500 Vac 3-phase,
  0.5 to 20 HP (0.37 kW to 15 kW)
- 525 Vac to 600 Vac 3-phase,
  1 to 20 HP (0.75 kW to 15 kW)

Lexium 32 servo drive

- Voltage range:
  1-phase 100 – 120 Vac or 200 – 240 Vac
  3-phase 200 – 240 Vac or 380 – 480 Vac
- Power: 0.5 to 8 HP (0.4 to 6 kW)
- Rated torque: 0.5 to 36 Nm
- Rated speed: 1500 to 8000 RPM
- The compact design allows for space-saving installation of the drive in control cabinets or machines.
- Features the "Power Removal" (Safe Stop) functional safety function, which prevents the motor from being started accidentally. Category 3 with machine standard EN 13849-1
- Lexium 32 servo amplifiers are fitted with a brake resistor as standard (an external brake resistor is optional)
- Quick control loop scan time: 62.5 µs for current control loop, 250 µs for speed control loop and 250 µs for position control loop
- Operating modes: Point-to-point positioning (relative and absolute), electronic gears, speed profile, speed control and manual operation for straightforward setup.
- Control interfaces:
  CANopen, Modbus or Profibus DP
  Analog reference inputs with ± 10 Vdc
  Logic inputs and outputs

Advantys OTB distributed I/O OTB1CODM9LP

Interface module for Advantys OTB I/O-Island with the following technical specifications:

- Bus parameterization via bus backplane module
- Integrated macros for rapid startup
- 16-channel input
- Removable screw terminal block

Advantages when integrating or replacing module

- Slim line design
- Plug-in contacts
- Controller sends configuration every time the power supply is connected

- CANopen connector Sub-D9
- Up to 7 expansion modules can be connected
- Very compact
- 12 Digital Inputs
- 6 Relay Outputs
- 2 Transistor Outputs (Source)
- 2 Remote Fast Counters
- 2 Remote Very Fast Counters
- 2 Impulsion Generators

Advantys Configuration Software STBSPU1000

Software to configure the Advantys OTB, (STB, FTB and FTM).

- Parameterize all the I/O modules of the Advantys OTB platform (digital, analog and intelligent modules) with standard functions.
- Generating of export EDS files for SoMachine

SoMachine OEM Machine Programming Software: MSDCHNSFNV31 or MSDCHLLMUV31S0

SoMachine is the OEM solution software for developing, configuring and commissioning the entire machine in a single software environment, including logic, motion control, HMI and related network automation functions.

SoMachine allows you to program and commission all the elements in Schneider Electric's Flexible and Scalable Control platform, the comprehensive solution-oriented offer for OEMs, which helps you achieve the most optimized control solution for each machine's requirements.

Flexible and Scalable Control platforms include:

Controllers:
HMI controllers:
- Magelis XBTGC HMI controller
- Magelis XBTGT HMI controller
- Magelis XBTGK HMI controller

Logic controllers:
- Modicon M238 Logic controller
- Modicon M258 Logic controller

Motion controller
- Modicon LMC058 Motion controller

Drive controller:
- Altivar ATV-IMC Drive controller

HMI:
HMI Magelis graphic panels:
- XBTGT
- XBTGK

SoMachine is a professional, efficient, and open software solution integrating Vijeo-Designer.
It integrates also the configuring and commissioning tool for motion control devices.

It features all IEC 61131-3 languages, integrated field bus configuration, expert diagnostics and debugging, as well as outstanding capabilities for maintenance and visualization.

SoMachine provides you:
- One software package
- One project file
- One cable connection
- One download operation

Contact

Homepage	http://www.schneider-electric.com

As standards, specifications and designs change from time to time, please ask for confirmation of the information given in this publication.