Automation solution for simple machines
Startup Guide

05/2005

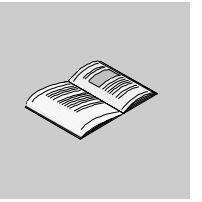
DIA3CD3050101F eng

DIA3CD3050101F

a brand of Schneider Electric

Telemecanique

Table of Contents



Safety Information 7

Part I Presentation of the automation solution for simple machines 9

At a Glance 9

Chapter 1 Introduction 11

At a Glance 11
General 12
Technical specifications 14
Description 17
Required hardware 18

Part II Main System 21

At a Glance 21

Chapter 2 Hardware implementation of the main system 23

At a Glance 23
Main cabling 24
HMI cabling 24
CANopen network cabling 25
I/O cabling 28
Power supply 30

Chapter 3 Software implementation of the main system 31

At a Glance 31
Installing Software and Loading Applications 32
ATV31 communications configuration 35

Chapter 4 Presentation of the application 37

Operating mode 37

Part III Add a second ATV31 variable speed controller 41
At a Glance 41

Chapter 5 Hardware implementation of a second ATV31 speed controller 43
At a Glance 43
Second ATV31 cabling 44
Cabling of the CANopen network. 45
Power supply 46

Chapter 6 Software implementation of a second ATV31 speed controller 47
Add a second ATV31 in the Twido controller application. 47

Part IV Add an Advantys OTB I/O module 57
At a Glance 57

Chapter 7 Hardware implementation of an Advantys OTB module 59
At a Glance 59
Advantys OTB outputs cabling. 60
Cabling of the CANopen network. 61
Advantys OTB I/O cabling 62
Power supply 62

Chapter 8 Software implementation of an Advantys OTB module 63
At a Glance 63
Configuration of Advantys OTB communications 64
Add an Advantys OTB module in the Twido controller application 66

Chapter 9 Add expansion modules to the Advantys OTB. 75
Principle 75
Mount expansion modules 76
Add expansion modules to the Advantys OTB in the Twido controller application. 76

Part V Add an Advantys FTB I/O splitter 83
At a Glance 83

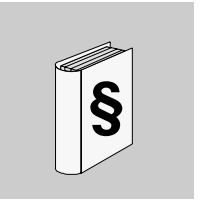
Chapter 10 Hardware implementation of an Advantys FTB splitter 85
At a Glance 85
Cabling of the Advantys FTB I/O splitter 86
Cabling of the CANopen network 87
Power supply 88

Chapter 11 Software implementation of an Advantys FTB splitter 89
At a Glance 89
Configuration of Advantys FTB communications 90
Add an Advantys FTB I/O splitter to the Twido controller application. 91

Appendices 97
At a Glance 97

Appendix A List of symbols used in the application 99
List of symbols used in the application 99

Safety Information



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 or Warning 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, serious injury, or equipment damage.

⚠ WARNING

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

⚠ CAUTION

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

PLEASE NOTE

Electrical equipment should be serviced only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material. This document is not intended as an instruction manual for untrained persons.

© 2005 Schneider Electric. All Rights Reserved.

User Comments

We welcome your comments about this document. You can reach us by e-mail at techpub@schneider-electric.com

Presentation of the automation solution for simple machines

I

At a Glance

Overview

This document presents installation of an automation solution for simple machines.

What's in this Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
1	Introduction	11

Introduction

1

At a Glance

Overview

This chapter presents the automation system described in this document.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
General	12
Technical specifications	14
Description	17
Required hardware	18

General

Introduction

Dedicated to simple installations and small compact machines, the programmable controller Twido, the Altivar variable speed controller, the Magelis XBTN displayer and the new Advantys OTB and FTB distributed I/O are already renowned for their ability to benefit you in terms of capacity, simplicity and competitiveness.

Today their association is a real solution, that guarantees the ease and speed of implementation as well as the prefect functioning of the whole, thanks to a package complete with cabling and well-integrated software (functional blocks integrated in TwidoSoft).

⚠ CAUTION

This document does not replace in any manner the individual documentation of each product.
It describes in simplified manner the installation, setup and implementation presented.
Descriptions and functional specifications of a specific application are not part of this document.
Nonetheless, this document presents a typical automated solution that can be used.
To facilitate system implementation, the setup files and applications necessary for the presented solution are delivered with the products.
Failure to follow this instruction can result in injury or equipment damage.

Abbreviations / Terminology

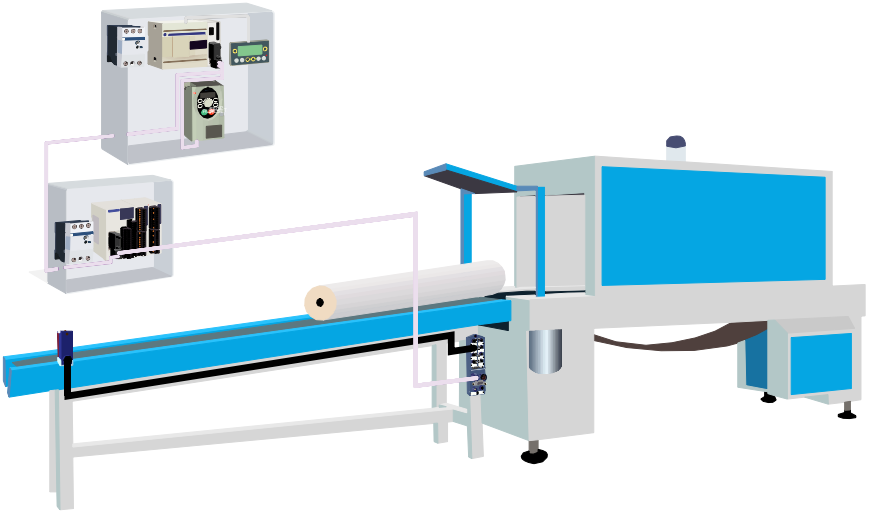
Here are the abbreviations and terms used in this document:

Abbreviation	Description
PC	Personal
XBTN	Displayer with alphanumeric screen
ATV31	Altivar 31 family variable speed controller
TAP	Derivation housing
OTB	Advantys OTB IP 20 I/O module
FTB	Advantys FTB IP67 I/O splitters
HMI	Human/Machine Interface
AC	AC power
DC	DC power
I/O	Input/Output.
Altivar	Name of the family that includes the Schneider variable speed controllers
Magelis	Name of the family that includes part of the Schneider HMI
Twido	Name of the family that includes part of the Schneider controllers

Technical specifications

Example of an automation solution

The diagram below shows an example of an automation solution:



The procedure for automating a simple machine of this type is shown in the following table:

For	Use
Monitor the operation, coordinate, configure and control the different sensors actuators	Twido controller
View and/or configure the different parameters of the application	A Magelis Human/Machine Interface
Operate the belt	An Altivar variable speed controller
Detect and control the products	of the Advantys remote inputs/outputs linked to the sensors/actuators

To respond to these simplified technical specifications, we propose a complete solution outlined in this guide.

Objective

The automation solution presented in this document shows how to control one or several motors. To do this, we propose to link an ATV31 variable speed controller with a Twido controller on the CANopen bus. In fact, the Twido controller includes the Macro Drive developed for ATV31s. This Macro Drive includes micro applications that allow easy development of an application for an ATV31.

It is possible to assign a name to each of the variables used in the Twido application, called SYMBOL.

To display the parameters of the system and change the speed of a motor, a XBTN operator display is added. There is a dynamic link between the list of symbols used in the Twido application and that of the XBTN application. This dynamic link facilitates the development of the operator display application.

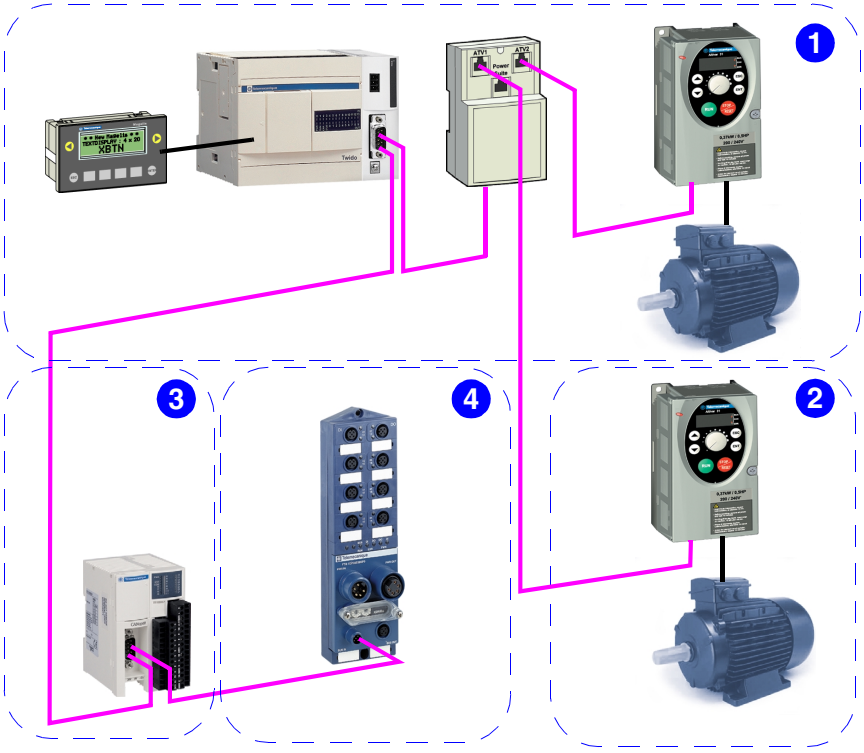
To control the system and give different kinds of information, we shall next add an Advantys OTB IP20 I/O module to the system as well as an Advantys FTB IP67 I/O splitter.

The automation solution includes the following elements:

Element	Description
Twido Controller	The Twido controller associated with the CANopen master supports the following: ● management and communication of the CANopen slaves: the ATV31 variable speed controller and the I/S modules, ● diagnostics for the ATV31 variable speed controller and the I/S modules, ● management of the XBTN 400 operator display.
HMI	The XBTN 400 is used for the following: ● to display the system parameters: motor speed, information on the I/O, system diagnostics, etc. ● to change the system parameters: motor speed.
Variable controller	The ATV31 variable speed controller is used for the following: ● to manage motor Start / Stop mode, ● to change the motor speed.
I/O modules	The I/O modules provide the following: ● controller information to start/stop the motor, ● predefined motor speeds, ● information on the state of the system (led indicators).

Diagram of the Principal

Here is the general diagram of the automation solution presented:



The present guide is divided into several sections depending on the modifications in the automation solution:

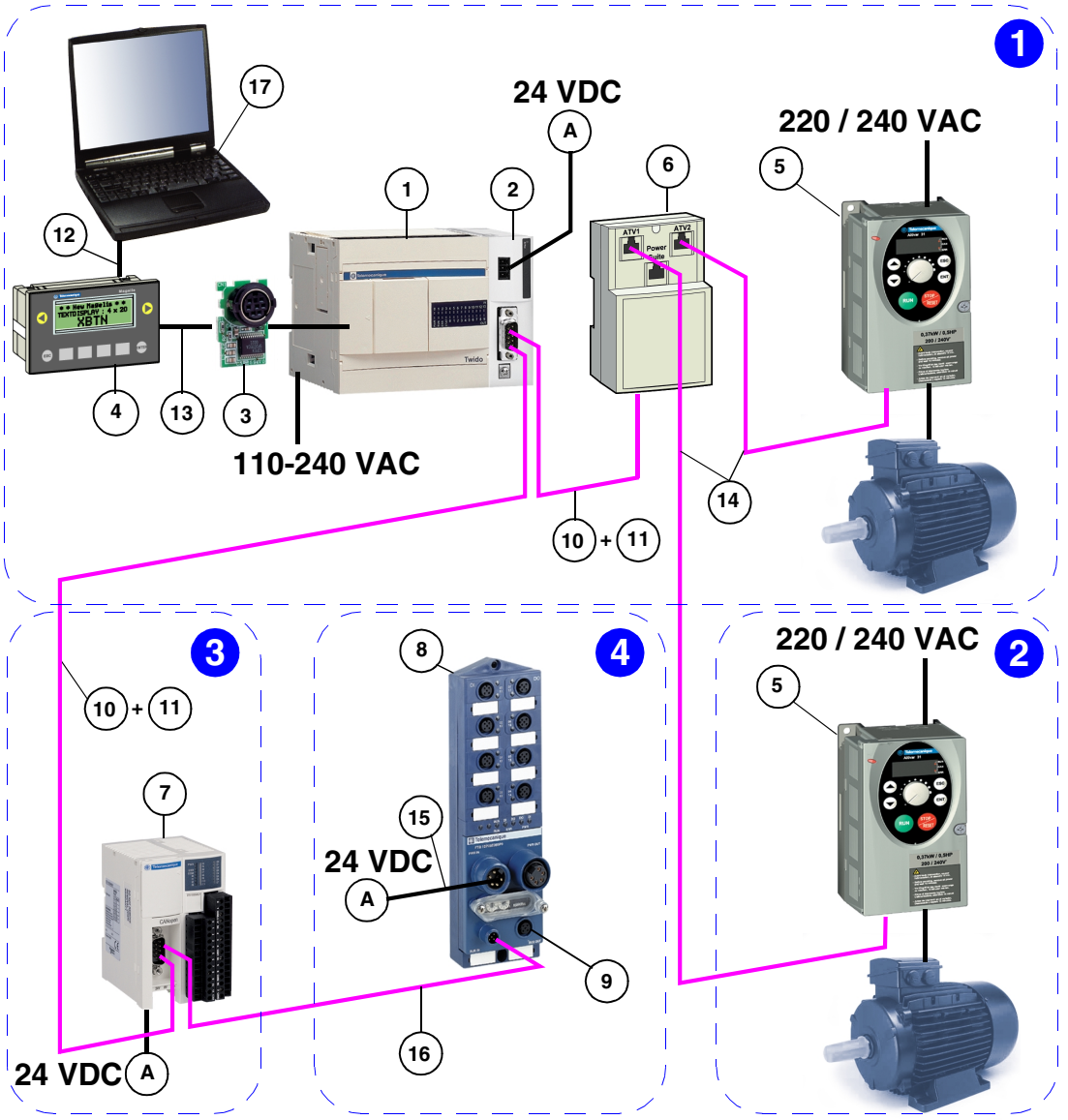
Steps	Description
1	Main system
2	Addition of a second variable speed controller: ATV31
3	Addition of a remote IP20 I/O module: Advantys OTB
4	Addition of a remote IP67 I/O splitter: Advantys FTB

Note: If an application requires more products, check the corresponding user guides to see if they are compatible (for example: maximum number of elements managed by the Twido controller).

Description

Installation diagram

The diagram below shows the automation solution:



Required hardware

Products

List of products:

TYPE	Reference	Description	Quantity	Label
Controller	TWDLCAA24DRF	Compact Twido	1	1
Controller	TWDNCO1M	Twido CANopen master	1	2
Controller	TWDNAC485D	Optional mini DIN RS485	1	3
HMI	XBTN400	Compact display, 4 line, 20 character	1	4
Variable controller	ATV31H18Mxx	0.18 kW/0.5 HP Monophase 200-240 V ATV31	1	5
Accessories	VW3CANTAP2	Derivation housing (TAP)	1	6
I/O	OTB1C0DM9LP	Advantys OTB CanOpen 12E 6S remote IP 20 I/O module	1	7
I/O	FTB1CN08E08SP0	Advantys FTB CANopen 08E 08S remote IP67 I/O splitter	1	8
I/O	FTXCNTL12	CanOpen M12 line end	1	9

Power supply and protection

List of recommended power supplies and protections

Reference	Description	Quantity	Num.
ABL7RM2401	Supply 24VDC	1	A
MULTI 9 -C10	20725 power breaker	1	-

Note: Calibrate the power supply based on the application.

Cables

List of cables:

TYPE	Reference	Description	Quantity	Label
CANopen	TSXCANCA50 *	CANopen cable - bare wire 50 m	1	10
CANopen	TSXCANKCDF90T *	CANopen SUB-D 9 female connector with line end terminal	2	11
HMI	XBTZ945	Configuration cable (PC - HMI)	1	12
HMI	XBTZ9780	Link cable (HMI - Twido controller)	1	13
Variable controller	VW3CANCARR03	0.3 m CAN RJ45 cable for ATV31	1	14
Variable controller	VW3CANCARR01	1 m CAN RJ45 cable for ATV31	1	14
I/O	FTXDP2115	1 m Advantys FTB power cable	1	15
I/O	FTXCN3130	3 m M12 - free wire Advantys FTB power cable	1	16
I/O	FTXCN3150	5 m M12 - free wire Advantys FTB power cable	1	16

* : or equivalent, reference available from 4th quarter 2005

Software

List of programming and configuration software:

TYPE	Reference	Description	Label
Controller	TWDSPU1004V10M	TwidoSoft ≥ V3.2 including a BlueTooth link*	17
HMI	XBTL1001M	XBTL1000 light ≥ V4.4	
I/O	FTXES00 (≥ V3.0)	Advantys Configuration Tool ≥ V1.3	

* : If the PC is not Bluetooth equipped, obtain the USB key adaptor reference VW3A8115.

Configuration

The present guide describes installation and implementation of the system, focusing on the main elements:

- A Twido TWDLCAA24DRF controller,
- An XBTN400 display,
- An ATV31H18Mxx variable speed controller,
- An Advantys OTB1C0DM9LP I/O module,
- An Advantys FTB1CN08E08SP0 I/O splitter,

Main System

II

At a Glance

Overview

This section of the document presents the installation of the main system.

Objective

The objective is to control an ATV31 variable speed controller via the CANopen bus using a Twido CANopen master. The XTBN allows to display / change the instruction of the speed controller and the I/O status information.

Application files

On the CD-ROM BUNDLE (DIA3CD3050101F), there are files for this configuration in the "Applicative files\Partie_II" folder for the Twido controller application and in the "Applicative files\XBTN_XBTR" folder for the XBTN operator display application.

What's in this Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
2	Hardware implementation of the main system	23
3	Software implementation of the main system	31
4	Presentation of the application	37

Hardware implementation of the main system

2

At a Glance

Overview

This chapter describes hardware implementation of the main system of the automation solution.

What's in this Chapter?

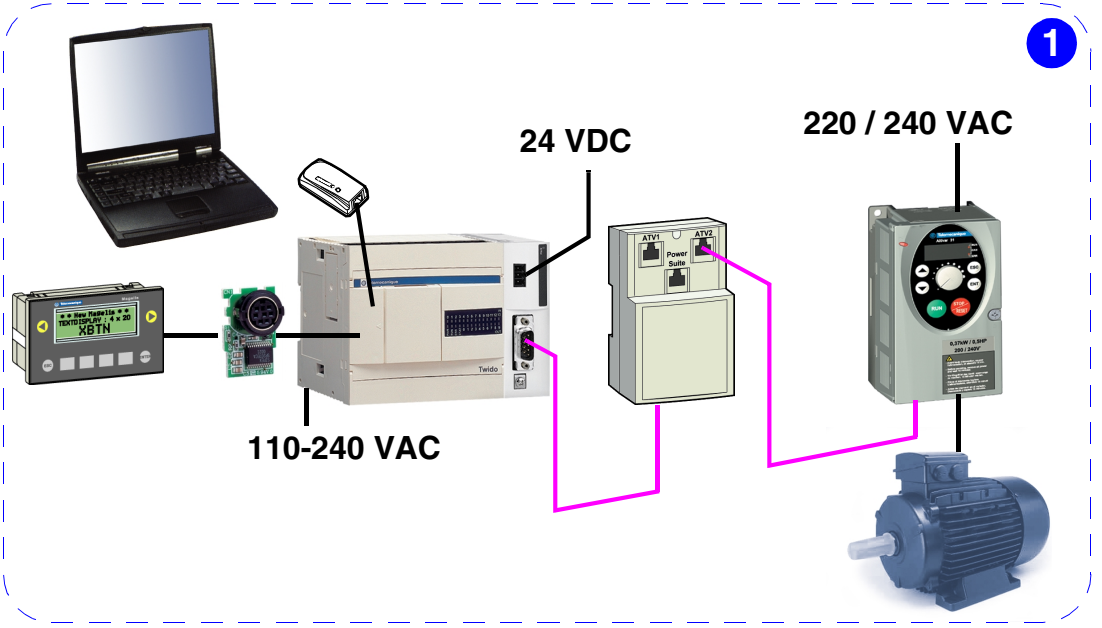
This chapter contains the following topics:

Topic	Page
Main cabling	24
HMI cabling	24
CANopen network cabling	25
I/O cabling	28
Power supply	30

Main cabling

Installation diagram

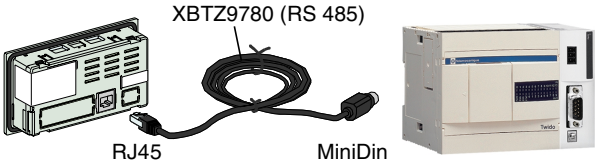
This part of the document covers the following mounting:



HMI cabling

XBTN-Twido cabling

The link between the XBTN and the Twido controller is made with the XBTZ9780 cable:



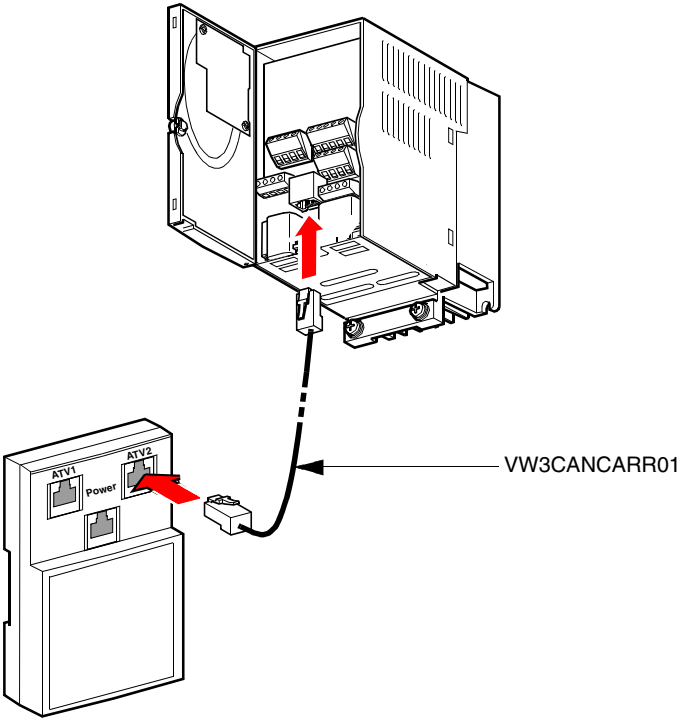
For easier use, use the additional interface mounted on the Twido controller.

For further information, see the Magelis XBTN / XBTR Operation Guide reference 1681028.

CANopen network cabling

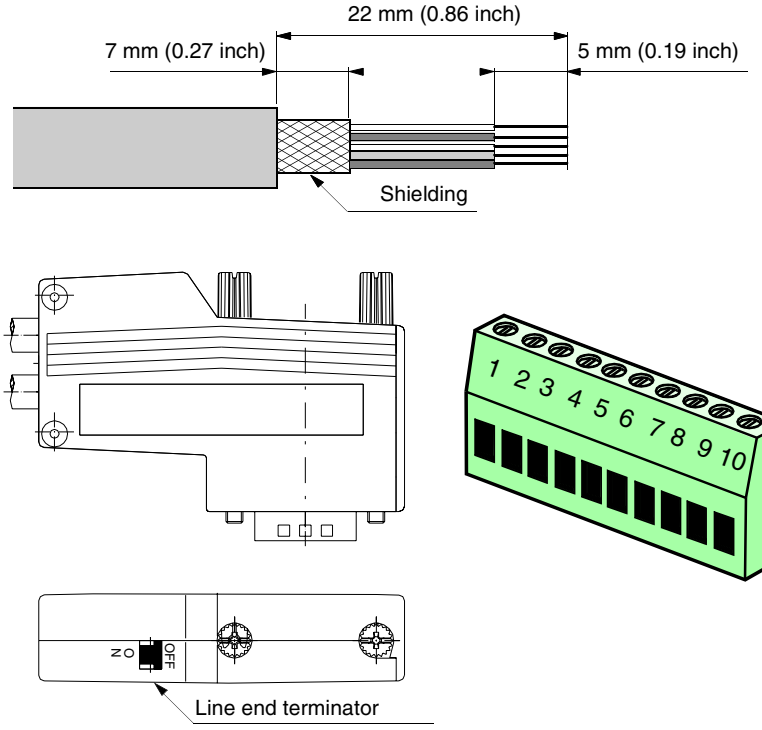
ATV31-TAP Cabling

To connect the ATV31 to the TAP, use the type RJ45 - RJ45 cable so that:



TAP-Twido Cable Preparation

To connect the TAP to the Twido controller, connect a SUB-D 9 TSXCANKCDF90T connector with the TSXCANCA50 cable in the following manner:



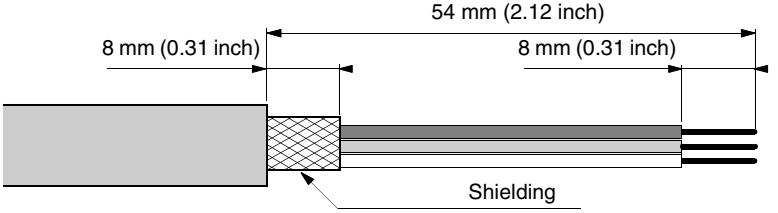
Cable pin assignment in the terminal of the SUB-D 9 connector:

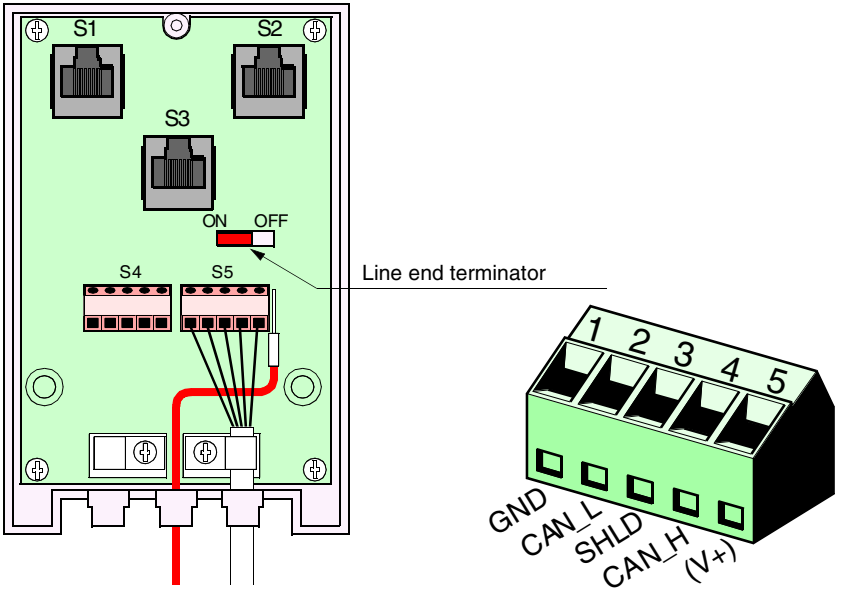
Label	Signal	Cable	Wire color
1	V+	TAP/Twido	Red
2	GND	TAP/Twido	Black
3	SHLD (CAN-GND)	TAP/Twido	(Bare cable shielding)
4	CAN_L	TAP/Twido	Blue
5	CAN_H	TAP/Twido	White

Note: Toggle the line end terminator of the connector to "On".

TAP-Twido cabling

The SUB-D 9 end of the previously prepared cable is connected to the Twido controller. The "bare wire" end of the cable is connected in the TAP in the following manner:





Cable pin assignment in the TAP terminal:

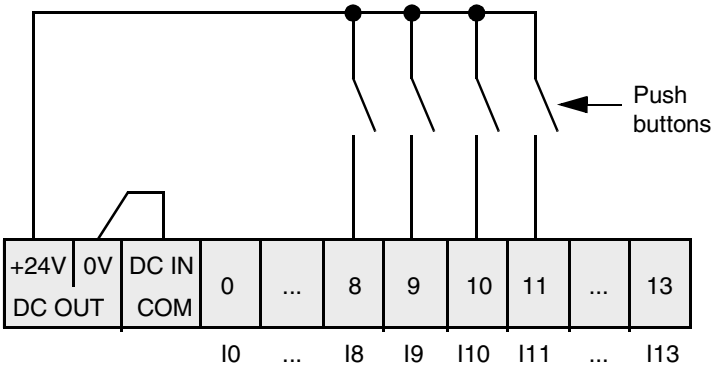
Label	Signal	Wire color	Description
1	GND	Black	Weight
2	CAN_L	Blue	CAN_L polarity
3	SHLD (CAN-GND)	(Bare cable shielding)	Optional shielding
4	CAN_H	White	CAN_H polarity
5	(V+)	Red	Optional power supply (1)

Note: Toggle the line end terminator of the TAP to "On".

I/O cabling

Twido inputs cabling

Diagram of inputs cabling:

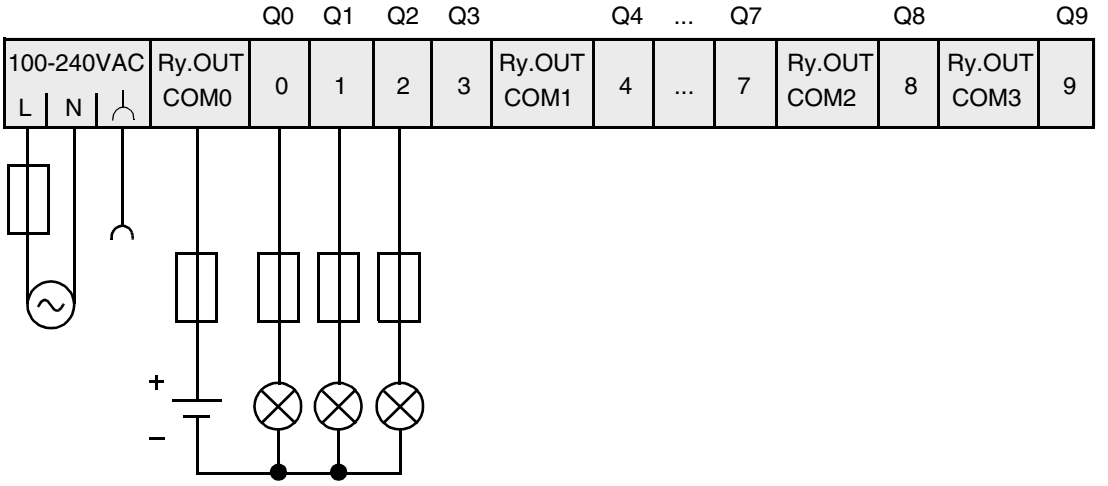


Additional information on inputs used in the application:

Input	Symbol used in the Twido application	Action
%I0.8	PB_START_FORWARD	Start forward motor rotation
%I0.9	PB_START_REVERSE	Start reverse motor rotation
%I0.10	PB_STOP	Stop motor
%I0.11	PB_SLOW_FAST	Set speed fast or slow: • Input = 1 for fast • Input = 0 for slow

Twido outputs cabling

Diagram of outputs cabling:



Additional information on outputs:

Output	Symbol used in the Twido application	Indication displayed
%Q0.0	SD_RUN_FORW	Forward motor rotation
%Q0.1	SD_RUN_REV	Reverse motor rotation
%Q0.2	SD_STOPPED	Motor stopped

Power supply

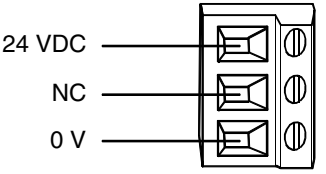
Power on the Twido

To power on the Twido controller, follow the recommendations provided in the "Hardware implementation guide."

Power on the CAN Master

To power on the CANopen master, follow the recommendations in the Service Instructions provided with the product.

Cabling Diagram:



Power on the ATV31

⚠ DANGER

To power on the ATV31 variable speed controller, follow the recommendations provided in the Installation Guide ref. VVDED303043.

Failure to follow this instruction will result in death or serious injury.

Software implementation of the main system

3

At a Glance

Overview

This chapter describes software implementation of the main system of the automation solution.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Installing Software and Loading Applications	32
ATV31 communications configuration	35

Installing Software and Loading Applications

applications

The CD-ROM BUNDLE (DIA3CD3050101F) contains the Applications files corresponding to the automation solution. Loading these files will allow a simple startup for the automation solution.

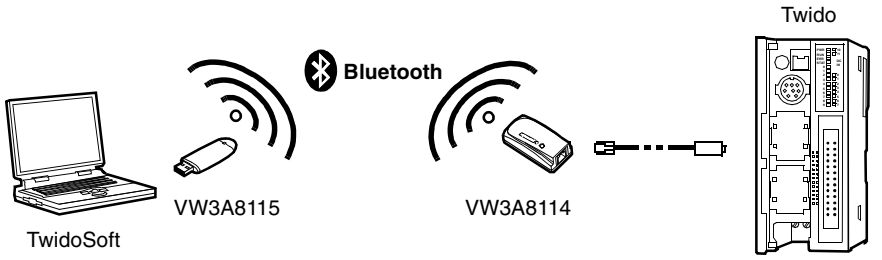
Software Installation

It is necessary, at the start, to install the following software:

- An XBT-L1000 for the XBTN operator display
- TwidoSoft for the Twido controller

Twido Application

Connect the Twido controller to a PC in the following manner:



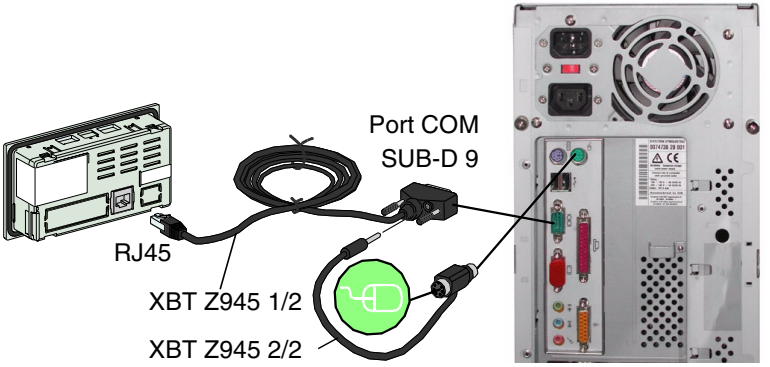
Note: It is recommended to add an RS485 additional interface to the Twido controller. This will allow the PC to be connected to the Twido controller without having to unplug the XBTN operator display.

The following method allows to load the application provided for the Twido controller:

Step	Action
1	From the CD-ROM BUNDLE (DIA3CD3050101F), **COPY** the file "Partie II\QUICK_START_TWIDO_CAN part II conf1 v1_1.TWD"
2	**PASTE** this file into the directory "C:\Program Files\Schneider Electric\TwidoSoft\Applications"
3	Open the file using TwidoSoft software.
4	Connect the Twido controller to the PC (see above diagram). The controller must be powered on.
5	From the Twido software menu bar, select "Controller/Connect...": • If the firmware of the Twido controller is not up-to-date, the following message will appear: "The controller firmware (2.XX) is not compatible with the generated application." Select "Tools/Update PLC microprogram" to launch the update. In step 2/4, click on "Browse" to select, on the TwidoSoft CD-ROM, the "Firmware\X.XX\"TWIDOYYY_V0XXXP00.tfw" file for the controller XXX that corresponds to the firmware version (XXX ≥ 3.2). • If the applications of the PC and the controller are different, a window will appear. Click on the "PC => Controller" button to start loading the program in the Twido controller.
6	Wait until the program is finished loading. From the TwidoSoft menu, select "Controller/Disconnect".

XBTN

To connect the XTBN to a PC, use the XBTZ945 cable:



For further information, see the Magelis XBTN / XBTR Operation Guide reference 1681028.

The following method allows to load the program provided for the XBTN:

Step	Action
1	From the CD-ROM BUNDLE (DIA3CD3050101F), **COPY** the file "XBTN_XBTR\QS_XBT_N400.DOP"
2	**PASTE** this file into the directory "C:\Program Files\Schneider Electric\XBT-L1000\appli"
3	Open the file using the XBT-L1000 software.
4	Connect the XBTN operator display to the PC (see above diagram). The XBTN operator display must be powered on (for more information, see the operation guide of the operator display. The XBTN displays "WAITING FOR TRANSFER".
5	From the XBT-L1000 software menu bar, select "Transfer/Export".
6	Wait until the program is finished loading to disconnect the XTBN from the PC.

ATV31 communications configuration

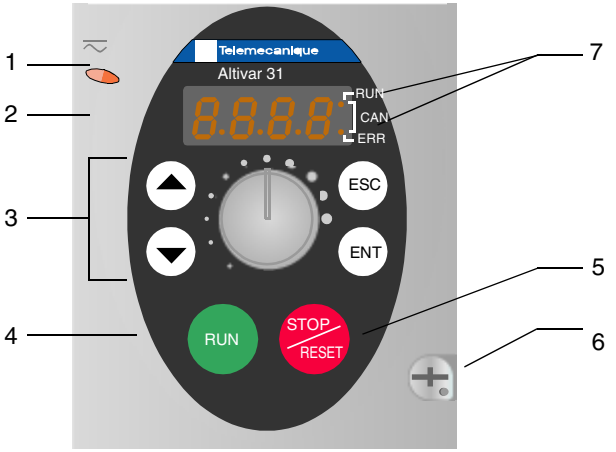
Principle

The system products must be configured according to the Twido application controller as follows:

- Address 1: ATV31
- Transmission Speed: 125 kBits/s

Description

Front panel of the ATV31 variable speed controller:



No.	Description
1	The red LED on indicates that the direct bus is powered on.
2	4-digit "7 segments" operations display.
3	Central programming terminal.
4	"RUN" key, for starting motor in forward mode.
5	"STOP/RESET" key, for stopping the motor and resetting the current defaults.
6	To lock/unlock the front panel of the speed controller, a flat or cross-slot screwdriver is needed.
7	These two LEDs indicate the status of communications ("RUN") and the presence of a possible fault ("ERR") on the CANopen bus.

Method

Configuration of ATV31 communication parameters:

Step	Action
1	Press on the "ENT" key to enter the ATV31 configuration menu.
2	Use the "Arrows" keys to select the "COM" Communication menu then confirm using the "ENT" key.
3	Use the "Arrows" keys to select the "AdCO" menu then confirm using the "ENT" key. Enter the value "1" (Address on the CANopen bus). Confirm using the "ENT" key then exit the menu using the "ESC" key.
4	Use the "Arrows" keys to select the "bdCO" menu then confirm using the "ENT" key. Enter the value "125" (Speed on the CANopen bus). Confirm using the "ENT" key then exit the menu using the "ESC" key.
5	Use the "ESC" key to exit the configuration menu.

Note: The configuration may be modified only when the motor is stopped and when the variable speed controller is locked (cover closed). Any modification entered will become effective after an "Off/On" cycle of the speed controller.
For more information, see the reference document VVDED303042.

Special Note

In the event of use WITHOUT MOTOR (simulation mode), the ATV31 variable speed controller will display an error message: "OPF". Change the configuration in the "FLt / OPL" submenu and send the value "YES" to "OAC".

Communication Parameters

Use the "COM" communication menu to access configuration of CANopen communication functions of the ATV31 connected to a Twido controller:

Parameter	Possible Values	Display on terminal	Values to input for the application
CANopen AdC0 address	0 to 15	1 to 16	1
CANopen bdC0 speed	125 kBits/s	125.0	125 kBits/s
	250 kBits/s	250.0	
	500 kBits/s	500.0	

Presentation of the application

4

Operating mode

Commands

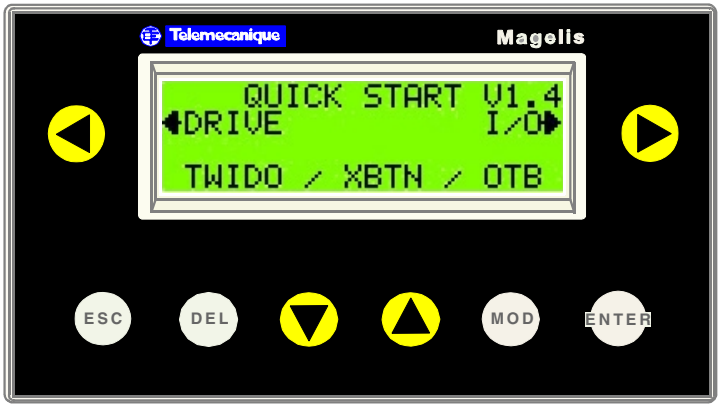
When system installation is complete, the system may be controlled using three push buttons and one two-position button.

Input Corresponding symbol	Function	Description	Output Corresponding symbol
%I0.8 PB_START_FORWARD	Forward start	The corresponding LED turns on when the selected speed is reached.	%Q0.0 SD_RUN_FORW
%I0.9 PB_START_REVERSE	Reverse start	The corresponding LED turns on when the selected speed is reached.	%Q0.1 SD_RUN_REV
%I0.10 PB_STOP	Stop motor	The corresponding LED turns on when the motor stops.	%Q0.2 SD_STOPPED
%I0.11 PB_SLOW_FAST	Predefined motor speeds	Apply a predefined speed when changing position: • %I0.11 is set to 1: 3300 tr/min • %I0.11 is set to 0: 88 tr/min	No output to match input

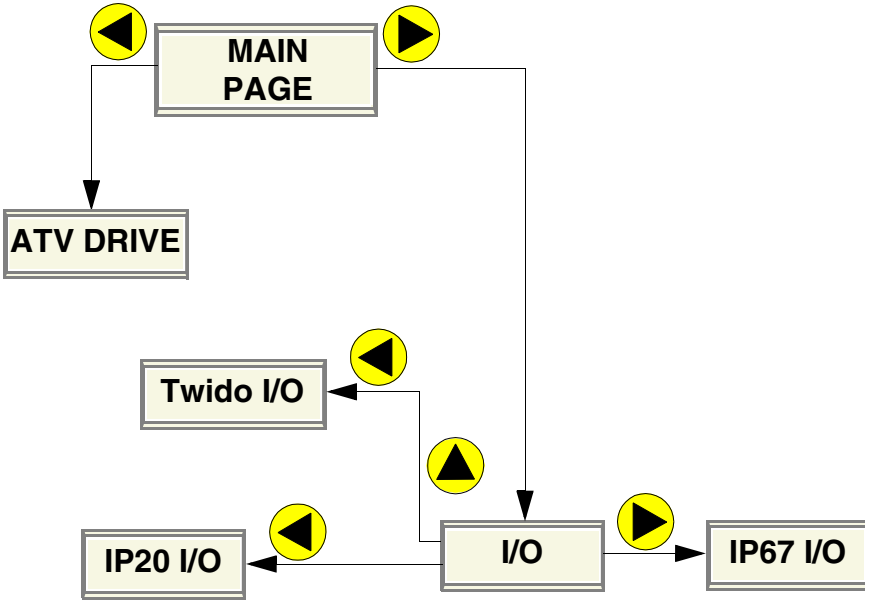
Note: In our sample application, the push button for %I0.10 must be pushed (Stop motor) in order to change the direction of motor rotation.

HMI

XBTN main page:



Organigram of XBTN pages:



Note: Press on the ESC key to return to the preceding page.

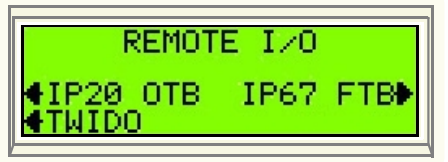
XBTN "ATV DRIVE" page:

The "ATV DRIVE" page displays the speed instruction sent to the ATV31 by the Twido controller.

To modify speed, do the following:

Step	Action
1	Press on the MOD key.
2	Use the ▲ and ▼ arrows to change the speed (immediate write).
3	Press on the ENTER key to confirm.

XBTN "I/O" page:



The "I/O" screen provides access to 3 screens that display the status of:

- I/O of the Twido controller,
- I/O of the Advantys OTB I/O module (see *Add an Advantys OTB I/O module, p. 57*),
- I/O of the Advantys FTB I/O splitter (see *Add an Advantys FTB I/O splitter, p. 83*).

Add a second ATV31 variable speed controller

III

At a Glance

Overview

This section of the document presents a possible evolution of the main system, that is the addition of a second ATV31 speed controller.

These two speed controllers function in exactly the same manner.

Application files

Files for this configuration may be found on the CD-ROM BUNDLE (DIA3CD3050101F) in the "Applicative files\Partie_III" folder for the Twido controller application.

What's in this Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
5	Hardware implementation of a second ATV31 speed controller	43
6	Software implementation of a second ATV31 speed controller	47

Hardware implementation of a second ATV31 speed controller

5

At a Glance

Overview

This chapter describes hardware implementation of a second ATV31 variable speed controller.

What's in this Chapter?

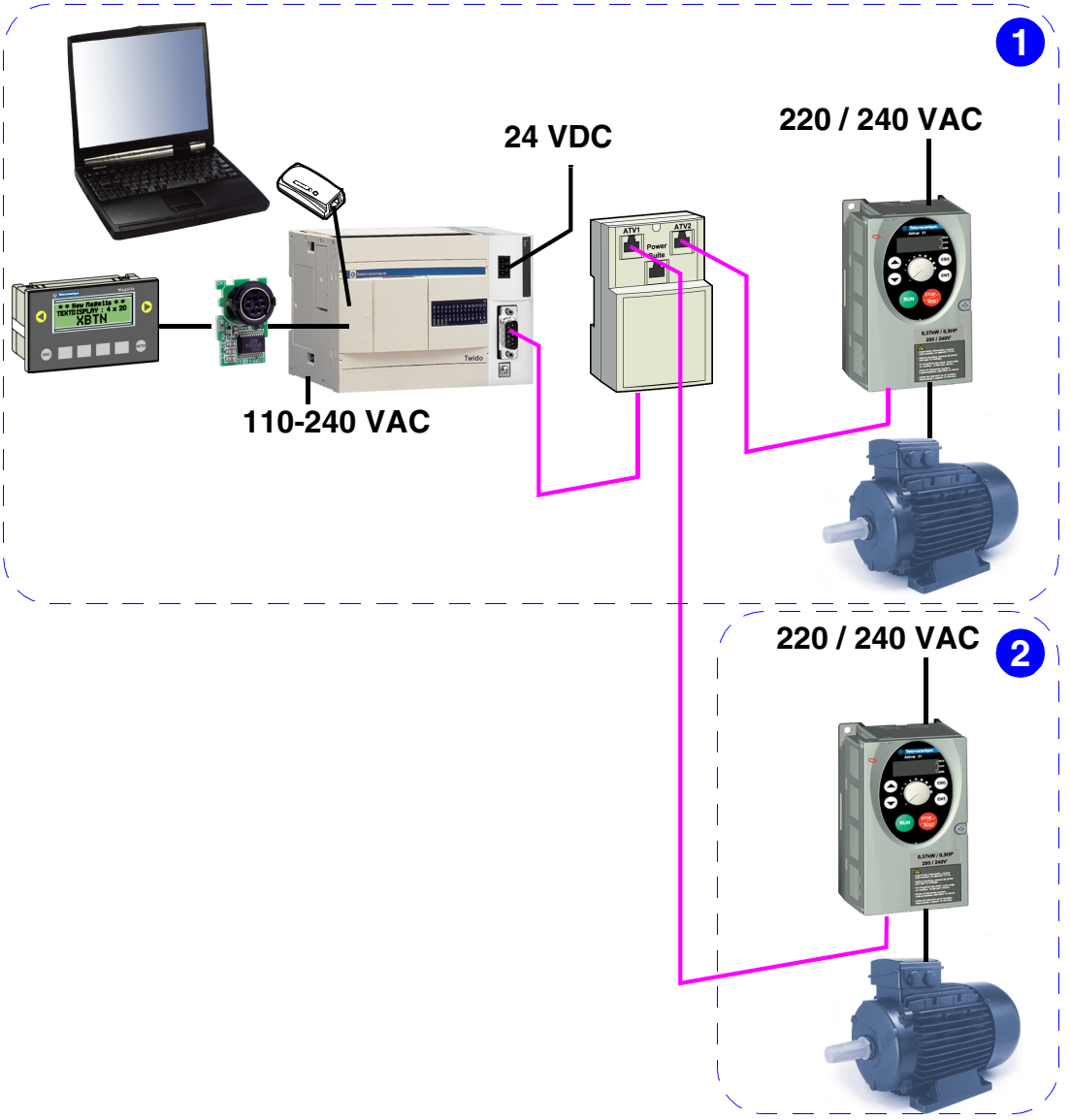
This chapter contains the following topics:

Topic	Page
Second ATV31 cabling	44
Cabling of the CANopen network	45
Power supply	46

Second ATV31 cabling

Installation diagram

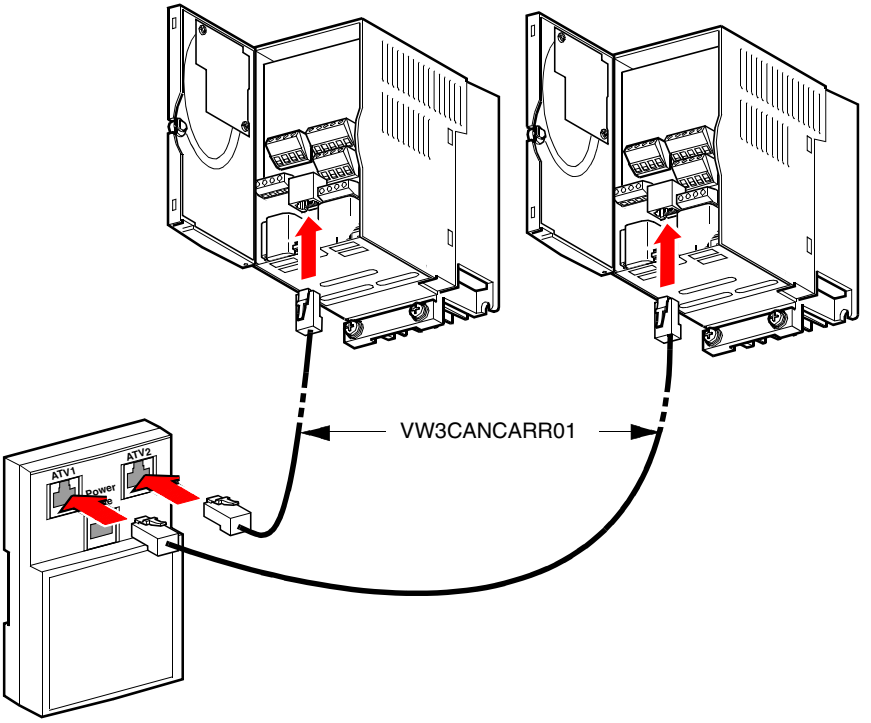
This part of the document covers the following mounting:



Cabling of the CANopen network

ATV31-TAP Cabling

To connect the ATV31 to the TAP, use the type RJ45 - RJ45 cable so that:



Power supply

Power on the ATV31

⚠ DANGER

To power on the ATV31 variable speed controller, follow the recommendations provided in its Installation Guide reference VVDED303043.
Failure to follow this instruction will result in death or serious injury.

Software implementation of a second ATV31 speed controller

6

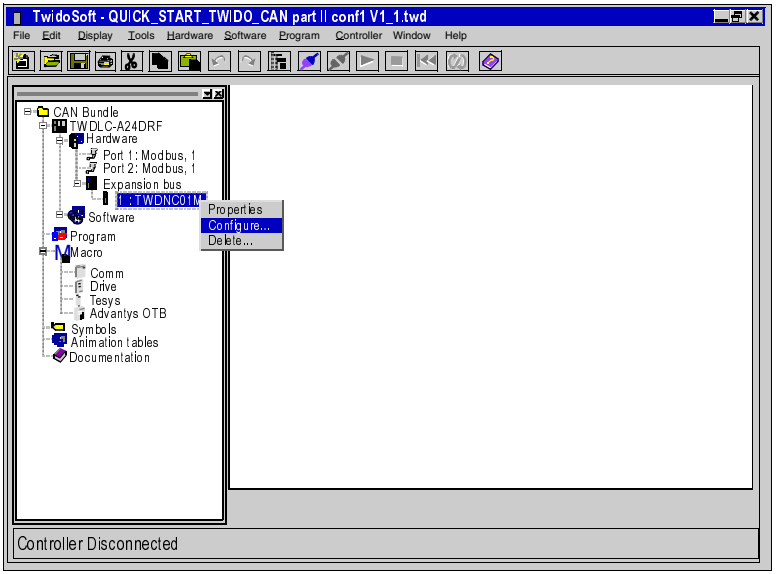
Add a second ATV31 in the Twido controller application

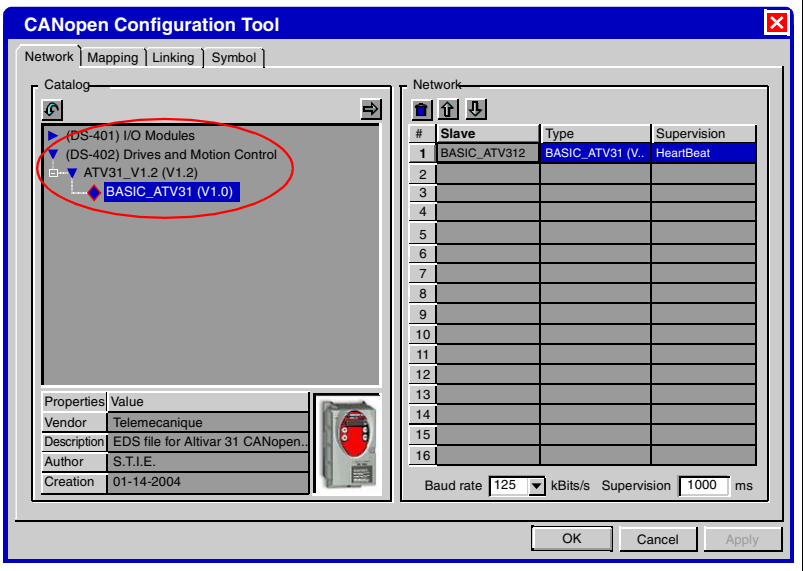
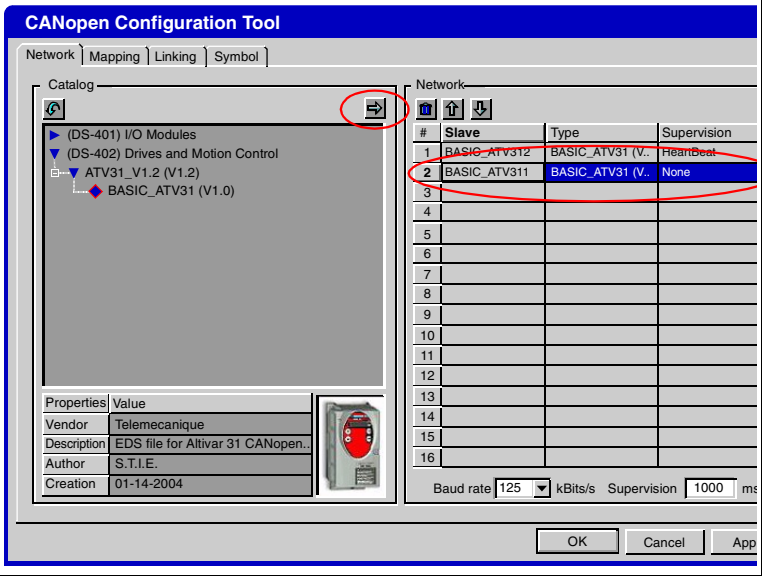
Principle

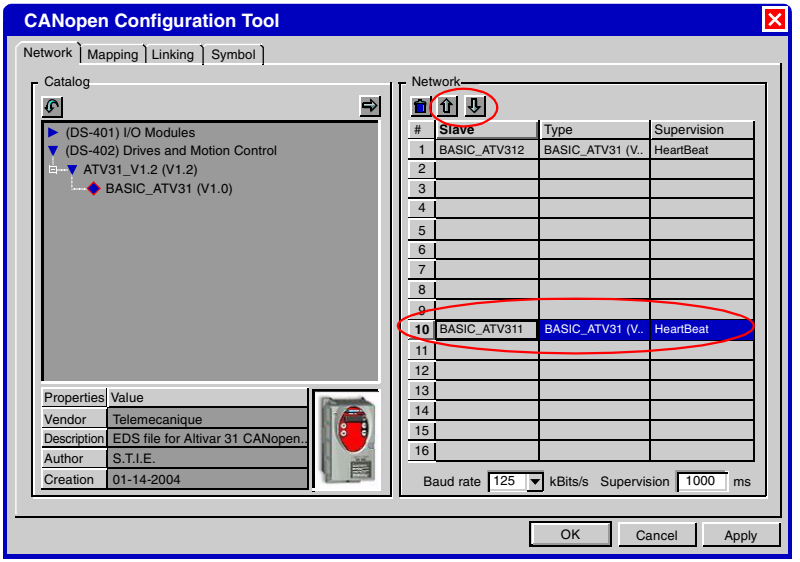
The object is to modify the existing application in order to be able to control the second ATV31 the same as the first ATV31.

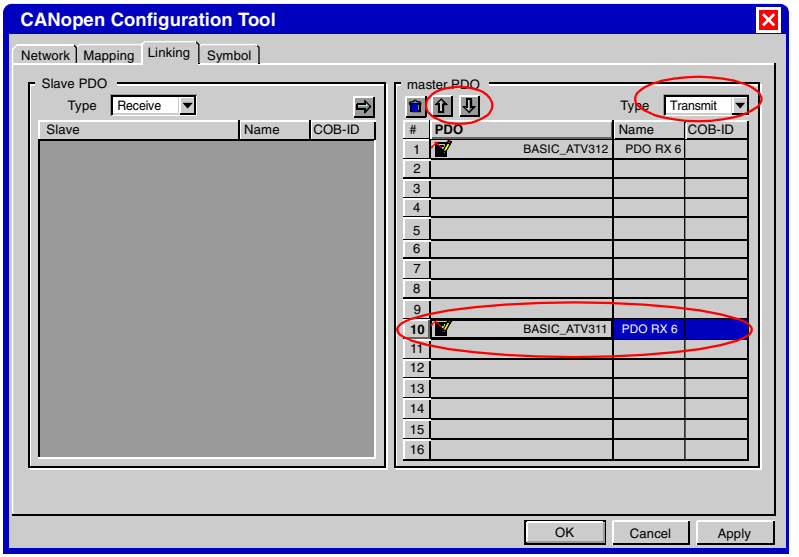
Declare the second ATV31

The following method allows declaration of the second ATV31 variable speed controller in the Twido controller and in the CANopen master.

Step	Action
1	Open the application "BUNDLE\Partie_II\QUICK_START_TWIDO_CAN part II conf1 v1_1.TWD" using the TwidoSoft software (TwidoSoft must be in disconnected mode). Right click on the expansion bus and select "Configure". 

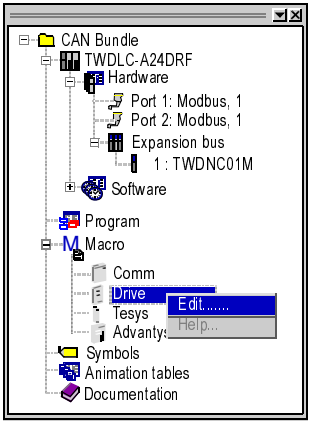
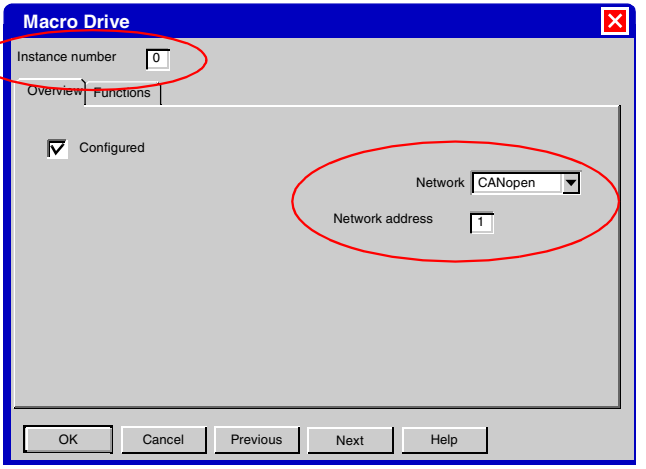
Step	Action
2	Expand the tree of the Drives and Motion Control (DS-402) profile: 
3	Select "BASIC_ATV31 (V1.0)" in the "Catalog" area and click on to add the new ATV31 speed controller in the Network area. 

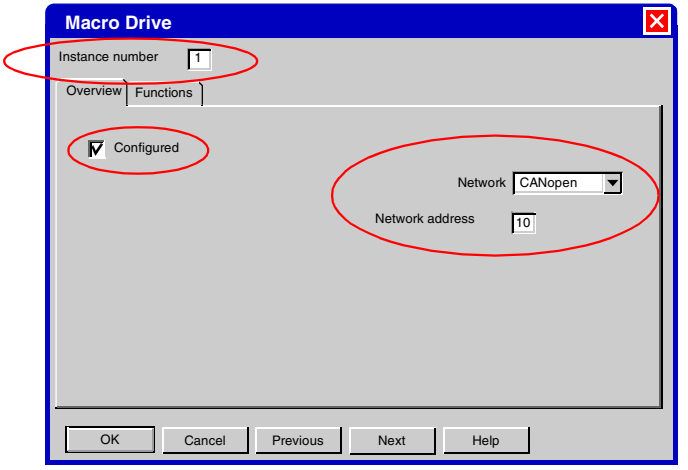
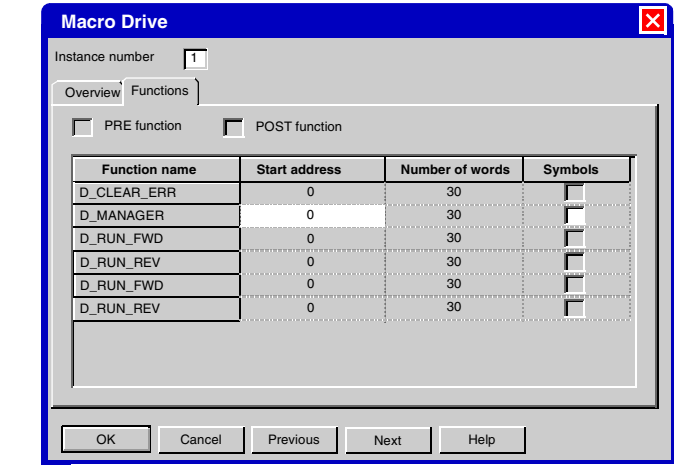
Step	Action
4	Click on to declare the new speed controller at the network address 10:  **Important note**: The Supervision type stays on "None", the motor will continue to turn even if the communication between the CANopen bus and the ATV31 is interrupted. **Change the Heartbeat value**.

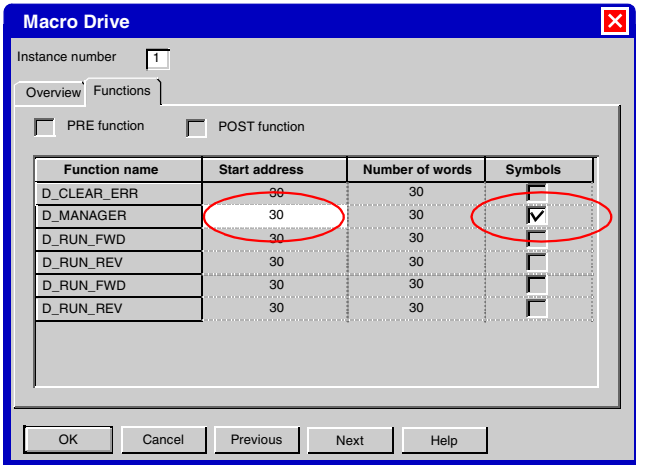
Step	Action
5	Select the "Linking" tab. Use and to assign the new speed controller to the PDO number 10, both in "Receive" and "Send": 
6	Click on the "OK" button to confirm. The new speed controller is then declared in the Twido controller and in the CANopen master.

Macro Drive of the second ATV31

The following method allows configuration of the second ATV31 variable speed controller in the Macro Drive of the application in order to be able to control it.

Step	Action
1	Right click on "Macro/Drive" and select "Edit". 
2	The Macro Drive window appears:  The first ATV31 speed controller is declared in the Macro Drive under number no. 0. It can be seen that its CANopen network address is indeed 1.

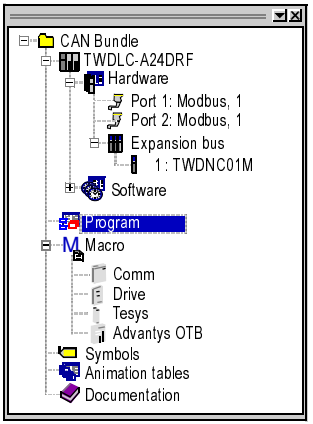
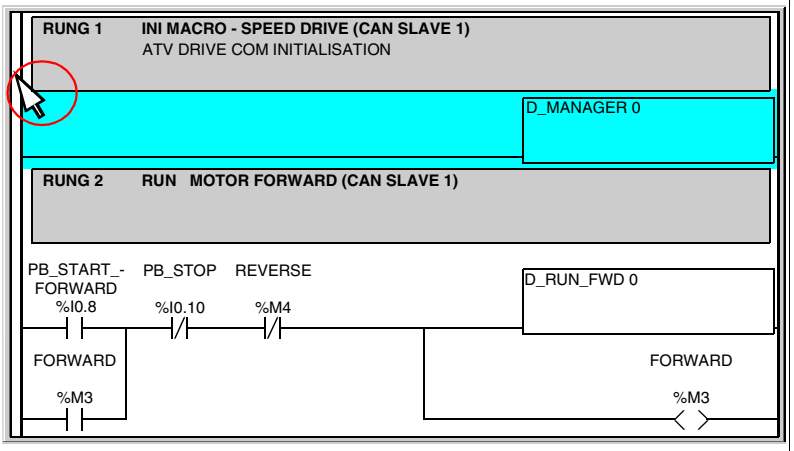
Step	Action
3	Click on the "Next" button to display instance no. 1. Check the "Configure" box. Set the CANopen network address to 10. 
4	Click on the "Functions" tab. 

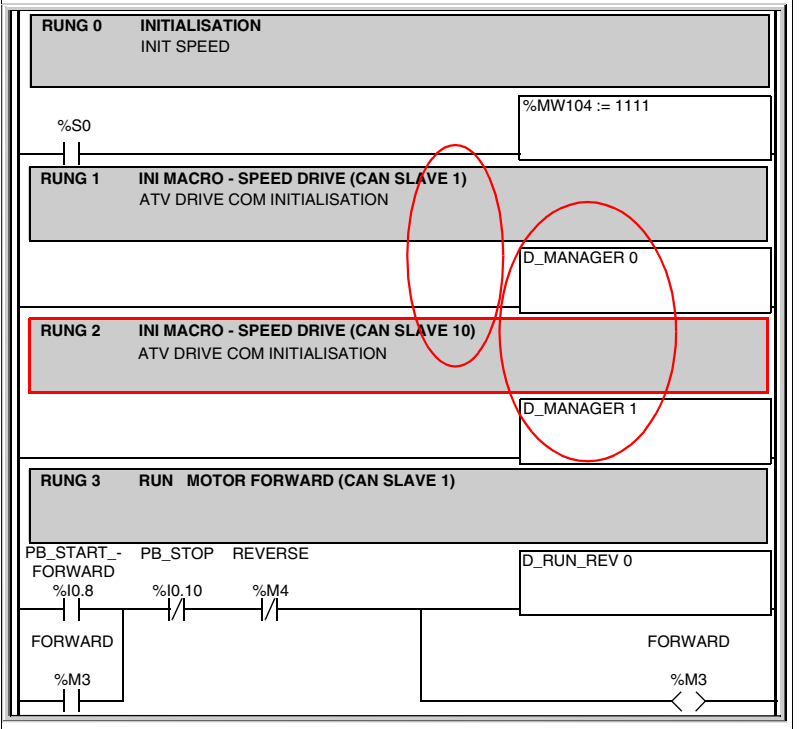
Step	Action
5	Click in the white area (Beginning address) and enter the value 30. Check the "Symbols" box:  **Note**: Instance no. 1 (first ATV31) uses 30 words starting at address 0; i.e. words %MW0 to %MW29. Instance no. 1 should have a different address range. Instance no. 1 uses words %MW30 to %MW59.
6	Click on the "OK" button to confirm.

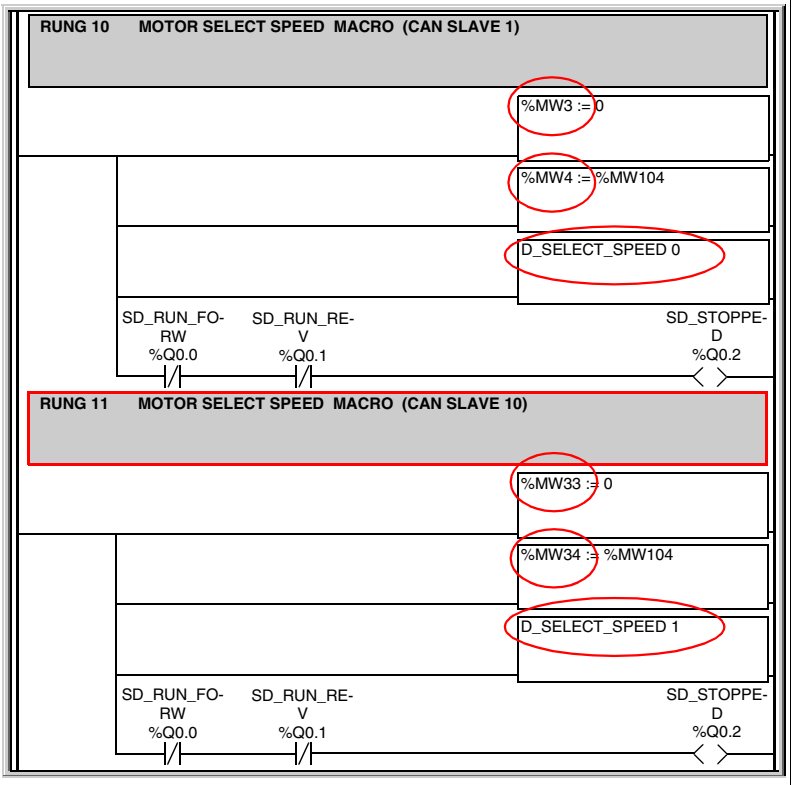
Non-reversible

The following method allows modification of the application to control of the second ATV31 variable speed controller.

The method involves Copy/Paste operations of all the lines that allow control of the first ATV speed controller then changing them to control the second speed controller in the same way.

Step	Action
1	Double click on "Program" to open the programming window: 
2	Click on the left side of the line "RUNG 1" to select the whole line. The selected line turns blue: 
3	Copy the line: "Edit / Copy" or "Ctrl+C". Select the new line ("RUNG 2" for example). Paste the new line: "Edit / Paste" or "Ctrl+V".

Step	Action
4	Select the new line. Double click on the function "D_MANAGER_0" and change it to "D_MANAGER_1" in order to match the instance no. of the second ATV31. Confirm this modification with the "ENTER" key on the keyboard. Click on the line header to change the comment. Next click on the "OK" button to confirm.  The first line allowing to control the second ATV31 is created.

Step	Action
5	Redo steps 2 to 4 so that all lines include the following functions: • D_CLEAR_ERR • D_RUN_FWD • D_RUN_REV • D_STOP • D_SELECT_SPEED On each new line, the words %MW0 to %MW29 relating to the first ATV31 should also be modified respectively as %MW30 to %MW59 (See *Macro Drive of the second ATV31, p. 51*). Here is the example of managing the speed of the motors: 
6	Perform a "File / Save".
7	Load the new application into the Twido controller: Perform a "Controller/Connect" and click on the "PC -> Automate" button.

Add an Advantys OTB I/O module

IV

At a Glance

Overview

This section of the document presents a possible evolution of the main system, that is the addition of an Advantys OTB I/O module.

Note: If you are not using a speed controller in your application, erase the associated elements in the CANopen configuration as well as in the Twido controller application program.

Application principle

The following example will allow to control outputs of the Advantys OTB I/O module the same as those of the Twido controller.

Files

Files for this configuration may be found on the CD-ROM BUNDLE (DIA3CD3050101F) in the "Applicative files\Partie_IV" folder for the Twido controller application.

What's in this Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
7	Hardware implementation of an Advantys OTB module	59
8	Software implementation of an Advantys OTB module	63
9	Add expansion modules to the Advantys OTB	75

Hardware implementation of an Advantys OTB module

7

At a Glance

Overview

This chapter describes hardware implementation of an Advantys OTB I/O module added to the main system.

What's in this Chapter?

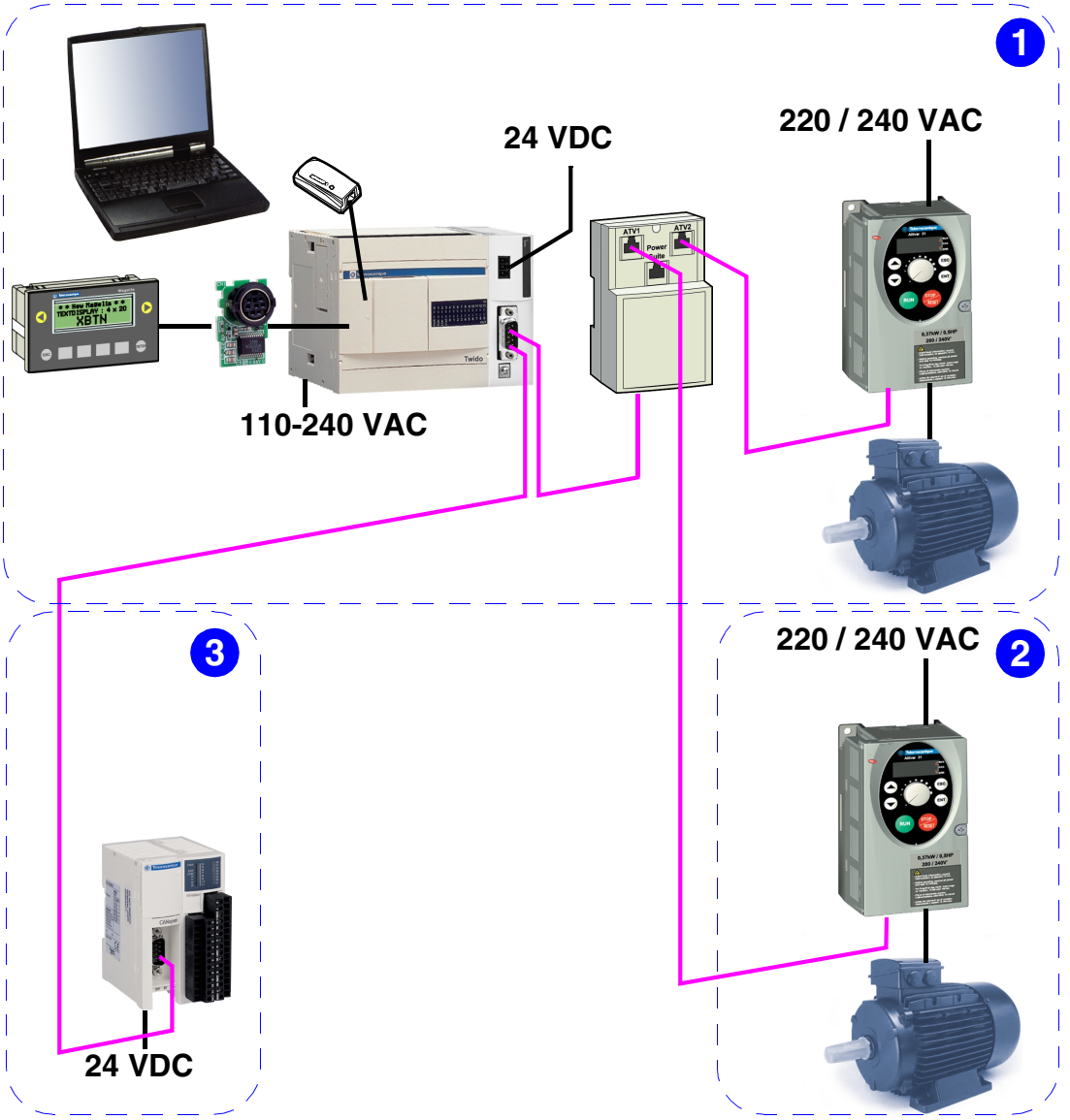
This chapter contains the following topics:

Topic	Page
Advantys OTB outputs cabling	60
Cabling of the CANopen network	61
Advantys OTB I/O cabling	62
Power supply	62

Advantys OTB outputs cabling

Installation diagram

This part of the document covers the following mounting:



Cabling of the CANopen network

Principle

The objective is to connect the Advantys OTB to the Twido controller using a cable that first must be prepared.

OTB-Twido cable preparation

Connect a SUB-D 9 TSXCANKCDF90T connector with the TSXCANCA50 cable (see *TAP-Twido Cable Preparation, p. 26*).

Cable pin assignment in the terminal of the SUB-D 9 connector:

No.	Signal	Cable	Wire color
1	V+	OTB/Twido	Red
2	GND	OTB/Twido	Black
3	SHLD (CAN-GND)	OTB/Twido	(Bare cable shielding)
4	CAN_L	OTB/Twido	Blue
5	CAN_H	OTB/Twido	White

Note: Toggle the line end terminator of the connector on the Advantys OTB side to "On".

OTB side cabling

The SUB-D 9 end of the previously prepared cable is connected to the Advantys OTB.

Twido side cabling

The "bare wire" end of the previously prepared cable is connected to the SUB-D 9 connector joined to the Twido controller. This mounting is performed in the same manner as previously with the following pin assignment:

No.	Signal	Cable	Wire color
6	CAN_L	OTB/Twido	Blue
7	CAN_H	OTB/Twido	White
8	SHLD (CAN-GND)	OTB/Twido	(Bare cable shielding)
9	V+	OTB/Twido	Red
10	V+GND	OTB/Twido	Black

Note: Toggle the line end terminator of the SUB-D 9 connector on the Twido controller side to "Off".

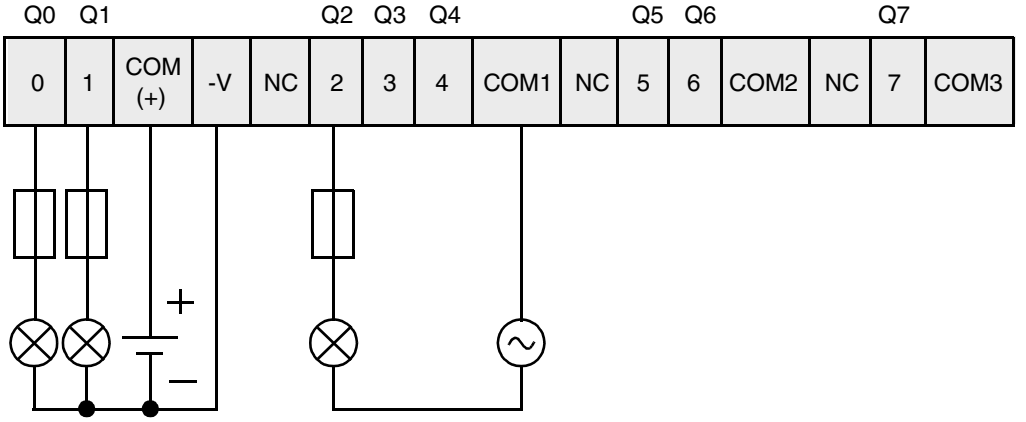
Advantys OTB I/O cabling

OTB inputs cabling

Inputs of the Advantys OTB I/O module are not used in this example. For more information, see the reference guide 160384.

OTB outputs cabling

Diagram of outputs cabling:



Additional information on outputs:

Output	Indication displayed	Information
%QWC1.1.0:X0	Forward motor rotation	Source output
%QWC1.1.0:X1	Reverse motor rotation	Source output
%QWC1.1.0:X2	Motor stopped	Relay output

Power supply

Power on the Advantys OTB

To power on the Advantys OTB I/O module, follow the recommendations provided in reference guide 1606384.

Software implementation of an Advantys OTB module

8

At a Glance

Overview

This chapter describes software implementation of an Advantys OTB I/O module added to the main system.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Configuration of Advantys OTB communications	64
Add an Advantys OTB module in the Twido controller application	66

Configuration of Advantys OTB communications

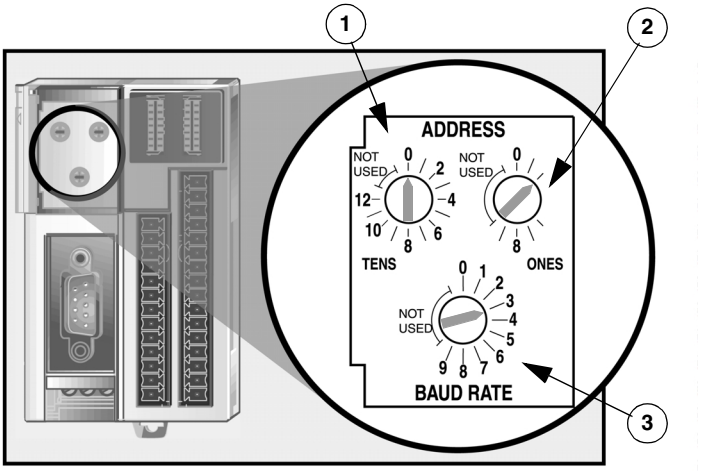
Principle

The system products must be configured according to the Twido application controller as follows:

- Address 1: ATV31
- Address 2: Advantys OTB
- Address 10: second ATV31
- Transmission Speed: 125 kBits/s

Description

Front panel of the Advantys OTB I/O module:



No.	Function
1	Network address (Node-ID x10) encoder wheel
2	Network address (Node-ID x1) encoder wheel
3	Transmission speed encoder wheel

Method

Configuration of Advantys OTB communication parameters:

Step	Action
1	Cut off all power to the Advantys OTB.
2	Set the lower encoder wheel 3 to the position corresponding to the required baud rate. In the example: encoder wheel 3 is set to position 3 (125 kBits/s).
3	Set encoder wheels 1 and 2 to the position corresponding to the desired network address. In the example, the address is 2 (10x0 + 1x2): ● encoder wheel 1: position 0 (tens) ● encoder wheel 2: position 2 (units)
4	Reapply power to the OTB in order to implement the new configuration.

Note: After changing any features of the components of the CANopen network, the bus should be reinitialized by cycling the Twido controller On/Off.

Add an Advantys OTB module in the Twido controller application

Principle

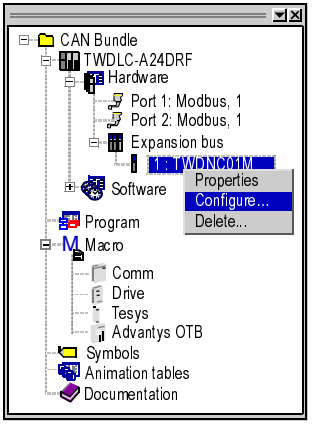
The objective is to modify the existing application in order to be able to manage the I/O of the Advantys OTB module.

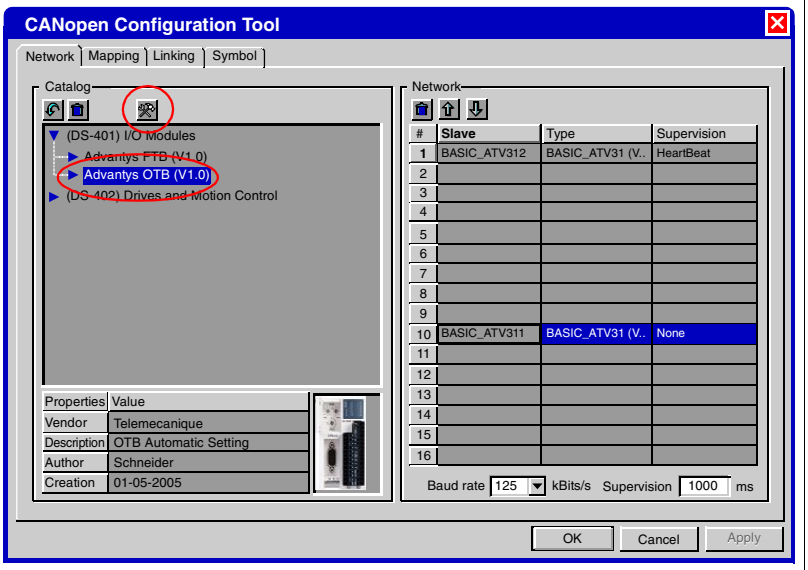
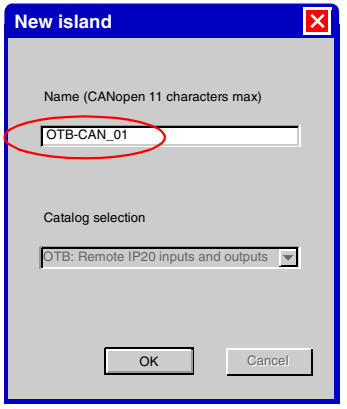
Software Installation

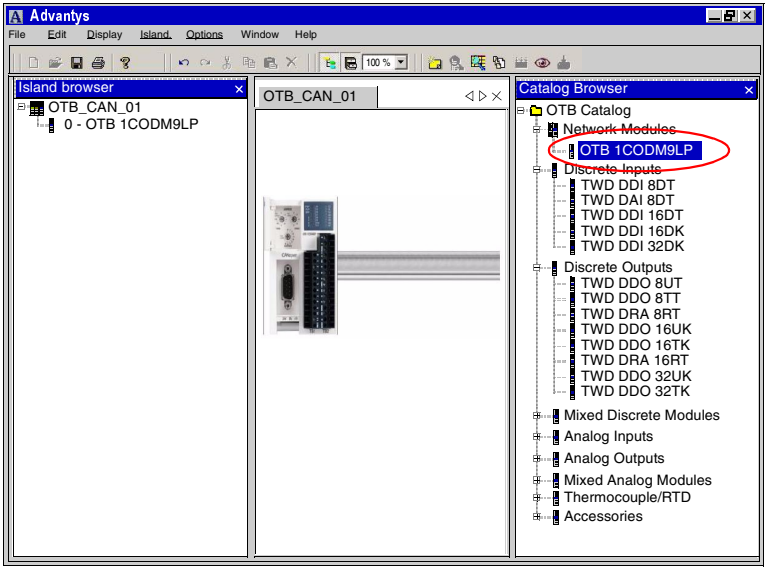
It is first necessary to install the Advantys Configuration Tool software that allows configuration of Advantys OTB and FTB I/O modules and splitters.

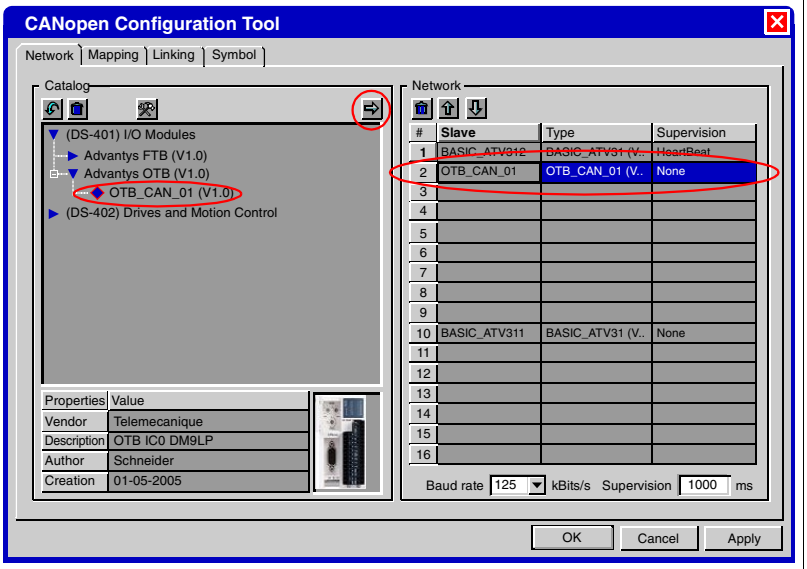
Declare the Advantys OTB module

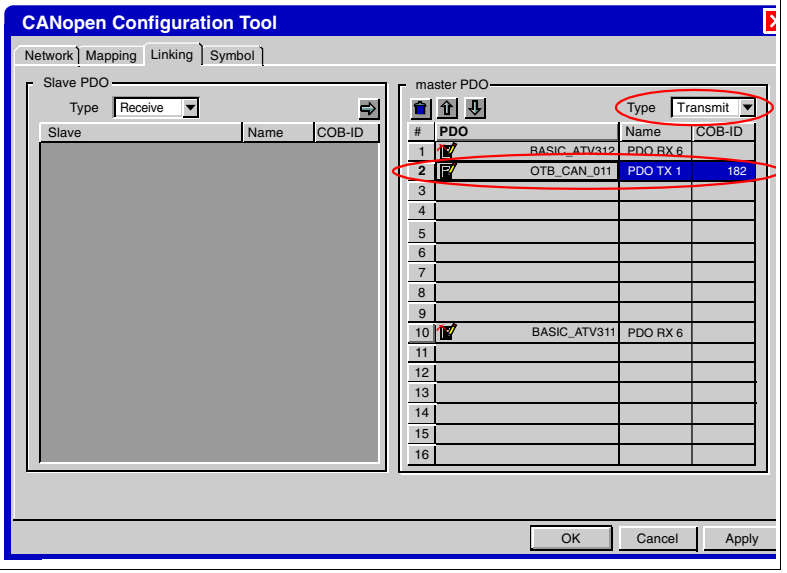
The following method allows declaration of the Advantys OTB I/O module in the Twido controller application.

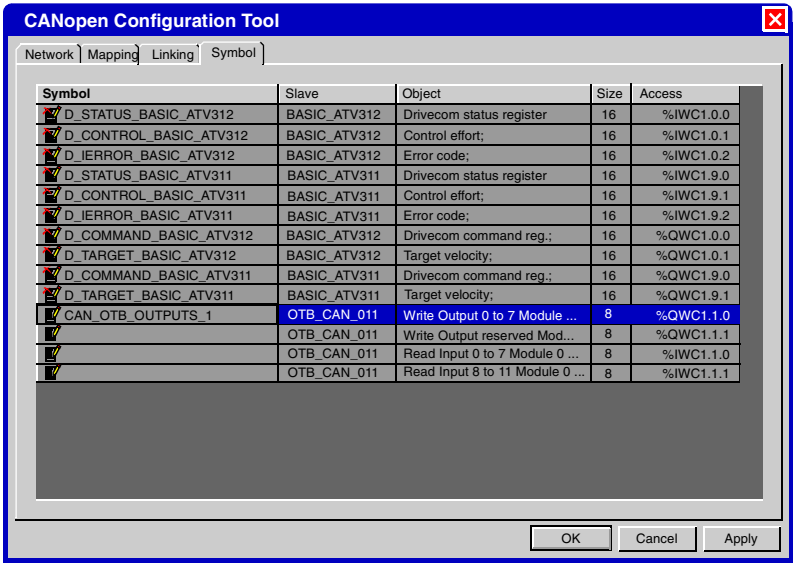
Step	Action
1	Open the existing application using the TwidoSoft (TwidoSoft must be in disconnected mode). Right click on the expansion bus and select "Configure". 

Step	Action
2	Expand the tree of the I/O Modules (DS-401) profile, select "Advantys OTB (V1.0)" and click on to launch configuration and addition of a new element: 
3	In the "New Island" window, enter "OTB_CAN_01" matching the name of the equipment that will be used in TwidoSoft. Next click on the "OK" button to confirm and launch the Advantys Configuration Tool application: 

Step	Action
4	Double click on the module "OTB1C0DM9LP" to add it in the new island. 
5	Perform a "File / Save". Perform a "File / Quit". The new island containing the Advantys OTB I/O module is then added to the catalog under the name "OTB_CAN_01".

Step	Action
6	Select "OTB_CAN_01 (V1.0)" in the "Catalog" area and click on to add the Advantys OTB in the Network area.  Note: The Advantys OTB is configured by default to the first free network address, that is the network address 2 in our example.

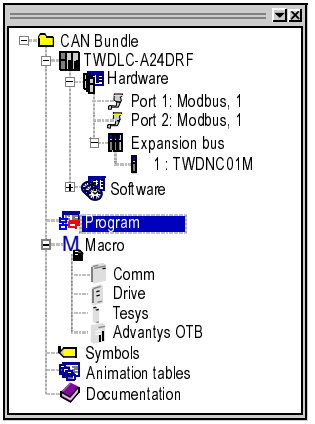
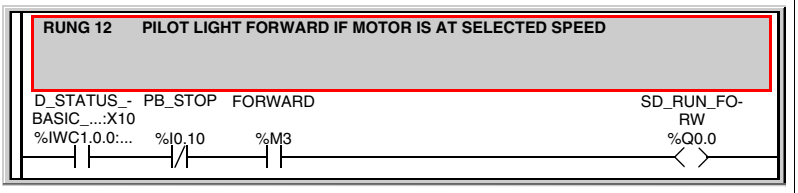
Step	Action
7	Select the "Linking" tab. Use to assign the new Advantys OTB I/O module to the PDO no. 2, in both "Receive" and "Send": 

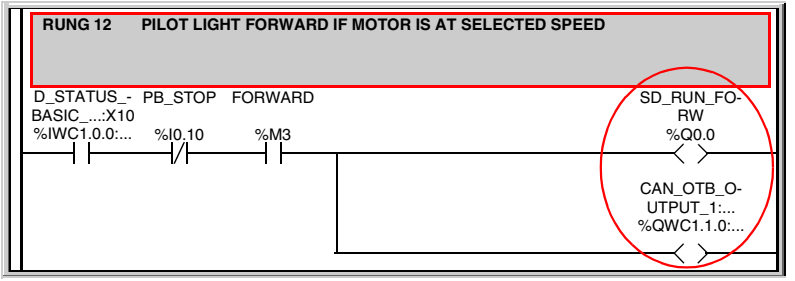
Step	Action
8	Select the "Symbol" tab. Enter, if necessary a "symbol" for the objects associated with the Advantys OTB I/O module: ● Double click in the object's "Symbol" field, ● Type the name of the "Symbol", ● Confirm using the "ENTER" key of the keyboard. 
9	Next click on the "OK" button to confirm the changes. The Advantys OTB I/O module is then declared in the Twido controller application.

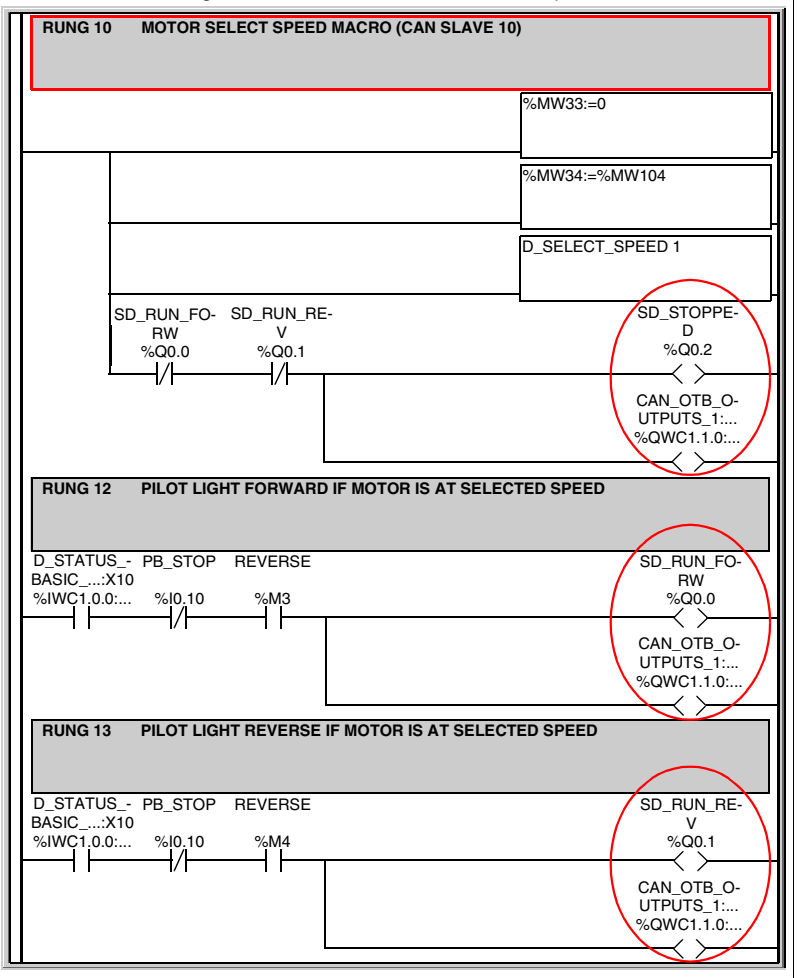
Non-reversible

The following method allows modification of the application to control outputs of the Advantys OTB I/O module the same as those of the Twido controller.

The following method consists in changing the lines that manage the outputs of the Twido controller by adding the outputs of the Advantys OTB.

Step	Action
1	Double click on "Program" to open the programming window: 
2	Select a program line corresponding to the control of outputs of the Twido controller: 
3	Double-click on the electrical diagram to edit it.

Step	Action
4	Add, parallel to the output coil of the Twido controller (output %Q0.0), a coil to which the OTB output is applied: "%QWC1.1.0:X0". Accept Changes The control program line that simultaneously maintains the two outputs: 

Step	Action
5	Redo the same changes to be able to control the other outputs X1 and X2: 
6	Perform a "File / Save".
7	Load the new application into the Twido controller: Perform a "Controller/Connect" and click on the "PC -> Automate" button.

Add expansion modules to the Advantys OTB

9

Principle

Overview

This section describes how to add I/O expansion modules to the Advantys OTB. These expansion modules allow to manage the Discrete and Analog I/O.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Mount expansion modules	76
Add expansion modules to the Advantys OTB in the Twido controller application.	76

Mount expansion modules

Principle

See reference guide no. 160384.

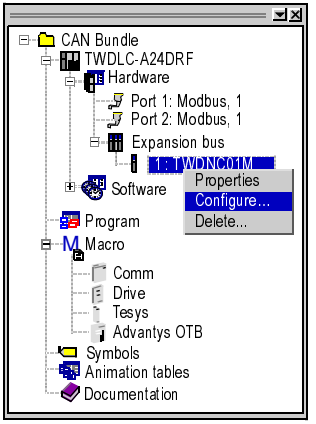
Add expansion modules to the Advantys OTB in the Twido controller application.

Principle

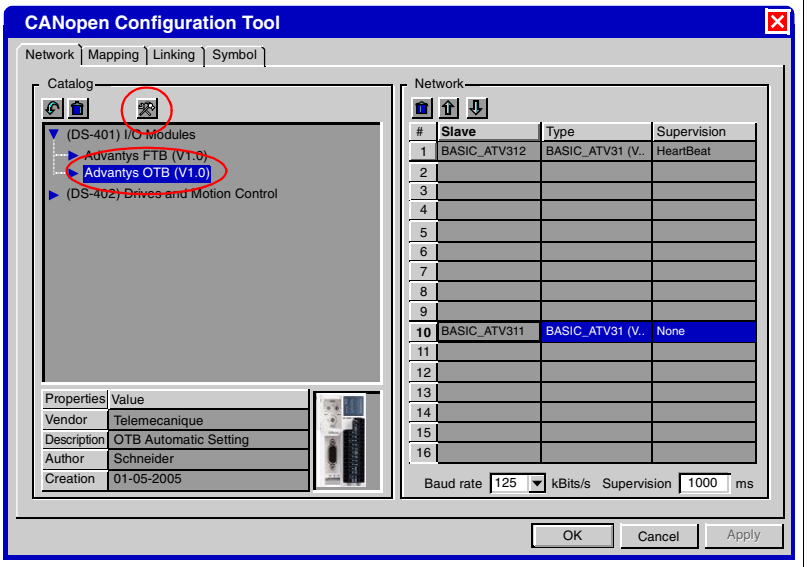
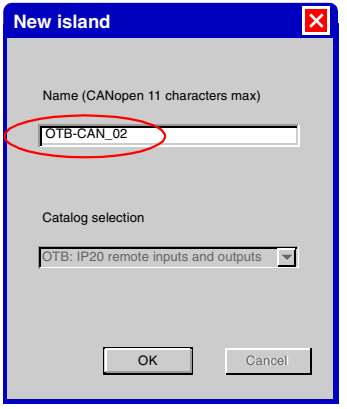
The objective is to modify the existing application in order to be able to manage the I/O of the Advantys OTB I/O module associated with two expansion modules.

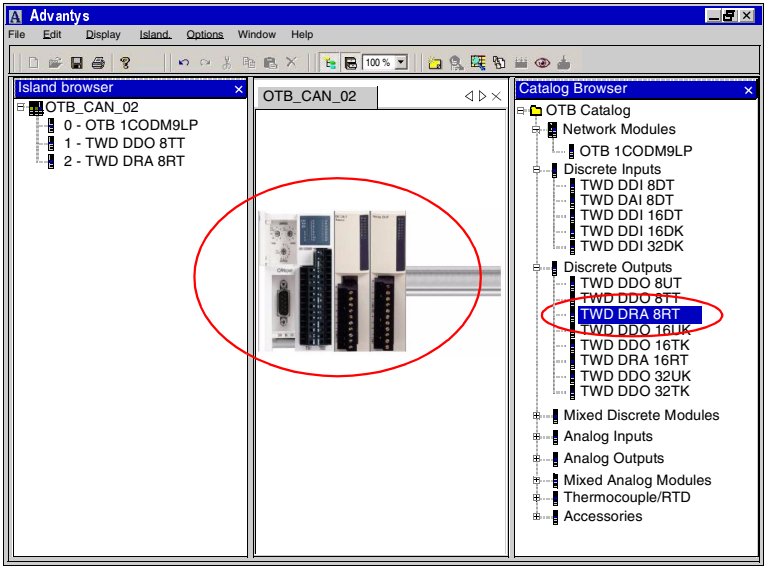
Declare the Advantys OTB module

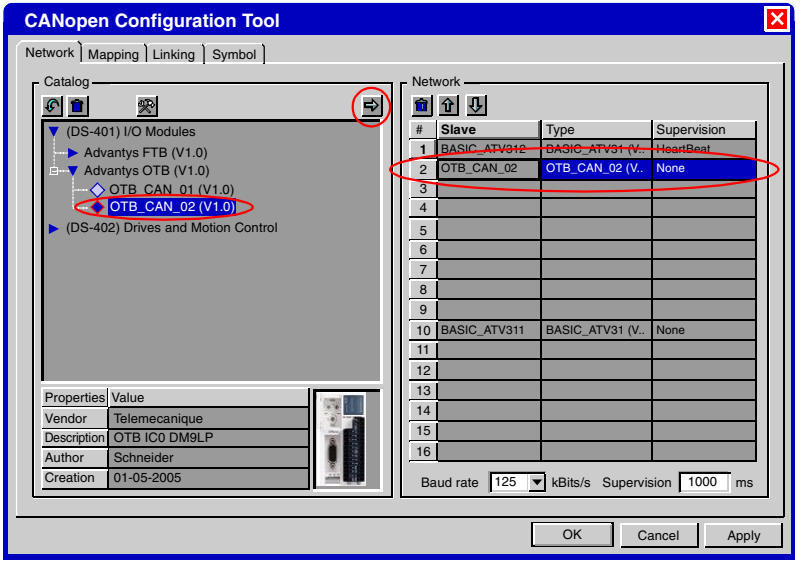
The following method allows declaration of the Advantys OTB I/O module in the Twido controller application.

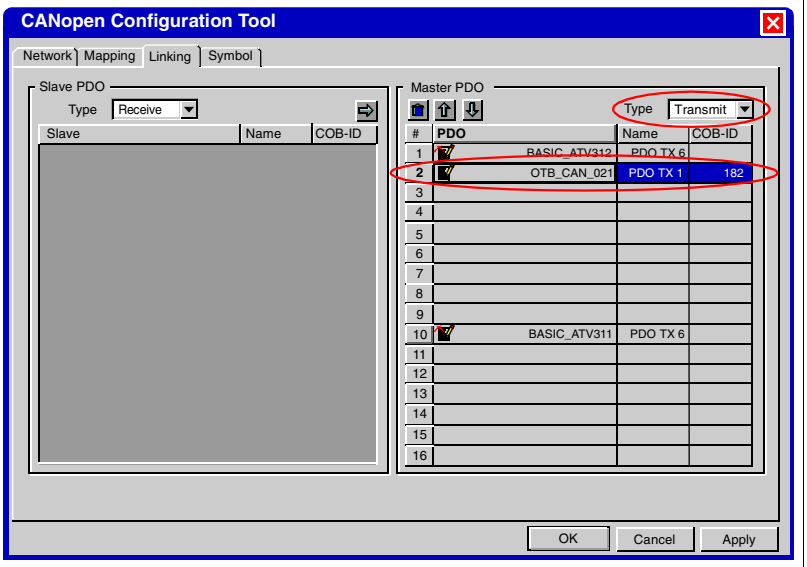
Step	Action
1	Open the existing application using the TwidoSoft (TwidoSoft must be in disconnected mode). Right click on the expansion bus and select "Configure". 

Step	Action
2	Select the module "OTB_CAN_01" in the "Network" area and click on to delete it:

Step	Action
3	Expand the tree of the I/O Modules (DS-401) profile, select "Advantys OTB (V1.0)" and click on to launch configuration and addition of a new element: 
4	In the "New Island" window", enter "OTB_CAN_02" matching the name of the equipment that will be used in TwidoSoft. Next click on the "OK" button to confirm and launch the Advantys Configuration Tool application: 

Step	Action
5	Double click on the "OTB1C0DM9LP" I/O module to add it in the new island: Do the same with the "TWDDDO8TT" and "TWDDRA8RT" expansion modules to add them to the new island: 
6	It is possible to double click on the image of each module to set the parameters (Mask, polarity, management of fallback modes, etc.). Perform a "File / Save". Perform a "File / Quit". The new island containing the Advantys OTB I/O module and its expansion modules is then added to the CANopen catalog under the name "OTB_CAN_02".

Step	Action
7	Select "OTB_CAN_02 (V1.0)" in the "Catalog" area and click on to add the Advantys OTB in the "Network" area:  Note: The Advantys OTB is configured by default to the first free network address, i.e., the network address 2 in our example.

Step	Action
8	Select the "Linking" tab. Use to assign the new Advantys OTB I/O module and its two expansion modules to the PDO no. 2, in both "Receive" and "Send" modes: 
9	Select the "Symbol" tab. Enter, if necessary a "symbol" for the objects associated with the Advantys OTB I/O module: ● Double click in the object's "Symbol" field, ● Type the name of the "Symbol", ● Confirm using the "ENTER" key of the keyboard.
10	Click on the "OK" button to confirm. The Advantys OTB I/O module and its expansion modules are then declared in the Twido controller application.

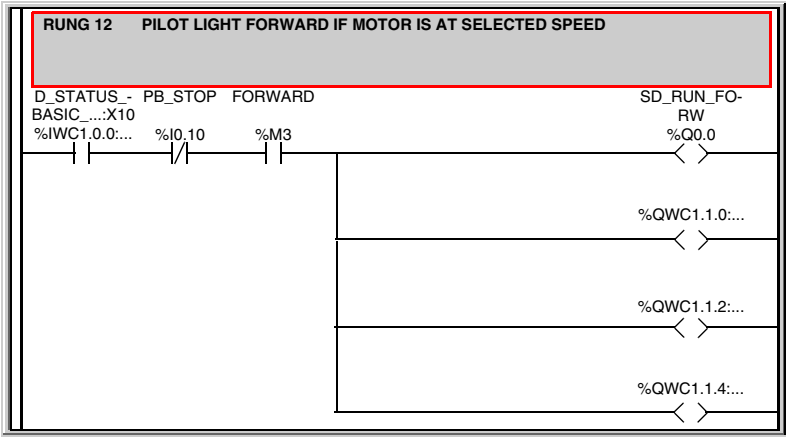
Non-reversible

The goal is to modify the application to manage the outputs of the Advantys OTB I/O module and of its expansion module the same as those of the Twido controller.

The method is identical to that described for the addition of an Advantys OTB by integrating the new outputs of the new island:

- %QWC1.1.0:X0 to X7 for the Advantys OTB "OTB1C0DEM9LP" module,
- %QWC1.1.2:X0 to X7 for the "TWDDO08TT" expansion module,
- %QWC1.1.4:X0 to X7 for the "TWDRA08RT" expansion module.

Here, for example, the line of the modified application managing the "FOWARD START" indicator LEDs:



Perform a "File / Save".

Load the new application into the Twido controller: Perform a "Controller/Connect" and click on the "PC -> Automate" button.

Add an Advantys FTB I/O splitter

V

At a Glance

Overview

This section of the document presents a possible evolution of the main system, that is the addition of an Advantys FTB I/O splitter.

Note: If you are not using a speed controller or Advantys OTB I/O module in your application, erase the associated elements in the CANopen configuration as well as in the Twido controller application program.

Application files

Files for this configuration may be found in CD-ROM BUNDLE (DIA3CD3050101F) in the "Applicative files\Partie_V" folder for the Twido controller application.

What's in this Part?

This part contains the following chapters:

Chapter	Chapter Name	Page
10	Hardware implementation of an Advantys FTB splitter	85
11	Software implementation of an Advantys FTB splitter	89

Hardware implementation of an Advantys FTB splitter

10

At a Glance

Overview

This chapter describes hardware implementation of an Advantys FTB I/O splitter added to the main system.

What's in this Chapter?

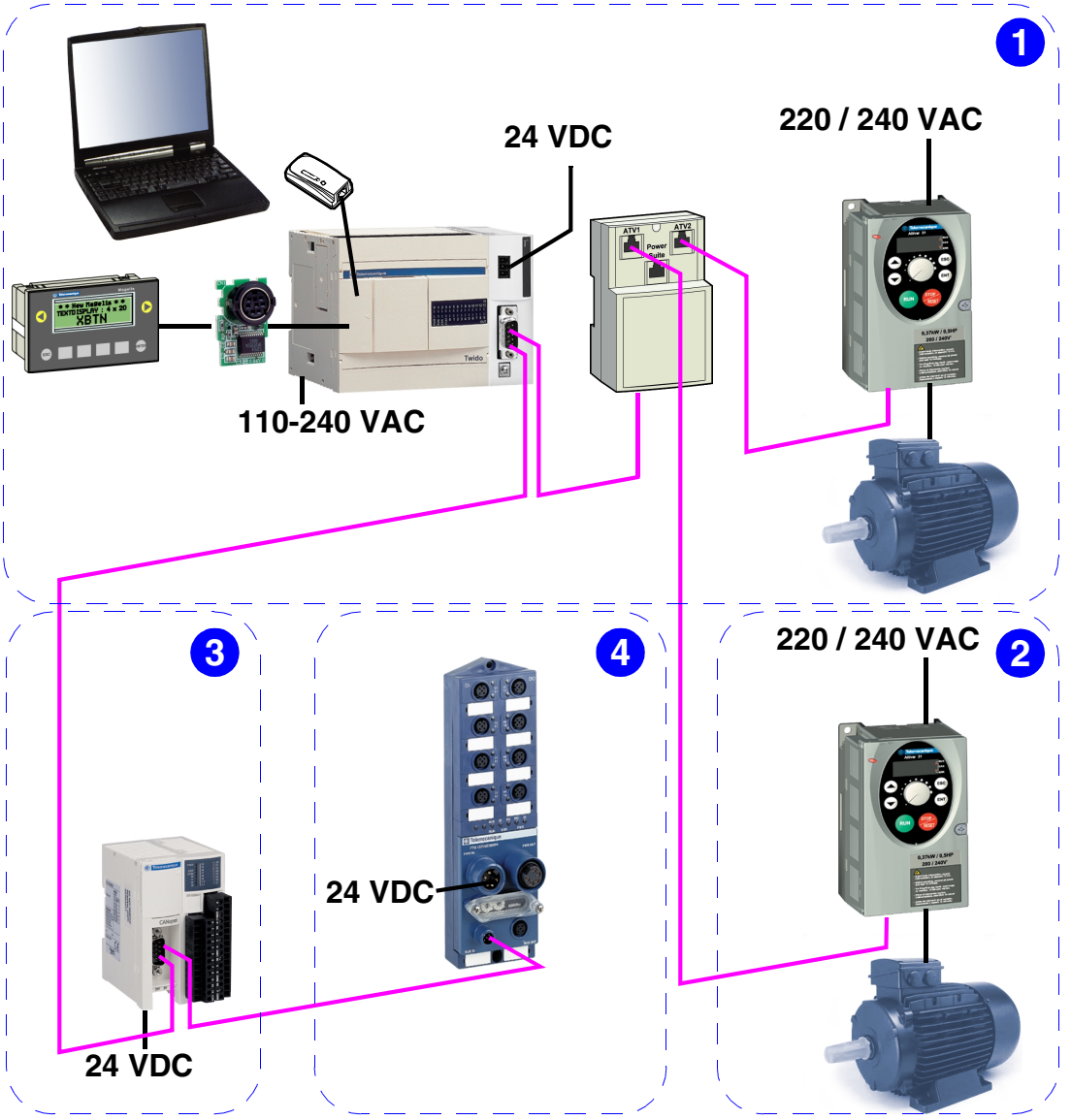
This chapter contains the following topics:

Topic	Page
Cabling of the Advantys FTB I/O splitter	86
Cabling of the CANopen network	87
Power supply	88

Cabling of the Advantys FTB I/O splitter

Installation diagram

This part of the document covers the following mounting:



Cabling of the CANopen network

Principle

The objective is to connect the Advantys FTB to the Advantys OTB using a cable.

Select a cable

To connect the Advantys FTB to the Advantys FTB, select one of the following cables:

Reference	Cable length (m)
FTXCN3130	3
FTXCN3150	5

Advantys FTB side cabling

The M12 end of the cable is connected to the Advantys FTB on the "BUS IN" base.

Advantys OTB side cabling

The "bare wire" end of cable is connected to the SUB-D 9 connector joined to the Advantys OTB (see *TAP-Twido Cable Preparation, p. 26*).

Cable pin assignment in the terminal of the SUB-D 9 connector:

No.	Signal	Cable	Wire color
6	CAN_L	FTB/OTB	Blue
7	CAN_H	FTB/OTB	White
8	SHLD (CAN-GND)	FTB/OTB	(Bare cable shielding)
9	V+	FTB/OTB	Red
10	GND	FTB/OTB	Black

Note: Toggle the line end terminator of the SUB-D 9 connector on the Advantys OTB to "OFF".
Connect the FTXCNTL12 line end terminator to the "BUS OUT" base of the Advantys FTB.

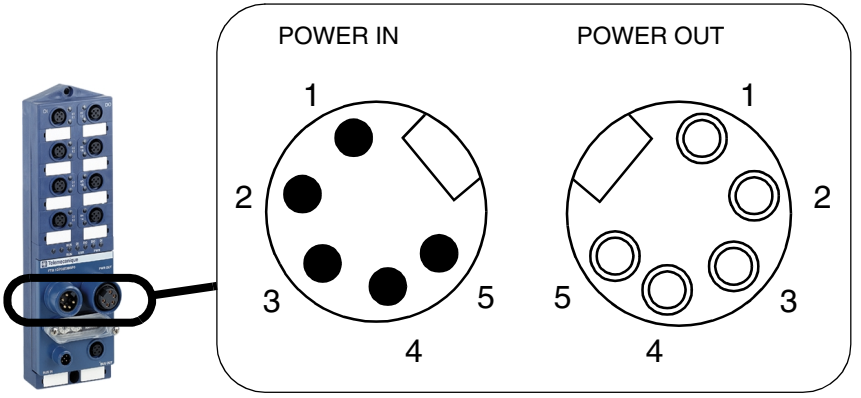
Power supply

Power on the Advantys FTB

Power on the Advantys FTB I/O splitter using one of the following cables:

Reference	Cable length (m)
FTXDP2115	1,5
FTXDP2130	3
FTXDP2150	5

Wiring Diagram:



Pin Assignment:

PIN No.	Description	Wire color	Printing on wire
1	0 V	Black	1
2	0 V	Black	2
3	PE	Green/Yellow	-
4	+24 V	Black	3
5	+24 V	Black	4

Software implementation of an Advantys FTB splitter

11

At a Glance

Overview

This chapter describes software implementation of an Advantys FTB I/O splitter added to the main system.

What's in this Chapter?

This chapter contains the following topics:

Topic	Page
Configuration of Advantys FTB communications	90
Add an Advantys FTB I/O splitter to the Twido controller application.	91

Configuration of Advantys FTB communications

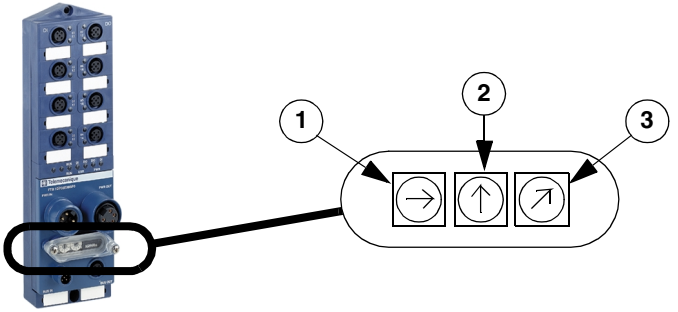
Principle

The system products must be configured according to the Twido application controller as follows:

- Address 1: ATV31
- Address 2: Advantys OTB
- Address 3: Advantys FTB
- Address 10: second ATV31
- Transmission Speed: 125 kBits/s

Description

Front panel of the Advantys FTB I/O splitter:



No.	Function
1	Transmission speed encoder wheel
2	Network address (Node-ID x10) encoder wheel
3	Network address (Node-ID x1) encoder wheel

Method

Configuration of Advantys FTB communication parameters:

Step	Action
1	Cut off all power to the element.
2	Set encoder wheel 1 to the position corresponding to the required baud rate. In the example: encoder wheel 1 is set to position 5 (125 kBits/s)
3	Set encoder wheels 2 and 3 to the position corresponding to the desired network address. In the example, the address is 3 (10x0 + 1x3): • encoder wheel 2: position 0 (tens) • encoder wheel 3: position 3 (units)
4	Reapply power to the Advantys FTB to implement the new configuration.

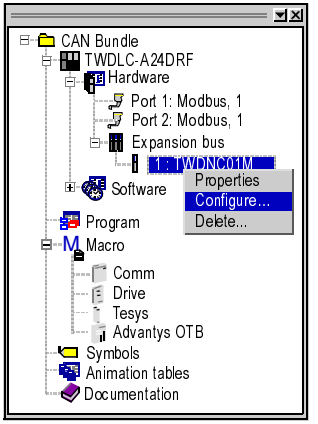
Add an Advantys FTB I/O splitter to the Twido controller application.

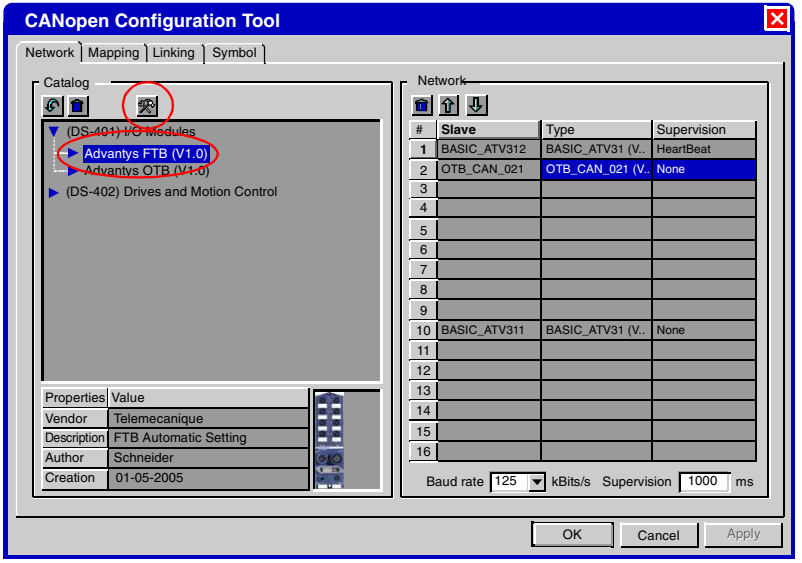
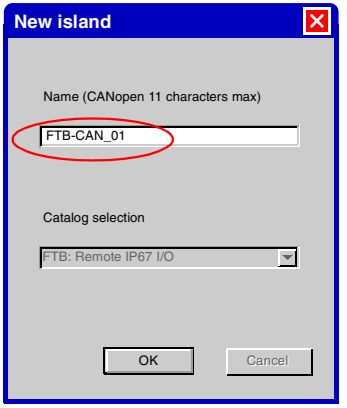
Principle

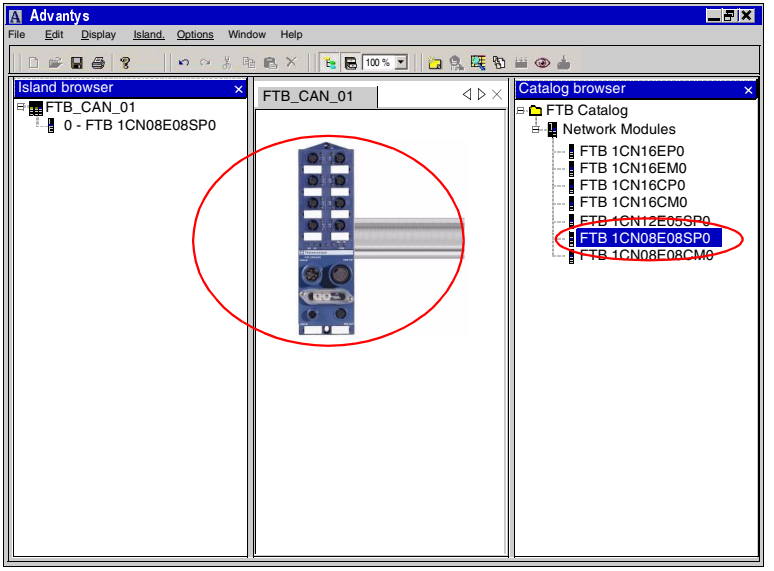
The objective is to modify the existing application in order to be able to manage the I/O of the Advantys FTB splitter.

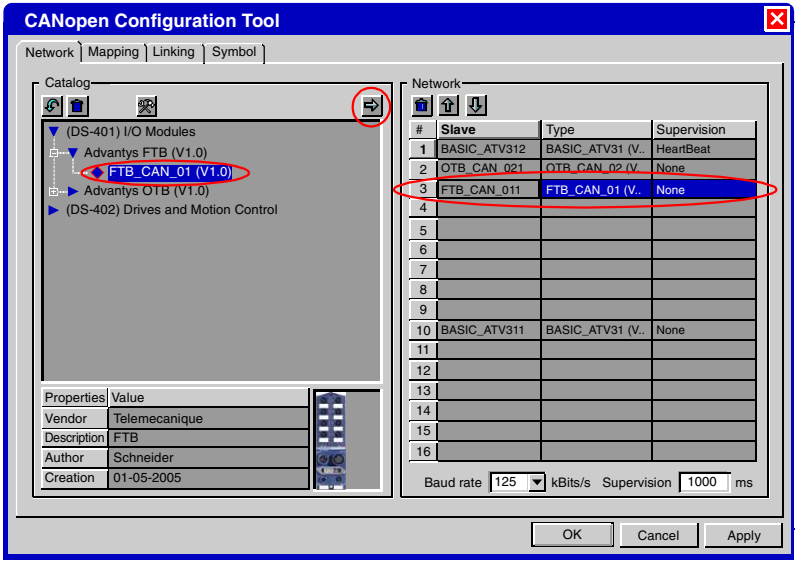
Declare the Advantys FTB splitter

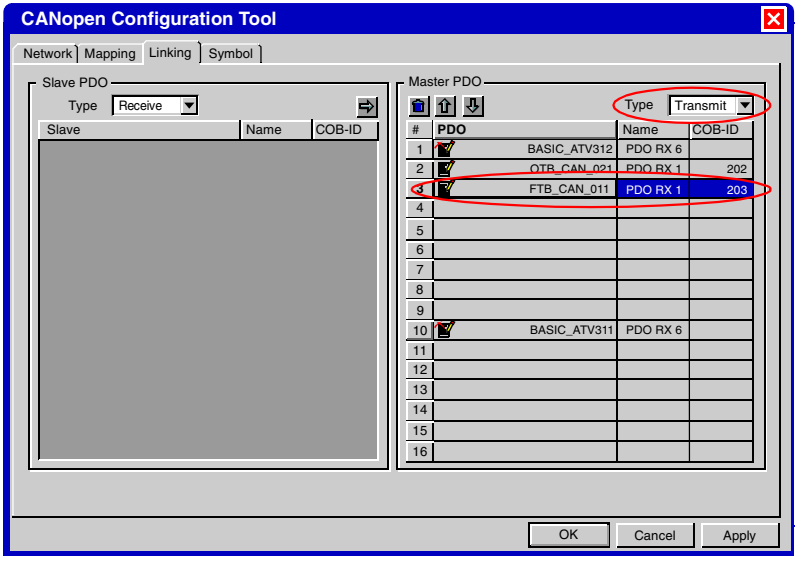
The following method allows declaration of the Advantys FTB I/O splitter in the Twido controller application.

Step	Action
1	Open the existing application using the TwidoSoft software (TwidoSoft must be in disconnected mode). Right click on the expansion bus and select "Configure". 

Step	Action
2	Expand the tree of the I/O Modules (DS-401) profile, select "Advantys FTB (V1.0)" and click on to launch configuration and addition of a new element: 
3	In the "New Island" window, enter "FTB_CAN_01" matching the name of the equipment that will be used in TwidoSoft. Next click on the "OK" button to confirm and launch the Advantys Configuration Tool application: 

Step	Action
4	Double click on the Advantys FTB I/O "FTB1CN08E08SP0" splitter to add it in the new island: 
5	It is possible to double click on the image of the splitter to set the parameters (Function, Mask, polarity, management of fallback modes, etc.). Perform a "File / Save". Perform a "File / Quit". The new island containing the Advantys FTB I/O splitter is then added to the CANopen catalog under the name "FTB_CAN_01".

Step	Action
6	Select "FTB_CAN_01 (V1.0)" in the "Catalog" area and click on to add the Advantys FTB splitter in the "Network" area.  Note: The Advantys FTB is configured by default to the first free network address, i.e., the network address 3 in our example.

Step	Action
7	Select the "Linking" tab. Use to assign the new Advantys FTB splitter to the PDO no. 3, in both "Receive" and "Send": 
8	Select the "Symbol" tab. Enter, if necessary, a "symbol" for the objects associated with the Advantys FTB I/O splitter: ● Double click in the object's "Symbol" field, ● Type the name of the "Symbol", ● Confirm using the "ENTER" key of the keyboard.
9	Click on the "OK" button to confirm. The Advantys FTB I/O splitter is then declared in the Twido controller application.

Non-reversible

The goal is to change the application to control the I/O of the Advantys FTB splitter the same as those of the Twido controller.

The method is identical to the one described for adding an Advantys OTB by integrating to the application the new I/O of the Advantys FTB splitter:

- %IWC1.2.0:X0 to X7 for inputs,
- %QWC1.2.0:X0 to X7 for the outputs.

Perform a "File / Save".

Load the new application into the Twido controller: Perform a "Controller/Connect" and click on the "PC -> Automate" button.

Appendices

At a Glance

List of symbols

The following pages contain the list of symbols used in the application described in part V of this document.

What's in this Appendix?

The appendix contains the following chapters:

Chapter	Chapter Name	Page
A	List of symbols used in the application	99

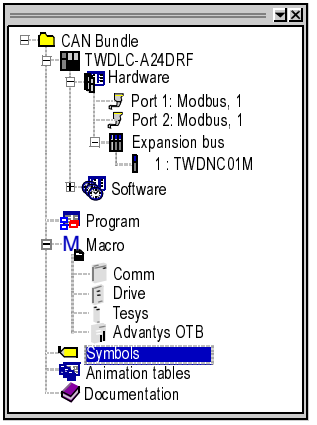
List of symbols used in the application

A

List of symbols used in the application

List of symbols

In the Twidosoft software, double click on "Symbols" to display the list of symbols used in the application.



Here, for example, is the list of symbols used in the application described in part V of this document:

Symbol	Number	Comment
PB_START_FORWARD	%I0.8	
PB_START_REVERSE	%I0.9	
PB_STOP	%I0.10	
PB_SLOW_FAST	%I0.11	
D_STATUS_BASIC_ATV312	%IWC1.0.0	
D_CONTROL_BASIC_ATV312	%IWC1.0.1	
D_IERROR_BASIC_ATV312	%IWC1.0.2	
OTB_CAN_INPUTS_1	%IWC1.1.0	
OTB_CAN_INPUTS_2	%IWC1.1.1	
FTB_CAN_INPUTS_1	%IWC1.2.0	
D_STATUS_BASIC_ATV313	%IWC1.9.0	
D_CONTROL_BASIC_ATV313	%IWC1.9.1	
D_IERROR_BASIC_ATV313	%IWC1.9.2	
FORWARD	%M3	
REVERSE	%M4	
PB_12_RF	%M12	
XBT_INIT_CAN_EDGE	%M20	test if the XBT function key was pressed to INIT CAN
XBT_OPER_CAN_EDGE	%M21	test if the XBT function key was pressed to INIT CAN
D_STATE_0	%MW0	Altivar state
D_CANSTATE_0	%MW1	Altivar CANOpen state
D_ERROR_0	%MW2	Altivar error code
D_SETPOINT_MODE_0	%MW3	Altivar set-point mode
D_SETPOINT_0	%MW4	Altivar set-point
D_STATE_1	%MW31	Altivar state
D_CANSTATE_1	%MW32	Altivar CANOpen state
D_ERROR_1	%MW33	Altivar error code
D_SETPOINT_MODE_1	%MW34	Altivar set-point mode
D_SETPOINT_1	%MW35	Altivar set-point
CAN_COM_PARAM_1	%MW80	Communication parameters used to go back in operational mode after a slave pb
CAN_COM_PARAM_2	%MW81	
SD_SPEED	%MW104	Speed entered with the XBT or selected by OTB input %IWC1.1.1:X3

Symbol	Number	Comment
XBT_FTB_OUTPUTS	%MW105	used to display information on the XBT
XBT_FTB_INPUTS	%MW106	used to display information on the XBT
XBT_OTB_OUTPUTS	ESPAÑOL	used to display information on the XBT
XBT_OTB_INPUTS	%MW108	used to display information on the XBT
XBT_DIALOG_TABLE_ALARM	%MW110	Used to manage Alarm display on the XBTN400
CAN_OPEN_SLAVE_STATUS_2_1	%MW200	
CAN_OPEN_SLAVE_STATUS_4_3	%MW201	
CAN_OPEN_SLAVE_STATUS_1	%MW202	
CAN_OPEN_SLAVE_STATUS_2	%MW203	
CAN_OPEN_SLAVE_STATUS_3	%MW204	
SD_RUN_FORW	%Q0.0	
SD_RUN_REV	%Q0.1	
SD_STOPPED	%Q0.2	
D_COMMAND_BASIC_ATV312	%QWC1.0.0	
D_TARGET_BASIC_ATV312	%QWC1.0.1	
OTB_CAN_OUTPUTS_1	%QWC1.1.0	
OTB_CAN_OUTPUTS_2	%QWC1.1.1	
OTB_CAN_OUTPUTS_3	%QWC1.1.2	
OTB_CAN_OUTPUTS_4	%QWC1.1.3	
OTB_CAN_OUTPUTS_5	%QWC1.1.4	
OTB_CAN_OUTPUTS_6	%QWC1.1.5	
FTB_CAN_OUTPUTS_1	%QWC1.2.0	
D_COMMAND_BASIC_ATV313	%QWC1.9.0	
D_TARGET_BASIC_ATV313	%QWC1.9.1	