Advantys OTB CANopen

Remote Inputs and Outputs

User Manual

2.0

12/2008

1606384 02

www.schneider-electric.com

Schneider Electric

Table of Contents

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 or serious injury.

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 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.

About the Book

At a Glance

Document Scope

This user manual contains the information for installing an Advantys OTB CANopen network interface module.

It has been designed to facilitate rapid familiarization with the system, while optimizing the system's features for the most advanced technology available.

Anyone installing this equipment must be familiar with the relevant communication protocol, and installation should only be performed by qualified personnel. Special points and warnings regarding safety are highlighted in the various chapters.

The early chapters provide information for designers and installers on installing the mechanical and electrical elements of the system.

The following chapters, from the section on "CANopen network interface", are specific to the communication protocol. They contain information on specific wiring for the network interface and all the necessary information for the software application programmer, and for the end user (diagnostics).

Chapter	Subject Dealt With
Introduction	General introduction to the network interface module and expansion modules
Description	Hardware installation, dimensions, installation and assembly of an island
Description, characteristics, and wiring of the Advantys OTB module	Description, electrical and mechanical characteristics and wiring diagrams for the OTB module
CANopen network interface	Introduction to the OTB module network interface Reminders on the communication protocol Managing the island's behavior on the network
Application-specific functions	Description of application-specific functions Remote I/O and dedicated function object dictionary
Software tools	Software installation help
Advantys OTB island diagnostics	Description of hardware diagnostics Description of software diagnostics How to perform diagnostics in the event of a failure
CANopen object dictionary	Description of the registers accessible for communication
Appendices	Presentation Appendix A: List of IEC symbols Appendix B: List of COB-IDs
Glossary	Acronyms Definitions

Record of Revisions

Version	List of Revisions
2.0	- Addition of TWD AMM6HT and TWD AMI2LT modules - M340 compatibility - Downloadable embedded software

Related Documents

Title of Documentation	Reference Number
Instruction sheet	1724121
Hardware guide for TM2 discrete I/O expansion modules	EIO0000000028
Hardware guide for TM2 analog I/O expansion modules	EIO0000000034
CANopen hardware installation manual	35010859
Updating embedded OTB software	35015001
FAQ: If you still have questions after reading this guide, visit the FAQ section on http://www.schneider-electric.com.	-

You can download these technical publications and other technical information from our website at www.schneider-electric.com.

User Comments

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

Introduction

1

Introduction

This chapter provides an overview of the Advantys OTB network interface modules, the different expansion modules, the maximum configuration and the specific functions of the module, as well as a communication architecture.

What's in this Chapter?

This chapter contains the following topics:

About Advantys OTB

Introduction

The Advantys OTB (Optimized Terminal Block) network interface module with built-in Inputs/Outputs is very compact. Its modularity, by adding I/O expansions, can optimize an application by providing the necessary number of I/O. The Advantys OTB module connects directly to a fieldbus or communication network.

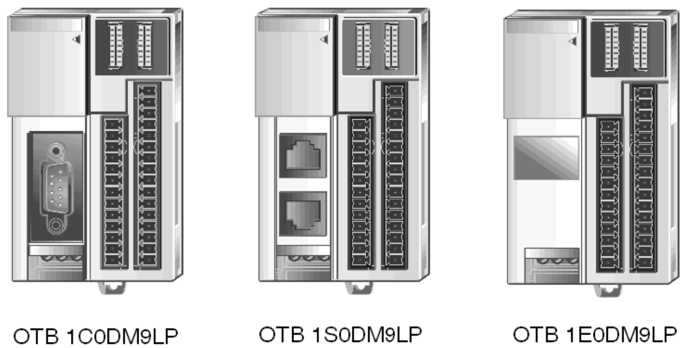
The available fieldbuses or networks are as follows:

Fieldbus or Network	OTB Module Reference
CANopen	OTB 1C0DM9LP
Modbus	OTB 1S0DM9LP
Ethernet	OTB 1E0DM9LP

The network interface module accepts up to 7 I/O expansion modules.

Network Interface Module with Built-In I/O

The illustration below shows the different network interface modules with built-in I/O:



Each network interface module with built-in I/O has:

- 12 discrete inputs
- 6 relay outputs
- 2 transistor outputs with positive logic (source)
- A 24 VDC power supply terminal block (Volt Direct Current)
- Communication bus connectors
- Indicator LEDs to display the communication status and I/O status

The table below lists the main characteristics of the network interface module:

Reference	Channel	Type of Channel	Input/Output type	Power Supply
OTB 1•0DM9LP	12	Inputs	24 VDC	24 VDC
	6 2	Outputs Outputs	Relay 24 VDC source transistor	

Discrete I/O Expansion Modules

The table below lists the discrete and relay I/O expansion modules:

Module Name	Reference	Channel	Type of Channel	Input/Output Type	Terminal Type
Input Modules					
8-channel input	TM2 DDI8DT	8	Inputs	24 VDC	Removable screw terminal block
8-channel input	TM2 DAI8DT	8	Inputs	120 VAC(1)	Removable screw terminal block
16-channel input	TM2 DDI16DT	16	Inputs	24 VDC	Removable screw terminal block
16-channel input	TM2 DDI16DK	16	Inputs	24 VDC	HE10 connector
32-channel input	TM2 DDI32DK	32	Inputs	24 VDC	HE10 connector
Output Modules					
8-channel output	TM2 DDO8TT	8	Outputs	Source transistor	Removable screw terminal block
8-channel output	TM2 DDO8UT	8	Outputs	Sink transistor	Removable screw terminal block
8-channel output	TM2 DRA8RT	8	Outputs	Relay	Removable screw terminal block
16-channel output	TM2 DDO16TK	16	Outputs	Source transistor	HE10 connector
16-channel output	TM2 DDO16UK	16	Outputs	Sink transistor	HE10 connector
16-channel output	TM2 DRA16RT	16	Outputs	Relay	Removable screw terminal block
32-channel output	TM2 DDO32TK	32	Outputs	Source transistor	HE10 connector
32-channel output	TM2 DDO32UK	32	Outputs	Sink transistor	HE10 connector
Mixed modules					
4-channel input/4-channel output	TM2 DMM8DRT	4	Inputs	24 VDC	Removable screw terminal block
		4	Outputs	Relay	
16-channel input/8-channel output	TM2 DMM24DRF	16	Inputs	24 VDC	Fixed spring terminal block
		8	Outputs	Relay	

(1) VAC : Volt Alternating Current

Analog I/O Expansion Modules

The table below lists the analog I/O expansion modules:

Module Name	Reference	Channel	Type of Channel	Details	Terminal Type
Input Modules					
2 inputs	TM2 AMI2HT	2	Inputs	12 bits 0-10 V 4-20 mA	Removable screw terminal block
2 inputs	TM2 AMI2LT	2	Inputs	12 bits thermocouple type J, K or T	Removable screw terminal block
4 inputs	TM2 AMI4LT	4	Inputs	12 bits Voltage/current RTD	Removable screw terminal block
8 inputs	TM2 AMI8HT	8	Inputs	10 bits Voltage/current	Removable screw terminal block
8 inputs	TM2 ARI8HT	8	Inputs	10 bits PTC NTC	Removable screw terminal block
8 inputs	TM2 ARI8LT	8	Inputs	12 bits PT100 PT1000	Removable screw terminal block
8 inputs	TM2 ARI8LRJ	8	Inputs	12 bits PT100 PT1000	8 x RJ11 connectors
Output Modules					
1 output	TM2 AMO1HT	1	Outputs	12 bits 0-10 V 4-20 mA	Removable screw terminal block
2 outputs	TM2 AVO2HT	2	Outputs	10 bits +/-10 V	Removable screw terminal block
Mixed Modules					
2 inputs/1 output	TM2 AMM3HT	2	Inputs	12 bits 0-10 V 4-20 mA	Removable screw terminal block
		1	Outputs		
2 inputs/1 output	TM2 ALM3LT	2	Inputs	12 bits RTD, thermocouple	Removable screw terminal block
		1	Outputs	12 bits 0-10 V 4-20 mA	
4 inputs/2 outputs	TM2 AMM6HT	4	Inputs	12 bits 0-10 V 4-20 mA	Removable screw terminal block
		2	Outputs		

Common Expansion Module

Module Name	Reference	Channel	Type of Channel	Details	Terminal Type
Common module	OTB 9ZZ61JP	16	Passive	2 x 8 contacts	Removable screw terminal block

Cables

Cable Name	Reference
Cable fitted with a 20-pin HE connector at both ends. (*AWG 28*/0.08 mm; length: 0.5 m/*1.64 ft*)	ABF T20E050
Cable fitted with a 20-pin HE connector at both ends. (*AWG 28*/0.08 mm; length: 1 m/*3.28 ft*)	ABF T20E100
Cable fitted with a 20-pin HE connector at both ends. (*AWG 28*/0.08 mm; length: 2 m/*6.56 ft*)	ABF T20E200

Maximum hardware configuration

Introduction

This section presents the maximum hardware configuration for the Advantys OTB network interface module.

The OTB module functions can be extended through the use of expansion modules, subject to the following limitations:
- The OTB module accepts up to 7 discrete I/O expansion modules
- The total consumption of the expansion modules must be less than 450 mA

⚠ WARNING

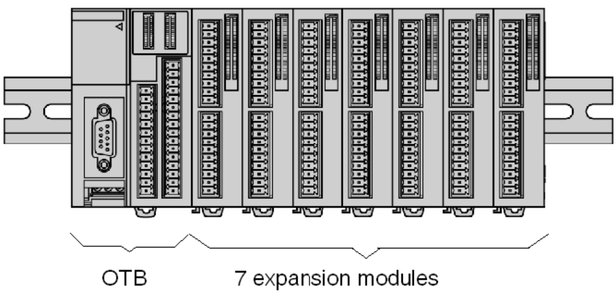
Risk of unintended equipment operation

Do not exceed the 450 mA current limit for the OTB module and its expansion modules. Exceeding this limit can cause the I/O power supplies to stop. The device can operate in an unintended manner depending on how the inputs and outputs are configured.

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

Each OTB module (whether or not associated with expansion modules) constitutes an island on the bus or the communication network. Each island offers a modular and flexible I/O solution.

The figure below is an example of an I/O island.



Maximum Number of Discrete I/O

The table below lists the maximum number of discrete I/O for the island:

Type of Built-in I/O	Maximum Number of Discrete I/O
Standard discrete inputs	12
Standard discrete outputs	8

Type of I/O with Expansion Modules	Maximum Number of Discrete I/O
Maximum discrete inputs (I/O module + exp I/O)	12+(7x32)=236
Maximum discrete outputs (I/O module + exp I/O)	8+(7x32)=232
Maximum discrete I/O (I/O module + exp I/O)	20+(7x32)=244
Maximum relay outputs	6 on base + 96 on expansion

Maximum Number of Analog I/O

The following table lists the maximum number of analog I/O by type for the island:

Type of Analog I/O	Maximum Number of Analog I/O
Analog inputs	24
Analog outputs	14

NOTE: No configuration should have more than 32 analog I/O.

Expansion Module Power Consumption

The total power consumption of the expansion modules must be less than 450 mA (See the above safety message). The table below lists the power consumption of each expansion module:

Expansion Module	Consumption
TM2 DDI8DT	25 mA
TM2 DAI8DT	60 mA
TM2 DDI16DT	40 mA
TM2 DDI16DT	35 mA
TM2 DDI32DK	65 mA
TM2 DDO8TT	10 mA
TM2 DDO8UT	10 mA
TM2 DRA8RT	30 mA
TM2 DDO16TK	10 mA
TM2 DDDO16UK	10 mA
TM2 DRA16RT	45 mA
TM2 DDO32TK	20 mA
TM2 DDO32UK	20 mA
TM2 DMM8DRT	25 mA
TM2 DMM24DRF	65 mA
TM2 AMI2HT	50 mA
TM2 AMO1HT	50 mA
TM2 AMM3HT	50 mA
TM2 AMM6HT	50 mA
TM2 ALM3LT	50 mA
TM2 AVO2HT	50 mA
TM2 AMI2LT	60 mA
TM2 AMI4LT	50 mA
TM2 AMI8HT	50 mA
TM2 ARI8HT	50 mA
TM2 ARI8LT	90 mA
TM2 ARI8LRJ	90 mA

Specific Functions of the Network Interface Module

Introduction

By default, all I/Os of the network interface module are configured as Discrete I/Os. However, certain I/Os can be assigned to remote functions.

Specific Functions

The following table lists the specific functions of the network interface module:

Function	Description
Fast counter: RFC	2 fast up/down counters: 5 kHz (1-phase)
Very fast counter: RVFC	2 very fast counters: Up/down counters - 20 kHz (2-phase)
Pulse generator: RPLS or RPWM	2 RPLS or RPWM pulse generators (Pulse width modulation): - RPLS pulse generator output, 7 kHz maximum - RPWM pulse width modulation, 7 kHz maximum
Programmable input filter	Input filter time can be changed during configuration No filtering or filtering at 3 ms or 12 ms

For further information, see *Application-Specific Functions, page 111*.

Communication Overview

Introduction

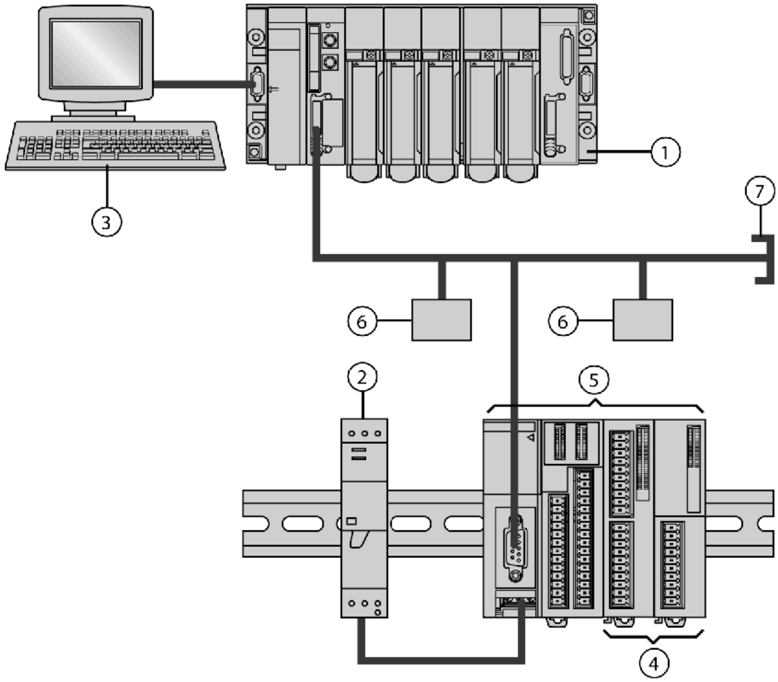
The Advantys OTB network interface modules are available for CANopen and Modbus field buses, and for the Ethernet network. They are used to exchange data from the built-in I/Os and expansion module I/Os with the bus master or client.

Field Bus or Network

A module with or without expansion constitutes an I/O island. The network interface module manages data transfers between the island and the master or client, via the field bus or network.

Communication Architecture

The following figure illustrates the different roles of the network interface module. This figure shows a network architecture with all the elements necessary for its implementation:



1 PLC with master and/or client
2 External 24 VDC electrical supply
3 PC with the PLC configuration software (API)
4 Expansion I/O modules
5 Advantys OTB island
6 Other islands or products on the field bus or network
7 Line terminator according to the field bus or network (if necessary)

Installation

2

Introduction

This chapter provides dimensions, installation, and mounting instructions for Advantys OTB network interface modules, and digital and analog expansion I/O modules.

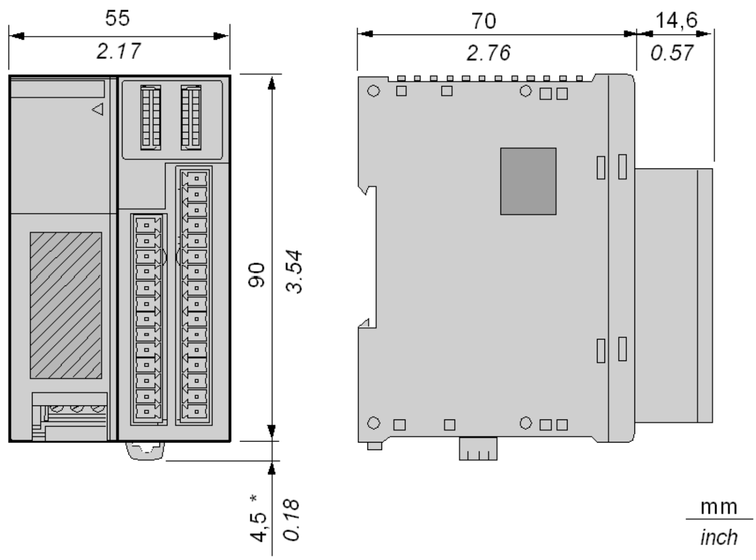
What's in this Chapter?

This chapter contains the following topics:

Dimensions of the Network Interface Modules

OTB 1•0DM9LP Dimensions

The following figure shows the dimensions of the Advantys OTB network interface module (OTB 1•0DM9LP):



NOTE: * 8.5 mm (*0.33 in*) when the clamp is pulled out.

How to Direct Mount on a Panel

Introduction

This section shows the positions of the mounting holes for each network interface module. Your module may differ from the illustrations in this procedure but the mechanism remains the same.

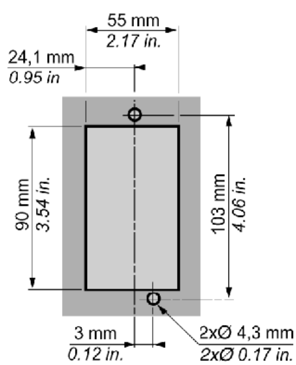
Installing a Mounting Strip

Mounting on a panel requires the use of a strip. The procedure below explains how to install a mounting strip (reference: TWD XMT5).

Step	Action
1	Remove the clamp at the rear of the module by pushing the clamp inward.
2	Insert the mounting strip, with the hook entering last, into the slot where the clamp was removed.
3	Slide the mounting strip into the slot until the hook enters the recess in the module.

Position of the Mounting Holes on the Network Interface Module

The diagram below shows the position of the mounting holes on the Advantys OTB network interface modules:



Installation Preparation

Introduction

The following section provides information on installing network interface modules and expansion I/O modules.

Before Starting

Before installing network interface modules, read the Safety Information at the beginning of this book.

⚠ WARNING

RISK OF UNINTENDED EQUIPMENT OPERATION

Do not add or remove an expansion module to or from the system before first removing all power. Adding or removing an expansion module while under power may cause damage to the module and the system, resulting in unexpected operation of inputs and outputs. Depending on I/O configuration, unintended equipment operation may occur.

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

NOTE: All options and expansion I/O modules should be installed in the network interface module before installing an island on a DIN rail, on a mounting plate, or in a cabinet. The island should be removed from a DIN rail, a mounting plate or a cabinet before disassembling the different modules.

Mounting Positions for the Network Interface Module and the Expansion Modules

Introduction

This section shows the correct and incorrect mounting positions for all network interface modules and expansion I/O modules.

⚠ WARNING

RISK OF UNEXPECTED EQUIPMENT OPERATION

Keep adequate spacing around the island for proper ventilation and to maintain an ambient temperature between 0 °C*(32°F)* and 55 °C *(131°F)*. Overheating of the OTB module and/or the I/O expansion modules may result in unexpected operation of inputs and outputs.

Depending on the I/O configuration, unintended equipment operation may occur.

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

⚠ WARNING

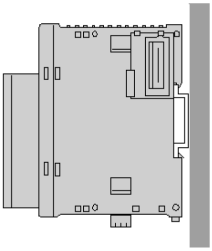
RISK OF UNEXPECTED EQUIPMENT OPERATION

Do not place heat generating devices such as transformers and supply blocks under the island. Heat generating devices in the proximity of the OTB module and I/O expansion modules could result in elevated temperatures and overheating, and may result in unexpected operation of the inputs and outputs. Depending on the I/O configuration, unexpected equipment operation may occur.

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

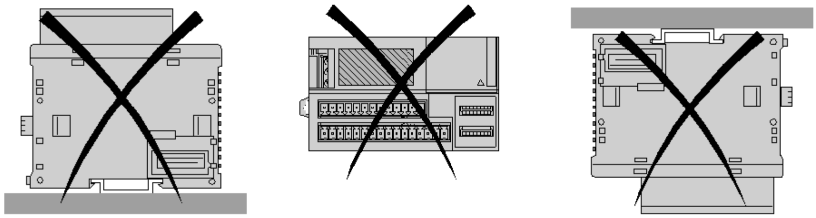
Correct Mounting Position

Network interface modules and expansion I/O modules must be mounted horizontally on a vertical plane as shown in the figures below.



Incorrect Mounting Position

The following diagrams show the incorrect mounting positions for the network interface modules and expansion modules.



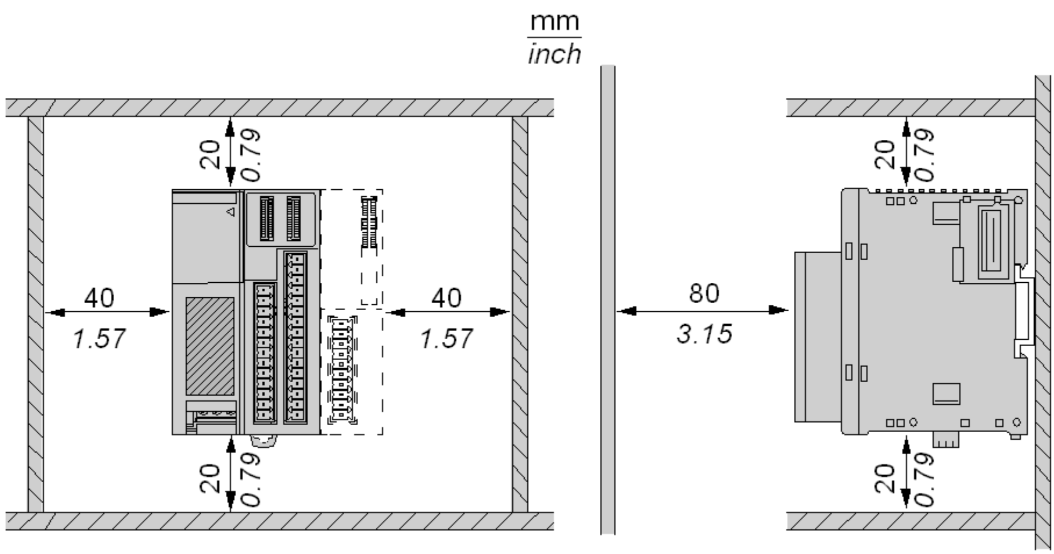
Assembly Precautions for an Island on a Panel or in a Cabinet

Introduction

This section presents the assembly precautions required for islands on a control panel or in a cabinet.

Required Assembly Space for an Island

In order for air to be able to circulate freely around the islands mounted in a control panel or in a cabinet, you must respect the minimum distances given in the following diagram.



Assembly of an Expansion Module to a Network Interface Module

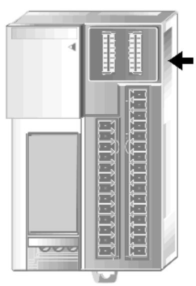
Introduction

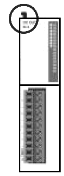
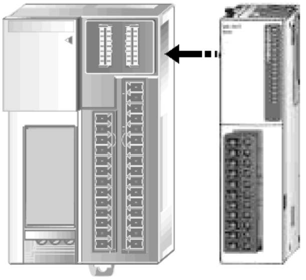
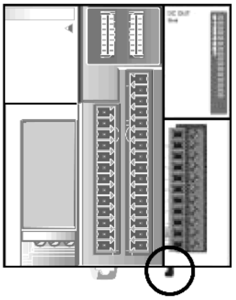
This section shows how to assemble an expansion module to a network interface module. Your network interface module or expansion module may differ to the ones shown in the illustrations for this procedure, but the mechanism remains the same.

⚠ WARNING
RISK OF UNEXPECTED EQUIPMENT OPERATION Do not change the hardware configuration without updating both the master device and the client application program. Do not reapply power after reconfiguring hardware until all updates to the master device and client application program have been accomplished and confirmed. Failure to make these required updates may cause unintended operation of the inputs and outputs. Depending on I/O configuration, unintended equipment operation may result. **Failure to follow these instructions can result in death, serious injury, or equipment damage.**

Assembly of an Expansion Module to a Network Interface Module

The following procedure must be performed with all products powered down. It shows how to assemble a network interface module to an expansion module.

Step	Action
1	Remove the protective label located on the side of the network interface module. 

Step	Action
2	Make sure the black latch button on the expansion module is in the up position. 
3	Align the connector on the left side of the expansion module to the connector on the right side of the network interface module. 
4	Press the expansion module to the network interface module until it "clicks" into place.
5	Push down the black latch button on the top of the expansion module to lock the modules together. 
6	Begin the operation again from step 1 for each expansion module to be added.

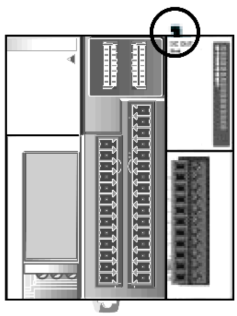
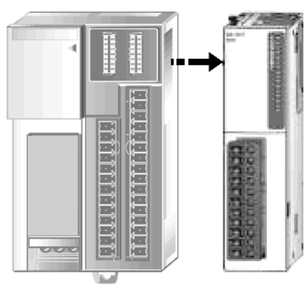
Disassembling an Expansion Module and a Network Interface Module

Introduction

This section shows how to disassemble an expansion interface from a network interface module. Your network interface module or expansion module may differ to the ones shown in the illustrations for this procedure, but the mechanism remains the same.

Disassembly of an Expansion Module from a Network Interface Module

The following procedure must be performed with all products powered down. It shows how to disassemble an expansion module from a network interface module.

Step	Action
1	Remove the island (network interface module + expansion module(s)) from the DIN rail before disassembling them. See *Introduction, page 43*.
2	Push the black latch from the bottom of the expansion module to disengage it from the network interface module. 
3	Separate the modules. 
4	Begin the operation again from step 2 for each expansion module to be separated.

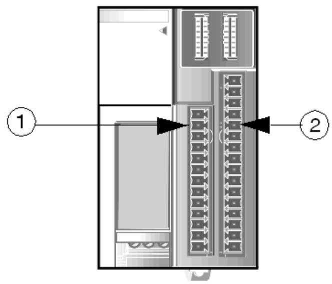
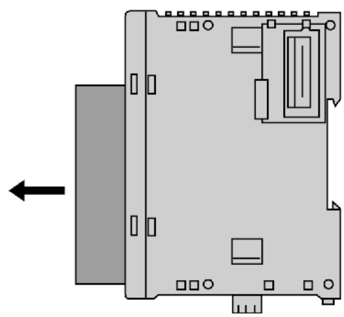
Removing a Terminal Block

Introduction

This section describes the removal of terminal blocks from Advantys OTB network interface modules.

Removing a Terminal Block

The following procedure describes how to remove terminal blocks from the network interface module.

Step	Action
1	Power down the network interface module and disconnect all wires. **Note:** The terminal block on the left (1) must be removed before the terminal block on the right (2). 
2	Remove the terminal block by holding the center of the terminal block and pulling it out straight. 

⚠ CAUTION

RISK OF EQUIPMENT DAMAGE

Do not attempt to remove the terminal by pulling on its top or bottom surfaces. Only pull on the terminal from the sides. Pulling from the top or bottom may cause the terminal to exit at an angle and damage the connector pins.

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

How to Install and Remove a Network Interface Module from a DIN Rail

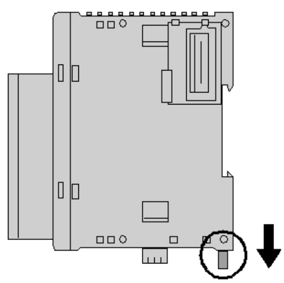
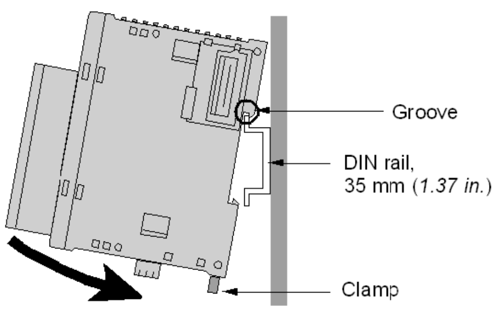
Introduction

This section shows how to install and remove an island from a DIN rail. Your island may differ from the illustrations in this procedure but the mechanism is the same.

NOTE: When mounting modules on a DIN rail, use two end stops, type AB1AB8P35 or equivalent.

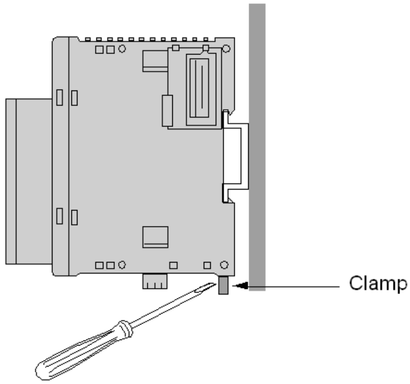
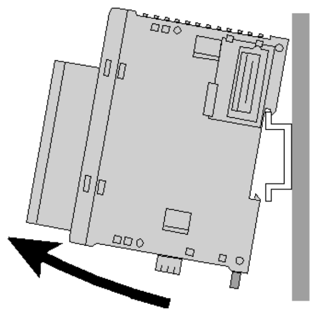
Installing an Island on a DIN Rail

The following procedure must be performed with all products powered down. It describes how to install an island on a DIN rail.

Step	Action
1	Fasten the DIN rail to a panel using screws.
2	Before any installation on a DIN rail, attach the communication module to the expansion modules. See *Assembly of an Expansion Module to a Network Interface Module, page 36*.
3	Pull out the clamp at the bottom of each module. 
4	Put the top groove of the island on the DIN rail and press it toward the rail. 
5	Push the clamp up to lock the island to the DIN rail.
6	Position the mounting stops of both sides of the modules to prevent the system from moving sideways.

Removing an island from a DIN Rail

The following procedure must be performed with all products powered down. It shows how to remove an island from the DIN rail.

Step	Action
1	Insert a flat screwdriver into the slot in the module clamp. 
2	Pull out the clamp.
3	Repeat steps 1 and 2 for each module comprising the island.
4	Pull the island to remove it from the DIN rail. 

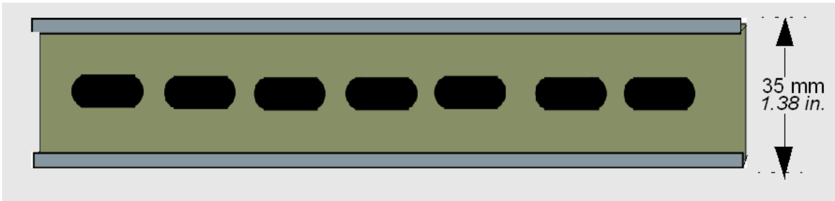
The DIN Rail

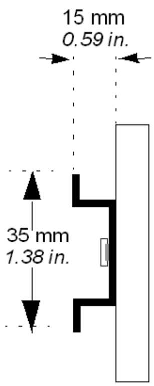
Introduction

The OTB modules are mounted on a DIN rail. A DIN rail can be attached to a smooth mounting surface or suspended from a EIA rack or in a NEMA cabinet.

Dimensions of the DIN Rail

The DIN rail measures 35 mm (*1.38 in.*) high and 15 mm (*0.59 in.*) deep, as illustrated below.





Recommended Equipment

You can order the appropriate DIN rail from Schneider Electric:

Rail Depth	Catalog Reference
15 mm (*0.59 in.*)	AM1 DE200

Description, characteristics, and wiring of the OTB module

3

Introduction

This chapter describes the wiring rules and recommendations, overviews, part references, characteristics and wiring diagrams for the Advantys OTB network interface module.

What's in this Chapter?

This chapter contains the following topics:

Wiring Rules and Recommendations

Introduction

There are several rules that must be followed when wiring a module or network interface. Recommendations, when needed, are provided on how to comply with the rules.

⚠ DANGER

ELECTRIC SHOCK

- Be sure to remove ALL power from ALL devices before connecting or disconnecting inputs or outputs to any terminal or installing or removing any hardware.
- Make sure you have COMPLETELY powered down ALL devices before connecting or disconnecting the bus or network.

Failure to follow these instructions will result in death or serious injury.

⚠ WARNING

RISK OF UNINTENDED EQUIPMENT OPERATION

If outputs should fail, outputs may remain on or off. Where personnel and or equipment hazards exist, use an appropriate hard-wired safety system.

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

Rules

- Each connector terminal accepts up to two wires fitted with cable ends or tags, with sections between 0.14 mm^2 and 1.5 mm^2, (*AWG26* and *AWG16*).
- Output module fusing is the responsibility of the user. It is not within the OTB network interface module itself. Select a fuse appropriate for the load with respect to the electrical codes.
- Depending on the load, a protection circuit may be needed for relay outputs on modules.
- The power supply wire should be between 0.14 mm^2 and 1.5 mm^2(*AWG26* and *AWG16*). Use the shortest wire length possible.
- The grounding wire should be 1.50 mm^2 (*AWG16*).
- Be sure to connect the grounding wire to a proper ground.
- Power supply wires routed inside the panel must be kept separate from I/O and communication wiring. Route wiring in separate cable ducting.

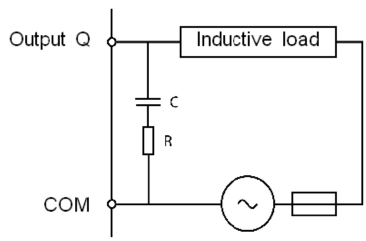
- Take care when wiring output modules that are designed to work as either source or sink. Incorrect wiring can cause equipment damage.
- Make sure that the operating conditions and environments are within the specification values.
- Use proper wire size to meet voltage and current requirements.
- Fit cable ends to the cables.

Contact Protection Circuit for Relay and Transistor Outputs

Depending on the load, a protection circuit may be needed for relay outputs. Choose a protection circuit, from the following diagrams, according to the power supply. Connect the protection circuit to the outside of the module for the relay outputs.

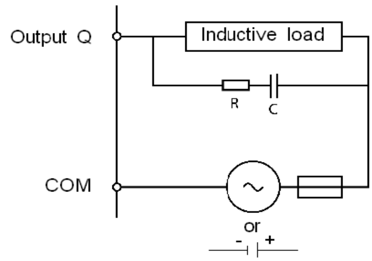
⚠ CAUTION
RISK OF EQUIPMENT DAMAGE
Apply a circuit protection to all outputs. Failure to add a circuit protection can result in the malfunction of the output(s) in the case of short circuit or overload condition.
Failure to follow these instructions can result in injury or equipment damage.

Protection Circuit A: This protection circuit can be used when the load impedance is smaller than the RC impedance in an AC load power circuit.



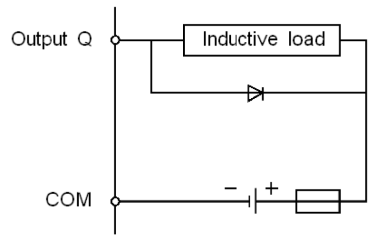
- C represents a value from 0.1 to 1 μF.
- R represents a resistor of approximately the same resistance value as the load.

Protection Circuit B: This protection circuit can be used for both AC and DC load power circuits.



- C represents a value from 0.1 to 1 μF.
- R represents a resistor of approximately the same resistance value as the load.

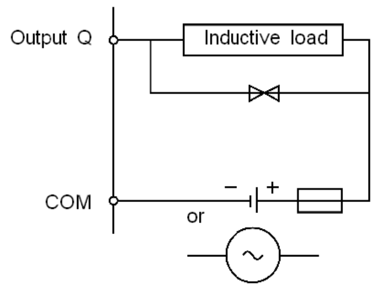
Protection Circuit C: this protection circuit can be used for DC load power circuits.



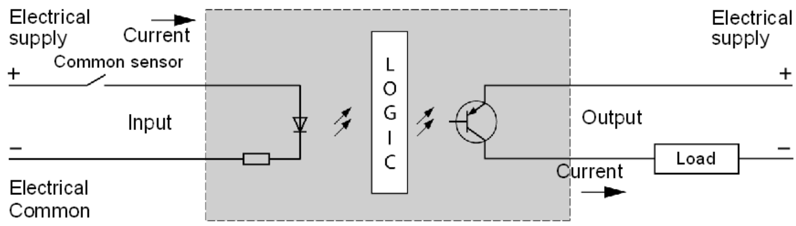
Use a diode with the following ratings:

- Reverse withstand voltage: power voltage of the load circuit x 10.
- Forward current: more than the load current.

Protection Circuit D: This protection circuit can be used for both AC and DC load power circuits.



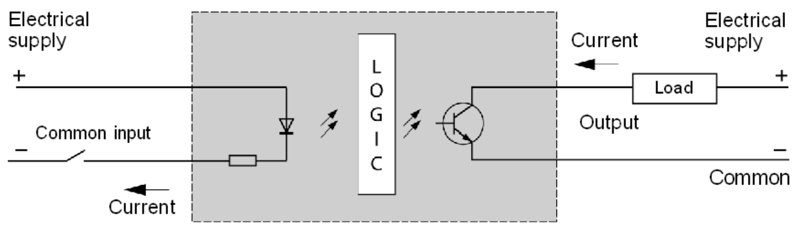
Operation of Source Inputs/Outputs



Source inputs/outputs

Input COM terminals are connected to the "-" terminal or common of the power supply. Output COM terminals are connected to the +24V power supply.

Operation of Sink Inputs/Outputs



Sink inputs/outputs

The input COM terminals are connected to the +24 V power supply. The output COM terminals are connected to the "-" terminal or common of the power supply

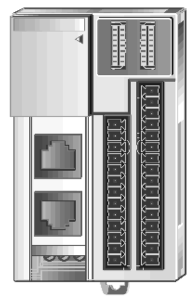
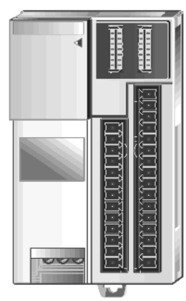
Overview of the Network Interface Modules

Introduction

This section describes the entire range of Advantys OTB network interface modules.

Illustrations

The following illustrations show the different network interface modules:

Module type	**Illustration**
Network interface module: • has 12 discrete inputs, 6 relay outputs, and 2 source transistor outputs • has terminal blocks for wiring • accepts up to 7 expansion I/O modules	 OTB1 C0DM9LP  OTB 1S0DM9LP  OTB 1E0DM9LP

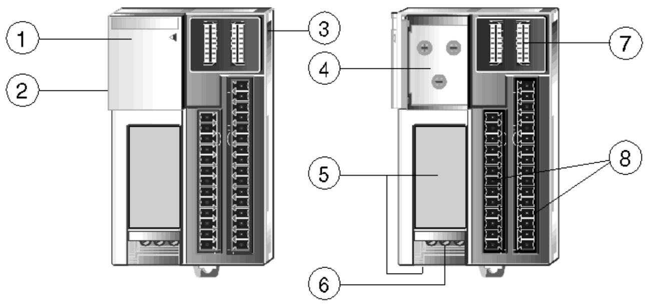
Physical Description of an Advantys OTB Network Interface Module

Introduction

This section describes the different sections of a network interface module. Only the communication section is dedicated to each field bus or network. This may differ from the illustrations, but the general description remains the same.

Physical Description of a Network Interface Module

The following illustration describes the different sections of a network interface module.



Legend

Label	Description
1	Hinged lid
2	OS update socket (RJ45 socket)
3	Expansion module connector
4	Encoder wheels (sets the island address and communication speed on the bus or network)
5	Communication bus connectors (the position depends on the reference)
6	24 VDC power supply terminals
7	Indicator light
8	I/O terminals

General Characteristics of the Network Interface Module

Introduction

This section describes the general characteristics common to the network interface modules.

WARNING
RISK OF EQUIPMENT DAMAGE OR UNEXPECTED EQUIPMENT OPERATION
The OTB modules were designed, manufactured and tested within specification limits as indicated in the following tables. Operating the product(s) outside of these limits may cause damage to the module and the system, resulting in unexpected operation on the inputs and outputs. Depending on the I/O configuration, unexpected equipment operation may result.
Failure to follow these instructions can result in death, serious injury, or equipment damage.

Normal Operating Specifications

Network interface module	**OTB 1•0DM9LP**
Operating temperature	0 ... 55 °C *(32 ... 131 °F)*
Storage temperature	-25 ... +70 °C *(-13 ... 158 °F)*
Relative humidity	30 ... 95% (non-condensing)
Degree of pollution	2 (IEC60664)
Degree of protection	IP20
Altitude	Operation: 0 ... 2 000 m *(0 ... 6561 ft)* Transport: 0 ... 3 000 m *(0 ... 9842 ft)*
Resistance to Vibration	When mounted on a DIN rail: from 10 to 57 Hz with amplitude of 0.075 mm *(0.0029 in.)*, from 57 to 150 Hz with acceleration of 9.8 m/s^2 (1G), 2 hours per axis on each of three mutually perpendicular axes. When mounted on a panel surface: from 2 to 25 Hz with amplitude of 1.6 mm *(0.062 in)*, from 25 to 100 Hz with acceleration of 39.2 m/s^2 (4G), 90 min per axis on each of three mutually perpendicular axes.
Impact strength	147 m/s^2 (15G), 11 ms duration, 3 shocks per axis, on three mutually perpendicular axes (IEC 61131).
Weight	185 g *(6.52 oz)*

Electrical Characteristics

Network interface module	**OTB 1•0DM9LP**
Rated power voltage	24 VDC
Allowable voltage range	20,4 ... 26.4 VDC (including ripple)
Consumed power	Communication module with 7 expansion modules
	19 W (26.4 VDC)
Allowable momentary power interruption	10 ms (24 VDC)
Dielectric strength	Between power and ground terminals: 500 VAC, 1 min Between I/O and ground terminals: 500 VAC, 1 min
Insulation resistance	Between power and ground terminals: 10 MΩ minimum (500 VDC) Between I/O and ground terminals: 10 MΩ minimum (500 VDC)
Noise resistance IEC 1131-2	DC power terminals: 1 kV, 50 ns to 1 µs I/O terminals (coupling clamp): 1,5 kV, 50 ns to 1 µs
Inrush current	50 A maximum (24 VDC)
Ground wiring	1 mm^2 *(AWG 18)*, 1.5 mm^2 *(AWG 16)*
Power supply wiring	0.14 mm^2 *(AWG 26)*, 1.5 mm^2 *(AWG 16)*
Tightening torque of the 24 VDC supply terminals	0.8 Nm *(7.08 lb-in)*
Tightening torque of the I/O terminals	0.6 Nm *(5.31 lb-in)*

Network Interface Module I/O Characteristics

Introduction

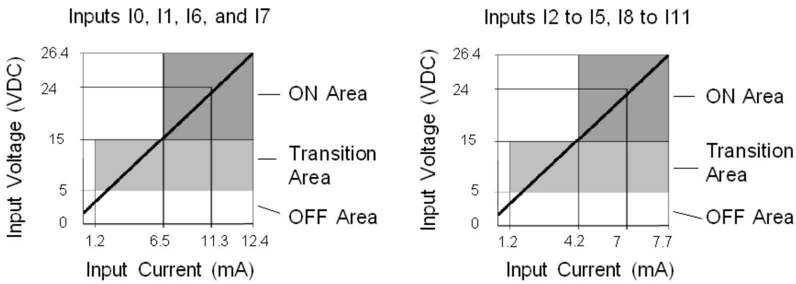
This section describes the characteristics of the Advantys OTB network interface module I/O.

Input Characteristics

Network Interface Module	**OTB 1•0DM9LP**
Number of inputs	12 inputs with common
Nominal input voltage	24 VDC source/sink input signal
Input voltage range	20.4 ... 26.4 VDC
Nominal input current	I0, I1, I6, I7: 5 mA/input (24 VDC) I2 to I5, I8 to I11: 7 mA/input (24 VDC)
Input impedance	I0, I1, I6, I7: 5 kΩ I2 to I5, I8 to I11: 3.4 kΩ
Switching time at high state (ON Time)	I0 to I7: 35 µs + filter value I8 to I11: 40 µs + filter value
Switching time at low state (OFF Time)	I0, I1, I6, I7: 45 µs + filter value I2 to I5, I8 to I11: 150 µs + filter value
Isolation	Between input terminals: not isolated Internal circuit: Optocoupler isolated (up to 500 VAC rms)
Filtering: 3 possibilities ● None ● 3 ms ● 12 ms	I0 to I11
Input type	Type 1 (IEC 61131)
External load for I/O interconnection	Not needed
Signal determination method	Static
Effect of incorrect input connection	The input signals can be both sink and source. But if any input exceeding the nominal value is applied, permanent damage may be caused. In all cases, the user is responsible for the wiring.
Input cable length	30 m *(98.4 ft)* for compliance with electromagnetic immunity
Average number of connector insertions/removals	100 times minimum

Input Operating Range

The module input operating range is shown below.



I/O Usage Limits

100% of the I/O can be used at 55°C *(131°F)*.

Q0 and Q1 Output Characteristics

Network Interface Module	**OTB 1•0DM9LP**
Output type	2 source transistor outputs
Number of outputs per common	2
Nominal load voltage	24 VDC
Maximum load current	1 A per common
Operating load voltage range	20.4 ... 28.8 VDC
Residual voltage (on voltage)	1 V maximum (voltage between COM and output terminals when output is active)
Nominal load current	0.3 A per output
Inrush current	1 A maximum
Leakage current	0.1 mA maximum
Limit voltage	39 V +/-1 V
Absorbed power	8 W
Inductive load	L/R = 10 ms (28.8 VDC, 1 Hz)
External current drawn	100 mA maximum, 24 VDC (voltage at the -V terminal)
Isolation	Between output terminals and internal circuit: Optocoupler isolated (up to 500 VAC rms) Between output terminals: not isolated
Average number of connector insertions/removals	100 times minimum
Output delay - turn-on time	300 µs maximum
Output delay - turn-off time	300 µs maximum

Q2 to Q7 Output Characteristics

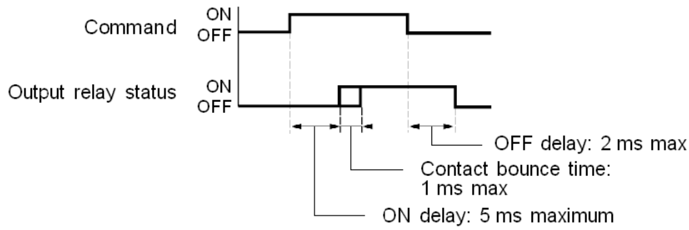
Network Interface Module	OTB 1•0DM9LP
Output type	6 relay outputs
Number of outputs per common - COM0	2 outputs
Number of outputs per common - COM1	3 NO contacts
Number of outputs per common - COM2	2 NO contacts
Number of outputs per common - COM3	1 NO contact
Maximum load current	2 A per output 8 A per common
Minimum switching load	0.1 mA/0.1 VDC (reference value)
Initial contact resistance	30 mΩ maximum
Mechanical life	20,000,000 operations minimum (estimated load 18,000 operations/hr)
Dielectric strength	1500 VACrms between the output and internal circuit, 1 min 750 VACrms between the output and COM terminal, 1 min
Average number of connector insertions/removals	100 times minimum
Closing delay	5 ms typ, 10 ms max
Opening delay	2 ms typ, 5 ms max
Closing bounce time	1 ms maximum

Operating Category	Nominal Load	Electrical Life (Number of Operations)
AC1 Resistive load control	500 VA(*)	10^5
AC14 Weak electromagnet load	250 VA	10^5
AC15 Electromagnet	200 VA	10^5
DC1 Resistive load control	60 W(*)	10^5
DC13 Electromagnet L/R=150 ms	30 W	10^5

(*) for AC1 & DC1 the powers indicated here take account of the maximum per point on OTB (2A).

Relay Output Delay

The output delay is illustrated below.



Wiring diagram for the network interface modules

Introduction

This section shows examples of wiring diagrams for Advantys OTB network interface modules.

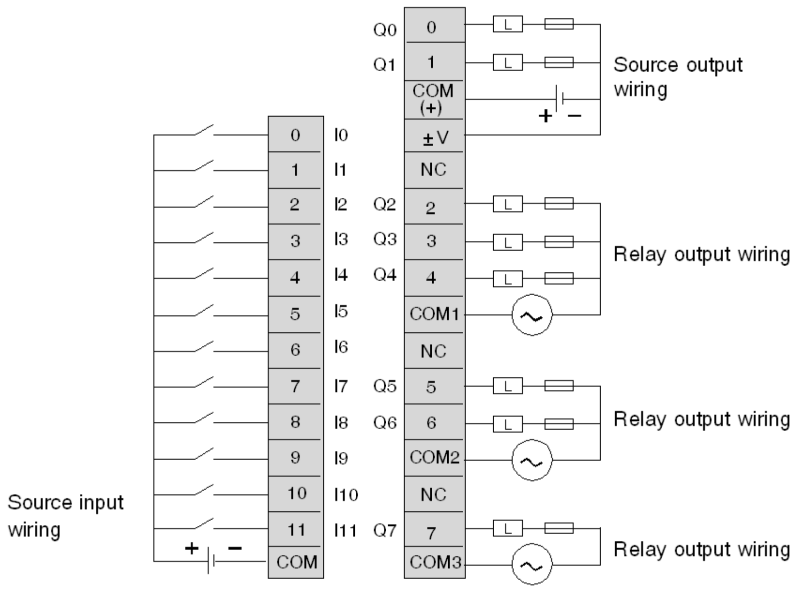
NOTE: These schematics are for external wiring only.

NOTE: The shaded boxes are markings on the module. The I and Q numbers are the input and output points.

NOTE: The inputs that are used as counting inputs must be connected with shielded cables.

Wiring Diagram for OTB Modules

This diagram applies to OTB 1•0DM9LP modules.



- Output points 0 and 1 are source transistor outputs, all other output points are relay.
- The COM terminals are **not** connected together internally.
- Connect an appropriate fuse for the load.

How to Connect the Power Supply

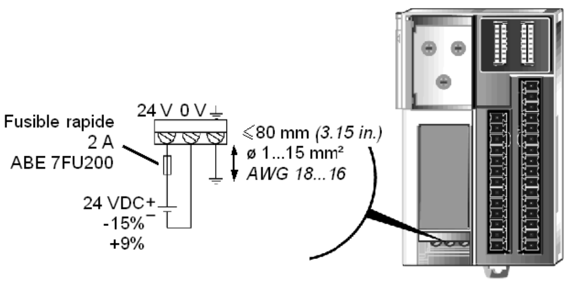
Introduction

This section describes how to connect the power supply to the network interface modules.

⚠ WARNING
RISK OF UNINTENDED EQUIPMENT OPERATION
When the power supply voltage is outside of the specified voltage range, outputs may not operate as expected. Use an appropriate externally-wired safety system to control and monitor the system voltage and ensure the specified voltage range is maintained.
Failure to follow these instructions can result in death, serious injury, or equipment damage.

Power Connection for a Network Interface Module

The following diagram describes the power connection for an Advantys OTB network interface module.



NOTE: Grounding wire length should not exceed 80 mm *(3.15 in.)*.

The sensor/actuator cables must be shorter than 30 m *(98.42 ft.)*.

⚠ WARNING
DANGER: UNEXPECTED EQUIPMENT OPERATION RISK
If the unit is not connected to the ground, or if the ground connection is made using an inappropriate cable, the product will be sensitive to electromagnetic disturbances. This may lead to unexpected equipment operation (*(see page 62)*).
Failure to follow these instructions can result in death, serious injury, or equipment damage.

Network Interface Module Power Supply Specifications

The following table describes the power supply specifications for the network interface module.

Item	Characteristics
Power supply voltage	Rated power voltage: 24 VDC Allowable range: from 20.4 to 26.4 VDC
Inrush current flow at power-up	50 A maximum
Power supply wiring	0.14 mm^2 *(AWG26)* or 1.5 mm^2 *(AWG16)* Make the power supply wiring as short as possible.
Ground wiring	1 mm^2 *(AWG18)* or 1.5 mm^2 *(AWG16)* Do not connect ground wire in common with ground wire of motor equipment. The grounding connection should be as short as possible < 8 cm *(3.15 in)*.

NOTE: Momentary power interruption for 10 ms or less at 24 VDC is not recognized as failure.

EMC Compatibility

Product Compliance

CE

This product complies with the European directive 89/336/EEC on "electromagnetic compatibility".

The products described in this manual meet all the conditions regarding electromagnetic compatibility and are compliant with the applicable standards. However, this does not mean that the electromagnetic compatibility of your installation is assured.

This is why it is strongly recommended to follow all instructions concerning an EMC-compliant installation. Only in these conditions and thanks to the exclusive use of CE approved components, will the devices used be deemed to comply with the EMC directives.

When handling the products, ensure that all safety measures related to electromagnetic compatibility and all conditions for the use of the products are complied with by all persons concerned. This is especially important when handling products sensitive to electrostatic discharges.

⚠ WARNING

RISK OF ELECTROMAGNETIC INTERFERENCE AND UNINTENDED EQUIPMENT OPERATION

The products described in this manual contain highly complex semiconductors that can be damaged or destroyed by electrostatic discharges (ESD). If, for example, they are used within the vicinity of devices rated as class A or B according to IEC 6100-4-4, the level of electromagnetic interference may be enough to cause the device to operate unexpectedly, and/or to damage it.

Damage may not necessarily cause a failure or malfunction that is immediately detectable. It may occur sporadically or in a delayed manner.

If there is a risk of electromagnetic interference, the system designer must implement the necessary protective measures.

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

Grounding

The grounding cable must be shorter than 8 cm *(3.15 in)*.

⚠ WARNING

RISK OF UNINTENDED EQUIPMENT OPERATION

If the unit is not connected to the ground, or if the ground connection is made using an inappropriate cable, the product will be sensitive to electromagnetic disturbance. This can lead to unintended equipment operation.

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

Cable Routing

Make sure that the following basic wiring rules are observed:

- Make sure there is a space of at least 10 cm *(3.94 in)* between the data cables and the power cables.
- The data cables and power cables must only cross at a right angle to one another.
- It is advisable to route the data cables and power cables through separate shielded ducts.
- When laying the cables, the noise voltage from other devices or wires must be considered. This particularly applies to frequency converters, motors and other devices or cables generating high frequency disturbance. High-frequency sources and the cables described in this manual must be as far apart from each other as possible.

⚠ WARNING

RISK OF UNINTENDED EQUIPMENT OPERATION

Please read and comply with the cabling rules listed above. Failure to comply with these wiring rules is a common cause of EMC problems! This can lead to unintended equipment operation.

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

Control of Inductive Loads

The outputs of the devices described in this manual are equipped with an integrated protective system against the high noise voltages that may be generated by inductive loads.

Integrated protective system against the high noise voltages generated by inductive loads

The varistor rapidly discharges the energy accumulated in the magnetic field of the inductive load.

The high voltages arising from the disconnection of inductive loads create large fields in the wires that may cause disturbances in nearby circuits or devices. It is advisable to fit an anti-interference device on the load. In this way, the voltage peak generated by the inductive load is short-circuited directly at the point where it occurs.

Field Bus or Network Connection

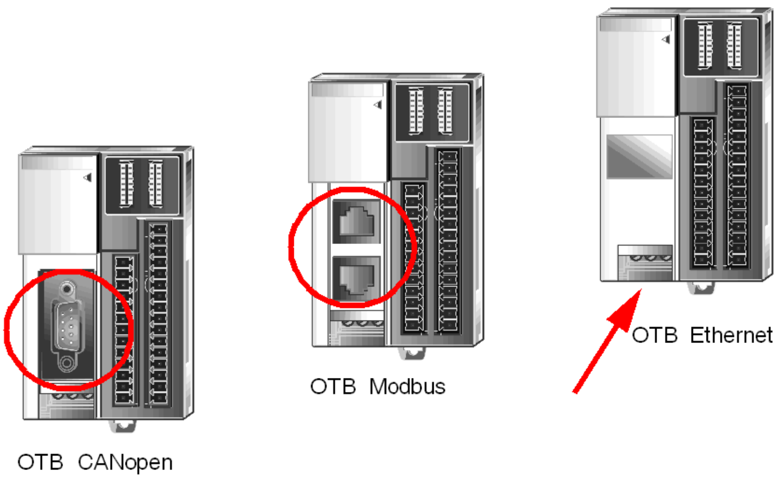
Overview

The specific types of cables and connectors for connecting the field bus or network of the OTB module vary according to the network used. Detailed cabling and connector information is given in Chapter 5 "Network Interface".

Field Bus or Network Connection

The field bus is connected between your master or server and the complete and physically installed OTB island. To make this connection, simply press the field bus connector into the specially-provided receptacle, and lock the connector in place.

There are three types of OTB modules: CANopen, Modbus and Ethernet. These represent the three available field bus protocols or networks. Below are the illustrations of the three types of different modules. Please note the differences between the field bus and network plugs.



Description, characteristics, and wiring of the expansion modules

4

Introduction

This chapter provides an overview of the analog and Discrete I/O expansion modules. Information on functions and wiring is given for each expansion module.

What's in this Chapter?

This chapter contains the following sections:

4.1 Discrete Expansion Modules

Discrete Expansion Modules

Reference Documents

Refer to the hardware installation manual for TM2 discrete I/O modules.

4.2 Analog Expansion Modules

Analog Expansion Modules

Reference Documents

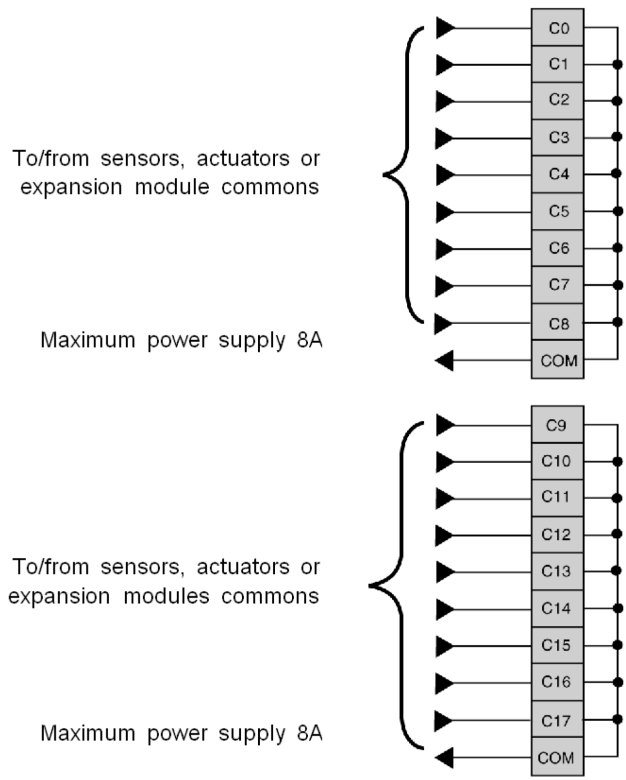
Refer to the hardware installation manual for TM2 analog I/O modules.

4.3 Expansion Module Common Blocks

Common Block Expansion Module Wiring Diagrams

OTB 9ZZ61JP Wiring Diagram

This diagram is for OTB 9ZZ61JP common block expansion modules. The common blocks can be used in a variety of ways, such as grouping I/O commons or power distribution.



⚠ DANGER

DANGER OF ELECTRIC SHOCK OR FIRE

Do not exceed 8A per common group.

Failure to follow these instructions will result in death or serious injury.

CANopen network interface

5

Introduction

This chapter describes the CANopen Advantys OTB network interface module, and gives a reminder of the main characteristics of the CANopen field bus protocol and the specific functions for managing the island.

What's in this Chapter?

This chapter contains the following sections:

5.1 General Principles

Introduction

This section addresses the general principles for operating and using the CANopen network.

What's in this Section?

This section contains the following topics:

About CANopen

Introduction

CANopen is a standard fieldbus protocol for industrial control systems. It is particularly well suited to real-time PLCs, as it provides an effective, low-cost solution for industrial applications.

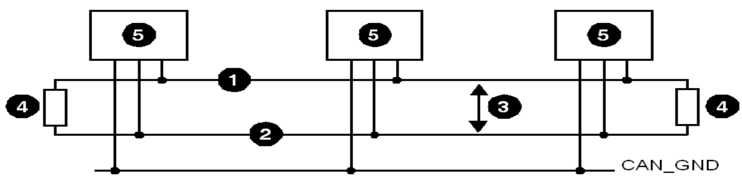
The CANopen Protocol

The CANopen protocol was created as a subset of CAL (CAN Application Layer). By profile definition, it is even more specifically adapted for use in standard industrial components. CANopen is a standard from the CiA (CAN in Automation) association that has been very quickly adopted by users since it first became available. In Europe, CANopen is now recognized as the industry standard for industrial systems based on a CAN design.

Physical Layer

CAN uses a differentially driven two-wire bus line (common ground). A CAN signal is the difference between the voltage levels of the CAN_H and CAN_L wires. (See figure below.)

The diagram below shows the components of the physical layer of a three-wire CAN bus:



1 CAN_H wire
2 CAN_L wire
3 Difference in voltage between CAN-H/CAN-L signals
4 120 Ω line terminator
5 Connected devices

The bus wires can be routed in parallel, twisted or shielded form in accordance with electromagnetic compatibility requirements.

CANopen Profiles

The communication profile

The CANopen communication protocol is based on a "communication profile", which specifies the main communication mechanisms and their description (DS301).

The device profile

The most important types of devices used in industrial automation are described in the "Device profiles". These also define the device functions.

Here are some examples of standard devices:

- Discrete and analog input/output splitter boxes (DS401)
- Motors (DS402)
- Control devices (DSP403)
- Closed loop controllers (DSP404)
- PLCs (DS405)
- Encoders (DS406)

Device Configuration via the CAN Bus

The possibility of configuring devices via the CANopen bus is one of the basic principles of the autonomy required by manufacturers (for each profile family).

General Specifications for CANopen Profiles

CANopen is a set of profiles for CAN systems with the following specifications:

- An open bus system
- Real-time data exchange without protocol overload
- A modular design with the possibility of resizing
- Interconnection and interchangeability of devices
- Support guaranteed by a large number of international manufacturers
- Standardized network configuration
- Access to all device parameters
- Synchronization and circulation of cyclic process data and/or event-driven data (possibility of short system response times).

CANopen Product Certification

All manufacturers offering CANopen-certified products on the market are members of the CiA (CAN in Automation) industrial consortium. As an active member of the CiA consortium, Schneider Electric develops its products in compliance with standard recommendations recognized internationally by the CiA consortium.

CAN Standards

CANopen specifications are defined by the CiA association and are available on the site www.can-cia.de. The master and slave source codes are available from different suppliers.

NOTE: To learn more about CANopen specifications and standard mechanisms, go to the CiA homepage (http//www.can-cia.de).

Communication on a CANopen Network

The communication profile is based on CAL (CAN Application Layer) services and protocols.

It provides the user with access to two types of exchange: SDO (Service Data Object) and PDO (Process Data Object).

On power-up, the device enters an initialization phase then goes into "Pre-operational" state. At this stage, only SDO communication is authorized. After receiving a startup command, the device switches to the "Operational" state. PDO and SDO communications are both authorized when the device is in the "Operational" state.

The Device Profile

List of Functions

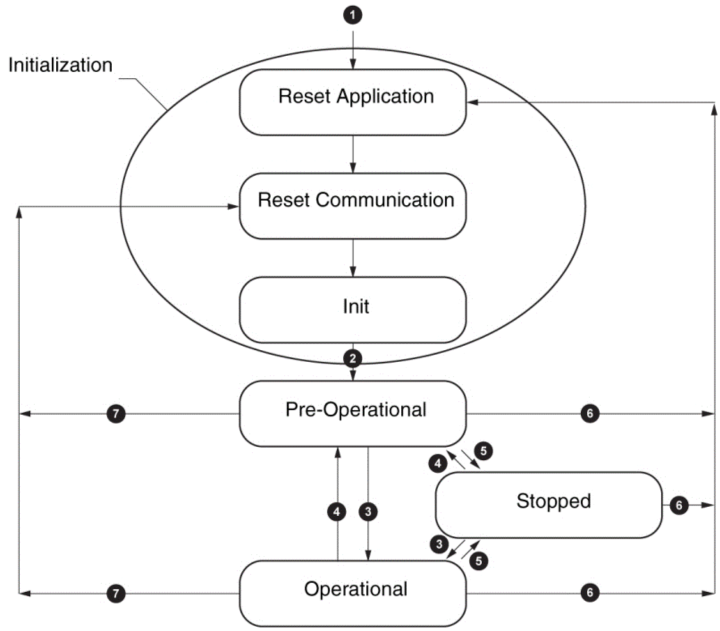
The list of functions supported and their codes are given in the table below:

Function	Function Code (binary)	Resulting COB-ID (Hex)	Resulting COB-ID (Dec)
NMT	0000	0	0
SYNC	0001	80	128
EMERGENCY	0001	81 - FF	129 - 255
TPDO1 (Tx)	0011	181 - 1FF	385 - 511
RPDO1 (Rx)	0100	201 - 27F	513 - 639
TPDO2 (Tx)	0101	281 - 2FF	641 - 767
RPDO2 (Rx)	0110	301 - 37F	769 - 895
TPDO3 (Tx)	0111	381 - 3FF	897 - 1023
RPDO3 (Rx)	1000	401 - 47F	1025 - 1151
TPDO4 (Tx)	1001	481 - 4FF	1153 - 1279
RPDO4 (Rx)	1010	501 - 57F	1281 - 1407
SDO (Tx)	1011	581 - 5FF	1409 - 1535
SDO (Rx)	1100	601 - 67F	1537 - 1663
NMT Error Control	1110	701 - 77F	1793 - 1919

CANopen "Boot-Up"

Procedure for "Boot-Up"

The minimum configuration of the equipment specifies a shortened boot procedure. This procedure is illustrated by the following diagram (excerpt of the DS 301 standard). Detailed device behavior is described in the following chapters :



Key

Number	Description
1	Device power-up
2	After initialization, the device automatically goes into the PRE-OPERATIONAL state
3	NMT service indication: START REMOTE NODE
4	NMT service indication: ENTER PRE-OPERATIONAL
5	NMT service indication: STOP REMOTE NODE
6	NMT service indication: RESET NODE
7	NMT service indication: RESET COMMUNICATION

NMT : Network Management Telegram

Active CANopen Objects depending on State Machine

The crosses in the table below indicate which CANopen objects are active for which states of the state machine:

	Initialisation	Pre-Operational	Operational	Stopped
PDO object:			X	
SDO object:		X	X	
Emergency		X	X	
Boot-Up	X			
NMT		X	X	X

"Reset Application"

The device goes into the "Reset Application" state:

- after the device starts up or,
- by "RESET NODE" (NMT service,Network Management Telegram).

In this state, the device profile is initialized, the device profile data is reset to the last saved value. When initialization is complete, the device automatically goes into the state "Reset Communication".

"Reset Communication"

The device goes into the "Reset Communication" state:

- after the "Reset Application" state,
- by "RESET COMMUNICATION" (NMT service).

In this state, all the parameters (standard value, depending on the device configuration) of the supported communication objects (1000H - 1FFFH) are saved in the object directory. The device then automatically goes into the "Init" state.

"Init"

The device goes into "Init" mode after being in the "Reset Communication" state.

This state enables you to:

- define the required communication objects (SDO, PDO, Sync, Emergency),
- install the corresponding CAL services
- configure the CAN-Controller.

Initialization of the device is complete and the device automatically goes into the "Pre-Operational" state and sends a "Boot-Up" message.

"Pre-Operational"

The device goes into the "Pre-Operational" state:

- after the "Init" state,
- on receiving the NMT "ENTER PRE-OPERATIONAL" indication if it was in the "Operational" or "Stopped" state.

When the device is in this state, its configuration can be modified. However, only SDOs can be used to read or write device-related data.

When configuration is complete, the device goes into one of the following states on receiving the corresponding indication:

- "Stopped" on receiving the NMT "STOP REMOTE NODE" indication.
- "Operational" on receiving the NMT "START REMOTE NODE" indication.

"Stopped"

The device goes into the "Stopped" state on receiving the "STOP REMOTE NODE" indication (NMT service) if it was in "Pre-Operational" or "Operational" state.

In this state, the device cannot be configured. No service is available to read and write device-related data (SDO). Only the slave monitoring function ("Node-Guarding" or "Heartbeat") remains active.

"Operational"

The device goes into the "Operational" state if it was in the "Pre-Operational" or "Stopped" state on receiving "START REMOTE NODE" indication.

During startup of the CANopen network using the NMT "START REMOTE NODE" services, all functions of the device can be used. Communication can be carried out via PDOs or SDOs.

⚠ WARNING

RISK OF UNINTENDED DEVICE OPERATION

Do not change the device configuration when it is in "Operational" state. Changing the equipment configuration while it is in the "Operational" state may result in the device behaving in an unexpected manner and/or in equipment damage or injury to personnel. If the device needs to be reconfigured, put it in the "Pre-Operational" state and check that this has been done correctly before proceeding to modify the configuration.

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

Process Data Object (PDO) Transmission

Definition of a PDO

PDOs are objects which provide the communication interface with process data and enable them to be exchanged in real time. The set of PDOs on a CANopen device describes the implicit exchanges between this device and its communication partners on the network.

The exchange of PDOs is authorized when the device is in "Operational" mode.

Types of PDO

There are two types of PDO:

- PDOs transmitted by the device ("Transmit PDO", "TPDO")
- PDOs received by the device ("Receive PDO", "RPDO")

PDO Consumer/Producer

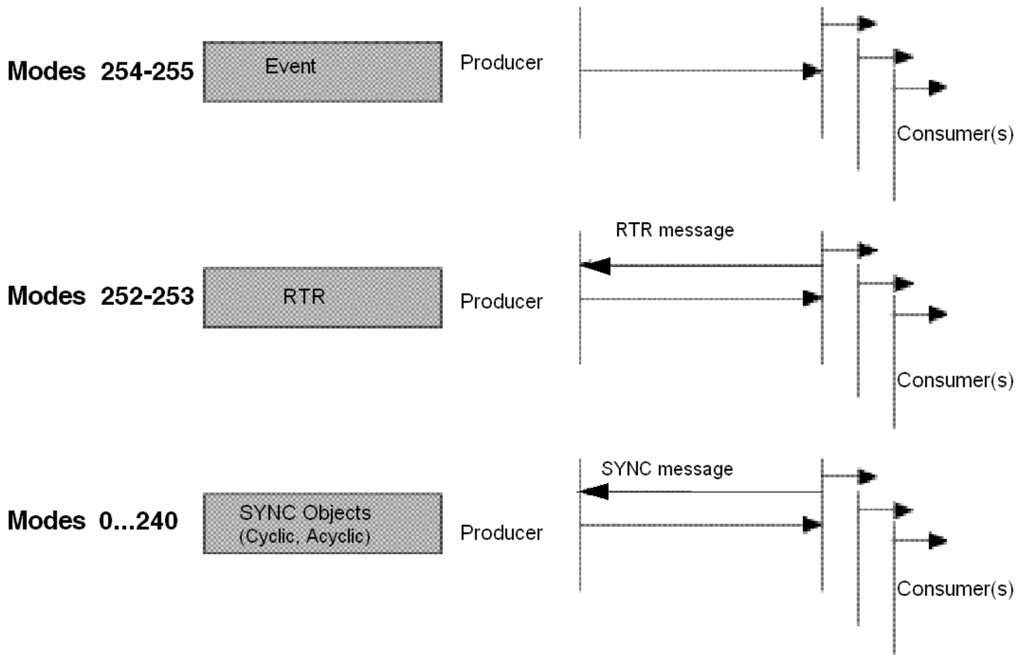
PDOs are based on the "Producer/consumer" model . The device which sends a PDO is called the producer, while the device receiving it is known as the consumer.

PDO Transmission Modes

In addition to data to be transported, it is possible to configure the type of exchange for each PDO.

The PDO transmission mode can be configured as described in the table below.

Transfer Code		**Transmission Mode**					**Notes**
Dec.	**Hex.**	**Cyclic**	**Acyclic**	**Synchro-nous**	**Asyn-chronous**	**only RTR**	
0	0		x	x			Send PDO on first SYNC message following an event
1 to 240	1 to F0	x		x			Send PDO every x SYNC messages
241 to 251	F1 to FB	Reserved					-
252	FC			x		x	Receive SYNC message and send PDO on RTR
253	FD				x	x	Data update and sending of PDO on RTR
254 to 255	FE to FF				x		Send PDO on event (Change of state mode)

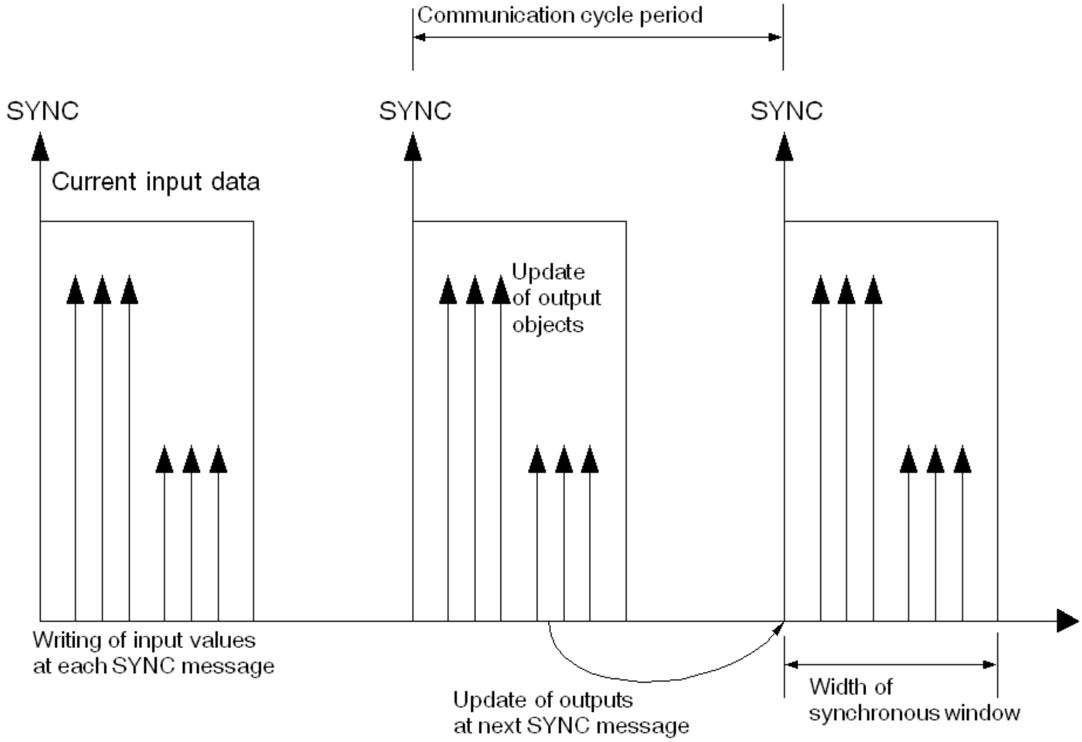


Synchronous (Mode 0 to 240)

For certain applications, synchronization between scanning of the inputs and activation of the outputs may be necessary.

For this reason, CANopen provides the "SYNC" object, a high-priority CAN message without any working data which, when it is received by the synchronized devices is used to trigger the reading of inputs or activation of outputs (Trigger).

The following diagram shows the time-related data for synchronized PDO transmission.



Synchronous RTR (Mode 252)

The master can poll slaves by using data request messages ("Remote-Frames", called RTR messages).

In mode 252 the device sends the TPDO upon receipt of the synchronization message that follows the RTR message.

Asynchronous RTR (mode 253)

In mode 253, the TPDOs are transmitted once the RTR message is received.

"Change of state" (Modes 254 and 255)

The asynchronous exchange of PDO in "Change of state" mode enables the rapid modification of an input value, followed by immediate confirmation of the change of value. This avoids the need to wait for the master to send a request.

A high priority bus status is assigned to the "Change of state" mode and only the updated input value is returned, not the image of the full process, thus considerably reducing traffic on the bus.

"Change of state" corresponds to modification of the input value (event control).

⚠ WARNING

RISK OF UNINTENDED EQUIPMENT OPERATION

The "Change of State" mode must not be used for I/O whose state changes continuously (such as analog inputs). The continual modification of I/O using the "Change of State" mode may block the transmission of other crucial commands, resulting in unintended operation of the device.

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

Inhibit Time and Event Timer

Inhibit Time

In event transmission mode, the Inhibit Time utility is used to define a minimum time delay before transmission of a new PDO. This avoids overloading the bus where a significant number of events occur in rapid succession.

The Inhibit Time is expressed in multiple of 100 µs.

Values (hex.)	Values (dec.)	Actual values (ms)
0000H	0	0000
000AH	10	1
0064H	100	10
03E8H	1000	100
2710H	10 000	1000
FFFFH	65 535	6553.5

Event Timer

In event transmission mode, the Event Timer is used to define an expiry time delay where transmission of a PDO will be forced, even if there has been no change in status .

The Event Timer is expressed in milliseconds.

Values (hex.)	Values (dec.)	Actual values (ms)
0000H	0	0 (deactivated)
000AH	10	10
0064H	100	100
01F4H	500	500
03E8H	1000	1000
1388H	5000	5000
2710H	10 000	10 000

Access to Data by Explicit Exchanges (SDO (Service Data Object))

What is an SDO?

An SDO allows a device's data to be accessed by using explicit requests.

The SDO service is available when the device is in "Operational" or "Pre-Operational" state.

Types of SDO

There are two types of SDO:
- Read SDOs (Download SDO)
- Write SDOs (Upload SDO)

The Producer/Consumer Model

The SDO protocol is based on a 'Producer/Consumer' model.

For a Download SDO

The client sends a request indicating the object to be read.

The server returns the data contained within the object.

For an Upload SDO

The client sends a request indicating the object to be written to and the desired value.

After the object has been updated, the server returns a confirmation message.

For an unprocessed SDO

In both cases, if an SDO was not able to be processed, the server returns an error code (Abort Code).

"Node-Guarding" and "Life-Guarding" Monitoring Protocols

Introduction

Error monitoring protocols are used to detect communication errors on the network. The default monitoring method, "Node-Guarding", consists in the master controlling the slaves. It is possible to add "Life-Guarding" control of the master by the slaves.

NOTE: The simultaneous use of both monitoring methods, "Guarding" and "Heartbeat", is impossible. Should both methods be activated at once, the equipment will only use the "Heartbeat" monitoring method.

Definition of "Life-Time"

The "Life-Time" parameter is calculated as follows:

"Life-Time"= "Guard-Time" x "Life-Time-Factor"

The object 100CH contains the "Guard-Time" parameter expressed in milliseconds. The object 100DH contains the "Life-Time-Factor" parameter.

Activation of Monitoring

When one of the two parameters "Life-Time-Factor" or "Guard-Time" is set to "0" (default configuration), the device does not perform monitoring (no "Life-Guarding").

To activate monitoring over time, you must enter a value (minimum 1) in the object 100DH and specify a time in ms in the object 100CH.

Common typical values for the "Guard-Time" parameter lie between 250 ms and 2 s.

Reliable Operation

To enable reliable and secure operation, the user must enter a "Life-Time-Factor" with a minimum value of 2.

When the value 1 is used, should a delay occur due to the processing of high priority messages or internal processing on the "Node-Guarding" master, the device switches back to the "Pre-Operational" default state without generating any errors.

⚠ WARNING

RISK OF UNINTENDED DEVICE OPERATION

Set the "Life-Time-Factor" (object 100DH) to a minimum value of 2 to prevent any inadvertent change of state to "Pre-Operational" state. Depending on the I/O configuration, an inadvertent change of state may result in unintended device operation.

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

Importance of Monitoring

These two monitoring mechanisms are especially important in the CANopen system. Devices connected to the bus do not regularly indicate their presence in operating mode, commanded by "Event".

Slave Monitoring

Monitoring is performed in the following way:

Phase	Description
1	The master sets "Remote-Frames" (or "Remote-Transmit-Request" request messages) on the "Guarding-CobID" of the slaves to be monitored.
2	The slaves concerned respond by sending the "Guarding" message. This message contains the "Status-Code" of the slave and the "Toggle-Bit", which changes after each message.
3	The NMT (Network Management Telegram) master compares the "Status"and "Toggle-Bit" information: If they are not in the expected state or if no response is received, the NMT master considers that an error has occurred on the slave.

Master Monitoring

If the master requests "Guarding" messages on a strictly cyclical basis, the slave can detect a master failure.

If the slave does not receive a request from the master within the defined "Life-Time" interval ("Guarding" error), it considers that a master failure has occurred ("Watchdog" function).

In this case, the corresponding outputs go into fallback mode and the slave switches back into "Pre-Operational" mode.

WARNING

RISK OF UNINTENDED DEVICE OPERATION

An unexpected change in state to "Pre-Operational" mode may occur when the slave does not successfully detect the master’s request even though a slave-master communication monitoring protocol is used.

Depending on the configuration of the slave’s inputs and outputs, this change in state may result in unintended device operation or in bodily injury or equipment damage. The person in charge of configuring the system is fully responsible for the configuration of the slave inputs/outputs and must ensure secure fallback operations in the event of a loss of master/slave communication. The person in charge of the configuration must also take all necessary steps to ensure equipment and personnel safety should it prove impossible to secure the fallback operations.

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

NOTE: Even if the monitoring function over time is disabled ("Guard-Time" and "Life-Time-Factor" registers set to 0), the slave will respond to a remote request from the master.

"Guarding" Protocol

The initial value of the "Toggle-Bit" sent in the first "Guarding'" message is "0".

Then, the "Toggle" bit changes in each subsequent "Guarding" message, which makes it possible to indicate if a message has been lost.

The network state of the device is indicated in the seven remaining bits:

Network state	Response in hex.
Stopped	04H or 84H
Pre-operational	7FH or FFH
Operational	05H or 85H

The "Heartbeat" Error Monitoring Protocol

Operation of "Heartbeat" Mechanism

The default monitoring method is "Node-Guarding". If a non-zero value is written in the object 1017H, the "Heartbeat" mechanism is used.

If the Heartbeat error monitoring protocol is selected, the producer transmits a "Heartbeat" message periodically, depending on the "Producer Hearbeat Time" parameter.

The devices responsible for monitoring this message ("Heartbeat Consumer") generate a "HeartBeat" event if the message is not received in the configured time ("Consumer Heartbeat Time").

NOTE: The simultaneous use of two monitoring methods, "Guarding" and "Heartbeat", is not possible. In case both methods are activated simultaneously, the device will only use the "Heartbeat" monitoring method.

Meaning of Possible Values

The message "Heartbeat" indicates the device status on a byte that is broken down as follows:

- The most significant bit is reserved and always has a value of 0
- The 7 least significant bits provide the status for the device producing the "Heartbeat" message.

The possible values are as follows:

Status of the "Heartbeat Producer"	Value (Decimal)
Boot-Up	0
Stopped	4
Operational	5
Pre-Operational	127

5.2 Wiring on the CANopen Bus

Introduction

The following section describes wiring on the CANopen bus.

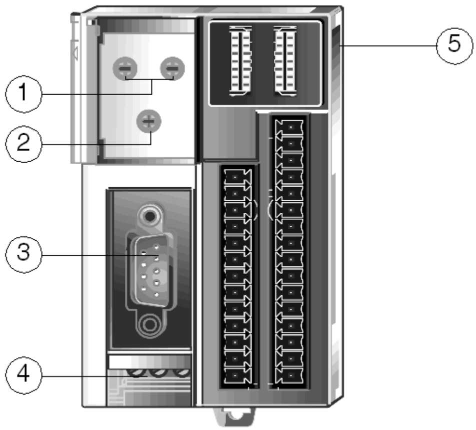
What's in this Section?

This section contains the following topics:

Description of the CANopen Network Interface Module

Introduction

The physical characteristics necessary for CANopen bus operation are shown in the following illustration:



The characteristics of the above illustration are given in the following table:

Description		Function	See
1	Upper encoder wheels	Definition of the node address for the CANopen field bus. ● Left encoder wheel: decimal encoding between 0 and 12 (tens), ● Right encoder wheel: decimal encoding between 0 and 9 (units).	Address *(see page 97)*
2	Lower encoder wheel	Selection of CANopen field bus speed.	Speed *(see page 99)*
3	Male 9-pin SUB-D connector	Connection to CANopen field bus.	Bus *(see page 96)*
4	Power supply terminal block	Connection of external 24 VDC supply for the network interface module.	Power supply *(see page 60)*
5	Indicator LED	Visual information on the operational state of the CANopen field bus and the network interface module.	Indicator LED *(see page 212)*

Topology

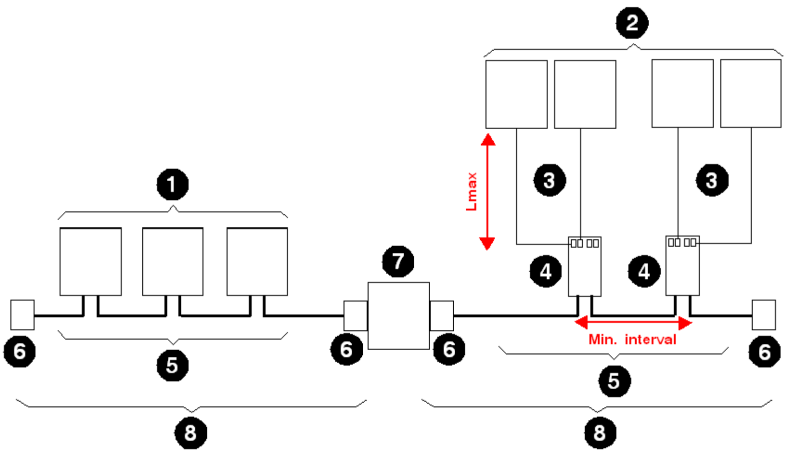
Architecture

The CANopen network architecture must comply with the following limitations:

- Bus length/transmission speed (See *Transmission Speed, page 95*)
- number of connected devices (See *Number of Connected Devices, page 94*)
- Length of the taps and the space between two taps (See *Tap Length, page 93*)
- Line terminator (See *Line Terminator Resistance, page 95*)

The connections to the CANopen bus may be the daisy-chaining or tap type.

The illustration below shows a CANopen network architecture:



The table below describes the components of a CANopen network:

Number	Description
1	CANopen devices connected by daisy-chaining
2	CANopen devices connected by taps
3	Drop cables (tap junction/device)
4	Tap junctions
5	Daisy-chaining cables
6	Line terminator
7	Repeater (identical arbitration on the various bus segments) or Bridge (different arbitration on the various bus segments)
8	CANopen bus segment

NOTE: A single-line architecture is recommended to reduce signal reflection. Avoid using star-type architectures.

Tap Length

A tap creates a signal reflection and thus its length must be limited on the following parameters:

Lmax is the maximum length of a tap.

Σ**LImax** is the maximum value of the sum of all taps on the same tap junction.

Min interval is the minimum distance necessary between two taps.

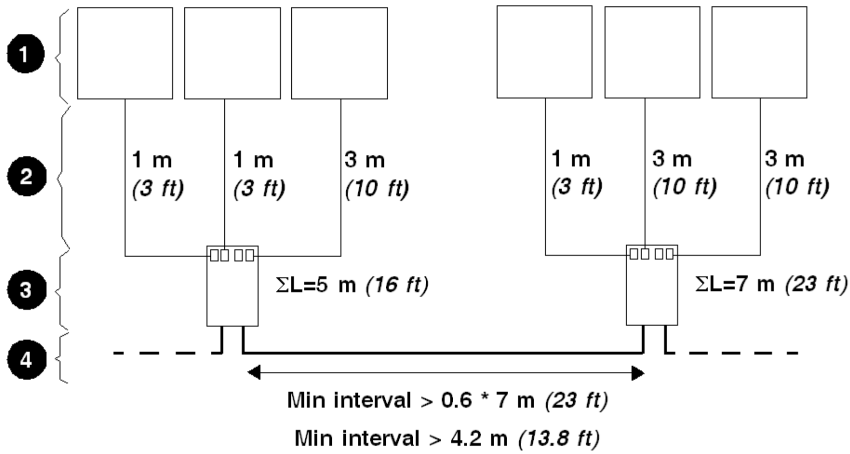
Σ**LGmax** is the maximum value of the sum of all taps on the segment.

The values to use are given in the table below:

Speed	**Lmax**	**ΣLImax**	**Min. interval 0.6xΣL local**	**ΣLGmax**
1 Mbps	0.3 m *(0.98 ft)*	0.6 m *(1.96 ft)*		1.5 m *(4.9 ft)*
800 Kbps	3 m *(9.8 ft)*	6 m *(19.6 ft)*	3.6 m *(11.8 ft)*(*)	15 m *(49 ft)*
500 Kbps	5 m *(16.4 ft)*	10 m *(32.80 ft)*	6 m *(19.6 ft)*(*)	30 m *(98.4 ft)*
250 Kbps	5 m *(16.4 ft)*	10 m *(32.80 ft)*	6 m *(19.6 ft)*(*)	60 m *(196.8 ft)*
125 Kbps	5 m *(16.4 ft)*	10 m *(32.80 ft)*	6 m *(19.6 ft)*(*)	120 m *(393.6 ft)*
50 Kbps	60 m *(196.8 ft)*	120 m *(393.6 ft)*	72 m *(236 ft)*(*)	300 m *(984 ft)*
20 Kbps	150 m *(492 ft)*	300 m *(984 ft)*	180 m *(590.5 ft)*(*)	750 m *(2460.6 ft)*
10 Kbps	300 m *(984 ft)*	600 m *(1968.4 ft)*	360 m *(1181 ft)*(*)	1500 m *(4921 ft)*
Legend:				
(*) The minimum cable length between two consecutive tap junctions must be greater than 60% of the largest of the two sums of the lengths of taps on each of the two junctions.				

Example

The illustration below shows the calculation of the length of a cable located between two tap junctions.



The table below describes the components of a CANopen network:

Number	Description
1	Connected CANopen devices
2	Drop cables (tap junction/device)
3	Tap junctions
4	Connection cables (tap junction/tap junction)

In this example, we have two tap junctions and 6 devices. We will begin by calculating the sum of the cable lengths for each tap junction; this gives us 5 m *(16 ft)* and 7 m *(23 ft)*. We keep the longest length, i.e. 7 m *(23 ft)*. The minimum cable length between the two tap junctions is 60% of 7 m or 4.2 m *(13.8 ft)*.

Number of Connected Devices

In addition to the length limitations over the whole of the CANopen bus, the following limitations apply:

- Whatever the case, no more than 64 devices may be connected on the same segment.
- Two segments must be separated by a repeater.
- The number of devices connected over an entire system depends on the CANopen master and cannot exceed 127.

Choice of System Cables

Transmission Speed

The maximum allowable transmission speeds are given in the table below:

Transmission Speed (Kbps)	Cable Length
1000	20 m *(65.62 ft)*
800	40 m *(131.23 ft)*
500	100 m *(328 ft)*
250	250 m *(820 ft)*
125	500 m *(1640.4 ft)*
50	1000 m *(3280 ft)*
20	2500 m *(8202 ft)*
10	5000 m *(16404 ft)*

Specific Resistance

The specific resistance of the cables should be below 70 mΩ/m.

Line Terminator Resistance

To minimize the voltage drop in the connection, it is advisable to use a higher line terminator resistance for cables longer than the length specified by the standard ISO11898-2. When configuring the system, the connector resistances must also be taken into consideration. The difference in potential at the CAN_GND connections of all the CANopen bus elements must not be greater than 2 VDC.

⚠ WARNING

RISK OF UNINTENDED EQUIPMENT OPERATION

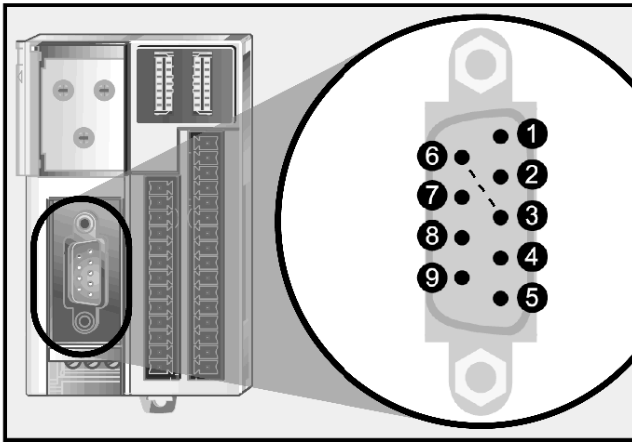
Connect a 120 Ω line terminator between CAN_H and CAN_L at the end of the line (see *Physical Layer, page 73*).

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

CANopen Fieldbus Interface

Connection to the Fieldbus

The connector is located in the lower part of the interface module:



Use a 9-way female SUB-D connector compliant with standard DIN 41652 or with the corresponding international standard. The connection with the OTB module must correspond to the following table:

Contacts	Signal	Description
1	Unused	Reserved
2	CAN_L	CAN_L bus line (Mandatory)
3	CAN_GND	CAN Ground (Mandatory)
4	Unused	Reserved
5	Unused	Reserved
6	GND	Ground
7	CAN_H	CAN_H bus line (Mandatory)
8	Unused	Reserved
9	Unused	Reserved
Shielding	CAN_SHLD	CAN shielding
Note: The contact numbers correspond to the key in the figure above.		

CANopen Network Cables

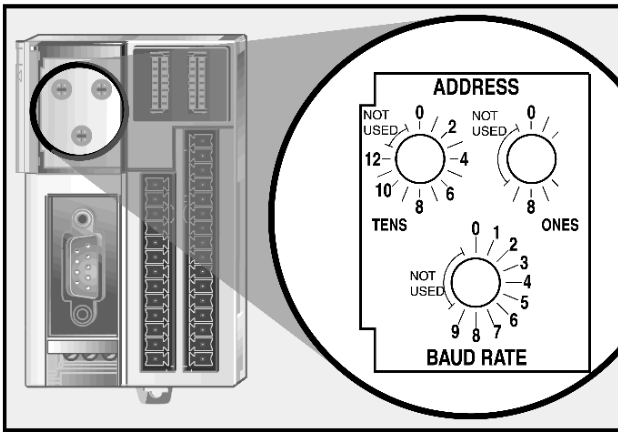
The CANopen network cables are shielded twisted pair cables complying with the CiA CANopen standard DR-303-1. No break in the wire is permitted in the bus cable. This enables the reserved contacts to be used in a future specification.

Island Network Address

Summary

The encoder wheels of the Advantys OTB CANopen module OTB 1C0DM9LP are used to define the address of the island on the network.

Physical Description



Island Addresses

The CANopen interface module reads the address of the island indicated by the upper encoder wheels every time the island is powered up.

The address of the island is a numerical value between 1 and 127, which must be different from all other island addresses on the network. If the configured address is prohibited, the network interface module will not communicate.

Configuring the Island Address

The instructions for configuring the address of the island are described in the following table.

Step	Action	Comment
1	All power supply to the island must be OFF.	The changes you make will be detected on the next power up.
2	Select an island address.	Select an address that is not currently being used on your field bus network.
3	Adjust the upper encoder wheels: • Left encoder wheel: 0 to 12 (tens), • Right encoder wheel: 0 to 9 (unit figures).	Use of addresses 0, 128 and 129 is prohibited.
4	Power up the island in order to implement the new configuration.	The network interface module reads the encoder wheel adjustments only on power up.

Field Bus Communication

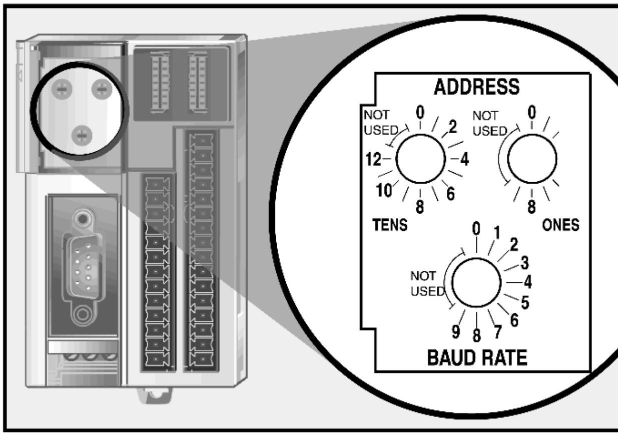
The Advantys OTB interface module communicates when the encoder wheels are configured to a valid and unique CANopen address. If the island has an invalid address, it cannot communicate with the master. To establish communication, configure the encoder wheels to a valid address and power up the island.

Network Speed

Summary

A thumbwheel on the Advantys OTB CANopen OTB 1C0DM9LP module is used to define the network speed.

Physical Description



Network Speed

The CANopen network interface module reads the address and baud rate indicated by the thumbwheels each time the module is powered up.

Network speed configuration

The instructions for configuring the baud rate are given in the table below.

Step	Action	Comment
1	All power supplies to the OTB network interface module must be off.	The changes you make will be detected on the next power-up.
2	Select the rate in bauds to be used for CANopen fieldbus communications.	The rate configuration depends on your system and network specifications.
3	Set the lower thumbwheel to the position corresponding to the required rate.	Use the following rate selection table.
4	Power up the OTB network interface module again to apply the new configuration.	The network interface module only reads the thumbwheel parameters on power-up.

Rate Selection Table

Position (Lower Thumbwheel)	Baud rate
0	10 Kbps
1	20 Kbps
2	50 Kbps
3	125 Kbps
4	250 Kbps
5	500 Kbps
6	800 Kbps
7	1 Mbps
8	Automatic detection
9	Default rate (250 kbps)

NOTE:

- If the rate selected is incorrect, the OTB module generates a Bus OFF.
- Position 8 is used to detect the bus transmission speed automatically. The search starts at 1 Mbps, then gradually reduces the baud rate in increments until communication is established on the bus. Automatic detection only works on an operational CANopen network with an active master.
- In automatic detection mode, at least one of the slaves present on the network must be configured with the same speed as the Master.

5.3 Advantys OTB CANopen,Network Interface Module Behavior

Introduction

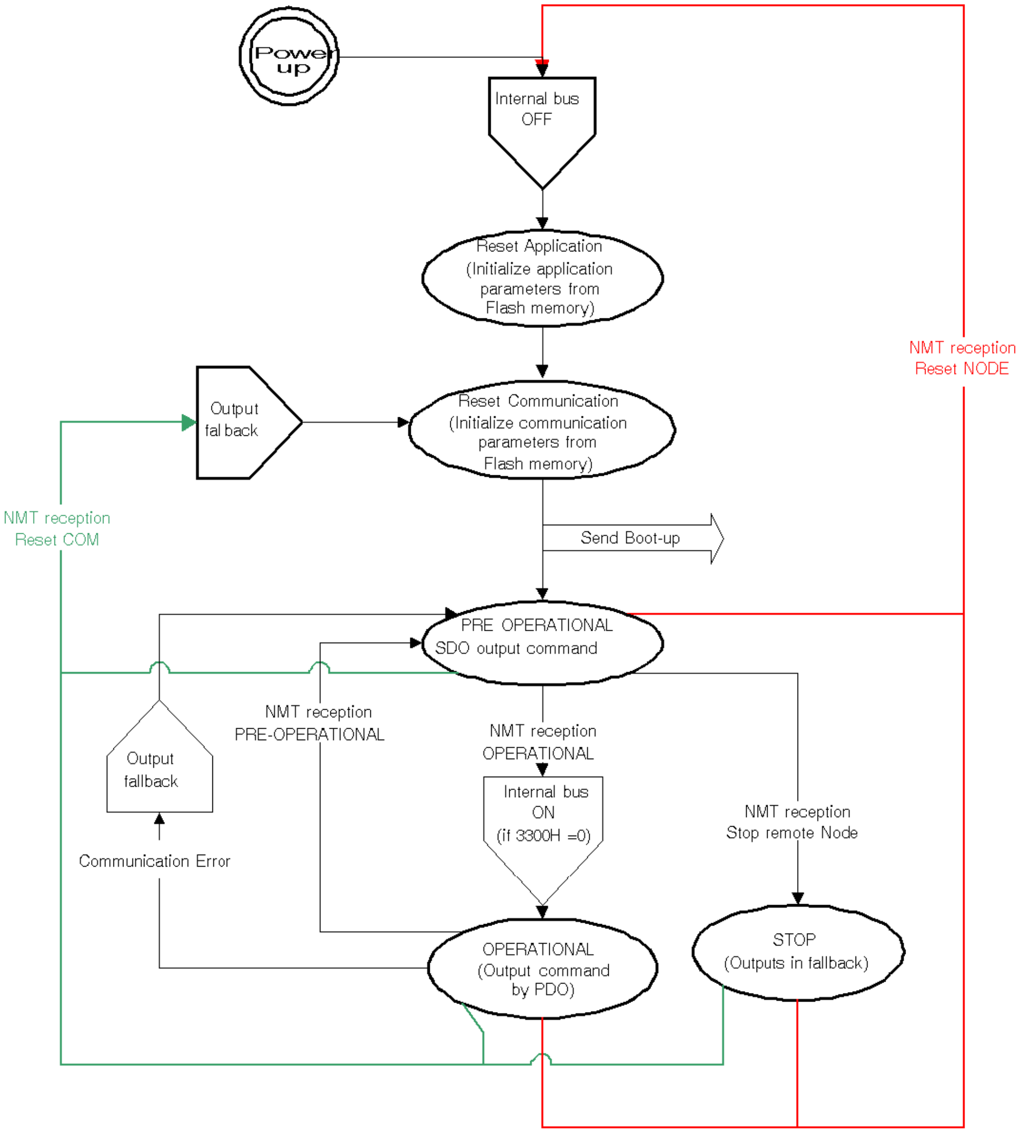
This section addresses the different behavior patterns of the Advantys OTB network interface module and the saving of different parameters.

What's in this Section?

This section contains the following topics:

Management of Island Behavior

Operating Diagram



Behavior at boot-up

Description

The behavior of the Advantys OTB CANopen network interface module at boot-up is in compliance with the "CANOPEN BOOT-UP *(see page 77)*" Diagram.

If a back-up configuration exists

Where a save has been carried out, the saved parameters are applied prior to switching to "Pre-Operational" status.

If a back-up configuration does not exist

If there is no back-up configuration, the Advantys OTB network interface module initializes the CANopen data with the default parameters.

Behavior in the Case of Communication Error

Description

In the event of a communication error detected by one of the error monitoring protocols ("Node-Guarding" or "Heartbeat"), fallback values are applied physically on the outputs until the next write of the output command object and when the communication error has disappeared.

Internal Bus Management

Switching the Internal Bus to the "Stop" State

The internal bus automatically switches from the ''Stop'' to the ''Run'' state when the communication module switches from the "Pre-operational" to the "Operational" state.

When the internal bus switches to the "Stop" state all the expansion module outputs are set to zero.

NOTE: When switching from "Stop" to "Run" the island outputs remain at 0 until the next time output is set.

Configuration of Expansion Modules

The internal bus used to update the configuration of the expansion module parameters.

The parameters are sent to the communication module when the bus is in the ''Stop'' state.

These new configuration parameters are acknowledged when the bus goes into the ''Run'' state.

Backup / Restore Configuration

Management of Saved Parameters

During initial power up, the Advantys OTB CANopen module is initialized with the default parameters. During subsequent power ups, it is initialized with the saved parameters.

NOTE:

- When the master detects the presence of the module on the network, the module parameters re-defined in the master's configuration tool are overwritten.
- During power up, the Advantys OTB module applies the saved parameters only if the physical configuration matches the configuration at the time they were saved. Otherwise (ex.: addition or removal of an expansion module), the default parameters are applied.

Updating Default Parameters

Saved parameters are only applied once the speed on the Advantys OTB CANopen module has been detected.

Saving Parameters

The back-up of parameters is performed by writing a signature to the object 1010H *(see page 237)* These parameters will be used during the next start-ups.

Restoring Parameters

The different possible restores are described in the following table:

To restart with...	it is necessary to write a signature to the object...	Implementation
the default configuration	1011H *(see page 238)*	After power up or "Reset Node"
the backed up configuration	3202H *(see page 360)*	Immediately

Recommendations to Avoid Data Losses

While writing or deleting saved parameters, the slave no longer processes communications received via the CANopen bus. During this operation, none of the messages transmitted to the slave are taken into account (this includes SDO or Node-Guarding messages).

In order to avoid equipment damage or injury to personnel as well as any losses of data, it is not advisable to initiate parameter saves or restitution when the equipment is in "Operational" mode.

⚠ WARNING

DANGER: UNEXPECTED EQUIPMENT OPERATION RISK

Do not change the Advantys OTB module configuration when it is in the "Operational" state. Changing the module configuration while it is in the "Operational" state may result in unexpected module behavior and/or equipment damage or injury to personnel. To reconfigure the Advantys OTB module, switch the module into the "Pre-Operational" state and confirm it has entered this state before attempting reconfiguration.

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

List of Saved Parameters

Application of Saved Parameters

In automatic speed detection mode, the saved parameters are only applied after speed detection.

Communication Profile Objects

The objects that are saved or reused on start-up are as follows:
- 1005H : COB-ID SYNC Message
- 1006H : Communication Cycle Period
- 100CH : Guard Time
- 100DH : Life Time Factor
- 1014H : COB-ID Emergency (EMCY) Message
- 1016H : Consumer Heartbeat Time
- 1017H : Producer Heartbeat Time
- 1400H...1407H : 1st to 8th Receive PDO Communication Parameter
- 1600H...1607H : 1st to 8th Receive PDO Mapping Parameter
- 1800H...1807H : 1st to 8th Transmit PDO Communication Parameter
- 1A00H...1A07H : 1st to 8th Transmit PDO Mapping Parameter

Manufacturer-specific Zone Objects

The objects that are saved or reused on start-up are as follows:
- 2000H : Local Digital Input Parameter
- 2100H : Analog Input Type
- 2101H : Analog Input Range
- 2102H : Analog Input Min
- 2103H : Analog Input Max
- 2104H : R_0 Value Register
- 2105H : T_0 Value Register
- 2106H : B Value Register
- 2200H : Analog Output Type
- 2201H : Analog Output Range
- 2202H : Analog Output Min
- 2203H : Analog Output Max
- 2300H : Counter Current Value
- 2304H : Counter Preset Value
- 2305H : Counter Delta Value
- 2306H : Counter Interrupt Trigger Selection
- 2307H : Counter Error Mode
- 2354H : Counter Preset Value
- 2355H : Counter Delta Value
- 2403H : Fast Counter Parameters Value
- 2404H : Fast Counter Preset Value

- 2405H : Fast Counter Delta Value
- 2406H : Fast Counter Interrupt Trigger Selection
- 2407H : Fast Counter Threshold S0
- 2408H : Fast Counter Threshold S1
- 2409H : Fast Counter Mode
- 240AH : Fast Counter Error Mode
- 2454H : Fast Counter Preset Value
- 2455H : Fast Counter Delta Value
- 2457H : Fast Counter Threshold S0
- 2458H : Fast Counter Threshold S1
- 2502H : PLS/PWM Time Base
- 2503H : PLS/PWM Preset Period
- 2504H : PLS/PWM Mode
- 2505H : PWM Ratio
- 2506H : PLS/PWM Interrupt Trigger Selection
- 2507H : PLS Number of Pulses
- 2508H : PLS/PWM Error Mode
- 2557H : PLS/PWM Time Base

Hardware Profile Objects

The objects that are saved or reused on start-up are as follows:

- 6102H : Polarity Input 16 Bits
- 6103H : Filter Mask Input 16 Bits
- 6302H : Polarity Output 16 Bits
- 6306H : Fallback Mode Output 16 Bits
- 6307H : Fallback Value Output 16 Bits
- 6308H : Filter Mask Output 16 Bits
- 6421H : Analog Input interrupt Trigger Selection
- 6424H : Analog Input interrupt Upper Limit Integer
- 6425H : Analog Input interrupt Lower Limit Integer
- 6426H : Analog Input interrupt Delta Value
- 6443H : Analog Output Fallback Mode
- 6444H : Analog Output Fallback Value

Application-Specific Functions

6

Introduction

This section describes the application-specific functions of the Advantys OTB modules. The information concerning I/O assignments, configuration and usage are provided for the OTB module and each expansion module.

What's in this Chapter?

This chapter contains the following sections:

6.1 Description of the OTB Module I/Os

Introduction

This section presents the I/O of the OTB module.

What's in this Section?

This section contains the following topics:

Description of the Advantys OTB Module Discrete I/Os

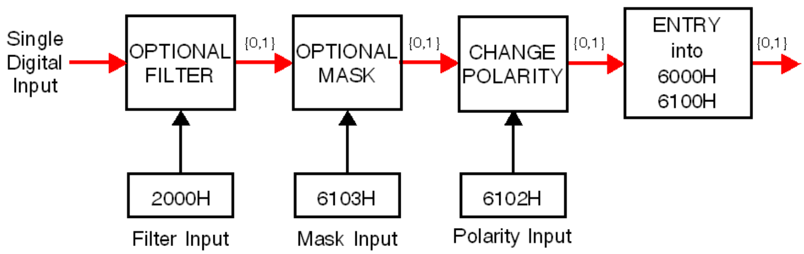
Discrete Inputs

Inputs are read by bytes (object 6000H) or by 16 bit words (object 6100H).

For each input, the following parameters may be modified:

- filter value (OFF, 3 or 12 ms) (object 2000H),
- Filter mask (object 6103H)
- Polarity (object 6102H).

The status read on inputs is defined as follows:



Discrete Outputs

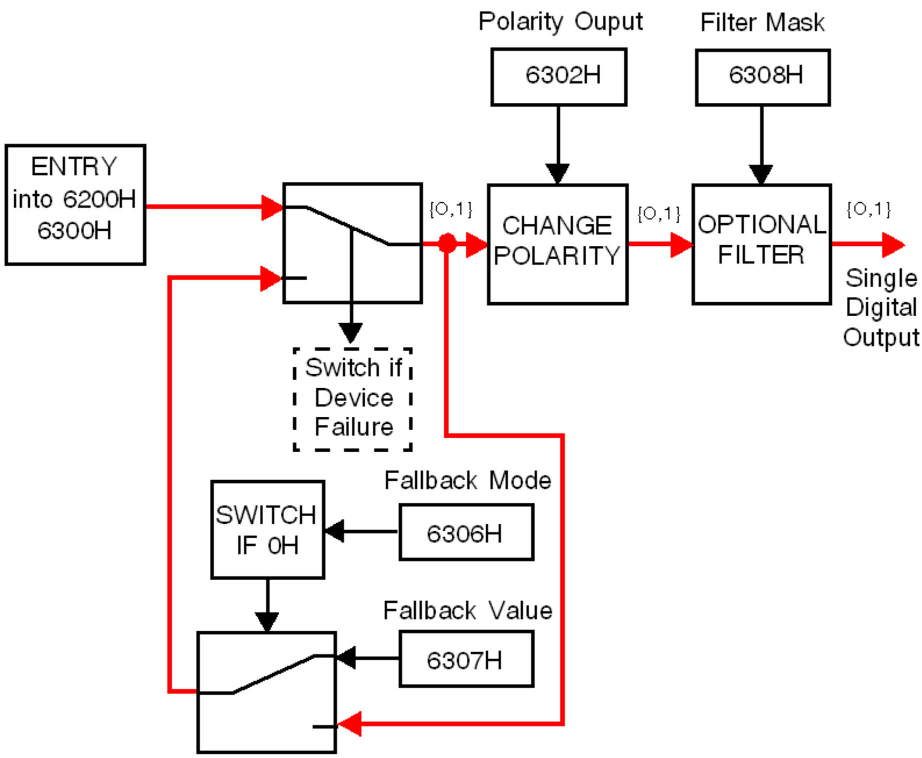
Outputs are written using bytes (object 6200H) or 16 bit words (object 6300H).

For each output, the following parameters may be modified:

- Polarity (object 6302H),
- Filter mask (object 6308H)
- Fallback mode (object 6306H),
- Fallback value (object 6307H).

In the event of an error (loss of communication with the master for example), the fallback mode is applied.

The state of the outputs is defined according to the configuration registers and equipment mode (according to the DS 401 device profile). See below:


Polarity Ouput
Filter Mask
6302H
6308H
ENTRY
into 6200H
6300H
{0,1}
CHANGE
POLARITY
{0,1}
OPTIONAL
FILTER
{0,1}
Single
Digital
Output
Switch if
Device
Failure
Fallback Mode
SWITCH
IF 0H
6306H
Fallback Value
6307H

Objects for the Discrete I/Os of the Advantys OTB Module

At a Glance

The discrete I/Os of the Advantys OTB network interface module use CANopen configuration objects.

Discrete I/O Access Objects

The following table describes the different objects reserved for the I/O read/write operations of the OTB module:

Object	Sub-index	Bit	Description	Parameter
6000H	1	Bit 0	Read input 0	8-bit word
		...	...	
		Bit 7	Read input 7	
	2	Bit 0	Read input 8	8-bit word
		...	...	
		Bit 3	Read input 11	
		Bit 4	Not used	
		...	...	
		Bit 7	Not used	
6100H	1	Bit 0	Read input 0	16-bit word
		...	...	
		Bit 11	Read input 11	
		Bit 12	Not used	
		...	...	
		Bit 15	Not used	
6200H	1	Bit 0	Write output 0	8-bit word
		...	...	
		Bit 7	Write output 7	
6300H	1	Bit 0	Write output 0	16-bit word
		...	...	
		Bit 7	Write output 7	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	

Discrete I/O Configuration Objects

The following table describes the different objects reserved for the I/O read/write operations of the OTB module:

Object	Sub-index	Bit	Description	Parameter
2000H	1	-	Filtering input 0	0 : None 1 : 3 ms (default value) 2 : 12 ms
	...		...	
	12	-	Filtering input 11	
6102H	1	Bit 0	Polarity input 0	0 : Normal input (default value) 1 : Negated input
		...	...	
		Bit 11	Polarity input 11	
		Bit 12	Not used	
		...	...	
		Bit 15	Not used	
6103H	1	Bit 0	Input mask 0	0 : Deactivated mask (default value) 1 : Activated mask
		...	...	
		Bit 11	Input mask 11	
		Bit 12	Not used	
		...	...	
		Bit 15	Not used	
6302H	1	Bit 0	Polarity output 0	0 : Normally Open output (NO) (default value) 1 : Normally Closed output (NC)
		...	...	
		Bit 7	Polarity output 7	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	
6306H	1	Bit 0	Fallback mode, output 0	0 : Maintain 1 : Fallback enabled (default value)
		...	...	
		Bit 7	Fallback mode, output 7	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	

Object	Sub-index	Bit	Description	Parameter
6307H	1	Bit 0	Fallback value, output 0	0 : Fallback to 0 (default value) 1 : Fallback to 1
		...	...	
		Bit 7	Fallback value, output 7	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	
6308H	1	Bit 0	Output 0 masking	0 : Maintain status 1 : Value of object 6300H (default value)
		...	...	
		Bit 7	Output 7 masking	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	

6.2 Specific Functions of the OTB Module

Introduction

This section describes the specific functions of the Advantys OTB module.

What's in this Section?

This section contains the following topics:

Specific Functions of the Advantys OTB Modules

Remote Fast Counter (RFC)

The Advantys OTB network interface module permits the use of a maximum of 2 fast counters. The functions RFC0 and RFC1 are allocated to inputs 18 and 19. These inputs can be used as standard discrete inputs if the function is not used.

Remote Very Fast Counter (RVFC)

The Advantys OTB network interface module permits the use of a maximum of 2 very fast counters. The functions RVFC0 and RVFC1 are allocated to inputs I0 to I3 and I4 to I7 respectively. These inputs can be used as standard discrete inputs if the function is not used.

Remote Pulse Generators (RPLS or RPWM)

The Advantys OTB network interface module permits the use of 2 pulse generators. The functions RPLS0/RPWM0 and RPLS1/RPWM1 are allocated to the outputs Q0 and Q1 respectively. These outputs can be used as standard discrete outputs if the function is not used.

Associated I/O and Functions

The I/Os associated with the pulse counters and generators are defined in the following table:

I/O	Very Fast counter 0 (RVFC0)	Very Fast counter 1 (RVFC1)	Fast counter 0 (RFC0)	Fast counter 1 (RFC1)	Pulse generator 0 (RPLS0/RPWM0)	Pulse generator 1 (RPLS1/RPWM1)
Input 0	X					
Input 1	X					
Input 2	X					
Input 3	X					
Input 4		X				
Input 5		X				
Input 6		X				
Input 7		X				
Input 8			X			
Input 9				X		
Output 0					X	
Output 1						X
Output 2	X					
Output 3	X					
Output 4		X				
Output 5		X				

Remote Fast Counter (RFC) Function

Introduction

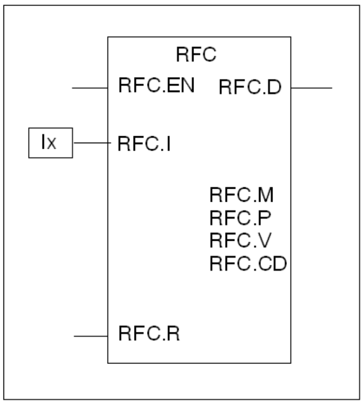
The remote fast counter (RFC) function can be used in up- or down-counting mode. It can count the pulses on the dedicated discrete inputs of frequencies up to 5 kHz.

Two remote fast counter functions are available. The fast counter functions RFC0 and RFC1 use the dedicated inputs I8 and I9 respectively. These inputs are not exclusively reserved for these functions, and may be used as standard inputs.

NOTE: The function representations are not pre-existing instructions in the programming software. They appear in these sections as graphical aides to understanding the parameters of these complex I/O functions.

Representation

The figure below shows a Remote Fast Counter (RFC) function.



Parameters

The following table shows the parameters for the Remote Fast Counter function.

Parameter	Function	Description
RFC.M	Counting Mode	Parameter used to select between: • 0: Disable function • Up counter • Down counter
RFC.P	Preset value	Threshold value to trigger the RFC.D Done bit and re-set RFC.V current value.
RFC.V	Current value	The current value increments or decrements according the counting mode selected. This value is between zero and the RFC.P preset value.
RFC.EN	Enter to enable	Validation of the RFC block operation.
RFC.R	Reset	Used to initialize the block. When set to 1, the current value is set to: • 0 if the block is configured in counting mode • RFC.P if the block is configured in downcounting mode
RFC.D	Done	Done switches to 1 if: • RFC.V reaches RFC.P in upcounting mode • RFC.V reaches zero in downcounting mode
RFC.CD	Reset Done	When set to 1, this bit is used to reset the RFC.D bit. If the user does not reset it to 0, the RFC.D bit is re-mains at 1.
RFC.I	Physical input	Input dedicated to up/down counting: • I8 for the RFC0 fast counter • I9 for the RFC1. fast counter

Operation

When the RFC function is configured to up-count, the current value is incremented by one when a rising edge appears at the dedicated input. When the preset value RFC.P is reached, the Done output bit RFC.D is set to 1 and the current value RFC.V is set to zero.

If the RFC function is configured to down-count, the current value is decreased by 1 when a rising edge appears at the dedicated input. When the value is zero, the Done output bit RFC.D is set to 1 and the current value RFC.P is set to the preset value.

Notes

The RFC function will only be activated after the RFC.R command is initialized and the RFC.EN input validated.

The selection or modification of the RFC.M counting mode will only be taken into account on activation of the RFC.R command.

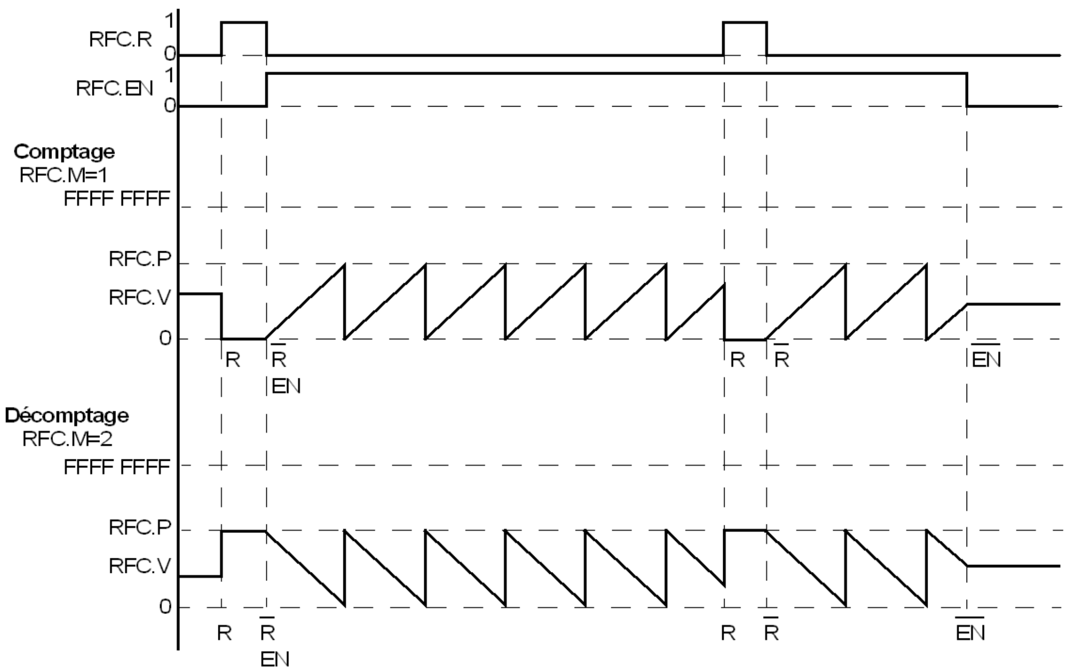
The RFC.P preset value modification is acknowledged at the end of the up counting or down counting cycle in progress without having to activate the RCF.R command.

Fallback Modes RFC.EM

The programmable fallback modes of the RFC function are as follows:
- Counter reset (equivalent of setting the RFC.R to 1)
- Set the current value of the RFC function counter (equivalent of setting RFC.EN to 0)
- Continue counting

Timing diagram

The timing diagram below illustrates the RFC function operation in up counting (RFC.M=1) and down counting (RFC.M=2) mode.



Configuration Objects of the Fast Counters (RFC)

At a Glance

The fast counters (RFC0 and RFC1) use the configuration objects defined below.

Objects 2300H to 2355H

Specific fast counter function 0 (RFC0).

Object	Sub-index	Parameter	Description	Format	Default value of the parameter
2300H	1	RFC.V	Current value	Word	0000H
2350H	1			DWord	0000 0000H
2301H	1	RFC.D	Bit [0]: ● upcounting: Preset value reached ● down counting: 0 reached	Word	0000H
2302H	1	RFC.EN RFC.R RFC.CD	Bit [0]: validation of the input EN Bit [1]: R (Reset) Bit [2]: reset of the RFC.D bit	Word	0000H
2303H	1	RFC.M	Counting mode: ● 0 : Not used ● 1 : Upcounter ● 2 : Down counter	Word	0000H
2304H	1	RFC.P	Preset value	Word	FFFFH
2354H	1			DWord	FFFF FFFFH
2305H	1	RFC.Delta	Delta value used to trigger PDO	Word	0000H
2355H	1			DWord	0000 0000H
2306H	1	Authorization of transmission from PDO	Bit 0: update of RFC.V Bit 1: Overrun Bit 2: update of RFC.V if superior to Delta	Byte	01H
2307H	1	RFC.EM	Fallback mode: ● 0 : Reset to zero of the counter ● 1 : Stop counting, save the last value read and freeze counter ● 2 : Continue counting	Word	0000H

The fast counter 1 (RFC1) specific function is identical to the fast counter 0 (RFC0) with the sub-index 2.

WARNING

DANGER: UNEXPECTED EQUIPMENT OPERATION RISK

In event transmission mode (modes 254 and 255) during the use of analog entries or RFC/RVFC counters, the use of DELTA or Inhibit Time is recommended.

If Delta or Inhibit Time are not used, the transmission of PDOs during each event may overload the bus and block the transmission of other crucial commands.

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

Remote Very Fast Counter (RVFC) function

Introduction

The Remote Very Fast Counter (RVFC) function can be configured to perform any of the following functions:
- Up/down counter
- 2-phase up/down counter
- Single up counter
- Single down counter
- Frequency meter

Two very fast counters are available. The RVFC function supports counting of dedicated discrete inputs from 0 to 20 kHz. Very fast counters RVFC0 and RVFC1 each use the I/O dedicated to these functions.

A Remote Very Fast Counter (RVFC) has a value range between 0 and 4,294,967,295.

Notes

The RVFC function will only be activated after the RVFC.R parameter is initialized and the RVFC.EN input enabled.

Selection or modification of the RVFC.M counting mode will only be taken into account on activation of the RVFC.R command.

Dedicated I/O Assignments

The Remote Very Fast Counter (RVFC) functions use dedicated inputs and outputs. These inputs and outputs are not exclusively reserved for these functions, and can be used as normal discrete I/O.

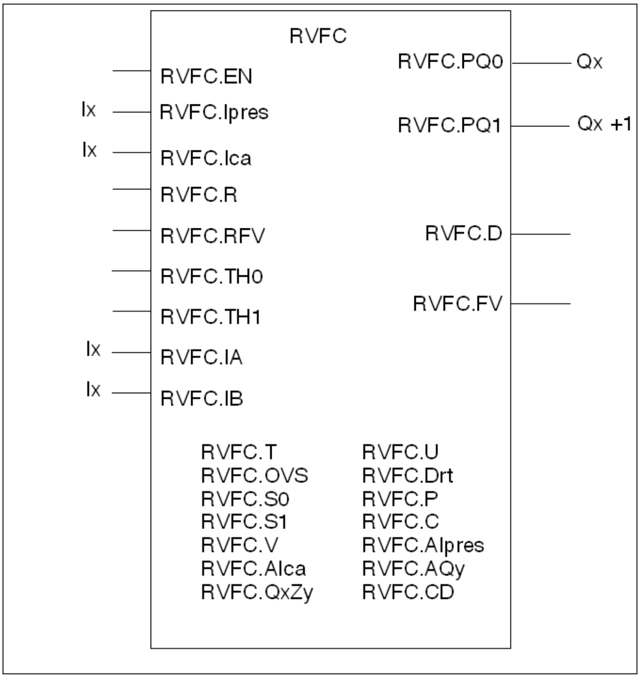
The table below summarizes the possible assignments:

		Main Inputs		**Auxiliary Inputs**		**Reflex Outputs**	
Operating mode		IA input	IB input	IPres (1)	Ica (1)	Output 0 (1)	Output 1 (1)
RVFC0	Up/down counter	I1 Pulse	I0 0=UP/1=Do wn	I2	I3	Q2	Q3
	Up/down 2-phase counter	I1 Phase A	I0 Phase B	I2	I3	Q2	Q3
	Single up counter	I1	Not used	I2	I3	Q2	Q3
	Single down counter	I1	Not used	I2	I3	Q2	Q3
	Frequency meter	I1	Not used	Not used	Not used	Not used	Not used
RVFC1	Up/down counter	I7 Pulse	I6 0=UP/1=DO	I5	I4	Q4	Q5
	Up/Down 2-phase counter	I7 Phase A	I6 Phase B	I5	I4	Q4	Q5
	Single up counter	I7	Not used	I5	I4	Q4	Q5
	Single down counter	I7	Not used	I5	I4	Q4	Q5
	Frequency meter	I7	Not used	Not used	Not used	Not used	Not used

Key:
(1) = Optional
IA input = Pulse input
IB input = Pulses or UP/Down
UP/Down = Up/down counting
Ipres = Preset input
Ica = Catch input
When not used by the function, the input or output remains a discrete I/O.

Representation

The figure below shows a Remote Very Fast Counter (RVFC) function.



NOTE: Qx depends on the type of Remote Very Fast Counter (RVFC) configured. For RVFC0, the physical reflex outputs are Q2 and Q3, For RVFC1, they are Q4 and Q5.

Parameters

The table shows the various parameters for the Remote Very Fast Counter (RVFC) function.

Parameter	Function	Description
RVFC.M	Counting mode	Parameter used to select between: • Not used • Up/down counter • 2-phase up/down counter • Up counter • Down counter • Frequency meter
RVFC.V	Current value	The current value increments or decrements according to the selected counting mode. This value can be set to the preset value (RVFC.P) using the preset input (RVFC.Ipres).
RVFC.Drt	Counting direction	This bit, which is only used in up/down counting mode, indicates the counting direction based on the previous current value: 0: Upcounting 1: Downcounting
RVFC.P	Preset value	The current value (RVFC.V) takes the preset value on an RVFC.Ipres input edge or on a RVFC.R counter reset in downcounting, up/down counting and two-phase modes. In upcounting and downcounting mode, 0 is forbidden. A change in the value will be taken into account at the end of the current cycle.
RVFC.C	Catch value	When the catch input (RVFC.Ica) is activated, the current value (RVFC.V) is stored in the catch value (RVFC.C). This function is only used in frequency meter mode.
RVFC.TH0	Threshold value S0	This parameter contains the value of threshold S0. This value must be lower than the value of threshold S1 (RVFC.TH1). This function is only used in frequency meter mode.
RVFC.TH1	Threshold value S1	This parameter contains the value of threshold S1. This value must be higher than the value of threshold S0 (RVFC.TH0). This function is only used in frequency meter mode.
RVFC.S0	Bit 0 threshold	This bit is set to 1 when the current value is >= the value of threshold S0 (RVFC.TH0). This function is only used in frequency meter mode.
RVFC.S1	Bit 1 threshold	This bit is set to 1 when the current value is >= the value of threshold S1 (RVFC.TH1). This function is only used in frequency meter mode.
RVFC.D	Done	The Done bit switches to 1 if: • RVFC.V reaches RVFC.P in upcounting mode • RVFC.V reaches zero in downcounting and up/down counting mode. The Done bit switches to 0 when the RVFC.R bit is activated or when Reset Done (RVFC.CD) =1.

Parameter	Function	Description
RVFC.CD	Reset Done	At state 1, this bit is used to reset the RVFC.D bit. This bit is processed depending on its level; if the user does not reset it to 0, the RVFC.D bit remains at 1.
RVFC.T	Frequency measure time base	Time base configuration element: ● 0 = 100 milliseconds ● 1 = 1 second This function is only used for the frequency measurement mode.
RVFC. Ipres	Preset physical input	On a rising edge, the current value (RVFC.V) is forced to the preset value. At state 0, up or down counting in progress.
RVFC. AIpres	Enable the Ipres input	Enables the preset value command.
RVFC.Ica	Physical catch input	On a rising edge, the current value (RVFC.V) is stored in the catch value (RVFC.C).
RVFC. AIca	Enable the Ica input	Enables the catch command.
RVFC.EN	Enable input	Activation of the RVFC function. At state 1, the current value (RVFC.V) is updated according to the pulses. At state 0, the current value (RVFC.V) is not updated according to the pulses.
RVFC.R	Reinitialization	The effect of this bit depends on the counting mode used when set to 1: ● Up/down counting, down counting and two-phase, the preset value (RVFC.P) is stored in the current value (RVFC.V). ● Upcounting, the current value is set to zero, ● Frequency meter, setting to zero of the current value and the valid frequency measurement bit (RVFC.FV). This function is also used to initialize the threshold outputs and acknowledge the threshold value modifications. RVFC.D bit reset to zero
RVFC.FV	Frequency measure valid	This bit is set to 1 when the frequency measurement is complete.
RVFC.RFV	Reset frequency measurement	This bit is set to 1 to reset the frequency measurement (RVFC.FV).
RVFC.Q0	Reflex output Qx	-
RVFC.AQ0	Activation of reflex output Qx	This parameter is used to activate the use of reflex output Qx.
RVFC.Q1	Reflex output Qx+1	-
RVFC.AQ1	Activation of reflex output Qx+1	This parameter is used to activate the use of reflex output Qx+1.
RVFC.Q0Z1	State of reflex output 0 in zone 1	State of reflex output 0 (RVFC.Q0) when the current value (RVFC.V) is less than the threshold S0 value (RVFC.TH0)

Parameter	Function	Description
RVFC.Q0Z2	State of reflex output 0 in zone 2	State of reflex output 0 (RVFC.Q0) when the current value (RVFC.V) is between the threshold S0 value (RVFC.TH0) and the threshold S1 value (RVFC.TH1) RVFC.TH0 ≤ RVFC.V ≤ RVFC.TH1.
RVFC.Q0Z3	State of reflex output 0 in zone 3	State of reflex output 0 (RVFC.Q0) when the current value (RVFC.V) is greater than the threshold S1 value (RVFC.TH1)
RVFC.Q1Z1	State of reflex output 1 in zone 1	State of reflex output 1 (RVFC.Q1) when the current value (RVFC.V) is less than the threshold S0 value (RVFC.TH0)
RVFC.Q1Z2	State of reflex output 1 in zone 2	State of reflex output 1 (RVFC.Q1) when the current value (RVFC.V) is between the threshold S0 value (RVFC.TH0) and the threshold S1 value (RVFC.TH1) RVFC.TH0 ≤ RVFC.V ≤ RVFC.TH1
RVFC.Q1Z3	State of reflex output 1 in zone 3	State of reflex output 1 (RVFC.Q1) when the current value (RVFC.V) is greater than the threshold S1 value (RVFC.TH1)

Description of Upcounting and Downcounting Functions

When the RVFC function is configured for upcounting, the current value is incremented by one when a rising edge appears at the dedicated input. When the preset value RVFC.P is reached, the RVFC.D Done bit is set to 1 and the RVFC.V current value is set to zero.

When the RFVC function is configured for downcounting, the current value is decreased by 1 when a rising edge appears on the dedicated input. When the value is zero, the Done bit RVFC.D is set to 1 and the current value RVFC.P equals the preset value.

Upcounting or downcounting operations are made on the rising edge of pulses, but only if the counting function has been activated (RVFC.EN). The RVFC.ICa and RVFC.IPres inputs are optional.

NOTE: These remarks do not apply in frequency meter mode.

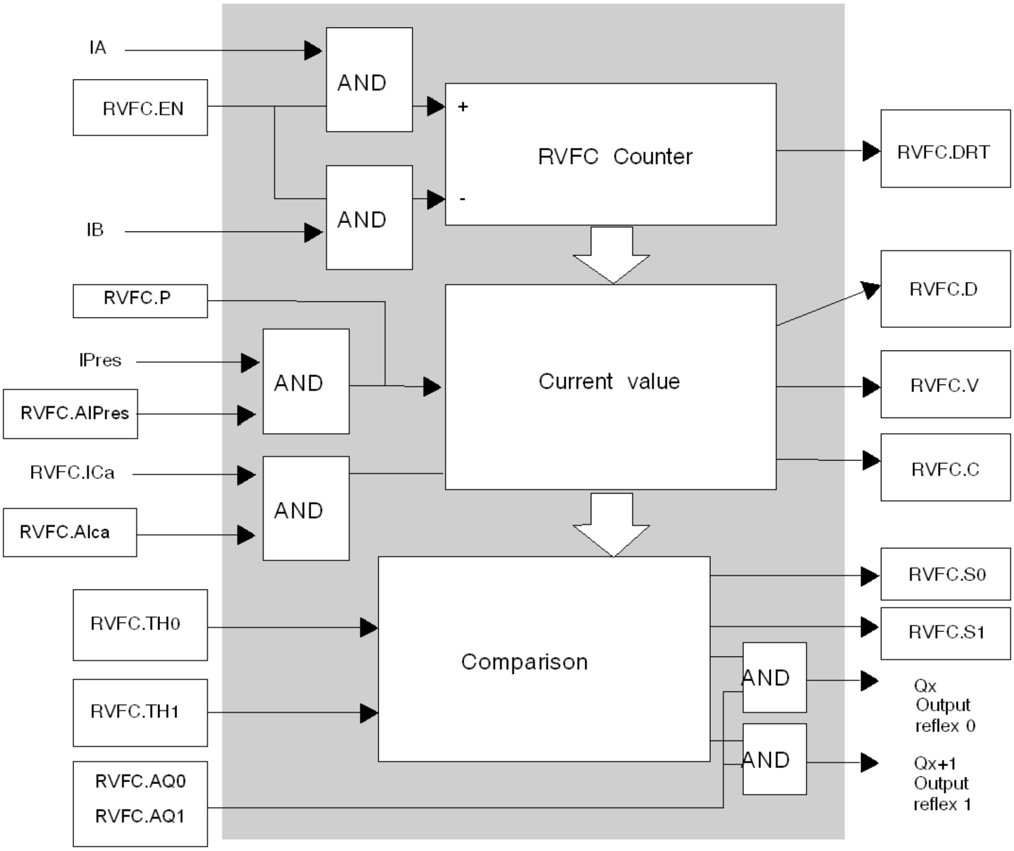
Notes on Function Outputs

The current value is compared with two threshold values (RVFC.TH0 and RVFC.TH1). The states of both threshold bits (RVFC.S0 and RVFC.S1) depend on the results of this comparison. State 1 if the current value is greater than or equal to the threshold value and 0 if the current value is less than the threshold value. Reflex outputs (if configured) are activated in accordance with these comparisons. It is possible to configure zero, one or two reflex outputs.

NOTE: These remarks do not apply in frequency meter mode.

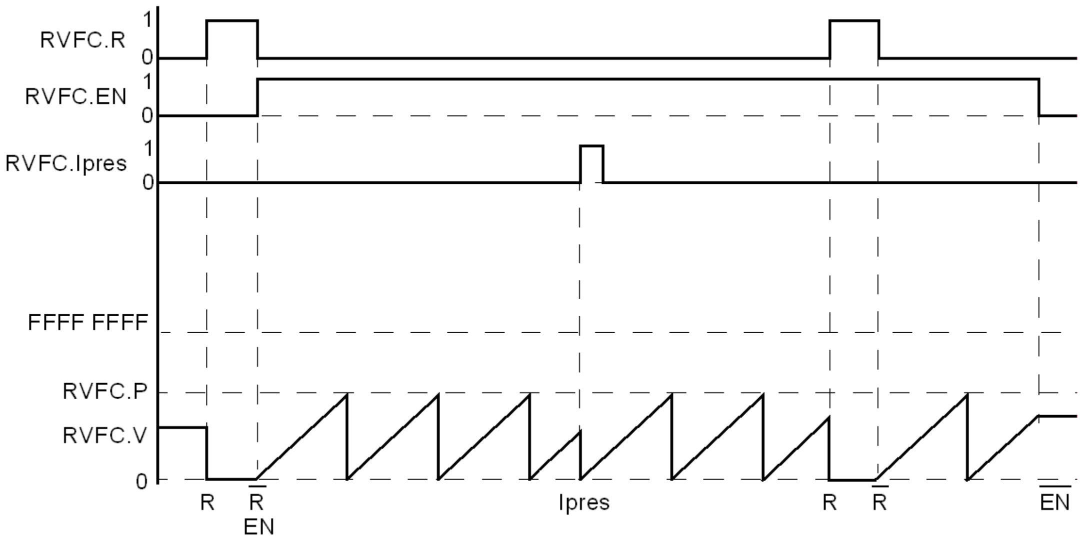
Counting Function Diagram

The following is a counting function diagram:



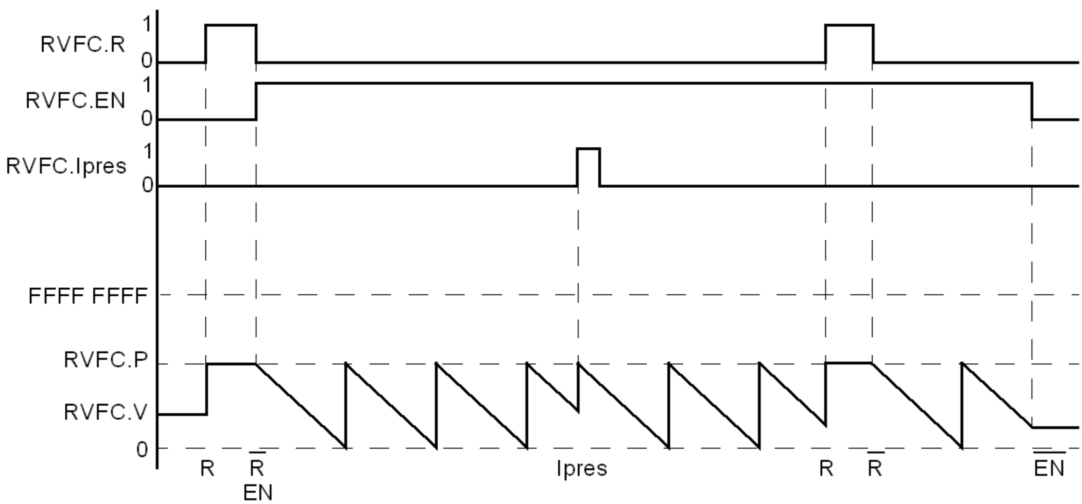
Use in Simple Upcounting Mode

Here is an example of an RVFC function timing diagram in simple upcounting mode (RVFC.M = 3):



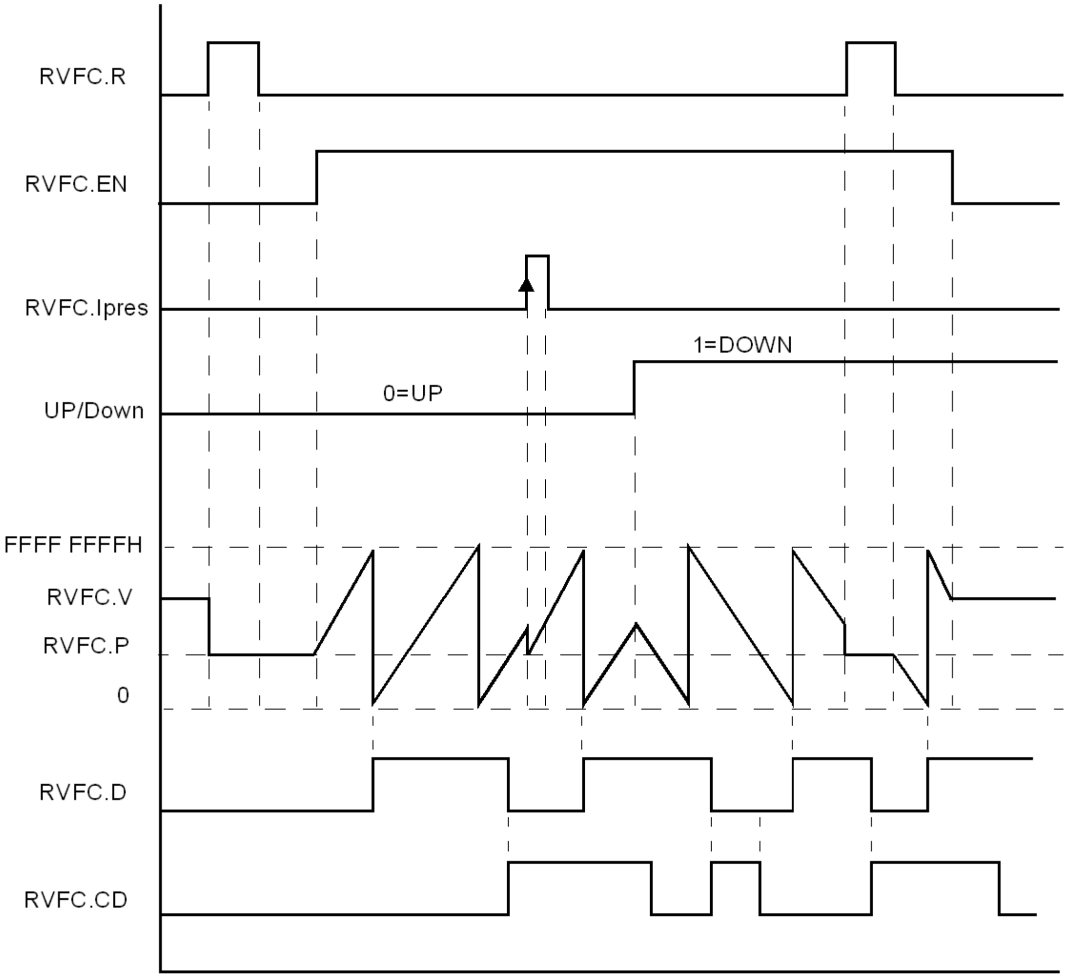
Use in Simple Downcounting Mode

Here is an example of an RVFC function timing diagram in simple downcounting mode (RVFC.M = 4):



Use in Up/Down Counting Mode

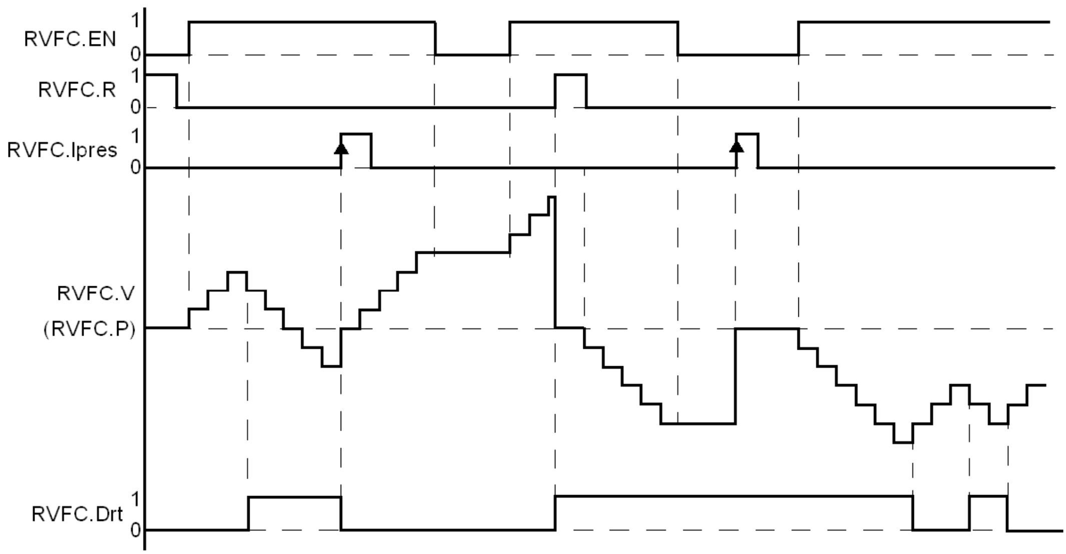
Here is an example of an RVFC function timing diagram in up/down counting mode (RVFC.M = 1):



Use in Two-phase Counting Mode

The two-phase counting mode is mainly dedicated to the use of incremental encoders. Channel A of the encoder is connected to the RVFC.IA input, channel B to the RVFC.IB input and channel Z (zero marker) to the RVFC.Ipres input.

The timing diagram below illustrates operation of the RVFC function in two-phase counting mode (RVFC.M=2).



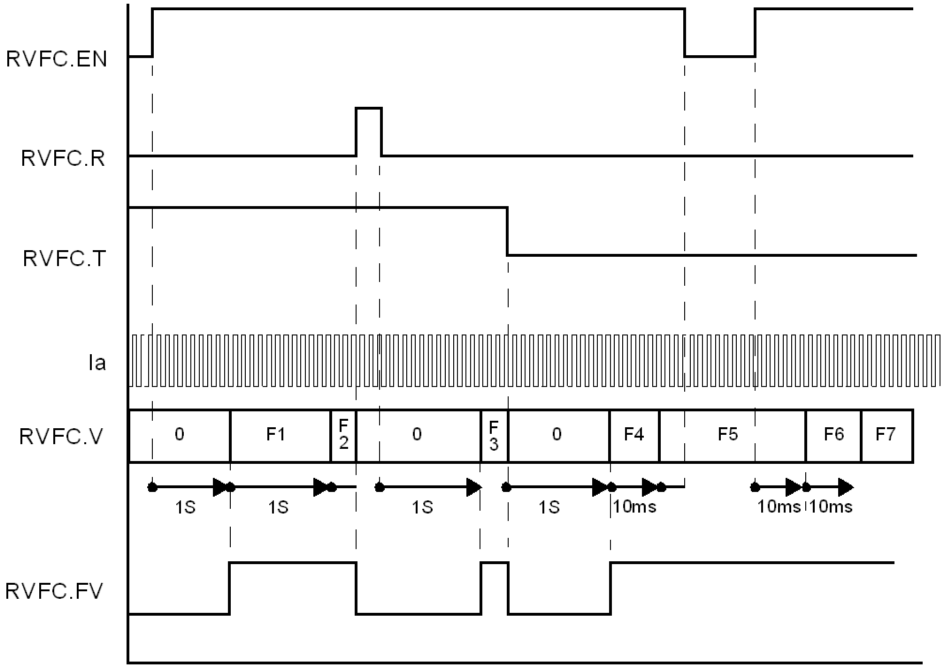
Frequency Meter Function Description

The frequency meter function of a RVFC is used to measure the frequency in Hz of a periodic signal on input IA. The range of frequencies which can be measured is between 1 Hz/10 Hz and 20 kHz. The user can choose between two time bases. This choice is made by a new object RVFC.T (Time base). 0 corresponds to a time base of 100 ms and 1 corresponds to a time base of 1 second.

Time base	Measurement Range	Accuracy	Update
100 ms	10 Hz to 20 kHz	0.05% for 20 kHz, 10% for 100 Hz	10 times per second
1 s	1 Hz to 20 kHz	0.005% for 20 kHz, 10% for 10 Hz	Once per second

Use in Frequency Meter Mode

The following is an example of a timing diagram of the use of RVFC in Frequency meter mode:



NOTE: The RVFC function uses a complete period to measure the frequency. RVFC.T time base modifications are acknowledged at the end of the current measurement cycle.

Fallback Modes

When the PLC stops or detects a communication error, the RVFC function may operate differently according to the programmed fallback mode.

The RVFC function programmable fallback modes are as follows:
- Very fast counter reset to zero (equivalent to setting RVFC.R to 1)
- Freeze the current counter value and stop the RVFC function counter (equivalent to setting RVFC.EN to 0)
- Continue counting

Configuration Objects for Remote Very Fast Counters (RVFC)

Presentation

Very fast counters (RVFC0 and RVFC1) use the configuration objects for the supported functions.

Objects 2400H to 2458H

Fast counter 0 (RFC0) dedicated function.

Object	Sub-Index	Parameter	Description	Format	Default Parameter Value
2400H	1	RVFC.V	Current value	Word	0000H
2450H	1			DWord	0000 0000H
2401H	1	RVFC.Drt	Bit [0]: Count direction 0: Up counter 1: Down counter	Word	0000H
		RVFC.D	Bit [1]: Output overshoot		
		RVFC.S0	Bit [2]: Threshold S0 reached. At state 1, it indicates that the current value is greater than S0.		
		RVFC.S1	Bit [3]: Threshold S1 reached. At state 1, it indicates that the current value is greater than S1.		
		RVFC.FV	Bit [4]: Measurement frequency valid		
2402H	1	RVFC.C	Catch value	Word	0000H
2452H	1			DWord	0000 0000H
2403H	1	RVFC.AQ0	Bit [0]: Activates the reflex output 0	Word	04C0H
		RVFC.AQ1	Bit [1]: Activates the reflex output 1		
		RVFC.T	Bit [2]: Frequency measure time base 0: 100 ms 1: 1 s		
		RVFC.AIpres	Bit [3]: Enables the preset input		

Object	Sub-Index	Parameter	Description	Format	Default Parameter Value
2403H	1	RVFC.AIca	Bit [4]: Enables the catch input	Word	
		RVFC.Q0Z1	Bit [5]: Status of reflex output 0 when the value is in zone 1		
		RVFC.Q0Z2	Bit [6]: Status of reflex output 0 when the value is in zone 2		
		RVFC.Q0Z3	Bit [7]: Status of reflex output 0 when the value is in zone 3		
		RVFC.Q1Z1	Bit [8]: Status of reflex output 1 when the value is in zone 1		
		RVFC.Q1Z2	Bit [9]: Status of reflex output 1 when the value is in zone 2		
		RVFC.Q1Z3	Bit [10]: Status of reflex output 1 when the value is in zone 3		
2404H	1	RVFC.P	Preset value	Word	FFFFH
2454H	1			DWord	FFFF FFFFH
2405H	1	RVFC.Delta	Delta value used to trigger the PDO	Word	0000H
2455H	1			DWord	0000 0000H
2406H	1	Authorize sending the PDO	Bit 0: Change of RFC.V	Byte	0001H
			Bit 1: Overrun		
			Bit 2: Change of RFC.V greater than Delta		
			Bit 3: RVFC.Ica		
			Bit 4: RVFC.S0		
			Bit 5: RVFC.S1		
2407H	1	RVFC.TH0	Threshold value S0	Word	0000H
2457H	1			DWord	0000 0000H
2408H	1	RVFC.TH1	Threshold value S1	Word	FFFFH
2458H	1			DWord	FFFF FFFFH
2409H	1	RVFC.M	Counting mode: ● 0: Not used ● 1: Up/down counter ● 2: 2-phase counter ● 3: Single up counter ● 4: Single down counter ● 5: Frequency meter	Word	0000H

Object	Sub-Index	Parameter	Description	Format	Default Parameter Value
240AH	1	RVFC.EM	Fallback mode: • 0: Counter reinitialization (0 if up counter, preset if down counter) • 1: Stops counting, saves the last value read and freezes counter • 2: Continues counting	Byte	00H
240BH	1	RVFC.EN RVFC.R RVFC.RFV RVFC.CD	Bit [0]: Enable input Bit [1]: Input set to 0 Bit [2]: Valid measurement frequency status set to 0 (RVFC.FV) Bit [3]: RVFC.D bit set to 0	Word	0001H

The description of very fast counter 1 (RVFC1) is identical to very fast counter 0 (RVFC0) with the sub-index 2.

⚠ WARNING

RISK OF UNINTENDED EQUIPMENT OPERATION

In event transmission mode (modes 254 and 255) when analog inputs or RFC/RVFC counters are being used, the use of DELTA or Inhibit Time is recommended.

If Delta or Inhibit Time are not used, sending PDOs on each event can overload the bus and block the transmission of other crucial commands.

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

Pulse Generator Output Function (RPLS)

Introduction

The RPLS function is used to generate a sequence of square wave signals.

There are two RPLS functions available. The RPLS0 function uses the dedicated output Q0 and the RPLS1 function uses the dedicated output Q1. The RPLS and RPWM functions share the same dedicated outputs. You must choose one or other of the functions for each output.

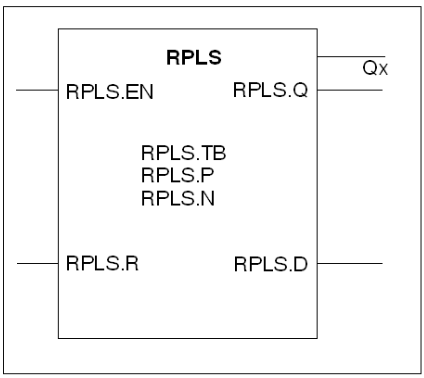
Notes

The function will only be activated after the RPLS.R input is initialized and the RPLS.EN input enabled.

Selection or modification of the RPLS.M counting mode will only be taken into account on activation of the RPLS.R command.

Representation

The figure below shows a pulse generator function block:



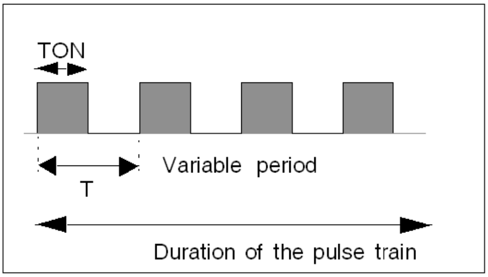
Parameters

The table below shows the different parameters of the RPLS pulse generator function.

Parameter	Designation	Description
RPLS.TB	Time base	This parameter can take the following time base values: • 0.127 ms (default value) • 0.508 ms • 10 ms • 1 s
RPLS.P	Period coefficient value	Authorized values for the preset period P: • 0: Function inactive • 0 < RPLS.P ≤ 255 with a time base of 0.127 ms or 0.508 ms • 1 < RPLS.P ≤ 65535 (FFFF H) with a time base of 10 ms or 1 s.
RPLS.N	Number of pulses	The number of pulses to be generated over a period T can be limited to 0 < RPLS.N ≤ 4, 294, 967, 295 (FFFF FFFF H). The default value is set to 0. To produce an unlimited number of pulses, set RPLS.N to zero.
RPLS.EN	Enable the pulse generator	Enables RPLS block operation. At state 0, this block is inhibited and the RPLS.Q output reset to zero.
RPLS.R	Reset at state 1	At state 1, outputs RPLS.Q and RPLS.D are set to 0. The number of pulses generated over a period T is set to 0.
RPLS.Q	Pulse generation in progress	At state 1, this indicates that the pulse signal is generated on the dedicated output channel.
RPLS.Qx	Dedicated outputs	Physical output to which the pulse train is applied.
RPLS.D	Pulse generation done output	At state 1, signal generation is complete. The desired number of pulses have been generated. This is reset by activating RPLS.R

Operation

The diagram below illustrates the RPLS function:



T= Period
TON = Time On

Duration of the pulse train: RPLS.N * T

The output signal period is set at the time of configuration, by selecting the time base RPLS.TB and the period coefficient value RPLS.P.

- T = RPLS.P * RPLS.TB
- TON = T/2 for time bases 0.127 ms and 0.508 ms
  = (RPLS.P * RPLS.TB)/2
- TON = [whole part (RPLS.P)/2] * RPLS.TB for the 10 ms to 1 s time bases.

NOTE:

- To obtain a good level of accuracy with time bases of 0.508 ms and 0.127 ms, it is advisable to have an RPLS.P $\geq$ 3.
- To obtain a good level of accuracy from the duty cycle with time bases of 10 ms and 1 s, it is advisable to have an RPLS.P $\geq$ 100 if P is odd.
- Any modification of the RPLS.P coefficient value is immediately taken into account.
- Where the RPLS function is used, the writing of the Q0 and Q1 outputs does not interrupt signal generation.

Period Ranges Available

The available period ranges are as follows:
- 0.127 ms to 32.38 ms in steps of 0.127 ms (30.9 Hz to 7.87 kHz)
- 0.508 ms to 129.54 ms in steps of 0.508 ms (7.72 Hz to 1.97 kHz)
- 20 ms to 5 min 27 s in steps of 10 ms
- 2s to 18 hrs 12 min 14 s in steps of 1 s

RPLS.EM Fallback Modes

When the PLC stops or detects a communication error, the RPLS function may operate differently according to the programmed fallback mode.

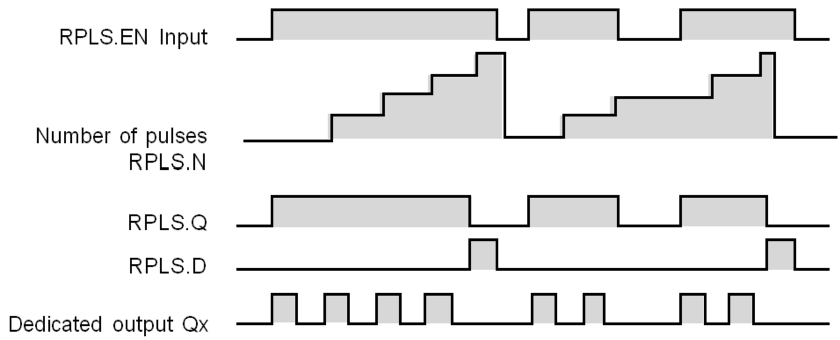
The programmable fallback modes of the RPLS function are as follows:

- Generator reset with output reset (equivalent of setting RPLS.R to 1)
- Stop at the end of the current pulse (equivalent of setting RPLS.EN to 0)
- Continue generating pulses

NOTE: The dedicated function fallback modes have priority and overwrite all other possible fallback conditions.

Example of a Pulse Generator

The illustration below represents a RPLS function pulse diagram.



Remote Pulse Width Modulator Function (RPWM)

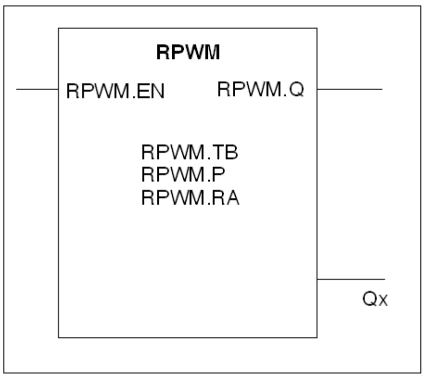
Introduction

The remote pulse width modulator (RPWM) function generates a rectangular signal on the dedicated outputs The signal duty cycle is variable.

There are two RPWM functions available. The RPWM0 function uses the dedicated output Q0 and the RPMW1 function uses the dedicated output Q1. The RPLS and RPWM functions share the same dedicated outputs. You must choose one or other of the functions for each output.

Representation

The following figure shows a remote pulse width modulator function:



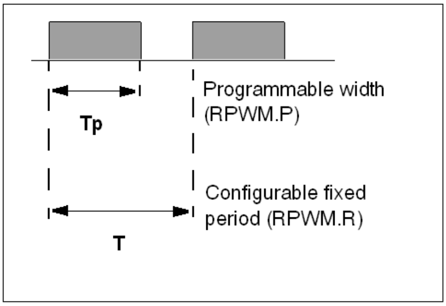
Parameters

The following table shows the different parameters of the remote pulse width modulator (RPWM) function:

Parameter	Designation	Description
RPWM.TB	Time base	This parameter can take the following time base values: ● 0.127 ms (default value) ● 0.508 ms ● 10 ms ● 1 s.
RPWM.P	Period coefficient value	Authorized values for the preset period P: ● 0 : Not used ● 0 < RPWM.P ≤ 255 with a time base of 0.127 ms or 0.508 ms ● 1 < RPWM.P ≤ 65535 (FFFF H) with a time base of 10 ms or 1 s
RPWM.RA	Duty cycle	This value establishes the percentage (0% ≤ RA ≤ 100%) of the signal of state 1 (activated) in a period T. The default value is 50%.
RPWM.EN	Validation of the pulse generator	Validation of the RPWM block operation. When set to 0, this block is inhibited and the RPLS.Q output reset to zero.
RPWM.Q	Generation of the pulses in progress	When set to 1, this indicates that the pulse signal is generated at the dedicated output channel.
RPWM.Qx	Dedicated outputs	Physical output to which the pulse train is applied.

Operation

The following diagram illustrates the RPWM function:



The output signal period is set on configuration, by selecting the time base RPWM.TB and the period coefficient value PWM.P. Modifying the RPWM.RA duty cycle in the program enables the signal width to be modulated.

Range of Periods

The coefficient value and the time base can be modified during configuration. They are used to set the signal period T = RPWM.P * TB. The range of periods available:
- 0.127 ms to 32.38 ms in steps of 0.127 ms (30.9 Hz to 7.87 kHz)
- 0.508 ms to 129.54 ms in steps of 0.508 ms (7.72 Hz to 1.97 kHz)
- 10 ms to 5 min 27 s in steps of 10 ms
- 2s at 18hrs 12min 14s in steps of 1 s

Pulse Modulation

Calculation of the Tp width: Tp = T * (RPWM.RA/100)

If the signal period is programmed to 500 ms, then
- Where the RPWM.RA ratio is set to 20%, the duration of the signal at state 1 is then: 20 % x 500 ms = 100 ms
- Where the RPWM.RA ratio is set to 50 % (duration = 250 ms)
- Where the RPWM.RA ratio is set to 80 % (duration = 400 ms)

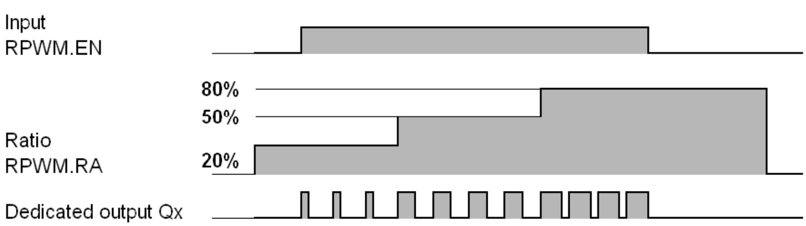
Fallback Modes RPWM.EM

When the PLC stops or detects a communication error, the RPWM function may operate differently according to the programmed fallback mode.

The programmable fallback modes of the RPWM function are as follows:
- Generator reset with output reset
- Stop at the end of the current interval (equivalent of setting RPWM.EN to 0)
- Continue generating pulses

Example of a Pulse Generator with Pulse Width Modulation

Below is an illustration of a pulse diagram for the RPWM function with varying duty cycles.



Configuration Objects of the Remote Pulse Generators (RPLS, RPWM)

At a Glance

The pulse generators (RPLS) and pulse width modulation generators (RPWM) use the configuration objects of the supported functions.

Objects 2500H to 2557H

Specific Objects of the Pulse Generator Function (RPLS and RPWM).

Object	Sub-index	Parameter	Description	Format	Default value of the parameter
2500H	1	RPLS.Q RPWM.Q RPLS.D RPWM.D	Bit [0]: Q output. At state 1, it indicates that the pulse signal is generated at the dedicated output configured. Bit [1]: Output D. When set to 1, signal generation is complete. The number of desired pulses has been reached.	Word	-
2501H	1	RPLS. EN RPWM.EN RPLS. R RPWM.R	Bit [0]: pulse generation. When set to 1, the pulse generation is produced on the dedicated output. When set to 0, the output is set to 0. Bit [1]: Reset to zero of the generator When set to 1, outputs Q and D are reset to 0. The number of pulses generated over a period T is reset to 0.	Word	0000H
2502H	1	RPLS.TB RPWM.TB	Time base: ● 0 : 0.127 ms ● 1 : 0.508 ms ● 2 : 10 ms ● 3 : 1 s	Word	0000H
2503H	1	RPLS.P RPWM.P	Preset period: P ● 0 : Not used ● 0< P < 255 with a time base of 0.127 ms or 0.508 ms ● 1< P < 65535 (FFFFH) with a time base of 10 ms or 1 s	Word	0001H
2504H	1	RPLS.M RPWM.M	Operating mode: ● 0 : Not used ● 1 : RPLS ● 2 : RPWM	Word	0000H
2505H	1	RPWM.RA	Duty cycle: $0 \leq RA \leq 100\%$. Duration of high status / Period	Word	0032H (50%)

Object	Sub-index	Parameter	Description	Format	Default value of the parameter
2506H	1	RPLS authorization to send PDO	Bit 0: Done Bit 1: No	Word	0001H
2507H	1	RPLS.N	Number of pulses: ● 0 : Unlimited number of pulses ● 1 ≤ N ≤ 4,294,967,295 (FFFF FFFFH)	Word	0000H
2557H	1			DWord	0000 0000H
2508H	1	RPLS.EM RPWM.EM	Fallback mode: ● 0 : Generator reset with zeroing of output, ● 1 : Stop at the end of current interval, ● 2 : Continue generating pulses.	Word	0000H

The description of the pulse generator 1 (RPLS/RPWM 1) is identical to the pulse generator 0 (RPLS/RPWM 0) with the sub-index 2.

6.3 Discrete I/O of Expansion Modules

Introduction

This section shows how discrete I/Os of expansion modules are used.

What's in this Section?

This section contains the following topics:

Description of Expansion Module Discrete I/Os

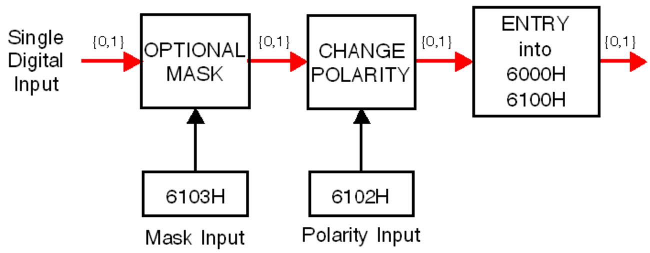
Discrete Inputs

Inputs are read by bytes (object 6000H) or by 16 bit words (object 6100H).

For each input, the following parameters may be modified:

- Filter mask (object 6103H)
- Polarity (object 6102H).

The status read on inputs is defined as follows:



Discrete Outputs

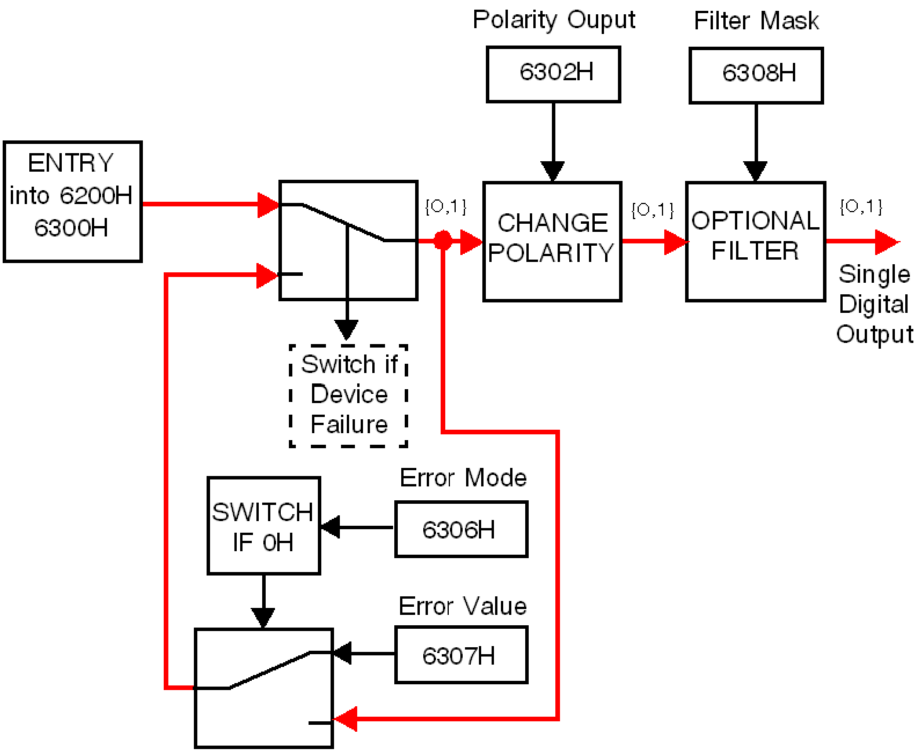
Outputs are written using bytes (object 6200H) or 16 bit words (object 6300H).

For each output, the following parameters may be modified:

- Polarity (object 6302H),
- Filter mask (object 6308H)
- Fallback mode (object 6306H),
- Fallback value (object 6307H).

In the event of an error (loss of communication with the master for example), the fallback mode is applied.

The state of the outputs is defined according to the configuration registers and equipment mode (according to the DS 401 device profile). See below:



Objects for TM2 DDI8DT and TM2 DAI8DT Expansion Modules

List of Objects

The sub-index number (N) depends on the location of the module in the island as well as the type of module *(see page 222)*.

The expansion module discrete inputs use the following objects:

Object	Sub-Index	Bit	Description	Value
6000H	N	Bit 0	Read input 0	-
		...	...	
		Bit 7	Read input 7	
6100H	N	Bit 0	Read input 0	-
		...	...	
		Bit 7	Read input 7	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	
6102H	N	Bit 0	Input 0 polarity	0: Normal input (default value) 1: Negated input
		...	...	
		Bit 7	Input 7 polarity	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	
6103H	N	Bit 0	Input 0 mask	0: Mask deactivated (default value) 1: Mask activated
		...	...	
		Bit 7	Input 7 mask	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	

Objects for TM2 DDI16DT and TM2 DDI16DK Expansion Modules

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

The expansion module discrete inputs use the following objects:

Object	Sub-Index	Bit	Description	Value
6000H	N	Bit 0	Read input 0	-
		...	...	
		Bit 7	Read input 7	
	N +1	Bit 0	Read input 8	-
		...	...	
		Bit 7	Read input 15	
6100H	N	Bit 0	Read input 0	-
		...	...	
		Bit 15	Read input 15	
6102H	N	Bit 0	Input 0 polarity	0: Normal input (default value) 1: Negated input
		...	...	
		Bit 15	Input 15 polarity	
6103H	N	Bit 0	Input 0 mask	0: Mask deactivated (default value) 1: Mask activated
		...	...	
		Bit 15	Input 15 mask	

Objects for the TM2 DDI32DK Expansion Module

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

The expansion module discrete inputs use the following objects:

Object	Sub-Index	Bit	Description	Value
6000H	N	Bit 0	Read input 0	-
		...	...	
		Bit 7	Read input 7	
	N +1	Bit 0	Read input 8	-
		...	...	
		Bit 7	Read input 15	
	N +2	Bit 0	Read input 16	-
		...	...	
		Bit 7	Read input 23	
	N +3	Bit 0	Read input 24	-
		...	...	
		Bit 7	Read input 31	
6100H	N	Bit 0	Read input 0	-
		...	...	
		Bit 15	Read input 15	
	N +1	Bit 0	Read input 16	-
		...	...	
		Bit 15	Read input 31	
6102H	N	Bit 0	Input 0 polarity	0: Normal input (default value) 1: Negated input
		...	...	
		Bit 15	Input 15 polarity	
	N +1	Bit 0	Input 16 polarity	
		...	...	
		Bit 15	Input 31 polarity	
6103H	N	Bit 0	Input 0 mask	0: Mask deactivated (default value) 1: Mask activated
		...	...	
		Bit 15	Input 15 mask	
	N +1	Bit 0	Input 16 mask	
		...	...	
		Bit 15	Input 31 mask	

Objects for TM2 DDO8TT, TM2 DDO8UT and TM2 DRA8RT Expansion Modules

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

The expansion module discrete outputs use the following objects:

Object	Sub-Index	Bit	Description	Value
6200H	N	Bit 0	Write output 0	-
		...	...	
		Bit 7	Write output 7	
6300H	N	Bit 0	Write output 0	-
		...	...	
		Bit 7	Write output 7	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	
6302H	N	Bit 0	Output 0 polarity	0: NO output (default value) 1: NC output
		...	...	
		Bit 7	Output 7 polarity	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	
6306H	N	Bit 0	Fallback mode, output 0	0: Maintain state 1: Fallback enabled (default value)
		...	...	
		Bit 7	Fallback mode, output 7	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	
6307H	N	Bit 0	Fallback value, output 0	0: Fallback to 0 (default value) 1: Fallback to 1
		...	...	
		Bit 7	Fallback value, output 7	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	

Object	Sub-Index	Bit	Description	Value
6308H	N	Bit 0	Output 0 mask	0: Maintain state 1: Output writing enabled (default value)
		...	...	
		Bit 7	Output 7 mask	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	

Objects for TM2 DDO16UK, TM2 DDO16TK and TM2 DRA16RT Expansion Modules

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

The expansion module discrete outputs use the following objects:

Object	Sub-Index	Bit	Description	Value
6200H	N	Bit 0	Write output 0	-
		...	...	
		Bit 7	Write output 7	
	N +1	Bit 0	Write output 8	-
		...	...	
		Bit 7	Write output 15	
6300H	N	Bit 0	Write output 0	-
		...	...	
		Bit 15	Write output 15	
6302H	N	Bit 0	Output 0 polarity	0: NO output (default value) 1: NC output
		...	...	
		Bit 15	Output 15 polarity	
6306H	N	Bit 0	Fallback mode, output 0	0: Maintain state 1: Fallback enabled (default value)
		...	...	
		Bit 15	Fallback mode, output 15	
6307H	N	Bit 0	Fallback value, output 0	0: Fallback to 0 (default value) 1: Fallback to 1
		...	...	
		Bit 15	Fallback value, output 15	
6308H	N	Bit 0	Output 0 mask	0: Maintain state 1: Output writing enabled (default value)
		...	...	
		Bit 15	Output 15 mask	

Objects for TM2 DDO32UK and TM2 DDO32TK Expansion Modules

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

The expansion module discrete outputs use the following objects:

Object	Sub-Index	Bit	Description	Value
6200H	N	Bit 0	Write output 0	-
		...	...	
		Bit 7	Write output 7	
	N +1	Bit 0	Write output 8	-
		...	...	
		Bit 7	Write output 15	
	N +2	Bit 0	Write output 16	-
		...	...	
		Bit 7	Write output 23	
	N +3	Bit 0	Write output 24	-
		...	...	
		Bit 7	Write output 31	
6300H	N	Bit 0	Write output 0	-
		...	...	
		Bit 15	Write output 15	
	N +1	Bit 0	Write output 16	-
		...	...	
		Bit 15	Write output 31	
6302H	N	Bit 0	Output 0 polarity	0: NO output (default value) 1: NC output
		...	...	
		Bit 15	Output 15 polarity	
	N +1	Bit 0	Output 16 polarity	0: NO output (default value) 1: NC output
		...	...	
		Bit 15	Output 31 polarity	

Object	Sub-Index	Bit	Description	Value
6306H	N	Bit 0	Fallback mode, output 0	0: Maintain state 1: Fallback enabled (default value)
		...	...	
		Bit 15	Fallback mode, output 15	
	N +1	Bit 0	Fallback mode, output 16	0: Maintain state 1: Fallback enabled (default value)
		...	...	
		Bit 15	Fallback mode, output 31	
6307H	N	Bit 0	Fallback value, output 0	0: Fallback to 0 (default value) 1: Fallback to 1
		...	...	
		Bit 15	Fallback value, output 15	
	N+1	Bit 0	Fallback value, output 16	0: Fallback to 0 (default value) 1: Fallback to 1
		...	...	
		Bit 15	Fallback value, output 31	
6308H	N	Bit 0	Output 0 mask	0: Maintain state 1: Output writing enabled
		...	...	
		Bit 15	Output 15 mask	
	N+1	Bit 0	Output 16 mask	0: Maintain state 1: Output writing enabled (default value)
		...	...	
		Bit 15	Output 31 mask	

Objects for the TM2 DMM8DRT Expansion Module

List of Discrete Input Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

The discrete inputs use the following objects:

Object	Sub-Index	Bit	Description	Value
6000H	N	Bit 0	Read input 0	-
		...	...	
		Bit 3	Read input 3	
		Bit 4	Not used	
		...	...	
		Bit 7	Not used	
6100H	N	Bit 0	Read input 0	-
		...	...	
		Bit 3	Read input 3	
		Bit 4	Not used	
		...	...	
		Bit 15	Not used	
6102H	N	Bit 0	Input 0 polarity	0: Normal input (default value) 1: Negated input
		...	...	
		Bit 3	Input 3 polarity	
		Bit 4	Not used	
		...	...	
		Bit 15	Not used	
6103H	N	Bit 0	Input 0 mask	0: Mask deactivated (default value) 1: Mask activated
		...	...	
		Bit 3	Input 3 mask	
		Bit 4	Not used	
		...	...	
		Bit 15	Not used	

List of Discrete Output Objects

The sub-index number (N) depends on the location of the module in the island.

The discrete outputs use the following objects:

Object	Sub-Index	Bit	Description	Value
6200H	N	Bit 0	Write output 0	-
		...	...	
		Bit 3	Write output 3	
		Bit 4	Not used	
		...	...	
		Bit 7	Not used	
6300H	N	Bit 0	Write output 0	-
		...	...	
		Bit 3	Write output 3	
		Bit 4	Not used	
		...	...	
		Bit 15	Not used	
6302H	N	Bit 0	Output 0 polarity	0: NO output (default value) 1: NC output
		...	...	
		Bit 3	Output 3 polarity	
		Bit 4	Not used	
		...	...	
		Bit 15	Not used	
6306H	N	Bit 0	Fallback mode, output 0	0: Maintain state 1: Fallback enabled (default value)
		...	...	
		Bit 3	Fallback mode, output 3	
		Bit 4	Not used	
		...	...	
		Bit 15	Not used	
6307H	N	Bit 0	Fallback value, output 0	0: Fallback to 0 (default value) 1: Fallback to 1
		...	...	
		Bit 3	Fallback value, output 3	
		Bit 4	Not used	
		...	...	
		Bit 15	Not used	

Object	Sub-Index	Bit	Description	Value
6308H	N	Bit 0	Output 0 mask	0: Maintain state 1: Output writing enabled (default value)
		...	...	
		Bit 3	Output 3 mask	
		Bit 4	Not used	
		...	...	
		Bit 15	Not used	

Objects for the TM2 DMM24DRF Expansion Module

List of Discrete Input Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

The mixed expansion module discrete inputs use the following objects:

Object	Sub-Index	Bit	Description	Value
6000H	N	Bit 0	Read input 0	-
		...	...	
		Bit 7	Read input 7	
	N +1	Bit 0	Read input 8	-
		...	...	
		Bit 7	Read input 15	
6100H	N	Bit 0	Read input 0	-
		...	...	
		Bit 15	Read input 15	
6102H	N	Bit 0	Input 0 polarity	0: Normal input (default value) 1: Negated input
		...	...	
		Bit 15	Input 15 polarity	
6103H	N	Bit 0	Input 0 mask	0: Mask deactivated (default value) 1: Mask activated
		...	...	
		Bit 15	Input 15 mask	

List of Discrete Output Objects

The sub-index number (N) depends on the location of the module in the island.

The mixed expansion module discrete outputs use the following objects:

Object	Sub-Index	Bit	Description	Value
6200H	N	Bit 0	Write output 0	-
		...	...	
		Bit 7	Write output 7	
6300H	N	Bit 0	Write output 0	-
		...	...	
		Bit 7	Write output 7	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	
6302H	N	Bit 0	Output 0 polarity	0: NO output (default value) 1: NC output
		...	...	
		Bit 7	Output 7 polarity	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	
6306H	N	Bit 0	Fallback mode, output 0	0: Maintain state 1: fallback value (default value)
		...	...	
		Bit 7	Fallback mode, output 7	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	
6307H	N	Bit 0	Fallback value, output 0	0: Fallback to 0 (default value) 1: Fallback to 1
		...	...	
		Bit 7	Fallback value, output 7	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	
6308H	N	Bit 0	Output 0 mask	0: Maintain state 1: Output writing enabled (default value)
		...	...	
		Bit 7	Output 7 mask	
		Bit 8	Not used	
		...	...	
		Bit 15	Not used	

6.4 Analog I/O of Expansion Modules

Introduction

This section shows how analog I/Os of expansion modules are used.

What's in this Section?

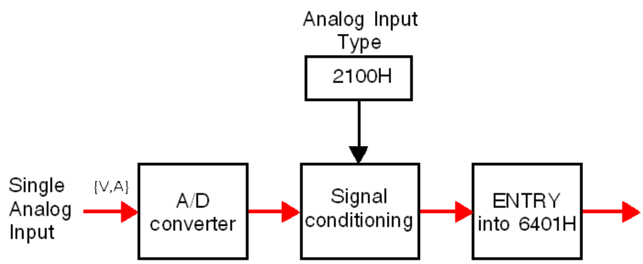
This section contains the following topics:

Description of Analog I/Os

Analog inputs

Analog inputs are read by 16 bit words. The value of each channel is contained in a sub-index of object 6401H.

The status read on inputs is defined as follows:



Use of the Delta Value (object 6426H)

The delta value is used to define a deadband, within which input signal value changes will not be reported. This is used to avoid a bus overload if data is transmitted about a change in value.

DELTA is expressed according to the unit configured in object 6426H.

Example

The last measurement value was 1000. By setting the delta value to 100, a new measurement value is only sent if it is below 900 or above 1100.

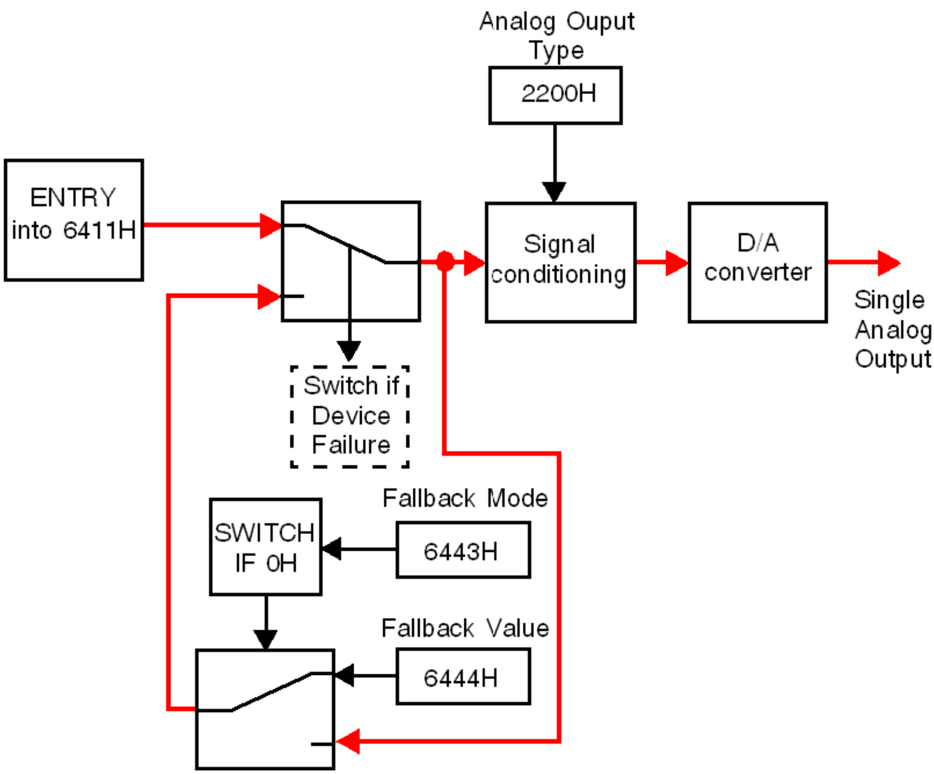
NOTE: If data is sent on a change of value, the object "Analog Input Global Interrupt Enable" (6423H) should be set to 1 by the user.

Analog Outputs

Outputs are written using 16 bit words. Each output byte is contained in the sub-index of object 6411H.

In the event of an error (loss of communication with the master for example), the fallback mode is applied.

The state applied to the output is defined as follows (according to the DS 401 device profile) :



NOTE: For further information on the various objects, go to the chapter *The Object Dictionary, page 219*.

Configuration Objects for Expansion Module Analog I/Os

At a Glance

The Analog I/Os of the expansion modules use the CANopen objects described in the following tables. The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

Objects Reserved for Analog I/Os

The analog expansion modules use different configuration objects according to their type.

The following table shows the possible values common to all channels and for each register.

Channel	Object	Description	Parameter
Channel X Input	2100H	Range	0 : Not used 1 : 0...20mA 2 : 4..20 mA. 3 : 0...10V 4 : +/- 10V 5 : Thermo K 6 : Thermo J 7 : Thermo T 8 : PT100 9 : PT1000 10 : NI100 11 : NI1000 12 : Reserved 13 : NTC / CTN 14 : PTC / CTP
	2101H	Unit	0 : Normal 1 : Customized 2 : Celsius (0.1°C) 3 : Fahrenheit (0.1°F) 4 : Resistance (ohms)
	2102H	Minimum value (where 2101H = 1)	Min.
	2103H	Maximum value (where 2101H = 1)	Max.
	2104H	R0 in NTC or high threshold in PTC	R0
	2105H	T0 in NTC or low threshold in PTC	T0
	2106H	Sensitivity in NTC or ignored in PTC	B
	6401H	Read value	

Channel	Object	Description	Parameter
	6421H	Reason for transmission from PDO	0 : High threshold exceeded 1 : Low threshold exceeded 2 : Delta exceeded
	6422H	Number of channel that caused the transmission from PDO	The active bit(s) indicate(s) which channel(s) caused the event
	6423H	Authorization of transmission from PDO	0 : Change in value 1 : Event **Note** Common to all the channels.
	6424H	High threshold	7FFFH
	6425H	Low threshold	0
	6426H	Delta value	0
Channel X Output	2200H	Range	0 : Not used 1 : 0...20mA 2 : 4..20 mA. 3 : 0...10V 4 : +/- 10V
	2201H	Unit	0 : Normal 1 : Customized
	2202H	Minimum value (where 2201H = 1)	Min.
	2203H	Maximum value (where 2201H = 1)	Max.
	6411H	Write value	
	6443H	Fallback mode	0 : Maintain state 1 : Fallback activated
	6444H	Fallback value	Fallback value

Irrespective of the analog module, the inputs are read on the 6401H object.

Irrespective of the analog module, the outputs are written to the 6411H object.

⚠ WARNING

RISK OF UNEXPECTED EQUIPMENT OPERATION

In event transmission mode (modes 254 and 255) during the use of analog entries or RFC/RVFC counters, the use of DELTA or Inhibit Time is recommended.

If Delta or Inhibit Time are not used, the transmission of PDOs during each event may overload the bus and block the transmission of other crucial commands.

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

Objects for the TM2 AMI2HT Expansion Module

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
Channel 0 (input)	2100H	N	Range	0: Not used 2: 4 ... 20 mA 3: 0 ... 10 V	0
	2101H	N	Unit	0: Normal 1: Customized	1
	2102H	N	Minimum value (if 2101H = 1)	Min.	0
	2103H	N	Maximum value (if 2101H = 1)	Max.	7FFFH
	6401H	N	Read value	Input	0
	6421H	N	Reason for sending the PDO	Bit 0: High threshold exceeded Bit 1: Low threshold exceeded Bit 2: Delta exceeded	7
	6422H	1	Number of channel that caused the PDO to be sent	An active bit indicates which channel caused the event	0
	6423H	0	Authorize sending the PDO	0: Change in value 1: Event	0
	6424H	N	High threshold		7FFFH
	6425H	N	Low threshold		0
	6426H	N	Delta value		0
Channel 1 (input)	Description identical to input V0 with the sub-index N+1.				

Objects for the TM2 AMO1HT Expansion Module

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
Channel 0 (output)	2200H	N	Range	0: Not used 2: 4...20 mA 3: 0...10 V	0
	2201H	N	Unit	0: Normal 1: Customized	1
	2202H	N	Minimum value (if 2201H = 1)	Min.	0
	2203H	N	Maximum value (if 2201H = 1)	Max.	7FFFH
	6411H	N	Write value	Output	0
	6443H	N	Fallback mode	0: Maintain state 1: Fallback activated	1
	6444H	N	Fallback value	Fallback value	0

Objects for the TM2 AMM3HT Expansion Module

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
Channel 0 (input)	2100H	N	Range	0: Not used 2: 4... 20 mA 3: 0...10 V	0
	2101H	N	Unit	0: Normal 1: Customized	1
	2102H	N	Minimum value (if 2101H = 1)	Min.	0
	2103H	N	Maximum value (if 2101H = 1)	Max.	7FFFH
	6401H	N	Read value	Input	0
	6421H	N	Reason for sending the PDO	Bit 0: High threshold exceeded Bit 1: Low threshold exceeded Bit 2: Delta exceeded	7
	6422H	1	Number of channel that caused the PDO to be sent		0
	6423H	0	Authorize sending the PDO	0: Change in value 1: Event	0
	6424H	N	High threshold		7FFFH
	6425H	N	Low threshold		0
	6426H	N	Delta value		0
Channel 1 (input)	Description identical to input V0 with the sub-index N+1.				
Channel 2 (outputs)	2200H	N	Range	0: Not used 2: 4...20 mA 3: 0...10 V	0
	2201H	N	Unit	0: Normal 1: Customized	1
	2202H	N	Minimum value (if 2201H = 1)	Min.	0
	2203H	N	Maximum value (if 2201H = 1)	Max.	7FFFH
	6411H	N	Write value	Output	0
	6443H	N	Fallback mode	0: Maintain state 1: Fallback activated	1
	6444H	N	Fallback value	Fallback value	0

Objects for the TM2 AMM6HT Expansion Module

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
Channel 0 (input)	2100H	N	Range	0: Not used 2: 4...20 mA 3: 0...10 V	0
	2101H	N	Unit	0: Normal 1: Customized	1
	2102H	N	Minimum value (if 2101H = 1)	Min.	0
	2103H	N	Maximum value (if 2101H = 1)	Max.	7FFFH
	6401H	N	Reading the value	Input	0
	6421H	N	Reason for sending the PDO	Bit 0: High threshold exceeded Bit 1: Low threshold exceeded Bit 2: Delta exceeded	7
	6422H	1	Number of channel that caused the PDO to be sent		0
	6423H	0	Authorize sending the PDO	0: Change in value 1: Event	0
	6424H	N	High threshold		7FFFH
	6425H	N	Low threshold		0
	6426H	N	Delta value		0
Channel 1 (input)	Description identical to the V0 input with the sub-index N+1.				
Channel 2 (input)	Description identical to the V0 input with the sub-index N+2.				
Channel 3 (input)	Description identical to the V0 input with the sub-index N+3.				

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
Channel 0 (output)	2200H	N	Range	0: Not used 2: 4...20 mA 3: 0...10 V	0
	2201H	N	Unit	0: Normal 1: Customized	1
	2202H	N	Minimum value (if 2201H = 1)	Min.	0
	2203H	N	Maximum value (if 2201H = 1)	Max.	7FFFH
	6411H	N	Writing the value	Output	0
	6443H	N	Fallback mode	0: Maintain state 1: Fallback activated	1
	6444H	N	Fallback value	Fallback value	0
Channel 1 (output)	Description identical to the V0 output with the sub-index N+1.				

Objects for the TM2 ALM3LT Expansion Module

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
V0 (input)	2100H	N	Range	0: Not used 5: Thermo K 6: Thermo J 7: Thermo T 8: PT100	0
	2101H	N	Unit	0: Normal 1: Customized 2: Celsius (0.1°C) 3: Fahrenheit (0.1°F)	1
	2102H	N	Minimum value (if 2101H = 1)	Min.	0
	2103H	N	Maximum value (if 2101H = 1)	Max.	7FFFH
	6401H	N	Read value	Input	0
	6421H	N	Reason for sending the PDO	Bit 0: High threshold exceeded Bit 1: Low threshold exceeded Bit 2: Delta exceeded	7
	6422H	1	Number of channel that caused the PDO to be sent		0
	6423H	0	Authorize sending the PDO	0: Change in value 1: Event	0
	6424H	N	High threshold		7FFFH
	6425H	N	Low threshold		0
	6426H	N	Delta value		0
V1 (input)	Description identical to input V0 with the sub-index N+1.				

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
V2 (outputs)	2200H	N	Range	0: Not used 2: 4...20 mA 3: 0...10 V	0
	2201H	N	Unit	0: Normal 1: Customized	1
	2202H	N	Minimum value (if 2201H = 1)	Min.	0
	2203H	N	Maximum value (if 2201H = 1)	Max.	7FFFH
	6411H	N	Write value	Output	0
	6443H	N	Fallback mode	0: Maintain state 1: Fallback activated	1
	6444H	N	Fallback value	Fallback value	0

Objects for the TM2 AVO2HT Analog Expansion Module

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
V0 (outputs)	2200H	N	Range	0: Not used 4: +/- 10 V	0
	2201H	N	Unit	0: Normal 1: Customized	1
	2202H	N	Minimum value (if 2201H = 1)	Min.	8000H
	2203H	N	Maximum value (if 2201H = 1)	Max.	7FFFH
	6411H	N	Write value	Output	0
	6443H	N	Fallback mode	0: Maintain state 1: Fallback activated	1
	6444H	N	Fallback value	Fallback value	0
V1 (outputs)	Description identical to the V0 output with the sub-index N+1.				

Objects for the TM2 AMI2LT Expansion Module

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
V0 (input)	2100H	N	Range	0: Not used 5: Thermo K 6: Thermo J 7: Thermo T	0
	2101H	N	Unit	0 : Normal 1: Customized 2: Celsius (0.1°C) 3: Fahrenheit (0.1°F)	1
	2102H	N	Minimum value (if 2101H = 1)	Min.	0
	2103H	N	Maximum value (if 2101H = 1)	Max.	7FFFH
	6401H	N	Reading the value	Input	0
	6421H	N	Reason for sending the PDO	Bit 0: High threshold exceeded Bit 1: Low threshold exceeded Bit 2: Delta exceeded	7
	6422H	1	Number of channel that caused the PDO to be sent		0
	6423H	0	Authorize sending the PDO	0: Change in value 1: Event	0
	6424H	N	High threshold		7FFFH
	6425H	N	Low threshold		0
	6426H	N	Delta value		0
V1 (input)	Description identical to the V0 input with the sub-index N+1.				

Objects for the TM2 AMI4LT Analog Expansion Module

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

NOTE: All inputs must be the same type (voltage, current or temperature).

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
V0 input	2100H	N	Range	0: Not used 2: 4...20 mA 3: 0...10 V 8: PT100 9: PT1000 10: NI100 11: NI1000	0
	2101H	N	Unit	0: Normal 1: Customized 2: Celsius (0.1°C) 3: Fahrenheit (0.1°F) 4: Resistance (ohms)	1
	2102H	N	Minimum value (if 2101H = 1)	Min.	0
	2103H	N	Maximum value (if 2101H = 1)	Max.	7FFFH
	6401H	N	Read value	Input	0
	6421H	N	Reason for sending the PDO	Bit 0: High threshold exceeded Bit 1: Low threshold exceeded Bit 2: Delta exceeded	7
	6422H	1	Number of channel that caused the PDO to be sent		0
	6423H	0	Authorize sending the PDO	0: Change in value 1: Event	0
	6424H	N	High threshold		7FFFH
	6425H	N	Low threshold		0
	6426H	N	Delta value		0
V1 (input)	Description identical to the V0 input with the sub-index N+1.				
V2 (input)	Description identical to the V0 input with the sub-index N+2.				
V3 (input)	Description identical to the V0 input with the sub-index N+3.				

Objects for the TM2 AMI8HT Analog Expansion Module

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

NOTE: All channels used must be the same type (voltage or current).

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
V0 (input)	2100H	N	Range	0: Not used 1: 0 ... 20 mA 3: 0...10 V	0
	2101H	N	Unit	0: Normal 1: Customized	1
	2102H	N	Minimum value (if 2101H = 1)	Min.	0
	2103H	N	Maximum value (if 2101H = 1)	Max.	7FFFH
	6401H	N	Read value	Input	0
	6421H	N	Reason for sending the PDO	Bit 0: High threshold exceeded Bit 1: Low threshold exceeded Bit 2: Delta exceeded	7
	6422H	1	Number of channel that caused the PDO to be sent		0
	6423H	0	Authorize sending the PDO	0: Change in value 1: Event	0
	6424H	N	High threshold		7FFFH
	6425H	N	Low threshold		0
	6426H	N	Delta value		0
V1 (input)	Description identical to the V0 input with the sub-index N+1.				
V2 (input)	Description identical to the V0 input with the sub-index N+2.				
V3 (input)	Description identical to the V0 input with the sub-index N+3.				
V4 (input)	Description identical to the V0 input with the sub-index N+4.				
V5 (input)	Description identical to the V0 input with the sub-index N+5.				
V6 (input)	Description identical to the V0 input with the sub-index N+6.				
V7 (input)	Description identical to the V0 input with the sub-index N+7.				

Objects for the TM2 ARI8HT Analog Expansion Module

NTC Probe

The temperature (T) varies in relation to the resistance (R) according to the equation below:

$$T(R) = \frac{1}{\frac{1}{T_0} + \frac{1}{B}\ln\left[\frac{R}{R_0}\right]}$$

Where:

- T = temperature measured by the probe, in Kelvin (object 6401H)
- R = physical value of the resistance in Ohm
- R_0 = reference resistance in Ohms at temperature T_0 (object 2104H)
- T_0 = reference temperature in Kelvin (object 2105H)
- B = sensitivity of NTC probe in Kelvin (object 2106H)

R_0, T_0 and B should at least be equal to 1.

If the resistance is selected as a unit, the displayed value must be equal to the resistance of the probe.

NOTE: 25°C = 77°F = 298.15°K

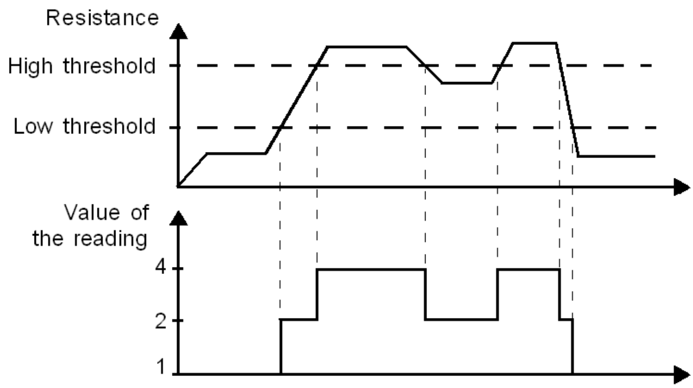
PTC Probe

- R_0 = high threshold (object 2104H)
- T_0 = low threshold (object 2105H)

Read value = 1 if resistance value < T_0

Read value = 2 if T_0 < resistance value < R_0

Read value = 4 if resistance value > R_0



List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
V0 (input)	2100H	N	Range	0: Not used 13: NTC 14: PTC	0
	2101H	N	Unit	0: Normal 1: Customized 2: Celsius (0.1°C) 3: Fahrenheit (0.1°F) 4: Resistance (ohms)	1
	2102H	N	Minimum value (if 2101H = 1)	Min.	0
	2103H	N	Maximum value (if 2101H = 1)	Max.	7FFFH
	2104H	N	R_0 in NTC or high threshold in PTC	R_0	014AH
	2105H	N	T_0 in NTC or low threshold in PTC	T_0 (0.01°K)	7477H
	2106H	N	Sensitivity in NTC or ignored in PTC	B (0.01°K)	0DF1H
	6401H	N	Reading the value	Input	0
	6421H	N	Reason for sending the PDO	Bit 0: High threshold exceeded Bit 1: Low threshold exceeded Bit 2: Delta exceeded	7
	6422H	1	Number of channel that caused the PDO to be sent		0
	6423H	0	Authorize sending the PDO	0: Change in value 1: Event	0
	6424H	N	High threshold		7FFFH
	6425H	N	Low threshold		0
	6426H	N	Delta value		0
V1 (input)	Description identical to the V0 input with the sub-index N+1.				
V2 (input)	Description identical to the V0 input with the sub-index N+2.				
V3 (input)	Description identical to the V0 input with the sub-index N+3.				
V4 (input)	Description identical to the V0 input with the sub-index N+4.				

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
V5 (input)	Description identical to the V0 input with the sub-index N+5.				
V6 (input)	Description identical to the V0 input with the sub-index N+6.				
V7 (input)	Description identical to the V0 input with the sub-index N+7.				

Objects for TM2 ARI8LT and TM2 ARI8LRJ Analog Expansion Modules

List of Objects

The sub-index number (N) depends on the location of the module in the island *(see page 222)*.

Channel	Object	Sub-Index	Description	Parameter	Default Parameter Value
V0 (input)	2100H	N	Range	0: Not used 8: PT100 9: PT1000	0
	2101H	N	Unit	0 : Normal 1: Customized 2: Celsius (0.1°C) 3: Fahrenheit (0.1°F)	1
	2102H	N	Minimum value (if 2101H = 1)	Min.	0
	2103H	N	Maximum value (if 2101H = 1)	Max.	7FFFH
	6401H	N	Read value	Input	0
	6421H	N	Reason for sending the PDO	Bit 0: High threshold exceeded Bit 1: Low threshold exceeded Bit 2: Delta exceeded	7
	6422H	1	Number of channel that caused the PDO to be sent		0
	6423H	0	Authorize sending the PDO	0: Change in value 1: Event	0
	6424H	N	High threshold		7FFFH
	6425H	N	Low threshold		0
	6426H	N	Delta value		0
V1 (input)	Description identical to the V0 input with the sub-index N+1.				
V2 (input)	Description identical to the V0 input with the sub-index N+2.				
V3 (input)	Description identical to the V0 input with the sub-index N+3.				
V4 (input)	Description identical to the V0 input with the sub-index N+4.				
V5 (input)	Description identical to the V0 input with the sub-index N+5.				
V6 (input)	Description identical to the V0 input with the sub-index N+6.				
V7 (input)	Description identical to the V0 input with the sub-index N+7.				

Software Tools

7

Introduction

This chapter deals with the installation of an Advantys OTB island with the PL7 or Unity software workshop and Advantys Configuration Tool (FTX ES0•).

What's in this Chapter?

This chapter contains the following sections:

7.1 Introduction to Software Tools

Introduction

General

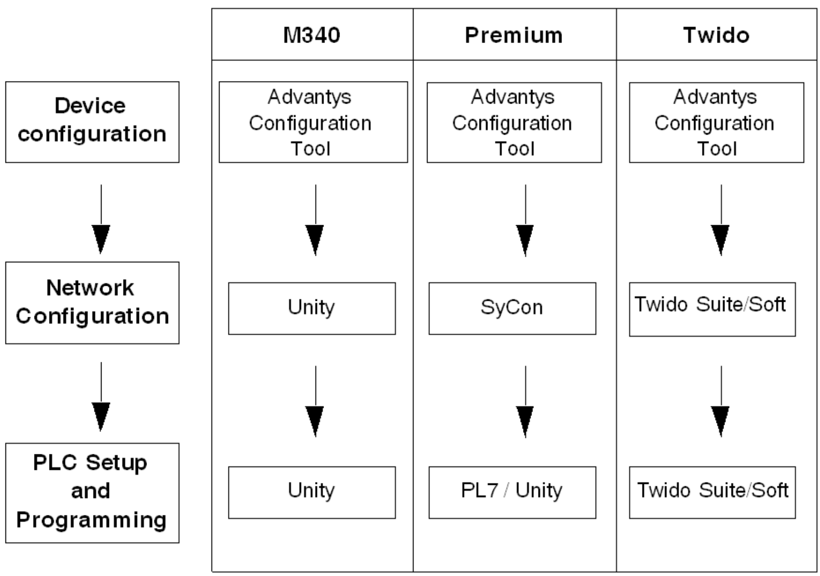
The products in the Advantys range must be configured to be able to operate correctly on the network. There are three stages in the configuration process:

- Configuration of the Advantys devices and the desired parameters.
- Configuration of the network (master and slaves).
- PLC setup and programming: I/O, startup of the network and subsequent use.

NOTE: For more information, please consult the appropriate documentation for the other network devices that may be required, the Advantys Configuration Tool online help (FTX ES 0•), the PLC manual etc.

Software Tools

The software to be used depends on the PLC software workshop. Certain PLC software workshops can configure the network. The following diagram shows the software to be used for three Telemecanique PLC software workshops:



NOTE: With Twido Suite, Advantys Configuration Tool is run directly by Twido Suite to create or modify an island.

Configuring Advantys Devices

The first phase is accomplished by using the Advantys Configuration Tool (FTX ES 0•). This tool is used to set the I/O parameters and functions and to generate the configuration files and image files (.dib) required by the master.

Configuration Files

There are two types of configuration file:

- EDS (Electronic Data Sheet) files, which define the structure of the data available in a splitter box (see the object dictionary).
- DCF (Device Configuration File) files which, in addition to the information contained in an EDS file, also contain parameter-setting data (see standard CAN-CiA DS 306).

NOTE: For further information on configuration file creation, please refer to the user manual or to the Advantys Configuration Tool online help.

Network Configuration

This phase may be carried out by a specific software application (e.g. SyCon) or by certain PLC software workshops (e.g. Unity, Twido Suite, etc). This phase involves integrating all devices into the network, and defining the network (master configuration) so as to create a functional network.

PLC Setup and Programming

This phase is carried out by the operator, via the PLC software workshop.

Software Installation

Before installing the software, please refer to the relevant manuals.

7.2 Product Configuration

Introduction

This section describes the tools and operating modes that generate the EDSs and DCFs of the Advantys range of devices using the Advantys Configuration Tool (FTX ES 0•).

The software generates one file per island. An island represents a node on the network with a separate network address. An island can correspond to:

- An OTB module (with or without expansion modules),
- An FTB splitter box,
- A modular FTM splitter (module with or without splitters).

What's in this Section?

This section contains the following topics:

Characteristics of an EDS File

Description

The EDS file describes all configurable objects for CANopen products. These configurable objects are used to identify the product and specify the appropriate behavior. The parameters of an EDS file contain all the important information relating to the product. For example:

- The product type
- The manufacturer
- The identification of the vendor
- The item number
- The software version
- The hardware version
- The details of all the configurable objects
- etc.

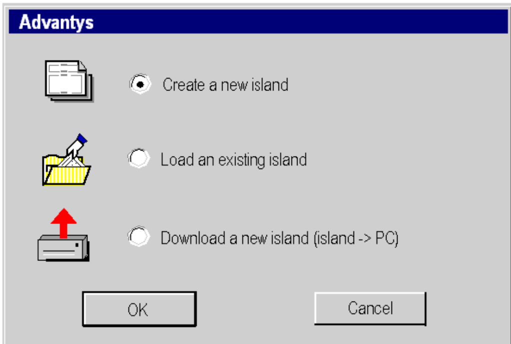
Each EDS file is specific to a product type and cannot be re-used on other products as this will result in the incorrect I/O configuration. It is up to the user to make sure that the correct EDS file is used.

Creating a New EDS and DCF Configuration File

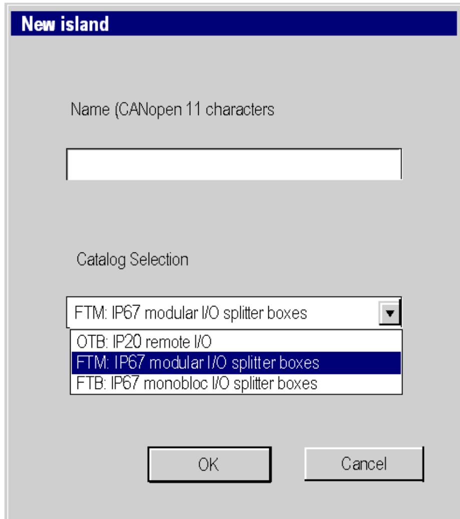
Introduction

Once you have installed and registered the Advantys Configuration Tool (ACT), the process of creating configuration (*.EDS / *.DCF) files for the island can begin.

Creating a New EDS and DCF Configuration File

Step	Action
1	Launch the Advantys Configuration Tool software. A window appears:  ● Select **Create new island** ● Click on the **OK**button.

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>2</td><td>The New island window appears:



The creation of an island must be in line with the physical configuration of your installation:

Enter the name of the island in the fieldName. The name of the island must correspond to the name of the EDS configuration file.
Select the catalog in the Catalog selection drop-down menu.
Confirm your selection by clicking on the OK button
</td></tr>
<tr><td>3</td><td>Building the Island
A browser window appears. A representative model of the island can be built in this window. At this point, this is an image of an empty 35mm (1.37in) DIN rail.
The catalog browser contains all the references of the catalog selected.
Building the island is a "drag and drop" operation:

Click on the reference in the catalog browser window and, while holding down the left mouse button, drag the reference over to the DIN rail and drop it (release the mouse button).
</td></tr>
</table>

Step	Action
4	**Island Configuration** Once the island has been built, you can set its parameters. The parameters you need to define will depend on the I/O functions you wish to use. The islands are configured in the configuration window: ● Open the configuration window by double clicking on the island (or by selecting the island and then the **Island/Module Editor** menu. ● Modify the required parameter(s). ● Click **OK** to save the changes and close the configuration window. **Notes:** The values given in the configuration window define the behavior of the island. PDOs are configured in such a way as to transport the island process data. The list of data contained in the PDOs is visible in the **I/O Assignment** tab of the configuration window.
5	**Saving the Island and Generating an EDS or DCF Configuration File** ● Select the **Save** command from the **File** menu. The `*.ISL` island file is saved. ● A **Generate** window appears. Select the file type to generate (*.EDS/*.DCF/Symbol table) and the configuration used (Twido/Premium/M340/Other). Click **OK** to generate and save the configuration files. **Notes:** The data in the symbol table can be modified only for M340.

7.3 Network Configuration

Setting the Network Parameters

Description

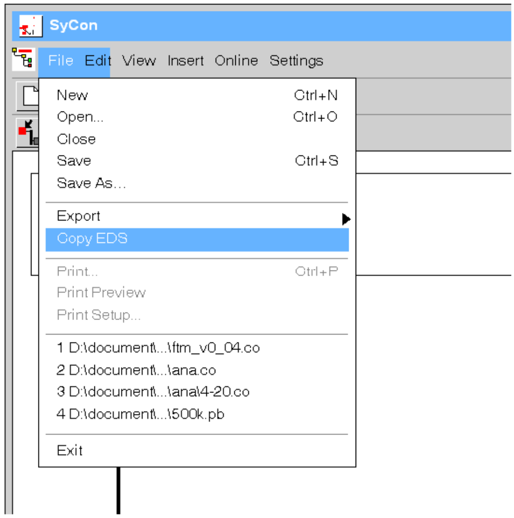
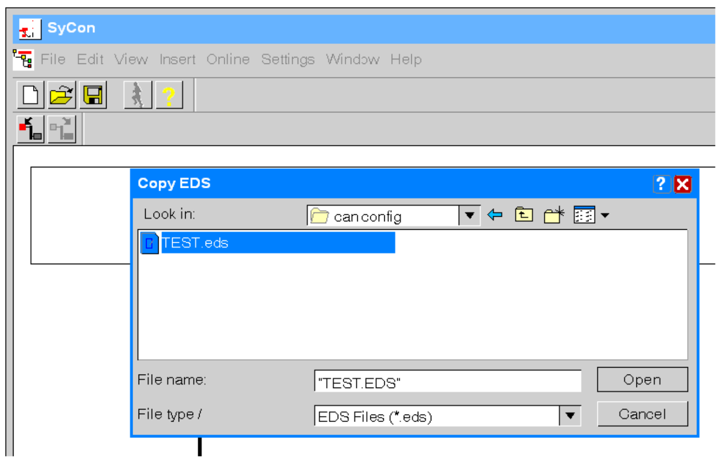
The configuration tool is used to draw diagrams of networks using a graphic representation of the network nodes. It is then used to generate the complete configuration of the network that has been drawn.

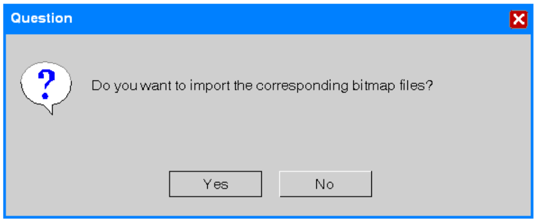
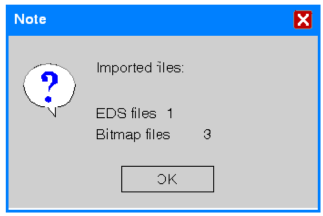
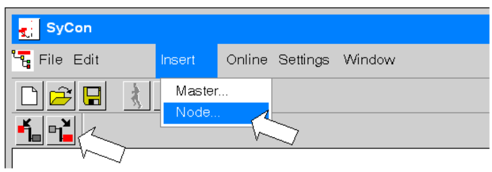
It provides access to the various configuration parameters and communication parameters by PDO.

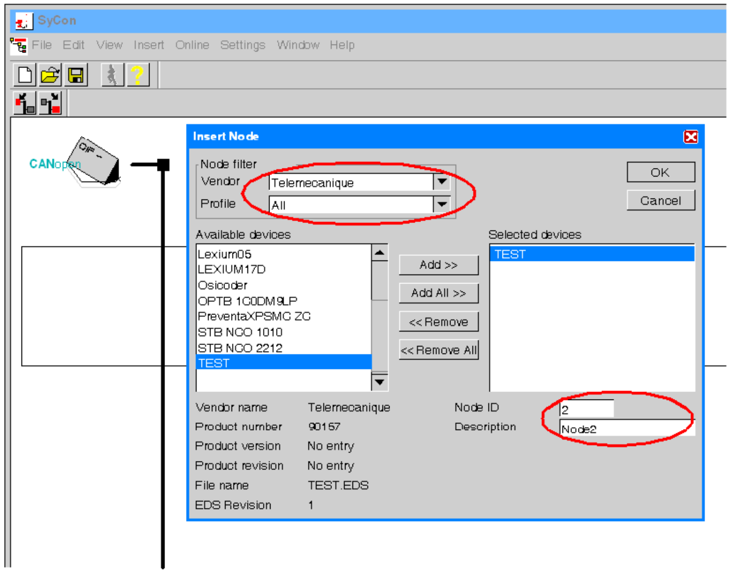
Below is an example of how to use the SyCon configuration tool:

Method

Within the PL7 programming software or Unity, launch the SyCon network tool and follow the steps below:

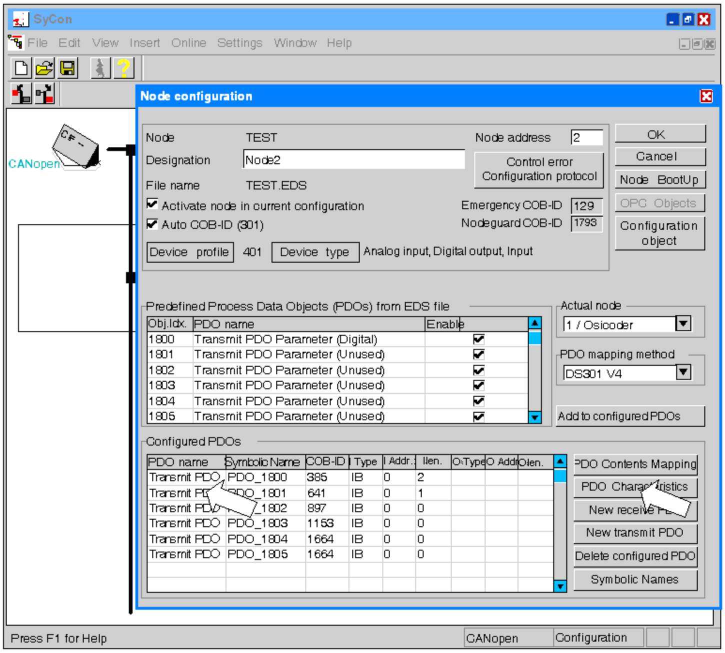
<table>
<tr><th>Steps</th><th>Actions</th></tr>
<tr><td>1</td><td>Open a CANopen type file.</td></tr>
<tr><td>2</td><td>Click on "File" and select "Copy EDS".

</td></tr>
<tr><td>3</td><td>Select the file to be imported and click on "Open":

</td></tr>
</table>

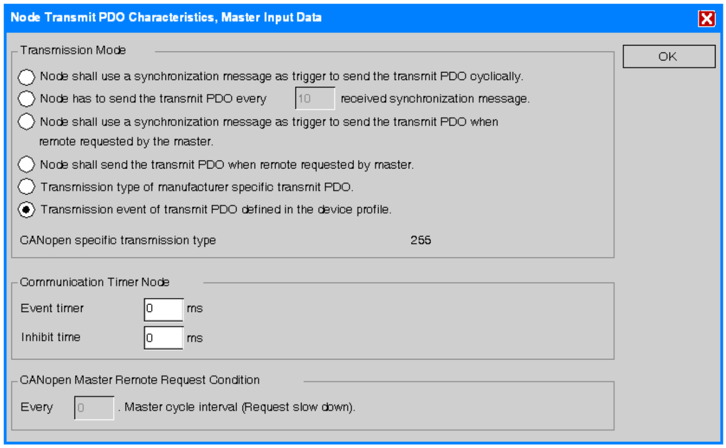
Steps	Actions
4	Click on "Yes" to import the 3 associated image files. 
5	If the image files are in the same directory as the EDS file, they are found automatically:  Click "OK".
6	Click on "Insert" and select "Node" or click on the associated button. 

Steps	Actions
7	Select the devices to be inserted in the network, enter the node address (given on the product) and the node description, and click OK:  Note: The name given in the list is the "comment associated with communication block" defined with CANConfig.

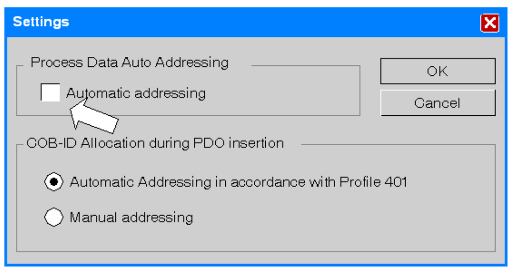
Configuring the PDOs

Follow the steps below:

Step	Action
1	Double-click on the image of the island to be configured. The configuration window appears.
2	Select a configured PDO and click on "PDO characteristics": 

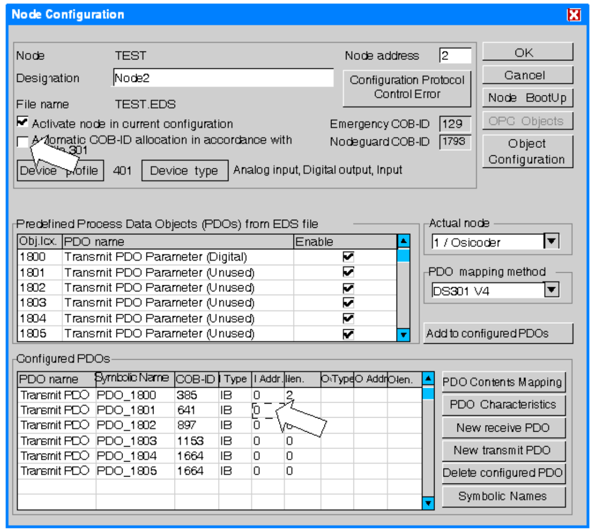
<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>3</td><td>Select the required transmission mode and click OK:

</td></tr>
<tr><td>4</td><td>If you want to define the addresses of the activated PDOs manually:

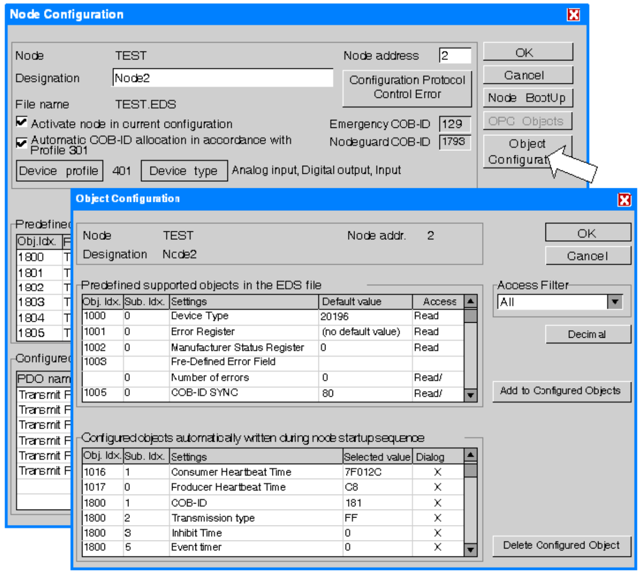
Select the master,
Click on "Settings" and select "Global settings",
Deselect "Automatic addressing" in the "Process Data Auto Addressing" area,
Click OK.

Illustration


Otherwise, go directly to step 6.</td></tr>
</table>

Step	Action
5	Enter the required values in the "I Adrr" and "O Addr" boxes opposite the activated PDO.



Step	Action
6	Click on "Object Configuration":  This window is used to configure the default values of the configured objects that will be sent to the device the next time the node is powered up. For further information on the various objects see *The Object Dictionary, page 219*
7	Select the objects to be sent to the device, click on "Add to Configured Objects" then click OK.
8	Select "File/Save": A *.CO configuration file is created, which contains the complete network architecture and the initial configuration of each node. This file is used by PLC programming software (e.g. PL7, Unity, etc.).

7.4 PLC Programming

Introduction

This chapter describes how to integrate the CANopen network configuration file and configuring under PL7.

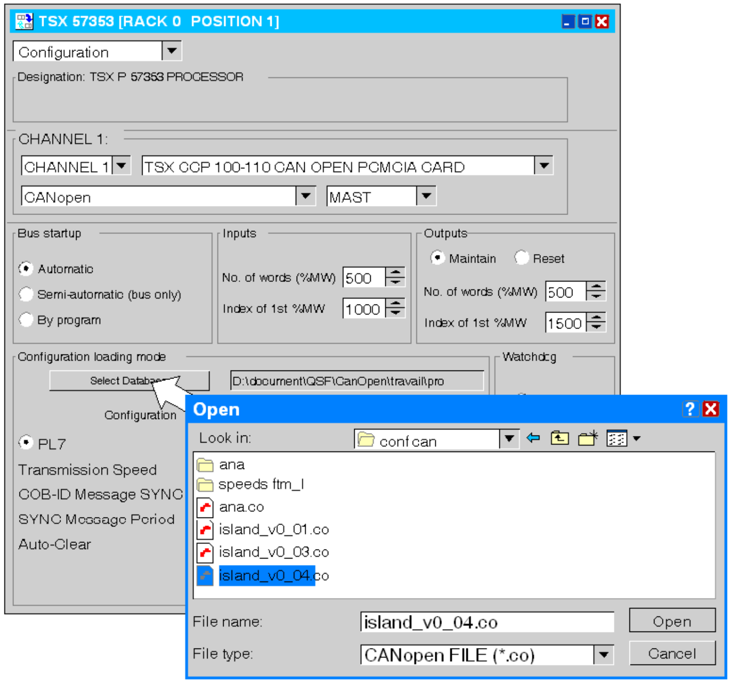
What's in this Section?

This section contains the following topics:

Integration and Use under PL7

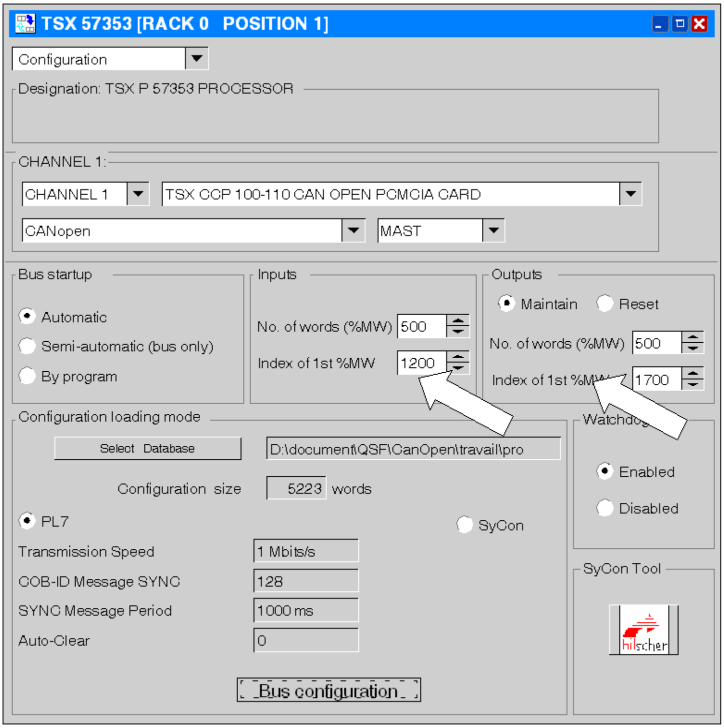
Configuration

Follow the steps below:

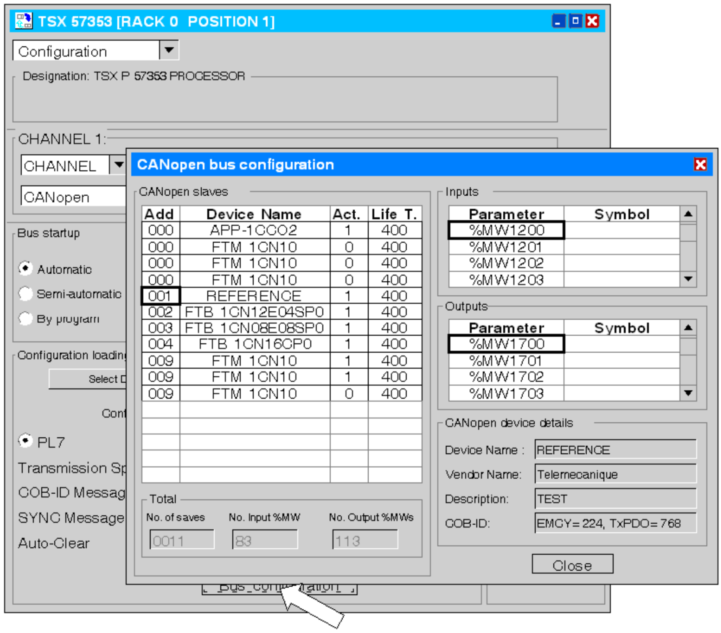
Steps	Actions
1	In the master configuration window, select the network configuration file generated with SYCON: 

<table>
<tr><th>Steps</th><th>Actions</th></tr>
<tr><td>2</td><td>

Complete the fields of the "Input" boxes (input data exchange area) and "Output" boxes (output data exchange area):

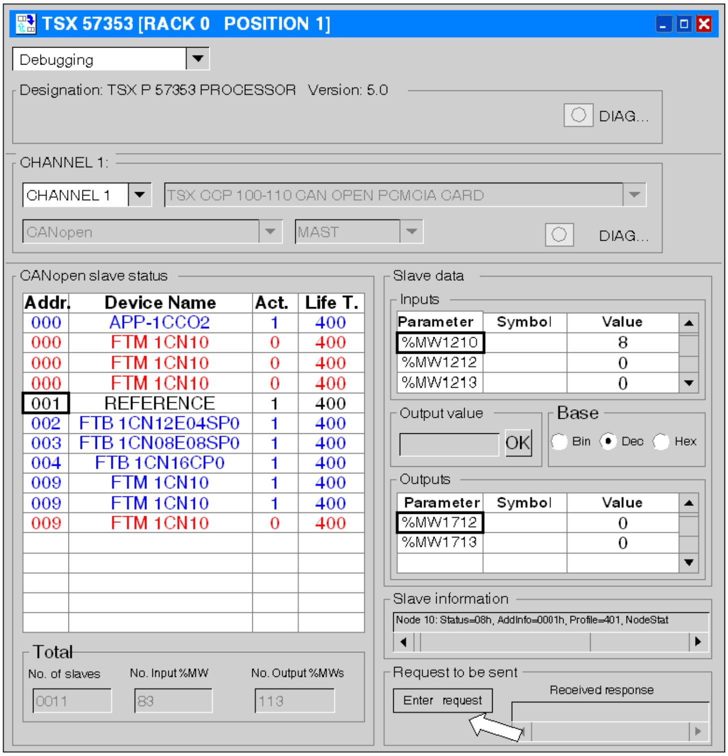


</td></tr>
</table>

Steps	Actions
3	Click on the "Bus Configuration" button:  The bus configuration window is used to display the exact address of the data associated with the devices. The start address of each PDO is defined by the start address of the exchange area configured using PL7, to which the PDO offset defined using SyCon is added.
4	Execute the required SDO requests (either from the debug screen, or with a program).

SDO Request from the Debug Screen

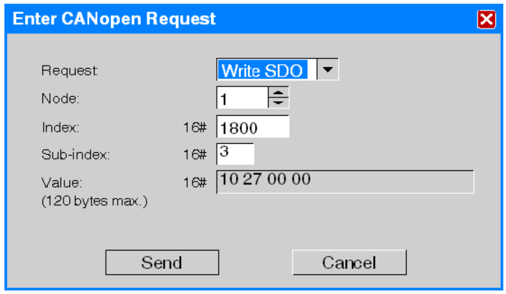
Follow the steps below:

Step	Action
1	Click on the "Enter request" button in the bottom-right of the debug screen: 

<table>
<tr><th>Step</th><th>Action</th></tr>
<tr><td>2</td><td>

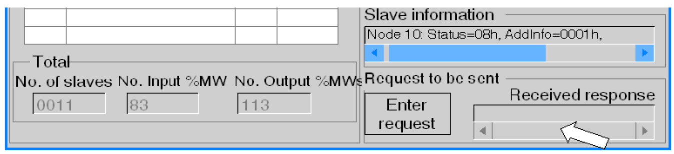
- Complete the fields:
  - Request: "Write SDO" or "Read SDO"
  - Node: address of the device on the CANopen network
  - Index: index of the object to read or write
  - Sub-index: sub-index of the object to read or write
  - Value: entry area for the data to be sent, for write only
- Click "Send".

Here is an example of how to configure the Inhibit Time to 1000 ms:



The value "10 27 00 00" corresponds to the number 2710 in hexadecimals, which is 1000 ms (see *Inhibit Time and Event Timer, page 84*).
</td></tr>
<tr><td>3</td><td>

After a "Read SDO", read the value given in the "Received response" area in the bottom-right of the debug screen:



</td></tr>
</table>

Examples of SDO Requests

Programmed SDO Request: Example 1

This example gives the program for reading object 1000H. After a request is made, the data obtained is read in the table Diag0:120 (defined below).

Variables used and function parameters

Variable	Type	Description
Read_sdo	Boolean	Request launch bit.
Index	Word	Index of the object to poll (LSB of the double word "Index_dw").
Sub-index	Word	Sub-index of the object to poll (MSB of the double word "Index_dw").
Slave_add	Word	Address of the slave to poll.
Diag0:120	Word table	Data exchange area.
Status0:4	Word table	Control and exchange status words.
ADR#y.SYS	Immediate value	Master board address.
‘SDO’	Character string	Type of SDO object (SDO always in upper case).
Index_dw	Double word	MSB = sub-index. LSB = index.
Node_Id	Word	Word or value identifying the destination device on the CANopen bus.

Program

```
Slave_add:=2     (*node at address 2 on the CANopen network*)
Index:=16#1000;      (*index 1000H*)
Sub_index:=0;      (*sub-index 0*)
IF Read_sdo THEN
   (*initialize command*)
                    Read_sdo:=FALSE;
  (*Parameter update*)
       Node_Id:=Slave_add; (*Slave address*)
      Diag0:120:=16#FFFF; (*Initialize diagnostics reception
table*)
       Status2:=0; (*Initialize exchange report*)
       Status3:=6; (*Time-out*)
   (*request*)
```

```
        READ_VAR(ADR#y.1.SYS,'SDO', _
    Index_dw,Node_Id,Diag0:120,Status1:4);
END_IF;
```

Programmed SDO Request: Example 2

This example shows the program for saving parameters with object 1010H. The data to be sent is contained in the table Diag0:4 (defined below).

Variables used and function parameters

Variable	Type	Description
Write_sdo	Boolean	Request launch bit.
Index	Word	Index of the object to poll (LSB of the double word "Index_dw").
Sub-index	Word	Sub-index of the object to poll (MSB of the double word "Index_dw").
Slave_add	Word	Address of the slave to poll.
Diag0:120	Word table	Data exchange area.
Status0:4	Word table	Control and exchange status words.
ADR#y.SYS	Immediate value	Master board address.
‘SDO’	Character string	Type of SDO object (SDO always in upper case).
Index_dw	Double word	MSB = sub-index. LSB = index.
Node_Id	Word	Word or value identifying the destination device on the CANopen bus.

Program

```
Slave_add:=2     (*node at address 2 on the CANopen network*)
Index:=16#1010;      (*index 1010H*)
Sub_index:=1;     (*sub-index 1*)
Diag0:=16#6173;      (*'as'*)
Diag0[1]:=16#6576;      (*'ev'*)
IF write_sdo THEN
    (*initialize command*)
                        write_sdo:=FALSE;
  (*Parameter update*)
        Node_Id:=Slave_add; (*Slave address*)
        Status2:=0; (*Initialize exchange report*)
```

```
        Status3:=6; (*Time-out*)
(*request*)
        WRITE_VAR(ADR#y.1.SYS,'SDO', _
   Index_dw,Node_Id,Diag0:4,Status1:4);
END_IF;
```

Diagnostics of the Advantys OTB Island

8

At a Glance

Diagnostics enables us to analyze the behavior of the network Advantys OTB island by:

- The LEDs indicating the communication and I/O status,
- Reading the diagnostics objects.

What's in this Chapter?

This chapter contains the following topics:

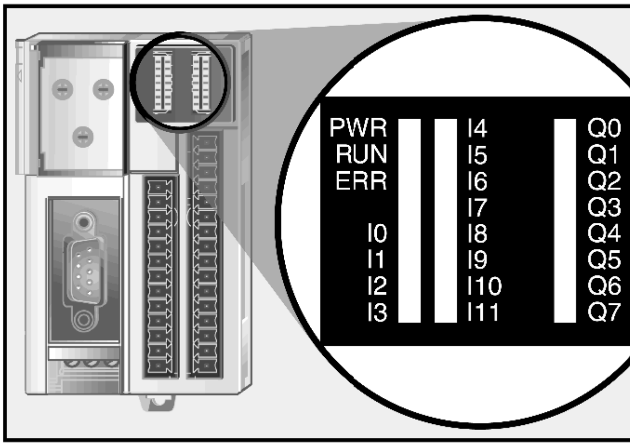
CANopen LEDs

Introduction

23 LEDs of the OTB 1C0DM9LP module visually reflect the operational status of the island on a CANopen network. The LEDs are located in the upper part of the network interface module.

Description

The illustration below shows the LEDs used by the CANopen Advantys OTB network interface module:



Meaning of the LEDs

- The PWR LED indicates the presence of a 24 VDC power supply to the network interface module.
- The LEDs 2 (CAN RUN) and 3 (CAN ERR) *(see page 213)* indicate the data exchange status between the CANopen fieldbus master and the Advantys OTB island.
- LEDs I0 to I11 and Q0 to Q7 reflect the status of the network interface module I/O.
- LED 4 is not used.

NOTE: When you consult the table, check the status of the Power LED.

CANopen Communication LEDs

Standard DRP303-3 defines the RUN and ERR LEDs and their different states. The table below describes the conditions being displayed, and the colors and types of flashing used by the CAN ERR and CAN RUN LEDs to display the normal operating modes and error conditions of an Advantys OTB CANopen network interface module.

LED	Type of Flashing	Description
CAN ERR (red)	Off	No error.
	Random flashing	Auto-baud: Automatic search for the bus communication speed.
	Flashing: 1 flash	Warning limit reached. An internal error counter in the CAN controller has reached or exceeded the error frame limit threshold (error frame).
	Flashing: 2 flashes	Error control event. Detection of a guard event (NMT-Slave or NMT-master) or a heartbeat event (Heartbeat consumer)
	Flashing: 3 flashes	Synchronization error: message not received within the defined period. (See Object 1006H *(see page 232)*).
	On	Bus OFF. OTB module status: Bus OFF.
CAN RUN (green)	Random flashing	Auto-baud: Automatic search for the bus communication speed.
	Continuous flashing	Module status: Pre-operational.
	Flashing: 1 flash	Module status: Stopped.
	On	Module status: Operational.

Power Supply and I/O Status LEDs

The table below describes the conditions as well as the colors used for the PWR, I0 to I11, Q0 to Q7 LEDs to display the normal operating modes and the error conditions for the power supply and I/O on the CANopen Advantys OTB network interface module.

Name	LED Color	Status	Function
PWR (power)	Green	On	Presence of 24 VDC for the OTB module
I0 to I11	Green	On	Input at 24 V
		Off	Input at 0V
Q0 to Q7	Green	On	Output at 24 V or contact closed
		Off	Output at 0V or contact open

CANopen Objects Diagnostics

Description

When the OTB module detects an error, the following objects are updated. These objects are described in more detail in the "Object Dictionary" chapter:

- **1001H: Error Register** See Object Dictionary *(see page 227)*,
- **1002H: Manufacturer Status Register** See Object Dictionary *(see page 228)*,
- **1003H: Pre=defined Error Field** See Object Dictionary *(see page 229)*,
- **3000H: Module Specific Diagnostics** See Object Dictionary *(see page 352)*,
- **3001H: Special Function Counter Diagnostics** See Object Dictionary *(see page 355)*,
- **3002H: Special Function Fast Counter Diagnostics** See Object Dictionary *(see page 356)*,
- **3003H: Special Function PLS/PWM Diagnostics** See Object Dictionary *(see page 357)*.

EMCYMessage Structure

For each error the EMCY message is sent by the module that detected the fault via the network (see table structure below).

Once the error has been cleared an EMCY message is sent again, incorporating an "Error code" = 0.

The EMCY message consists of 8 data bytes outlined in the following table:

Byte	**0-1**	**2**	**3**	**4**	**5**	**6**	**7**
Contents	Emergency Error Code	Error register	No. of the module,	Manufacturer Status Register			
Corresponding object	1003H	1001H	-	1002H			

Error Codes (EMCY bytes 0-1)

The table below lists the error codes and their meanings:

Error code	Diagnostics code	Cause
0000H	ERROR_RESET_OR_NO_ERROR	Clearing of one, or all, errors
1000H	GENERIC_ERROR	Internal communication error
6101H	SOFTWARE_RX_QUEUE_OVERRUN	The receive buffer has exceeded its internal memory capacity
6102H	SOFTWARE_TX_QUEUE_OVERRUN	The transmit buffer has exceeded its internal memory capacity
8100H	COMMUNICATION	Synchronization, transmit/receive error counter > 96
8120H	CAN_IN_ERROR_PASSIVE_MODE	CAN controller interrupted
8130H	LIFE_GUARD_ERROR	Node-Guarding error
8140H	BUS_OFF	The transmit error counter has exceeded its capacity

Status Register (EMCY Byte 2)

The object 1001H (Error register) is a byte used by the device to display internal errors when an error is detected:

Bit	Meaning	Comments
0	Generic error	Detail in object 1003H
1	Reserved	Not monitored
2	Reserved	Not monitored
3	Reserved	Not monitored
4	Communication error	Detail in object 1003H
5	Reserved	Not monitored
6	Reserved	Not monitored
7	Specific to the manufacturer	Detail in object 3000H

Manufacturer status register (EMCY Byte 4-5-6-7)

The data contained in bytes 4-5 corresponds to the expansion module status, while data contained in bytes 6-7 corresponds to the island status.

The following table indicates the assignment of the 32 bit set:

Bits	Function	Description
Bits 0...15	Island status word	Bits 0...8 : not used Bit 9: communication fault or external fault Bits 10...12: not used Bits 13: configuration fault (expansion modules missing or badly configured) Bits 14...15: not used
Bits 16...31	Expansion module status word	Bit 16: state of network interface module Bit 17: state of first expansion module Bit 18: state of second expansion module Bit 19: state of third expansion module Bit 20: state of fourth expansion module Bit 21: state of fifth expansion module Bit 22: state of sixth expansion module Bit 23: state of seventh expansion module Bit 24...31: not used

Expansion Module Identification Codes

List of Identification Codes

Table of identification codes for each type of expansion module:

Type of Discrete I/O Module	Identification Code
8 inputs	0004H
16 inputs	0000H
32 inputs	0200H
8 outputs	0005H
16 outputs	0001H
32 outputs	0301H
8 mixed I/O	0006H
24 mixed I/O	0205H

Type of Analog I/O Module	Identification Code
TM2 AMI2HT	6002H
TM2 AM01HT	6003H
TM2 AMM3HT	6001H
TM2 AMM6HT	6008H
TM2 ALM3LT	6000H
TM2 AVO2HT	6007H
TM2 AMI2LT	600AH
TM2 AMI4LT	6004H
TM2 AMI8HT	6005H
TM2 ARI8HT	6006H
TM2 ARI8LT	600CH
TM2 ARI8LRJ	600BH

The Object Dictionary

9

Introduction

This chapter provides a description of the object dictionary, the list of objects concerning the communication profile, the hardware profile and the specific manufacturer zone, with a detailed description of each.

What's in this Chapter?

This chapter contains the following sections:

9.1 The Object Dictionary

Introduction

This section describes the three object dictionary zones.

What's in this Section?

This section contains the following topics:

Object Dictionary

Index Ranges

There are three zones in the object dictionary:

Index (hexadecimal)	**Zone**	**Documentation**
1000-1FFF	Communication profile zone	*Objects of the Communication Profile 1000H to 1FFFH, page 224*
2000-5FFF	Manufacturer-specific zone	*Manufacturer-specific Zone Objects 2000H to 5FFFH, page 294*
6000-9FFF	Device-specific profile zone	*Hardware Profile Objects 6000H to 9FFFH, page 364*

Use of Sub-Indexes

Discrete Expansion Module

The table below gives the sub-index number used by each discrete expansion module:

Object	TM2 DDI8DT TM2 DDI16DT TM2 DDI16DT	TM2 DDI32DK	TM2 DMM8DRT TM2 DMM24DRF	TM2 DDO8•T TM2 DRA•RT TM2 DDO16•T	TM2 DDO32UK TM2 DDO32TK
1027H	1	1	1	1	1
6000H	2	4	2	-	-
6100H	1	2	1	-	-
6102H	1	2	1	-	-
6103H	1	2	1	-	-
6200H	-	-	2	2	4
6300H	-	-	1	1	2
6302H	-	-	1	1	2
6306H	-	-	1	1	2
6307H	-	-	1	1	2
6308H	-	-	1	1	2
3000H	1	1	1	1	1

Analog Expansion Module

The table below gives the sub-index number used by each analog expansion module:

Object	AMI2HT AMI2LT	AM01HT	AMM3HT ALM3LT	AMM6HT	AV02HT	AMI4LT	AMI8HT ARI8HT	ARI8LT ARI8LRJ
1027H	1	1	1	1	1	1	1	1
6401H	2	-	2	4	-	4	8	8
6411H	-	1	1	2	2	-	-	-
6421H	2	-	2	4	-	4	8	8
6424H	2	-	2	4	-	4	8	8
6425H	2	-	2	4	-	4	8	8
6426H	2	-	2	4	-	4	8	8
6443H	-	1	1	2	2	-	-	-
6444H	-	1	1	2	2	-	-	-
2100H	2	-	2	4	-	4	8	8

Object	AMI2HT AMI2LT	AM01HT	AMM3HT ALM3LT	AMM6HT	AV02HT	AMI4LT	AMI8HT ARI8HT	ARI8LT ARI8LRJ
2101H	2	-	2	4	-	4	8	8
2102H	2	-	2	4	-	4	8	8
2103H	2	-	2	4	-	4	8	8
2200H	-	1	1	2	2	-	-	-
2201H	-	1	1	2	2	-	-	-
2202H	-	1	1	2	2	-	-	-
2203H	-	1	1	2	2	-	-	-
3000H	1	1	1	1	1	1	1	1

9.2 Objects of the Communication Profile 1000H to 1FFFH

Introduction

This section lists the objects relating to the communication profile. Each object, with all its technical characteristics, is described according to the CANopen standard.

What's in this Section?

This section contains the following topics:

Object 1000H: Device Type

Description

This object indicates the device type and its functionalities.

The Least Significant Byte indicates the profile number (401 or 191H, for CANopen standard inputs / outputs).

The Most Significant Byte is known as the "additional information" and provides details of the device's functionalities:

Bit No.	Meaning if bit = 1
0	The device has discrete inputs
1	The device has discrete outputs
2	The device has analog inputs
3	The device has analog outputs

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED32	*	ro	no	no

* Depends on the island configuration

Object 1001H: Error Register

Description

This object allows the device to indicate internal faults. When a fault is detected, the corresponding bit is therefore activated.

The following faults can be displayed:

Bit	Meaning	Comments
0	Generic error	Set to 1 when a fault, including configuration faults, is found on the communication module and its expansions.
1	Current	Unchecked
2	Voltage	Unchecked
3	Temperature	Unchecked
4	Communication error	CANopen error indicator. This may be supported by the stack.
5	Reserved	Unchecked
6	Reserved	Unchecked
7	Manufacturer specific	Set to 1 when a fault is detected between the communication module and its expansions (internal bus fault).

These bits represent the Boolean "OR" for the faults present on the island.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED8	-	ro	yes	no

Object 1002H: Manufacturer Status Register

Description

Diagnostics data is saved in this double word. This object is contained in the message EMCY.

The least significant byte contains the error code.

The most significant word contains additional information.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED32	-	ro	yes	no

Bit assignment

Registers	Function	Description
Bits 0...15	Island status word	Bits [0...8]: not used Bit [9]: communication fault or external fault Bits [10...12]: not used Bit [13]: configuration fault (expansion modules missing or badly configured) Bits [14...15]: not used
Bits 16...31	Expansion module status word	Bit [16]: state of network interface module Bit [17]: state of first expansion module Bit [18]: state of second expansion module Bit [19]: state of third expansion module Bit [20]: state of fourth expansion module Bit [21]: state of fifth expansion module Bit [22]: state of sixth expansion module Bit [23]: state of seventh expansion module Bits [24..0.31]: not used

NOTE:

Bit values:
- 0 : no fault
- 1 : fault

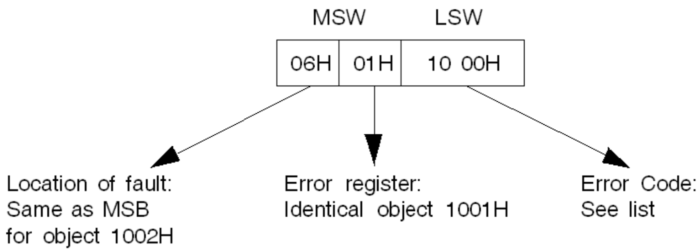
Object 1003H: Pre-defined Error Field (PEF)

Description

This object stores the most recent faults, as well as their characteristics:

- The Error Code is stored to the least significant word,
- "Additional Information" to the most significant word (number of the faulty module, 0 for the communication module),
- Sub-index 0 contains the number of recorded errors.

The diagram below shows the structure of sub-indexes n+1 for object 1003H :



Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-indexes = Number of recorded errors	UNSIGNED8	0	rw	no	no
1	Most recent error	UNSIGNED32	-	ro	no	no
2	Second to last error	UNSIGNED32	-	ro	no	no
...						
10						

Appearance of a New Fault

When a new fault appears, the codes already present are moved into the upper level sub-indexes: the fault in sub-index 1 is moved to sub-index 2, the fault in sub-index 2 is moved to sub-index 3, etc.

Clearing Faults

The fault code history can only be cleared by writing the value 0 in the sub-index 0 of object 1003H.

NOTE: Removing the cause a fault does not delete the error code from the PEF.

Indicating Faults

All faults are indicated by sending an "Emergency" message (EMCY message). Once the source of the fault has been cleared, an EMCY message with the No-error content is sent (Error-Code 0000H).

List of error codes

Error code (Hex)	Diagnostics	Cause
0000H	ERROR_RESET_OR_NO_ERROR	An error has been rectified
1000H	GENERIC_ERROR	Internal communication error
6101H	SOFTWARE_RX_QUEUE_OVERRUN	Receive memory capacity exceeded
6102H	SOFTWARE_TX_QUEUE_OVERRUN	Transmit memory capacity exceeded
8100H	COMMUNICATION	Synchronization error from the transmission/ reception counter (EMCYmessage transmitted if the counter value is > 96)
8120H	CAN_IN_ERROR_PASSIVE_MODE	CAN controller interrupted
8130H	LIFE_GUARD_ERROR	Node-Guarding error
8140H	BUS_OFF	The CAN frame error counter has exceeded its capacity

Object 1005H: COB-ID SYNC Message

Description

This object contains the synchronization message identifier.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED32	80H	rw	no	yes

Object 1006H: Communication Cycle Period

Description

This object describes the time interval between two SYNC signals. This interval must be at least 10 ms with a minimum increment of 1ms. The entry must be a double word.

If unused, the value of this field is zero.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED32	0	ro	no	yes

Coding the Switching Cycle Period

The threshold values are specified in the following table:

Value type	Decimal values	Hexadecimal values	IntervalSYNCin ms
Standard value	0	0000 0000	-
Min. value	10 000	0000 2710	10
-	25 000	0000 61A8	25
-	250 000	0003 D090	250
-	1 000 000	000F 4240	1000
-	5 000 000	004C 4B40	5000
Max. value	10 000 000	0098 9680	10 000

Object 1008H: Manufacturer Device Name

Description

This object contains the device name.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	STRING	OTB1C0DM9LP	ro	no	no

Object 100AH: Manufacturer Software Version (MSV)

Description

This object contains the software version of the device, in the form 'Vxx.yy'.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	STRING	*	ro	no	no

*Dependent on embedded software version

Object 100CH: Guard Time

Description

The object 100CH contains the "Guard-Time" parameter expressed in milliseconds.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED16	0	rw	no	yes

Object 100DH: Life Time Factor

Description

This object contains the "Life-Time-Factor" parameter. It allows computation of "Life-Time".See *"Node-Guarding" and "Life-Guarding" Monitoring Protocols, page 86*.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED8	0	rw	no	yes

Reliable Operation

For reliable and secure operation, the user must enter a "Life-Time Factor" with a minimum value of 2.

When the value 1 is used, and in case of delays due to high priority messages or internal processing of the "Node-Guarding" master, the module switches to the default "Pre-Operational" state without generating errors.

⚠ WARNING

RISK OF UNEXPECTED DEVICE OPERATION

Set the "Life-Time-Factor" to a minimum value of 2 to prevent any inadvertent change of state to a "Pre-Operational" state. Depending on the I/O configuration, an inadvertent change of state may result in unintended device operation.

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

Object 1010H: Store Parameters

Description

This object is used to store the CANopen island's parameters in back-up memory.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	4	ro	no	no
1	Store all parameters	UNSIGNED32	-	rw	no	no
2	Store communication parameters (1000H–1FFFH)	UNSIGNED32	-	rw	no	no
3	Not used	-	-	-	-	-
4	Store default application parameters (2000H–9FFFH).	UNSIGNED32	-	rw	no	no

Operation

To save the parameters, the "save" string (6576 6173H) must be written to the corresponding index:

	Most significant word		Least significant word	
ISO 8859 (ASCII) signature	e	v	a	s
Hex value	65H	76H	61H	73H

Information on storage functionality is read from a sub-index. The result obtained, 0000 0001H, indicates that the module saves parameters only when it receives the command to do so.

Object 1011H: Restore Default Parameters

Description

This object is used to restore the CANopen island's "factory" parameters.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	4	ro	no	no
1	Restore all default parameters.	UNSIGNED32	-	rw	no	no
2	Restore default communication parameters (1000H–1FFFH).	UNSIGNED32	-	rw	no	no
3	Not used.	-	-	-	-	-
4	Restore default application parameters (2000H–9FFFH).	UNSIGNED32	-	rw	no	no

NOTE: Communication parameters are only acknowledged after :

- a power up,
- a Reset COM command,
- a Reset NODE command,
- a restoration of saved objects (3202H).

Application parameters are acknowledged only after:

- a power up,
- a Reset NODE command.
- a restoration of saved objects (3202H).

Operation

To restore the parameters, the "load" string (64616F6CH) must be written to the corresponding index:

	Most significant word		Least significant word	
ISO 8859 (ASCII) signature	d	a	o	l
Hex value	64H	61H	6FH	6CH

Information on whether it is possible to restore the module's default parameters is read from a sub-index. The result obtained, 0000 0001H, indicates that the factory parameters can only be restored when the module receives the command to do so.

Object 1014H: COB-ID Emergency (EMCY) Message

Description

This object contains the EMCY emergency message identifier.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED32	80H + NODE-ID	rw	no	yes

Object 1016H: Consumer Heartbeat Time

Description

This object is used to configure the time interval in ms within which the module must receive the Heartbeat message from monitored islands. The communication module is designed in such a way that it can only monitor one island at a time.

The value of this object must be greater than the value of object 1017H.

The time must be a multiple of 1ms.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	1	ro	no	yes
1	Consumer Heartbeat Time	UNSIGNED32	0	rw	no	yes

Content of Variable

The contents of sub-index 1 is as follows:

Bit	31 to 24	23 to 16	15 to 0
Value	0H (Reserved)	Address of monitored island	Monitoring time in ms

If the value of the object is 0, no island is monitored.

Object 1017H: Producer Heartbeat Time

Description

This object is used to configure the time interval in ms within which the module must produce the Hearbeat message.

The default monitoring method is Node-Guarding. If a non-zero value is written in this object the Heartbeat mechanism is used.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED16	0	rw	no	yes

If the Heartbeat error monitoring protocol is selected, the producer transmits a Heartbeat message periodically, depending on the "Producer Heartbeat Time" parameter. The devices responsible for monitoring this message (Heartbeat Consumer) generate a Heartbeat event if the message is not received in the configured time ("Consumer Heartbeat Time").

Object 1018H: Identity Object

Description

This object contains device information. It indicates the manufacturer's CiA identifier (vendor ID), the product code) and the device revision numbers.

The revision information is coded in two parts:

- the major revision part (most significant byte) indicates an evolution in CANopen functionalities,
- the minor revision part (least significant byte) indicates an evolution in device functionalities only.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of inputs	UNSIGNED8	3	ro	no	no
1	Vendor ID	UNSIGNED32	0500005AH	ro	no	no
2	Product code	UNSIGNED32	FEFBH	ro	no	no
3	Revision number	UNSIGNED32	Version dependant	ro	no	no

Object 1027H: Module List

Description

This object contains the list of modules connected to the island.

Object Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of sub-indexes (number of connected expansion modules)	UNSIGNED8	n	ro	no	no
1	Product code for first expansion module	UNSIGNED16	-	ro	no	no
...						
n	Product code for last expansion module	UNSIGNED16	See the table below	ro	no	no

List of Identification Codes

Table of identification codes for each type of expansion module:

Type of Discrete I/O Module	Identification Code
8 inputs	0004H
16 inputs	0000H
32 inputs	0200H
8 outputs	0005H
16 outputs	0001H
32 outputs	0301H
8 mixed I/O	0006H
24 mixed I/O	0205H

Type of Analog I/O Module	Identification Code
TM2 AMI2HT	6002H
TM2 AM01HT	6003H
TM2 AMM3HT	6001H
TM2 AMM6HT	6008H
TM2 ALM3LT	6000H
TM2 AVO2HT	6007H
TM2 AMI2LT	600AH
TM2 AMI4LT	6004H
TM2 AMI8HT	6005H
TM2 ARI8HT	6006H
TM2 ARI8LT	600CH
TM2 ARI8LRJ	600BH

Object 1200H: Server SDO Parameter

Description

This object contains message identifiers for communication by SDO.

Object Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of sub-indexes	UNSIGNED8	2H	ro	no	no
1	Client to Server	UNSIGNED32	600H + Node ID	ro	no	no
2	Server to Client	UNSIGNED32	580H + Node ID	ro	no	no

Object 1400H: 1st Receive PDO Communication Parameter

Description

This object contains the receive PDO identifier.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	2	ro	no	yes
1	COB-ID	UNSIGNED32	0000 0200H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes

Transmission mode

The PDO transmission mode can be configured as described in the table below.

Transfer code (Dec.)	Transmission mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For modes 254 and 255, the event triggering the send is defined by the message producer.

Object 1401H: 2nd Receive PDO Communication Parameter

Description

This object contains the receive PDO identifier.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	2	ro	no	yes
1	COB-ID	UNSIGNED32	x000 0300H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes

x = 0 or 8 depending on island configuration (0 = PDO activated, 8 = PDO inactivated)

Transmission mode

The PDO transmission mode can be configured as described in the table below.

Transfer code (Dec.)	Transmission mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For modes 254 and 255, the event triggering the send is defined by the message producer.

Object 1402H: 3rd Receive PDO Communication Parameter

Description

This object contains the receive PDO identifier.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	2	ro	no	yes
1	COB-ID	UNSIGNED32	x000 0400H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes

x = 0 or 8 depending on island configuration (0 = PDO activated, 8 = PDO inactivated)

Transmission mode

The PDO transmission mode can be configured as described in the table below.

Transfer code (Dec.)	Transmission mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For modes 254 and 255, the event triggering the send is defined by the message producer.

Object 1403H: 4th Receive PDO Communication Parameter

Description

This object contains the receive PDO identifier.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	2	ro	no	yes
1	COB-ID	UNSIGNED32	x000 0500H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes

x = 0 or 8 depending on island configuration (0 = PDO activated, 8 = PDO inactivated)

Transmission mode

The PDO transmission mode can be configured as described in the table below.

Transfer code (Dec.)	Transmission mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For modes 254 and 255, the event triggering the send is defined by the message producer.

Object 1404H: 5th Receive PDO Communication Parameter

Description

This object contains the receive PDO identifier.

Object Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	2H	ro	no	yes
1	COB-ID	UNSIGNED32	8000 0000H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes

NOTE: Starting with the 5th PDO, COB-IDs are not identified by default. They should be defined by the user, making sure that they are unique on the network. See *List of COB-IDs, page 389*.

Transmission mode

The PDO transmission mode can be configured as described in the table below.

Transfer code (Dec.)	Transmission mode					Notes
	Cyclic	**Acyclic**	**Synchro-nous**	**Asynchro-nous**	**only RTR**	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For modes 254 and 255, the event triggering the send is defined by the message producer.

Object 1405H: 6th Receive PDO Communication Parameter

Description

This object contains the receive PDO identifier.

Object Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Sub-index number	UNSIGNED8	2	ro	no	yes
1	COB-ID	UNSIGNED32	8000 0000H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes

NOTE: Starting with the 5th PDO, COB-IDs are not identified by default. They should be defined by the user, making sure that they are unique on the network. See *List of COB-IDs, page 389*.

Transmission Mode

The PDO transmission mode can be configured as described in the table below.

Transfer Code (Dec.)	Transmission Mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For modes 254 and 255, the event triggering sending is defined by the message producer.

Object 1406H: 7th Receive PDO Communication Parameter

Description

This object contains the receive PDO identifier.

Object Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Sub-index number	UNSIGNED8	2H	ro	no	yes
1	COB-ID	UNSIGNED32	8000 0000H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes

NOTE: Starting with the 5th PDO, COB-IDs are not identified by default. They should be defined by the user, making sure that they are unique on the network. See *List of COB-IDs, page 389*.

Transmission Mode

The PDO transmission mode can be configured as described in the table below.

Transfer Code (Dec.)	Transmission Mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For modes 254 and 255, the event triggering sending is defined by the message producer.

Object 1407H: 8th Receive PDO Communication Parameter

Description

This object contains the receive PDO identifier.

Object Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Sub-index number	UNSIGNED8	2H	ro	no	yes
1	COB-ID	UNSIGNED32	8000 0000H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes

NOTE: Starting with the 5th PDO, COB-IDs are not identified by default. They should be defined by the user, making sure that they are unique on the network. See *List of COB-IDs, page 389*.

Transmission Mode

The PDO transmission mode can be configured as described in the table below.

Transfer Code (Dec.)	Transmission Mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For modes 254 and 255, the event triggering sending is defined by the message producer.

Object 1600H: 1st Receive PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1601H: 2nd Receive PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1602H: 3rd Receive PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1603H: 4th Receive PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1604H: 5th Receive PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1605H: 6th Receive PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1606H: 7th Receive PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1607H: 8th Receive PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1800H: 1st Transmit PDO Communication Parameter

Description

This object contains the PDO transmit identifier.

Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Sub-index number	UNSIGNED8	5	ro	no	yes
1	COB-ID	UNSIGNED32	0000 0180H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes
3	Inhibit Time	UNSIGNED16	0	rw	no	yes
4	Not available					
5	Event Timer	UNSIGNED16	0	rw	no	yes

Transmission Mode

The PDO transmission mode can be configured as described in the table below.

Transfer Code (Dec.)	Transmission Mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For discrete I/O, the event is the change in value.

For analog inputs/outputs several events can be selected (see object 6421H):
- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta

For the current value of counters several events can be selected (see object 2306H and 2406H) :

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta
- Catching a value
- done,
- The measurement frequency is valid

For pulse generators the event is the "done" signal (see object 2506H).

COB-ID Structure

The structure of a COB-ID for CAN2.0 is shown in the following table:

Bit No.	Value	Meaning
31 (MSb)	0	The PDO object exists
	1	The PDO object does not exist
30	0	RTR mechanism authorized
	1	RTR mechanism not authorized
29	0	11-Bit ID (CAN 2.0A)
28 - 11	0	if bit 29 = 0
10 - 0 (LSb)	X	Bit 10 - 0 of the identifier

Inhibit Time (Sub-index 3)

In the case of PDO transmission (Transmit PDO), the Inhibit Time may be entered in this 16-bit field. After data has been changed, the PDO sender checks that an Inhibit Time has expired since the last transmission. A new PDO transmission can only take place if the Inhibit Time has expired. The Inhibit Time is useful during asynchronous transmission (transmission mode 255), to avoid overloading the CANopen bus. The Inhibit Time is a multiple of 100 μs of the value written in sub-index 3 of objects 1800H to 1807H.

The table below shows some examples of values.

Value	Inhibit Time in ms
0000H	0
64H	10
3E8H	100
1388H	500
2710H	1000
FFFFH	6553.5

Event Timer (Sub-index 5)

The Event Timer only works in asynchronous transmission mode (transmission mode 255). If the data changes before the Event Timer expires, a TPDO (Transmit PDO) is sent. If a value higher than 0 is written in the 16-bit field, the TPDO is sent after the Event Timer expires. The value written in sub-index 5 of objects 1800H to 1807H corresponds to the Event Timer in milliseconds. Data transfer occurs even if there are no changes to the data.

The table below shows some examples of values.

Value	Event Timer in ms
0000H	0
64H	100
3E8H	1000
1388H	5000
2710H	10000
FFFFH	65535

Object 1801H: 2nd Transmit PDO Communication Parameter

Description

This object contains the PDO transmit identifier.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	5	ro	no	yes
1	COB-ID	UNSIGNED32	x000 0280H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes
3	Inhibit Time	UNSIGNED16	0	rw	no	yes
4	Not available					
5	Event Timer	UNSIGNED16	0	rw	no	yes

x = 0 or 8 depending on island configuration (0 = PDO activated, 8 = PDO inactivated)

Transmission mode

The PDO transmission mode can be configured as described in the table below.

Transfer code (Dec.)	Transmission mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For discrete I/Os, the event is the change in value.

For analog inputs/outputs several events can be selected (see object 6421H):

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta

For the current value of counters several events can be selected (see object 2306H and 2406H) :

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta
- Value sensor
- done,
- The measurement frequency is valid

For pulse generators the event is the signal "done" (see object 2506H).

COB-ID Structure

The structure of a COB-ID for CAN2.0 is shown in the following table:

Bit No.	Value	Meaning
31 (MSb)	0	The PDO object exists
	1	The PDO object does not exist
30	0	RTR mechanism authorized
	1	RTR mechanism not authorized
29	0	11-Bit ID (CAN 2.0A)
28 - 11	0	if bit 29 = 0
10 - 0 (LSb)	X	Bit 10 - 0 of the identifier

Inhibit Time (Sub-index 3)

In the case of PDO (Transmit PDO) transmission, theInhibit Time may be entered in this 16 bit field. After data has been changed, the PDO sender checks that an Inhibit Time has expired since the last transmission. A new PDO transmission can only take place if the Inhibit Time has expired. The Inhibit Time is useful during asynchronous transmission (transmission mode 255), to avoid overloading the CANopen bus. The Inhibit Time is a multiple of 100 μs of the value written in sub-index 3 of objects 1800H to 1807H.

The following table shows some examples of values.

Value	Inhibit Time in ms
0000H	0
64H	10
3E8H	100
1388H	500
2710H	1000
FFFFH	6553.5

Event Timer (Sub-index 5)

TheEvent Timer only works in asynchronous transmission mode (transmission mode 255). If the data changes before the Event Timer expires, a TPDO (Transmit PDO) is sent. If a value higher than 0 is written in the 16-bit field, the TPDO is sent after the Event Timer expires. The value written in sub-index 5 of objects 1800H to 1807H corresponds to the Event Timer in milliseconds. Data transfer occurs even if there are no changes to the data.

The following table shows some examples of values.

Value	Event Timer in ms
0000H	0
64H	100
3E8H	1000
1388H	5000
2710H	10000
FFFFH	65535

Object 1802H: 3rd Transmit PDO Communication Parameter

Description

This object contains the PDO transmit identifier.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	5	ro	no	yes
1	COB-ID	UNSIGNED32	x000 0380H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes
3	Inhibit Time	UNSIGNED16	0	rw	no	yes
4	Not available					
5	Event Timer	UNSIGNED16	0	rw	no	yes

x = 0 or 8 depending on island configuration (0 = PDO activated, 8 = PDO inactivated)

Transmission mode

The PDO transmission mode can be configured as described in the table below.

Transfer code (Dec.)	Transmission mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For discrete I/Os, the event is the change in value.

For analog inputs/outputs several events can be selected (see object 6421H):

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta

For the current value of counters several events can be selected (see object 2306H and 2406H) :

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta
- Value sensor
- done,
- The measurement frequency is valid

For pulse generators the event is the signal "done" (see object 2506H).

COB-ID Structure

The structure of a COB-ID for CAN2.0 is shown in the following table:

Bit No.	Value	Meaning
31 (MSb)	0	The PDO object exists
	1	The PDO object does not exist
30	0	RTR mechanism authorized
	1	RTR mechanism not authorized
29	0	11-Bit ID (CAN 2.0A)
28 - 11	0	if bit 29 = 0
10 - 0 (LSb)	X	Bit 10 - 0 of the identifier

Inhibit Time (Sub-index 3)

In the case of PDO (Transmit PDO) transmission, theInhibit Time may be entered in this 16 bit field. After data has been changed, the PDO sender checks that an Inhibit Time has expired since the last transmission. A new PDO transmission can only take place if the Inhibit Time has expired. The Inhibit Time is useful during asynchronous transmission (transmission mode 255), to avoid overloading the CANopen bus. The Inhibit Time is a multiple of 100 μs of the value written in sub-index 3 of objects 1800H to 1807H.

The following table shows some examples of values.

Value	Inhibit Time in ms
0000H	0
64H	10
3E8H	100
1388H	500
2710H	1000
FFFFH	6553,5

Event Timer (Sub-index 5)

TheEvent Timer only works in asynchronous transmission mode (transmission mode 255). If the data changes before the Event Timer expires, a TPDO (Transmit PDO) is sent. If a value higher than 0 is written in the 16-bit field, the TPDO is sent after the Event Timer expires. The value written in sub-index 5 of objects 1800H to 1807H corresponds to the Event Timer in milliseconds. Data transfer occurs even if there are no changes to the data.

The following table shows some examples of values.

Value	Event Timer in ms
0000H	0
64H	100
3E8H	1000
1388H	5000
2710H	10000
FFFFH	65535

Object 1803H: 4th Transmit PDO Communication Parameter

Description

This object contains the PDO transmit identifier.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	5	ro	no	yes
1	COB-ID	UNSIGNED32	x000 0480H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes
3	Inhibit Time	UNSIGNED16	0	rw	no	yes
4	Not available					
5	Event Timer	UNSIGNED16	0	rw	no	yes

x = 0 or 8 depending on island configuration (0 = PDO activated, 8 = PDO inactivated)

Transmission mode

The PDO transmission mode can be configured as described in the table below.

Transfer code (Dec.)	Transmission mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For discrete I/Os, the event is the change in value.

For analog inputs/outputs several events can be selected (see object 6421H):

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta

For the current value of counters several events can be selected (see object 2306H and 2406H) :

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta
- Value sensor
- done,
- The measurement frequency is valid

For pulse generators the event is the signal "done" (see object 2506H).

COB-ID Structure

The structure of a COB-ID for CAN2.0 is shown in the following table:

Bit No.	**Value**	**Meaning**
31 (MSb)	0	The PDO object exists
	1	The PDO object does not exist
30	0	RTR mechanism authorized
	1	RTR mechanism not authorized
29	0	11-Bit ID (CAN 2.0A)
28 - 11	0	if bit 29 = 0
10 - 0 (LSb)	X	Bit 10 - 0 of the identifier

Inhibit Time (Sub-index 3)

In the case of PDO (Transmit PDO) transmission, theInhibit Time may be entered in this 16 bit field. After data has been changed, the PDO sender checks that an Inhibit Time has expired since the last transmission. A new PDO transmission can only take place if the Inhibit Time has expired. The Inhibit Time is useful during asynchronous transmission (transmission mode 255), to avoid overloading the CANopen bus. The Inhibit Time is a multiple of 100 μs of the value written in sub-index 3 of objects 1800H to 1807H.

The following table shows some examples of values.

Value	Inhibit Time in ms
0000H	0
64H	10
3E8H	100
1388H	500
2710H	1000
FFFFH	6553.5

Event Timer (Sub-index 5)

TheEvent Timer only works in asynchronous transmission mode (transmission mode 255). If the data changes before the Event Timer expires, a TPDO (Transmit PDO) is sent. If a value higher than 0 is written in the 16-bit field, the TPDO is sent after the Event Timer expires. The value written in sub-index 5 of objects 1800H to 1807H corresponds to the Event Timer in milliseconds. Data transfer occurs even if there are no changes to the data.

The following table shows some examples of values.

Value	Event Timer in ms
0000H	0
64H	100
3E8H	1000
1388H	5000
2710H	10000
FFFFH	65535

Object 1804H: 5th Transmit PDO Communication Parameter

Description

This object contains the PDO transmit identifier.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	5	ro	no	yes
1	COB-ID	UNSIGNED32	8000 0000H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes
3	Inhibit Time	UNSIGNED16	0	rw	no	yes
4	Not available					
5	Event Timer	UNSIGNED16	0	rw	no	yes

Transmission mode

The PDO transmission mode can be configured as described in the table below.

Transfer code (Dec.)	Transmission mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For discrete I/Os, the event is the change in value.

For analog inputs/outputs several events can be selected (see object 6421H):

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta

For the current value of counters several events can be selected (see object 2306H and 2406H) :

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta
- Value sensor
- done,
- The measurement frequency is valid

For pulse generators the event is the signal "done" (see object 2506H).

COB-ID Structure

The structure of a COB-ID for CAN2.0 is shown in the following table:

Bit No.	Value	Meaning
31 (MSb)	0	The PDO object exists
	1	The PDO object does not exist
30	0	RTR mechanism authorized
	1	RTR mechanism not authorized
29	0	11-Bit ID (CAN 2.0A)
28 - 11	0	if bit 29 = 0
10 - 0 (LSb)	X	Bit 10 - 0 of the identifier

Inhibit Time (Sub-index 3)

In the case of PDO (Transmit PDO) transmission, theInhibit Time may be entered in this 16 bit field. After data has been changed, the PDO sender checks that an Inhibit Time has expired since the last transmission. A new PDO transmission can only take place if the Inhibit Time has expired. The Inhibit Time is useful during asynchronous transmission (transmission mode 255), to avoid overloading the CANopen bus. The Inhibit Time is a multiple of 100 µs of the value written in sub-index 3 of objects 1800H to 1807H.

The following table shows some examples of values.

Value	Inhibit Time in ms
0000H	0
64H	10
3E8H	100
1388H	500
2710H	1000
FFFFH	6553,5

Event Timer (Sub-index 5)

TheEvent Timer only works in asynchronous transmission mode (transmission mode 255). If the data changes before the Event Timer expires, a TPDO (Transmit PDO) is sent. If a value higher than 0 is written in the 16-bit field, the TPDO is sent after the Event Timer expires. The value written in sub-index 5 of objects 1800H to 1807H corresponds to the Event Timer in milliseconds. Data transfer occurs even if there are no changes to the data.

The following table shows some examples of values.

Value	Event Timer in ms
0000H	0
64H	100
3E8H	1000
1388H	5000
2710H	10000
FFFFH	65535

Object 1805H: 6th Transmit PDO Communication Parameter

Description

This object contains the PDO transmit identifier.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	5	ro	no	yes
1	COB-ID	UNSIGNED32	8000 0000H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes
3	Inhibit Time	UNSIGNED16	0	rw	no	yes
4	Not available					
5	Event Timer	UNSIGNED16	0	rw	no	yes

Transmission mode

The PDO transmission mode can be configured as described in the table below.

Transfer code (Dec.)	Transmission mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For discrete I/Os, the event is the change in value.

For analog inputs/outputs several events can be selected (see object 6421H):

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta

For the current value of counters several events can be selected (see object 2306H and 2406H) :

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta
- Value sensor
- done,
- The measurement frequency is valid

For pulse generators the event is the signal "done" (see object 2506H).

COB-ID Structure

The structure of a COB-ID for CAN2.0 is shown in the following table:

Bit No.	Value	Meaning
31 (MSb)	0	The PDO object exists
	1	The PDO object does not exist
30	0	RTR mechanism authorized
	1	RTR mechanism not authorized
29	0	11-Bit ID (CAN 2.0A)
28 - 11	0	if bit 29 = 0
10 - 0 (LSb)	X	Bit 10 - 0 of the identifier

Inhibit Time (Sub-index 3)

In the case of PDO (Transmit PDO) transmission, theInhibit Time may be entered in this 16 bit field. After data has been changed, the PDO sender checks that an Inhibit Time has expired since the last transmission. A new PDO transmission can only take place if the Inhibit Time has expired. The Inhibit Time is useful during asynchronous transmission (transmission mode 255), to avoid overloading the CANopen bus. The Inhibit Time is a multiple of 100 µs of the value written in sub-index 3 of objects 1800H to 1807H.

The following table shows some examples of values.

Value	Inhibit Time in ms
0000H	0
64H	10
3E8H	100
1388H	500
2710H	1000
FFFFH	6553,5

Event Timer (Sub-index 5)

TheEvent Timer only works in asynchronous transmission mode (transmission mode 255). If the data changes before the Event Timer expires, a TPDO (Transmit PDO) is sent. If a value higher than 0 is written in the 16-bit field, the TPDO is sent after the Event Timer expires. The value written in sub-index 5 of objects 1800H to 1807H corresponds to the Event Timer in milliseconds. Data transfer occurs even if there are no changes to the data.

The following table shows some examples of values.

Value	Event Timer in ms
0000H	0
64H	100
3E8H	1000
1388H	5000
2710H	10000
FFFFH	65535

Object 1806H: 7th Transmit PDO Communication Parameter

Description

This object contains the PDO transmit identifier.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	5	ro	no	yes
1	COB-ID	UNSIGNED32	8000 0000H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes
3	Inhibit Time	UNSIGNED16	0	rw	no	yes
4	Not available					
5	Event Timer	UNSIGNED16	0	rw	no	yes

Transmission mode

The PDO transmission mode can be configured as described in the table below.

Transfer code (Dec.)	Transmission mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For discrete I/Os, the event is the change in value.

For analog inputs/outputs several events can be selected (see object 6421H):

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta

For the current value of counters several events can be selected (see object 2306H and 2406H) :

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta
- Value sensor
- done,
- The measurement frequency is valid

For pulse generators the event is the signal "done" (see object 2506H).

COB-ID Structure

The structure of a COB-ID for CAN2.0 is shown in the following table:

Bit No.	Value	Meaning
31 (MSb)	0	The PDO object exists
	1	The PDO object does not exist
30	0	RTR mechanism authorized
	1	RTR mechanism not authorized
29	0	11-Bit ID (CAN 2.0A)
28 - 11	0	if bit 29 = 0
10 - 0 (LSb)	X	Bit 10 - 0 of the identifier

Inhibit Time (Sub-index 3)

In the case of PDO (Transmit PDO) transmission, theInhibit Time may be entered in this 16 bit field. After data has been changed, the PDO sender checks that an Inhibit Time has expired since the last transmission. A new PDO transmission can only take place if the Inhibit Time has expired. The Inhibit Time is useful during asynchronous transmission (transmission mode 255), to avoid overloading the CANopen bus. The Inhibit Time is a multiple of 100 μs of the value written in sub-index 3 of objects 1800H to 1807H.

The following table shows some examples of values.

Value	Inhibit Time in ms
0000H	0
64H	10
3E8H	100
1388H	500
2710H	1000
FFFFH	6553.5

Event Timer (Sub-index 5)

TheEvent Timer only works in asynchronous transmission mode (transmission mode 255). If the data changes before the Event Timer expires, a TPDO (Transmit PDO) is sent. If a value higher than 0 is written in the 16-bit field, the TPDO is sent after the Event Timer expires. The value written in sub-index 5 of objects 1800H to 1807H corresponds to the Event Timer in milliseconds. Data transfer occurs even if there are no changes to the data.

The following table shows some examples of values.

Value	Event Timer in ms
0000H	0
64H	100
3E8H	1000
1388H	5000
2710H	10000
FFFFH	65535

Object 1807H: 8th Transmit PDO Communication Parameter

Description

This object contains the PDO transmit identifier.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	5	ro	no	yes
1	COB-ID	UNSIGNED32	8000 0000H + Node ID	rw	no	yes
2	Transmission mode	UNSIGNED8	FFH (255 Dec)	rw	no	yes
3	Inhibit Time	UNSIGNED16	0	rw	no	yes
4	Not available					
5	Event Timer	UNSIGNED16	0	rw	no	yes

Transmission mode

The PDO transmission mode can be configured as described in the table below.

Transfer code (Dec.)	Transmission mode					Notes
	Cyclic	Acyclic	Synchro-nous	Asynchro-nous	only RTR	
0		x	x			Send PDO on first SYNC message following an event
1 to 240	x		x			Send PDO every x SYNC messages
241 to 251	Reserved					-
252 to 253	Reserved					-
254				x		Send PDO on event
255				x		Send PDO on event

NOTE: For discrete I/Os, the event is the change in value.

For analog inputs/outputs several events can be selected (see object 6421H):

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta

For the current value of counters several events can be selected (see object 2306H and 2406H) :

- Change in value
- Overrun of upper threshold
- Overrun of lower threshold
- Change in value greater than configured delta
- Value sensor
- done,
- The measurement frequency is valid

For pulse generators the event is the signal "done" (see object 2506H).

COB-ID Structure

The structure of a COB-ID for CAN2.0 is shown in the following table:

Bit No.	Value	Meaning
31 (MSb)	0	The PDO object exists
	1	The PDO object does not exist
30	0	RTR mechanism authorized
	1	RTR mechanism not authorized
29	0	11-Bit ID (CAN 2.0A)
28 - 11	0	if bit 29 = 0
10 - 0 (LSb)	X	Bit 10 - 0 of the identifier

Inhibit Time (Sub-index 3)

In the case of PDO (Transmit PDO) transmission, theInhibit Time may be entered in this 16 bit field. After data has been changed, the PDO sender checks that an Inhibit Time has expired since the last transmission. A new PDO transmission can only take place if the Inhibit Time has expired. The Inhibit Time is useful during asynchronous transmission (transmission mode 255), to avoid overloading the CANopen bus. The Inhibit Time is a multiple of 100 µs of the value written in sub-index 3 of objects 1800H to 1807H.

The following table shows some examples of values.

Value	Inhibit Time in ms
0000H	0
64H	10
3E8H	100
1388H	500
2710H	1000
FFFFH	6553.5

Event Timer (Sub-index 5)

TheEvent Timer only works in asynchronous transmission mode (transmission mode 255). If the data changes before the Event Timer expires, a TPDO (Transmit PDO) is sent. If a value higher than 0 is written in the 16-bit field, the TPDO is sent after the Event Timer expires. The value written in sub-index 5 of objects 1800H to 1807H corresponds to the Event Timer in milliseconds. Data transfer occurs even if there are no changes to the data.

The following table shows some examples of values.

Value	Event Timer in ms
0000H	0
64H	100
3E8H	1000
1388H	5000
2710H	10000
FFFFH	65535

Object 1A00H: 1st Transmit PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1A01H: 2nd Transmit PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1A02H: 3rd Transmit PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1A03H: 4th Transmit PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1A04H: 5th Transmit PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1A05H: 6th Transmit PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1A06H: 7th Transmit PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

Object 1A07H: 8th Transmit PDO Mapping Parameter

Description

This object is used to describe the objects that will be transported by the PDO.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	Configuration dependent	rw	no	yes
1	First object in PDO	UNSIGNED32	-	rw	no	yes
2	2nd object in PDO	UNSIGNED32	-	rw	no	yes
...						
8	Most recent object in PDO	UNSIGNED32	-	rw	no	yes

Data Field Structure

Each data object to be transported is represented in the following manner:

Bits	31 to 16	15 to 8	7 to 0
Data	Index number of object to be transported	Sub-index number of object to be transported	Length of object to be transported (in bits)
Example	6200H	01H	08H

NOTE: The total length of data transported by the PDO is 8 bytes maximum.

9.3 Manufacturer-specific Zone Objects 2000H to 5FFFH

Introduction

This section lists the objects from the manufacturer-specific zone. Each object, with all its technical characteristics, is described according to the CANopen standard.

What's in this Section?

This section contains the following topics:

Object 2000H: Local Digital Input Parameter

Description

This object is only used by the communication module discrete inputs. It allows the discrete input filter value to be defined.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of local inputs	UNSIGNED8	12 Dec. (CH)	ro	no	yes
1	Communication module, input 0 parameter	UNSIGNED16	1	rw	no	yes
...	...	...	...	...	...	...
12	Communication module, input 11 parameter	UNSIGNED16	1	rw	no	yes

Bit Values

The parameter description is provided in the table below:

Value	Description
0	no filtering
1	3 ms filter (default value)
2	12 ms filter

Object 2001H: Local Input Used by Special Functions Status

Description

This object is used to determine the assignment of inputs used by a special function. The n bit is set to 1 if a special function uses the n input. Only bits 0 to 9 are significant.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED16	0	ro	no	no

Input Assignment

The following table gives the assignment of inputs:

I/O	RVFC0	RVFC1	RFC0	RFC1
Input 0	x			
Input 1	x			
Input 2	x			
Input 3	x			
Input 4		x		
Input 5		x		
Input 6		x		
Input 7		x		
Input 8			x	
Input 9				x

Object 2002H: Local Output Used by Special Functions Status

Description

This object is used to determine the assignment of outputs used by a special function. The n bit is set to 1 if a special function uses the n output. Only bits 0 to 5 are significant.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED16	0	ro	no	no

Output Assignment

The following table gives the assignment of outputs:

I/O	RPLS1 / RPWM0	RPLS1 / RPWM1	RVFC0	RVFC1
Output 0	x			
Output 1		x		
Output 2			x	
Output 3			x	
Output 4				x
Output 5				x

Object 2100H: Analog Input Type

Description

This object defines the range of analog inputs on the expansion modules.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-indexes (number of analog inputs)	UNSIGNED8	n	ro	no	yes
1	Input type of channel 0	UNSIGNED16	0	rw	no	yes
...						
n	Input type of last channel	UNSIGNED16	0	rw	no	yes

Possible Sub-index Values

0 : Not used

1 : 0 ... 20mA

2 : 4 ... 20mA

3 : 0 ... 10V

4 : +/- 10V

5 : Thermo K

6 : Thermo J

7 : Thermo T

8 : PT100

9 : PT1000

10 : NI100

11 : NI1000

12 : Reserved

13 : NTC / CTN

14 : PTC / CTP

NOTE: The configuration of an incompatible value (for example PT100 on a normal analog input) will trigger an " Abort Code ".

Object 2101H: Analog Input Range

Description

This object defines the measurement unit for each analog input on the expansion modules.

Properties

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-indexes (number of analog inputs)	UNSIGNED8	n	ro	no	yes
1	Unit of measurement for channel 0	UNSIGNED16	1H	rw	no	yes
...						
n	Unit of measurement for last channel	UNSIGNED16	1H	rw	no	yes

Possible Sub-index Values

0 : Normal

1 : Customized

2 : Celsius (0.1°C)

3 : Fahrenheit (0.1°F)

4 : Resistance (Ohms)

NOTE: Trying to configure an incompatible value (for example PT100 on a normal analog input) will trigger an " Abort Code ".

Object 2102H: Analog Input Min

Description

This object contains the minimum value of a user-defined measurement range (sub-index of object 2101H = 1).

Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of sub-indexes (number of analog inputs)	UNSIGNED8	n	ro	no	yes
1	Minimum measurement value for channel 0	INTEGER16	0	rw	no	yes
...						
n	Minimum measurement value for last channel	INTEGER16	0	rw	no	yes

Object 2103H: Analog Input Max

Description

This object contains the maximum value of a user-defined measurement range (sub-index of object 2101H = 1).

Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of sub-indexes (number of analog inputs)	UNSIGNED8	n	ro	no	yes
1	Maximum measurement value for channel 0	INTEGER16	7FFFH	rw	no	yes
...						
n	Maximum measurement value for last channel	INTEGER16	7FFFH	rw	no	yes

Object 2104H: R_0 Value Register

Description

This object contains the values of the R_0 parameter for each of the module channels.

NOTE: The object 2104H is only used for the TM2 ARI8HT expansion module.

Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of sub-indexes	UNSIGNED8	n	ro	no	yes
1	R_0 value for channel 0	UNSIGNED16	014AH	rw	no	yes
...						
n	R_0 value for last channel	UNSIGNED16	014AH	rw	no	yes

NTC Probe

The temperature (T) varies in relation to the resistance (R) according to the equation below:

$$T(R) = \frac{1}{\frac{1}{T_0} + \frac{1}{B}\ln\left[\frac{R}{R_0}\right]}$$

Where:

- T = temperature measured by the probe, in Kelvin (object 6401H)
- R = physical value of the resistance in Ohm
- R_0 = reference resistance in Ohms at temperature T_0 (object 2104H)
- T_0 = reference temperature in Kelvin (0.01°K) (object 2105H)
- B = sensitivity of the NTC probe in Kelvin (0.01°K) (object 2106H)

R_0, T_0 and B should at least be equal to 1.

If resistance is selected as a unit, the displayed value equals the resistance of the probe.

NOTE: 25°C = 77°F = 298.15°K

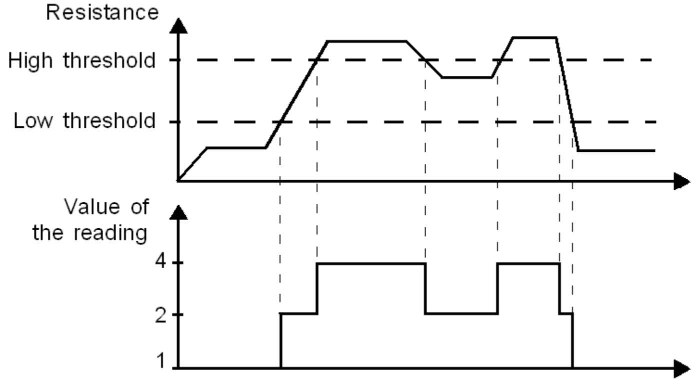
PTC Probe

- R_0 = high threshold (object 2104H)
- T_0 = low threshold (object 2105H)

Read value = 1 if resistance value < T_0

Read value = 2 if T_0 < resistance value < R_0

Read value = 4 if resistance value > R_0



Object 2105H: T_0 Value Register

Description

This object contains the values of the T_0 parameter for each of the module channels.

NOTE: The object 2105H is only used for the TM2 ARI8HT expansion module.

Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of sub-indexes (number of analog channels)	UNSIGNED8	n	ro	no	yes
1	T_0 parameter value for channel 0	UNSIGNED16	7477H	rw	no	yes
...						
n	T_0 parameter value for last channel	UNSIGNED16	7477H	rw	no	yes

NTC Probe

The temperature (T) varies in relation to the resistance (R) according to the equation below:

$$T(R) = \frac{1}{\frac{1}{T_0} + \frac{1}{B}\ln\left[\frac{R}{R_0}\right]}$$

Where:

- T = temperature measured by the probe, in Kelvin (object 6401H)
- R = physical value of the resistance in Ohm
- R_0 = reference resistance in Ohms at temperature T_0 (object 2104H)
- T_0 = reference temperature in Kelvin (0.01°K) (object 2105H)
- B = sensitivity of the NTC probe in Kelvin (0.01°K) (object 2106H)

R_0, T_0 and B should at least be equal to 1.

If resistance is selected as a unit, the displayed value equals the resistance of the probe.

NOTE: 25°C = 77°F = 298.15°K

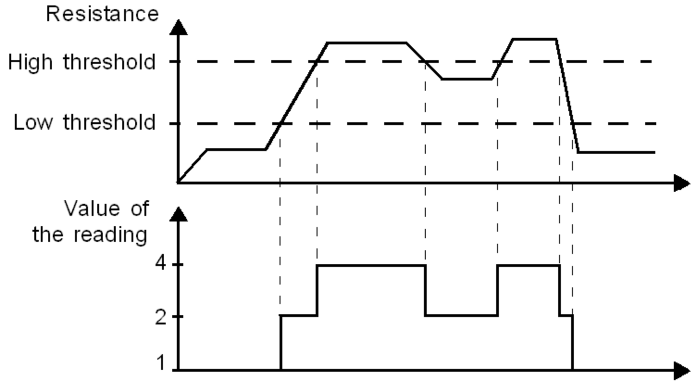
PTC Probe

- R_0 = high threshold (object 2104H)
- T_0 = low threshold (object 2105H)

Read value = 1 if resistance value < T_0

Read value = 2 if T_0 < resistance value < R_0

Read value = 4 if resistance value > R_0



Object 2106H: B Value Register

Description

This object contains the values of the B parameter for each of the module channels.

NOTE: The object 2106H is only used for the TM2 ARI8HT expansion module.

Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of sub-indexes (number of analog channels)	UNSIGNED8	n	ro	no	yes
1	B parameter value for channel 0	UNSIGNED16	0DF1H	rw	no	yes
...						
n	B parameter value for last channel	UNSIGNED16	0DF1H	rw	no	yes

NTC Probe

The temperature (T) varies in relation to the resistance (R) according to the equation below:

$$T(R) = \frac{1}{\frac{1}{T_0} + \frac{1}{B}\ln\left[\frac{R}{R_0}\right]}$$

Where:

- T = temperature measured by the probe, in Kelvin (object 6401H)
- R = physical value of the resistance in Ohm
- R_0 = reference resistance in Ohms at temperature T_0 (object 2104H)
- T_0 = reference temperature in Kelvin (0.01°K) (object 2105H)
- B = sensitivity of the NTC probe in Kelvin (0.01°K) (object 2106H)

R_0, T_0 and B should at least be equal to 1.

If resistance is selected as a unit, the displayed value equals the resistance of the probe.

NOTE: 25°C = 77°F = 298.15°K

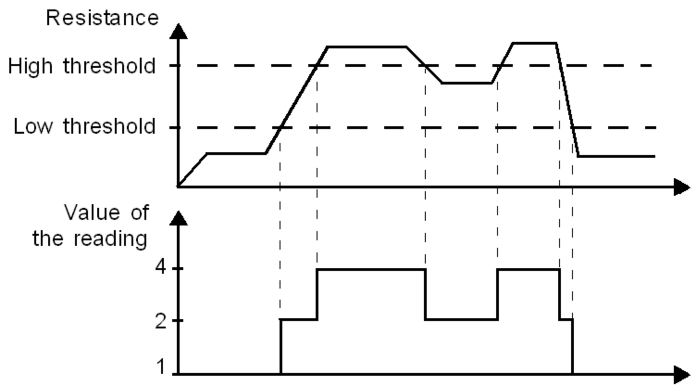
PTC Probe

- R_0 = high threshold (object 2104H)
- T_0 = low threshold (object 2105H)

Read value = 1 if resistance value < T_0

Read value = 2 if T_0 < resistance value < R_0

Read value = 4 if resistance value > R_0



Object 2200H: Analog Output Type

Description

This object defines the range of analog outputs on the expansion modules.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-indexes (Number of analog outputs)	UNSIGNED8	n	ro	no	yes
1	Output type of channel 0	UNSIGNED16	0	rw	no	yes
...						
n	Output type of last channel	UNSIGNED16	0	rw	no	yes

Possible Sub-index Values

0 : Not used

1 : 0 ... 20mA

2 : 4 ... 20mA

3 : 0 ... 10V

4 : +/- 10V

NOTE: The configuration of an incompatible value (for example 0...20 mA on a voltage only output) can trigger an "Abort Code".

Object 2201H: Analog Output Range

Description

This object defines the write range for each analog output on the expansion modules.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-indexes (Number of analog outputs)	UNSIGNED8	n	ro	no	yes
1	Unit of measurement for channel 0	UNSIGNED16	1H	rw	no	yes
...						
n	Unit of measurement for last channel	UNSIGNED16	1H	rw	no	yes

Possible Sub-index Values

0 : Normal

1 : Customized

Object 2202H: Analog Output Minimum

Description

This object contains the minimum value of a user-defined write range (sub-index of object 2101H = 1).

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-indexes (Number of analog outputs)	UNSIGNED8	n	ro	no	yes
1	Minimum measurement value for channel 0	INTEGER16	0	rw	no	yes
...						
n	Minimum value for last channel	INTEGER16	0	rw	no	yes

Object 2203H: Analog Output Max

Description

This object contains the maximum value of a user-defined write range (sub-index of object 2101H = 1).

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-indexes (number of analog inputs)	UNSIGNED8	n	ro	no	yes
1	Maximum value for channel 0	INTEGER16	7FFFH	rw	no	yes
...						
n	Maximum value for last channel	INTEGER16	7FFFH	rw	no	yes

Object 2300H: RFC Counter Current Value

Description

This RFC.V object contains the current value of the counter in 16 bits.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RFC counter)	UNSIGNED8	2	ro	no	no
1	Current RFC0 value	UNSIGNED16	0	ro	yes	no
2	Current RFC1 value	UNSIGNED16	0	ro	yes	no

Object 2301H: RFC Counter Status Value

Description

This object contains the counter status word(RFC.D).

NOTE: The bit "Done" is set to 1 by the counter, the user is responsible for resetting it.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RFC counter)	UNSIGNED8	2	ro	no	no
1	Status of RFC0	UNSIGNED16	0	ro	yes	no
2	Status of RFC1	UNSIGNED16	0	ro	yes	no

Object 2302H: RFC Counter Commands

Description

This object contains the counter command word.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of counters)	UNSIGNED8	2	ro	no	no
1	RFC0 control	UNSIGNED16	0	rw	yes	no
2	RFC1 control	UNSIGNED16	0	rw	yes	no

Bit Values

The parameter description is provided in the following table:

Parameter	Description
RFC.EN	Bit [0]: counter validation
RFC.R	Bit [1]: Reset to zero of the counter
RFC.CD	Bit [2]: reset to zero of bit RFC.D

Object 2303H: RFC Counter Mode

Description

This object (RFC.M) contains the counter use mode.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of counter)	UNSIGNED8	2	ro	no	yes
1	Mode of RFC0	UNSIGNED16	0	rw	no	yes
2	Mode of RFC1	UNSIGNED16	0	rw	no	yes

Counting modes

RFC.M parameter:
- 0 : Not used
- 1 : Upcounter
- 2 : Down counter

Object 2304H: RFC Counter Preset Value

Description

This object (RFC.P) contains the preset value of the counter in 16 bits.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RFC counter)	UNSIGNED8	2	ro	no	yes
1	RFC0 preset value	UNSIGNED16	FFFFH	rw	no	yes
2	RFC1 preset value	UNSIGNED16	FFFFH	rw	no	yes

Object 2305H: RFC Counter Delta Value

Description

This object defines the Delta value (coded on 16 bits) which will trigger a PDO send.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of counter)	UNSIGNED8	2	ro	no	yes
1	Delta value of RFC0	UNSIGNED16	0	rw	no	yes
2	Delta value of RFC1	UNSIGNED16	0	rw	no	yes

Object 2306H: RFC Counter Interrupt Trigger Selection

Description

This object defines the event that triggers the PDO transmission.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of counter)	UNSIGNED8	2	ro	no	yes
1	RFC0 counter trigger mode	UNSIGNED8	1	rw	no	yes
2	RFC1 counter trigger mode	UNSIGNED8	1	rw	no	yes

Bit assignment

Bit 0: RFC.V change

Bit 1: overrun

Bit 2: change of RFC.V by Delta value

Object 2307H: RFC Counter Fallback Mode

Description

This object (RFC.EM) indicates the behavior to adopt in case of a communication fault.

Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of sub-indexes (number of counters)	UNSIGNED8	2	ro	no	yes
1	Fallback mode of RFC0	UNSIGNED8	0	rw	no	yes
2	Fallback mode of RFC1	UNSIGNED8	0	rw	no	yes

Fallback Modes

RFC.EM:
- 0: Counter reinitialization (0 if upcounter, preset if downcounter)
- 1: Stops counting, saves the last value and freezes the counter
- 2: Continues counting

NOTE: Counting is restarted by a user action, not by the disappearance of the fault.

Object 2350H: RFC Counter Current Value

Description

This RFC.V object contains the current value of the counter in 32 bits.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RFC counter)	UNSIGNED8	2	ro	no	no
1	Current value of RFC0	UNSIGNED32	0	ro	yes	no
2	Current value of RFC1	UNSIGNED32	0	ro	yes	no

Object 2354H: RFC Counter Preset Value

Description

This object (RFC.P) contains the preset value of the counter in 32 bits.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RFC counter)	UNSIGNED8	2	ro	no	yes
1	RFC0 preset value	UNSIGNED32	FFFF FFFFH	rw	no	yes
2	RFC1 preset value	UNSIGNED32	FFFF FFFFH	rw	no	yes

Object 2355H: RFC Counter Delta Value

Description

This object defines the Delta value (coded on 32 bits) which will trigger a PDO send.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RFC counter)	UNSIGNED8	2	ro	no	yes
1	Delta value of RFC0	UNSIGNED32	0	rw	no	yes
2	Delta value of RFC1	UNSIGNED32	0	rw	no	yes

Object 2400H: RVFC Counter Current Value

Description

This (RVFC.V) object contains the current value of the counter in 16 bits.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	no
1	Current value of RVFC0	UNSIGNED16	0	ro	yes	no
2	Current value of RVFC1	UNSIGNED16	0	ro	yes	no

Object 2401H: RVFC Counter Status Value

Description

This object contains the counter status word.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	no
1	Status of RVFC0	UNSIGNED16	0	ro	yes	no
2	Status of RVFC1	UNSIGNED16	0	ro	yes	no

Bit assignment

Bit 0: RVFC.Drt (count direction)

Bit 1: RVFC.D (count threshold reached)

Bit 2: RVFC.S0 (current value above valeur S0)

Bit 3: RVFC.S1 (current value above S1)

Bit 4: RVFC.FV (valid frequency measurement)

NOTE: The bit "Done" is set to 1 by the counter, the user is responsible for resetting it.

Object 2402H: RVFC Counter Capture Value

Description

This object RVFC.C contains the current counter value stored, in 16 bit format, on a rising edge of the RVFC.Ica physical input.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	no
1	Capture value of RVFC0	UNSIGNED16	0	ro	yes	no
2	RVFC1 catch value	UNSIGNED16	0	ro	yes	no

Object 2403H: RVFC Counter Parameters Value

Description

This object contains the fast counter parameter values.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	yes
1	RVFC0 parameters	UNSIGNED16	04C0H	rw	no	yes
2	RVFC1 parameters	UNSIGNED16	04C0H	rw	no	yes

Bit assignment

Parameter	Description
RVFC.AQ0	Bit 0: activates the reflex output 0
RVFC.AQ1	Bit 1: activates the reflex output 1
RVFC.T	Bit 2: Frequency measure time base 0: 100ms 1: 1s
RVFC.AIpres	Bit 3: validates the preset input
RVFC.AIca	Bit 4: Validates the capture
RVFC.Q0Z1	Bit 5: status of reflex output 0 when the value is in zone 1
RVFC.Q0Z2	Bit 6: status of reflex output 0 when the value is in zone 2
RVFC.Q0Z3	Bit 7: status of reflex output 0 when the value is in zone 3
RVFC.Q1Z1	Bit 8: status of reflex output 1 when the value is in zone 1
RVFC.Q1Z2	Bit 9: status of reflex output 1 when the value is in zone 2
RVFC.Q1Z3	Bit 10: status of reflex output 1 when the value is in zone 3

Object 2404H: RVFC Counter Preset Value

Description

This object RVFC.P contains the counter preset value (coded on 16 bits).

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	yes
1	RVFC0 preset value	UNSIGNED16	FFFFH	rw	no	yes
2	RVFC1 preset value	UNSIGNED16	FFFFH	rw	no	yes

Object 2405H: RVFC Counter Delta Value

Description

This object (RVFC.Delta) defines the Delta value (coded on 16 bits) which will trigger a PDO send.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	yes
1	RVFC0 Delta value	UNSIGNED16	0	rw	no	yes
2	RVFC1 Delta value	UNSIGNED16	0	rw	no	yes

Object 2406H: RVFC Counter Interrupt Trigger Selection

Description

This object defines the event that triggers the PDO transmission.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	yes
1	Authorization of PDO transmission from RVFC0	UNSIGNED8	1	rw	no	yes
2	Authorization of PDO transmission from RVFC1	UNSIGNED8	1	rw	no	yes

Bit assignment

Bit 0: RVFC.V change

Bit 1: overrun (switch from 0 to 1 of RVCF.D) bit

Bit 2: RVFC.V change greater than RVFC.Delta

Bit 3: RVFC.Ica

Bit 4: RVFC.S0

Bit 5: RVFC.S1

Object 2407H: RVFC Counter Threshold TH0

Description

This object (RVFC.TH0) contains the value (coded on 16 bits) of the TH0 low threshold which will trigger a PDO send and/or switching of reflex outputs.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	yes
1	RVFC0TH0 threshold value	UNSIGNED16	0	rw	no	yes
2	RVFC1TH0 threshold value	UNSIGNED16	0	rw	no	yes

Object 2408H: RVFC Counter Threshold TH1

Description

This object (RVFC.TH1) contains the value (coded on 16 bits) of the TH1 high threshold which will trigger a PDO send and/or switching of reflex outputs.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	yes
1	RVFC0TH1 threshold value	UNSIGNED16	FFFFH	rw	no	yes
2	RVFC1TH1 threshold value	UNSIGNED16	FFFFH	rw	no	yes

Object 2409H: RVFC Counter Mode

Description

This object (RVFC.M) contails the counter use mode.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	yes
1	RVFC0 counting mode	UNSIGNED16	0	rw	no	yes
2	RVFC1 counting mode	UNSIGNED16	0	rw	no	yes

Counting modes

RVFC.M :
- 0 : Not used
- 1 : up/down counter
- 2 : 2-phase counter
- 3 : Single up counter
- 4 : Single down counter
- 5 : Frequency meter

Object 240AH: RVFC Counter Fallback Mode

Description

This object (RVFC.EM) indicates the behavior to adopt in case of a communication fault.

Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of sub-indexes (number of RVFC counters)	UNSIGNED8	2	ro	no	yes
1	Fallback mode of RVFC0	UNSIGNED8	0	rw	no	yes
2	Fallback mode of RVFC1	UNSIGNED8	0	rw	no	yes

Fallback Mode

RVFC.EM:

- 0: Counter reinitialization (0 if upcounter, preset if downcounter)
- 1: Stops counting, saves the last value and freezes the counter
- 2: Continues counting

NOTE: Counting is restarted by a user action, not by the disappearance of the fault.

Object 240BH: RVFC Counter Commands

Description

This object contains the counter command word.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	no
1	RVFC0 control	UNSIGNED16	0	rw	yes	no
2	RVFC1 control	UNSIGNED16	0	rw	yes	no

Bit assignment

Bit 0: RVFC.EN, counter validation

Bit 1: RVFC.R, reset of current value

Bit 2: RVFC.RFV, reset of valid measurement frequency status (RVFC.FV)

Bit 3: RVFC.CD, reset of RVFC.D

Object 2450H: RVFC Counter Current Value

Description

This object RVFC.V object contains the current value of the counter in 32 bits.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	no
1	RVFC0 current value	UNSIGNED32	0	ro	yes	no
2	Current value of RVFC1	UNSIGNED32	0	ro	yes	no

Object 2452H: RVFC Counter Capture Value

Description

This object RVFC.C contains the current memorized counter value, in 32 bits, during the front on the RVFC.Ica physical entry.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of fast counter)	UNSIGNED8	2	ro	no	no
1	RVFC0 catch value	UNSIGNED32	0	ro	yes	no
2	RVFC1 catch value	UNSIGNED32	0	ro	yes	no

Object 2454H: RVFC Counter Preset Value

Description

This object RVFC.P contains the counter preset value (coded on 32 bits).

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	yes
1	RVFC0 preset value	UNSIGNED32	FFFF FFFFH	rw	no	yes
2	RVFC1 preset value	UNSIGNED32	FFFF FFFFH	rw	no	yes

Object 2455H: RVFC Counter Delta Value

Description

This object defines the Delta value (coded on 32 bits) which will trigger a PDO send.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of counter)	UNSIGNED8	2	ro	no	yes
1	RVFC0 Delta value	UNSIGNED32	0	rw	no	yes
2	RVFC1 Delta value	UNSIGNED32	0	rw	no	yes

Object 2457H: RVFC Counter Threshold TH0

Description

This object (RVFC.TH0) contains the value (coded on 32 bits) of the TH0 low threshold which will trigger a PDO send and/or switching of reflex outputs.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	yes
1	RVFC0TH0 threshold value	UNSIGNED32	0	rw	no	yes
2	RVFC1TH0 threshold value	UNSIGNED32	0	rw	no	yes

Object 2458H: RVFC Counter Threshold TH1

Description

This object (RVFC.TH1) contains the value (coded on 32 bits) of the TH1 high threshold which will trigger a PDO send and/or switching of reflex outputs.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	yes
1	RVFC0TH1 threshold value	UNSIGNED32	FFFF FFFFH	rw	no	yes
2	RVFC1TH1 threshold value	UNSIGNED32	FFFF FFFFH	rw	no	yes

Object 2500H: RPLS/RPWM Status

Description

This object contains the pulse generator status word.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of pulse generator)	UNSIGNED8	2	ro	no	no
1	Status of RPLS0/RPWM0	UNSIGNED16	0	ro	yes	no
2	Status of RPLS1/RPWM1	UNSIGNED16	0	ro	yes	no

Bit assignment

Bit 0: RPLS.Q or RPWM.Q at state 1, it indicates that the pulse signal is generated at the dedicated output configured.

Bit 1: RPLS.D when set to 1, signal generation is complete. The number of desired pulses has been reached.

Object 2501H: RPLS/RPWM Commands

Description

This object contains the pulse generator command word.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of pulse generator)	UNSIGNED8	2	ro	no	no
1	RPLS0/RPWM0 parameters	UNSIGNED16	0	rw	yes	no
2	RPLS1/RPWM1 parameters	UNSIGNED16	0	rw	yes	no

Bit assignment

Bit 0: RPLS.EN or RPMW.EN. When set to 1, the pulse generation is authorized on the dedicated output. When set to 0, the output channel is set to 0.

Bit 1: RPLS.R or RPMW.R. When set to 1, outputs Q and D are reset to 0. The number of pulses generated over a period T is reset to 0.

Object 2502H: RPLS/RPWM Time Base

Description

This object RPxx.TB contains the pulse generator time base.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of pulse generator)	UNSIGNED8	2	ro	no	yes
1	RPLS0/RPWM0 time base	UNSIGNED16	0	ro	no	yes
2	RPLS1/RPWM1 time base	UNSIGNED16	0	ro	no	yes

Time base

RPxx.TB :
- 0 : 0.127 ms
- 1 : 0.508 ms
- 2 : 10 ms
- 3 : 1 s

Object 2503H: RPLS/RPWM Preset Period

Description

This object RPxx.P contains the pulse generator period. The period is equal to RPxx.P multiplied by RPxx.TB.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of pulse generator)	UNSIGNED8	2	ro	no	yes
1	RPLS0/RPWM0 preset value	UNSIGNED16	0	rw	no	yes
2	RPLS1/RPWM1 preset value	UNSIGNED16	0	rw	no	yes

Period coefficient

RPxx.P :
- 0 : Generator inactive
- 0< P $\leq$ 255 with a time base of 0.127 ms or 0.508 ms
- 1< P $\leq$ 65 535 (FFFFH) with a time base of 10 ms or 1 s

Object 2504H: RPLS/RPWM Mode

Description

This object RPxx.M contains the pulse generator operating mode.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of pulse generator)	UNSIGNED8	2	ro	no	yes
1	RPLS0/RPWM0 mode	UNSIGNED16	0	rw	no	yes
2	RPLS1/RPWM1 mode	UNSIGNED16	0	rw	no	yes

Operating mode

RPxx.M :

- 0 : Not used
- 1 : RPLS
- 2 : RPWM

Object 2505H: RPWM Ratio

Description

This object RPWM.RA contains the signal duty cycle (Duration of high status/ Period).

Duty cycle: $0\% \leq RA \leq 100\%$.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of pulse generator)	UNSIGNED8	2	ro	no	yes
1	RPWM0 duty cycle	UNSIGNED16	32H (50%)	rw	no	yes
2	RPWM1 duty cycle	UNSIGNED16	32H (50%)	rw	no	yes

Object 2506H: RPLS Interrupt Trigger Selection

Description

This object defines the event that triggers the PDO transmission.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of pulse generator)	UNSIGNED8	2	ro	no	yes
1	Trigger event for RPLS0	UNSIGNED8	1	rw	no	yes
2	Trigger event for RPLS1	UNSIGNED8	1	rw	no	yes

Bit assignment

Bit 0: PDO sent on change in value of RPxx.D bit

Object 2507H: RPLS Number of Pulses

Description

This object (RPLS.N) contains the number of pulses (coded on 16 bits) that should be created by the RPLS generator.

Number of pulses:
- 0 : Unlimited number of pulses
- 1< N $\leq$ 65 535 (FFFFH)

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of pulse generator)	UNSIGNED8	2	ro	no	yes
1	RPLS0 pulse number	UNSIGNED16	0	rw	no	yes
2	RPLS1 pulse number	UNSIGNED16	0	rw	no	yes

Object 2508H: RPLS/RPWM Fallback Mode

Description

This object (RPxx.EM) indicates the behavior to adopt in case of a communication fault.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of pulse generator)	UNSIGNED8	2	ro	no	yes
1	RPLS0/RPWM0 fallback mode	UNSIGNED8	0	rw	no	yes
2	RPLS1/RPWM1 fallback mode	UNSIGNED8	0	rw	no	yes

Fallback mode

RPxx.EM :

- 0 : Generator reset with zeroing of output
- 1 : Stop at the end of current interval
- 2 : Continue generating pulses

Object 2557H: RPLS Number of Pulses

Description

This object (RPLS.N) contains the number of pulses (coded on 32 bits) that should be created by the RPLS generator.

Number of pulses:
- 0 : Unlimited number of pulses
- 1< N $\leq$ 4 294 967 295 (FFFF FFFFH)

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of pulse generator)	UNSIGNED8	2	ro	no	yes
1	Number of pulses of RPLS0	UNSIGNED32	0	rw	no	yes
2	Number of pulses of RPLS1	UNSIGNED32	0	rw	no	yes

Object 3000H: Module Specific Diagnostic

Description

This object contains the diagnostics of the communication and expansion modules.

Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of sub-indexes (number of expansion modules)	UNSIGNED8	n	ro	no	no
1	Communication module diagnostics	UNSIGNED16	0	ro	yes	no
2	Diagnostics for first expansion module	UNSIGNED16	0	ro	yes	no
...		UNSIGNED16	0	ro		
n	Diagnostics for last expansion module	UNSIGNED16	0	ro	yes	no

Bit Assignment for Communication Module

Bit 2: Hardware fault (external power supply fault, common fault on all channels)

Bit 3: Module configuration fault

Bit 8: Value error on last write operation

Bit 9: Value inconsistent on last write operation

Bit Assignment for a Discrete I/O Module

Bit 2: Hardware failure (external power supply failure, common fault on all channels)

Bit 3: Module configuration fault

Bit 8: Value error on last write operation

Bit 9: Value inconsistent on last write operation

Bit Assignment for an Analog I/O Module

Type	Description of Diagnostics Register
Analog modules TM2 AMI2HT TM2 AMO1HT TM2 AMM3HT TM2 ALM3LT TM2 AMI2LT	Bit [0]: Fault present Bit [1]: Module being initialized (or initialization of the data on all channels) Bit [2]: Hardware fault (external power supply fault, common to all channels) Bit [3]: Analog expansion module configuration fault Bit [4]: Conversion of data input channel 0 in progress Bit [5]: Conversion of data input channel 1 in progress Bit [6]: Input thermocouple channel 0 not configured Bit [7]: Input thermocouple channel 1 not configured Bit [8]: Not used Bit [9]: Configuration inconsistent Bit [10]: Analog input data channel 0 beyond the range Bit [11]: Analog input data channel 1 beyond the range Bit [12]: Incorrect connection (analog input data channel 0 below the current range, current loop open) Bit [13]: Incorrect connection (analog input data channel 1 below the current range, current loop open) Bit [14]: Mixture of analog input types prohibited Bit [15]: Invalid parameter for the output channel
Analog module TM2 AMM6HT	Bit [1], Bit [0]: Channel 0 input ● 0 , 0: No fault ● 0 , 1: configuration fault ● 1 , 0: Incorrect value (module being initialized or conversion in progress) ● 1 , 1: incorrect value (outside range) Bit [3], Bit [2]: Channel 1 input description identical to channel 0 Bit [5], Bit [4]: Channel 2 input description identical to channel 0 Bit [7], Bit [6]: Channel 3 input description identical to channel 0 Bit [9], Bit [8]: Channel 0 output description identical to channel 0 input Bit [11], Bit [10]: Channel 1 output description identical to channel 0 input Bit [12...15]: Not used
Analog module TM2 AVO2HT	Bit 1, Bit 0: Channel 0 output ● 0 , 0: No fault ● 0 , 1: Configuration fault ● 1 , 0: incorrect value Bit 3, Bit 2: Channel 1 output ● 0 , 0: No fault ● 0 , 1: Configuration fault ● 1 , 0: Incorrect value Bit 4..0.15: Not used

Type	Description of Diagnostics Register
Analog module TM2 AMI4LT	Bit 1, Bit 0: Channel 0 input • 0 , 0: No fault • 0 , 1: Configuration fault • 1 , 0: Incorrect value (module being initialized or conversion in progress) • 1 , 1: Incorrect value (outside the range) Bit 3, Bit 2: Channel 1 input description identical to channel 0 Bit 5, Bit 4: Channel 2 input description identical to channel 0 Bit 7, Bit 6: Channel 3 input description identical to channel 0 Bit 8...15: Not used
Analog modules TM2 AMI8HT TM2 ARI8HT TM2 ARI8LT TM2 ARI8LRJ	Bit 1, Bit 0: Channel 0 input • 0 , 0: No fault • 0 , 1: Configuration fault • 1 , 0: Incorrect value (module being initialized or conversion in progress) • 1 , 1: Incorrect value (outside the range) Bit 3, Bit 2: Channel 1 input description identical to channel 0 Bit 5, Bit 4: Channel 2 input description identical to channel 0 Bit 7, Bit 6: Channel 3 input description identical to channel 0 Bit 9, Bit 8: Channel 4 input description identical to channel 0 Bit 11, Bit 10: Channel 5 input description identical to channel 0 Bit 13, Bit 12: Channel 6 input description identical to channel 0 Bit 15, Bit 14: Channel 7 input description identical to channel 0

NOTE: Bit values:

- 0: No error
- 1: Error

Object 3001H: RFC Counter Diagnostic

Description

This object contains the RFC counter diagnostics.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RFC counter)	UNSIGNED8	2	ro	no	no
1	RFC0 diagnostics	UNSIGNED16	0	ro	yes	no
2	RFC1 diagnostics	UNSIGNED16	0	ro	yes	no

Bit assignment

Bit 8: Value error occurred on last write

Bit 9: Value consistency error occurred on last write

Object 3002H: RVFC Counter Diagnostic

Description

This object contains the RVFC counter diagnostics.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of RVFC counter)	UNSIGNED8	2	ro	no	no
1	RVFC0 diagnostics	UNSIGNED16	0	ro	yes	no
2	RVFC1 diagnostics	UNSIGNED16	0	ro	yes	no

Bit assignment

Bit 8: Value error occurred on last write

Bit 9: Value consistency error occurred on last write

Object 3003H: RPSL/RPWM Diagnostic

Description

This object contains the pulse generator diagnostics.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (Number of fast counter)	UNSIGNED8	2	ro	no	no
1	RPSL0/RPWM0 diagnostics	UNSIGNED16	0	ro	yes	no
2	RPSL1/RPWM1 diagnostics	UNSIGNED16	0	ro	yes	no

Bit assignment

Bit 8: Value error occurred on last write

Bit 9: Value consistency error occurred on last write

Object 3200H: Parameters Status (OTB island)

Description

This object contains the parameters status.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED16	0	ro	no	no

Parameters used

Object value:

0: the default parameters are used

1: the saved parameters are used

2: the current (unsaved) parameters are used

Object 3201H: Configuration Stack Number

Description

This object contains the number of backups made since the latest restoration of default parameters.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED16	-	ro	no	no

Object 3202H: Restore Last Saved Parameters

Description

This object is used to restore the last saved parameters.

⚠ WARNING

RISK OF UNEXPECTED EQUIPMENT OPERATION

Restoring parameters during the "Operational" state can lead to unexpected equipment operation, equipment damage and/or injury to personnel. If this must be done, place the OTB module in "Pre-operational" mode or, if need be, ensure equipment and personnel safety before attempting any new operations.

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

Properties

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Restoring Parameters	UNSIGNED32	1	rw	no	no

Operation

To restore the parameters, the "load" ASCII string (6461 6F6CH) must be written to the corresponding index:

	Most significant word		Least significant word	
ISO 8859 (ASCII) signature	d	a	o	l
Value	64H	61H	6FH	6CH

NOTE:

- This command is ignored if restoration is already in progress (object 3202H set to 0).
- Parameters are immediately restored.

Restored communication profile parameters

The objects that are restored with saved parameters are as follows:

- 1005H : COB-ID SYNC Message
- 1006H : Communication Cycle Period
- 100CH : Guard Time
- 100DH : Life Time Factor
- 1014H : COB-ID Emergency (EMCY) Message
- 1016H : Consumer Heartbeat Time
- 1017H : Producer Heartbeat Time
- 1400H...1407H : 1st to 8th Receive PDO Communication Parameter
- 1600H...1607H : 1st to 8th Receive PDO Mapping Parameter
- 1800H...1807H : 1st to 8th Transmit PDO Communication Parameter
- 1A00H...1A07H : 1st to 8th Transmit PDO Mapping Parameter

Restored parameters of the manufacturer-specific zone

The objects that are restored with saved parameters are as follows:

- 2000H : Local Digital Input Parameter
- 2100H : Analog Input Type
- 2101H : Analog Input Range
- 2102H : Analog Input Min
- 2103H : Analog Input Max
- 2104H : R_0 Value Register
- 2105H : T_0 Value Register
- 2106H : B Value Register
- 2200H : Analog Output Type
- 2201H : Analog Output Range
- 2202H : Analog Output Min
- 2203H : Analog Output Max
- 2303H : Counter Mode
- 2304H : Counter Preset Value
- 2305H : Counter Delta Value
- 2306H : Counter Interrupt Trigger Selection
- 2307H : Counter Error Mode
- 2354H : Counter Preset Value
- 2355H : Counter Delta Value
- 2403H : Fast Counter Parameters Value
- 2404H : Fast Counter Preset Value
- 2405H : Fast Counter Delta Value
- 2406H : Fast Counter Interrupt Trigger Selection
- 2407H : Fast Counter Threshold S0
- 2408H : Fast Counter Threshold S1
- 2409H : Fast Counter Mode
- 240AH : Fast Counter Error Mode
- 2454H : Fast Counter Preset Value

- 2455H : Fast Counter Delta Value
- 2457H : Fast Counter Threshold S0
- 2458H : Fast Counter Threshold S1
- 2502H : PLS/PWM Time Base
- 2503H : PLS/PWM Preset Period
- 2504H : PLS/PWM Mode
- 2505H : PWM Ratio
- 2506H : PLS/PWM Interrupt Trigger Selection
- 2507H : PLS Number of Pulses
- 2508H : PLS/PWM Error Mode
- 2557H : PLS/PWM Time Base

Restored hardware profile parameters

The objects that are restored with saved parameters are as follows:
- 6102H : Polarity Input 16 Bits
- 6103H : Filter Constant Input 16 Bits
- 6302H : Polarity Output 16 Bits
- 6306H : Fallback Mode Output 16 Bits
- 6307H : Fallback Value Output 16 Bits
- 6308H : Filter Mask Output 16 Bits
- 6421H : Analog Input interrupt Trigger Selection
- 6424H : Analog Input interrupt Upper Limit Integer
- 6425H : Analog Input interrupt Lower Limit Integer
- 6426H : Analog Input interrupt Delta Value
- 6443H : Analog Output Fallback Mode
- 6444H : Analog Output Fallback Value

Object 3300H: Extension Bus Reset

Description

This object is used to authorize expansion expansion module parameter configuration updates:

- When the value is set to 1, the internal bus is stopped.
- When the value is set to 0, the internal bus is activated if the expansion module parameters are consistent.

WARNING

RISK OF UNINTENDED EQUIPMENT OPERATION

Stopping the internal bus sets all the outputs to 0.

Stopping the extension bus during the "Operational" state can lead to unexpected equipment operation, injury to personnel and/or equipment damage. If this must be done, place the OTB module in "Pre-operational" mode or, if need be, ensure equipment and personnel safety before any new operations.

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

Properties

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	-	UNSIGNED16	0	rw	no	no

9.4 Hardware Profile Objects 6000H to 9FFFH

Introduction

This section lists the objects relating to the hardware profile. Each object, with all its technical characteristics, is described according to the CANopen standard.

What's in this Section?

This section contains the following topics:

Object 6000H: Read Input 8 Bits

Description

This object contains the value of discrete inputs in 8 bit format.

Sub-index Management:
- Each discrete expansion module uses an even number of sub-indexes (alignment with 16 bit words).
- The expansion modules and their sub-indexes are numbered from left to right and from top to bottom.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-indexes (Number of 8 bit inputs)	UNSIGNED8	n	ro	no	no
1	Read input 0 to 7 of communication module	UNSIGNED8	-	ro	yes	no
2	Read input 8 to 15 of communication module	UNSIGNED8	-	ro	yes	no
3	Read input 0 to 7 of first expansion module	UNSIGNED8	-	ro	yes	no
...						
n	Read of last 8 inputs of last expansion module	UNSIGNED8	-	ro	yes	no

Object 6100H: Read Input 16 Bits

Description

This object contains the value of discrete inputs in 16 bit format.

NOTE: For a 32-input discrete expansion module two sub-indexes are used.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-in-dex	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (number of 16 bit inputs)	UNSIGNED8	-	ro	no	no
1	Read of inputs 0 to 15 of communication module	UNSIGNED16	-	ro	yes	no
2	Read inputs 0 to 15 of first expansion module	UNSIGNED16	-	ro	yes	no
...						
n	Read of last 16 inputs of last expansion module	UNSIGNED16	-	ro	yes	no

Object 6102H: Polarity Inputs 16 Bits

Description

This object is used to define the polarity of inputs:

- 0 = Input not reversed
- 1= reversed input

NOTE: I/O status LEDs continue to show the actual electrical state of connected hardware and are not affected by this object.

Properties

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-indexes (number of 16 bit inputs)	UNSIGNED8	n	ro	no	yes
1	Input polarity of communication module	UNSIGNED16	0	rw	no	yes
2	Input polarity of first expansion module	UNSIGNED16	0	rw	no	yes
...						
n	Input polarity of last expansion module	UNSIGNED16	0	rw	no	yes

Object 6103H: Filter Mask Input 16 Bits

Description

This object is used to configure the mask for inputs.

- 0 : input read
- 1 : input ignored

NOTE: I/O status LEDs continue to show the actual electrical state of connected hardware and are not affected by this object.

The change in polarity does not affect the behavior described above.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (number of 16 bit inputs)	UNSIGNED8	n	ro	no	yes
1	Input mask of communication module	UNSIGNED16	0	rw	no	yes
2	Input mask of first expansion module	UNSIGNED16	0	rw	no	yes
...						
n	Input mask of last expansion module	UNSIGNED16	0	rw	no	yes

Object 6200H: Write Outputs 8 Bits

Description

This object commands the status of discrete outputs.

Sub-index Management:

- Each discrete expansion module uses an even number of sub-indexes (alignment with 16 bit words).
- The expansion modules and their sub-indexes are numbered from left to right and from top to bottom.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	n	ro	no	no
1	Discrete output of communication module	UNSIGNED8	0	rw	yes	no
2	Not used as the communication module only has 8 outputs	UNSIGNED8	0	rw	yes	no
3	Discrete output of first expansion module	UNSIGNED8	0	rw	yes	no
...						
n	Discrete output of last expansion module	UNSIGNED8	0	rw	yes	no

Object 6300H: Write Outputs 16 Bits

Description

This object commands the status of discrete outputs.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (number of 16 bit outputs)	UNSIGNED8	n	ro	no	no
1	Discrete output of communication module	UNSIGNED16	0	rw	yes	no
2	Discrete output of first expansion module	UNSIGNED16	0	rw	yes	no
...						
n	Discrete output of last expansion module	UNSIGNED16	0	rw	yes	no

Object 6302H: Polarity Outputs 16 Bits

Description

This object commands the polarity for a group of 16 discrete outputs.

NOTE: I/O status LEDs continue to show the actual electrical state of connected hardware and are not affected by this object.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (number of 16 bit outputs)	UNSIGNED8	n	ro	no	yes
1	Polarity of communication module's discrete outputs	UNSIGNED16	0	rw	no	yes
2	Polarity of first expansion module's discrete outputs	UNSIGNED16	0	rw	no	yes
...						
n	Polarity of last expansion module's discrete outputs	UNSIGNED16	0	rw	no	yes

Polarity

The characteristics of the outputs are outlined in the following table:

Status	Description
0 (output not reversed)	• 0 V or contact open = 0 • 24 V or contact closed = 1
1 (output reversed)	• 0 V or contact open = 1 • 24 V or contact closed = 0

Object 6306H: Fallback Mode Outputs 16 Bits

Description

This object indicates the fallback mode adopted by outputs in the event of an internal fault or a communication fault.

Status	Description
0	Maintain value
1	Fallback value (defined in object 6307H)

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (number of 16 bit outputs)	UNSIGNED8	n	ro	no	yes
1	Fallback mode of communication module	UNSIGNED16	FFFFH	rw	no	yes
2	Fallback mode of first expansion module	UNSIGNED16	FFFFH	rw	no	yes
...						
n	Fallback mode of last expansion module	UNSIGNED16	FFFFH	rw	no	yes

Object 6307H: Fallback Value Output 16 Bits

Description

The object indicates the fallback value adopted by outputs in the event of an internal fault or a communication fault if the corresponding bit in the object 6306H is at 1.

If the object 6308H is at 0, the fallback is not operational.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (number of 16 bit outputs)	UNSIGNED8	n	ro	no	yes
1	Fallback value of communication module	UNSIGNED16	0	rw	no	yes
2	Fallback value of first expansion module	UNSIGNED16	0	rw	no	yes
...						
n	Fallback value of last expansion module	UNSIGNED16	0	rw	no	yes

Object 6308H: Filter Mask Output 16 Bits

Description

This object defines the filter mask for a group of 16 outputs.

Status	Description
0	Current output value is frozen
1	Authorizes writing to output (corresponding to the value of object 6200H or 6300H)

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-index (number of 16 bit outputs)	UNSIGNED8	n	ro	no	yes
1	Filter mask of communication module	UNSIGNED16	FFFFH	rw	no	yes
2	Filter mask of first expansion module	UNSIGNED16	FFFFH	rw	no	yes
...						
n	Filter mask of last expansion module	UNSIGNED16	FFFFH	rw	no	yes

Object 6401H: Read Analog Input 16 Bits

Description

Analog input values are stored in this object.

Sub-index Management:

- Each channel uses one sub-index.
- The expansion modules and their sub-indexes are numbered from left to right and from top to bottom.

NOTE: This rule applies to all objects relating to analog inputs.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number n of analog channels	UNSIGNED8	n	ro	no	no
1	Value of analog entry on channel 0 of first analog module	UNSIGNED16	0	ro	yes	no
...						
n	Value of analog entry on last channel of last analog module	UNSIGNED16	0	ro	yes	no

Object 6411H: Write Analog Output 16 Bits

Description

This object writes analog output values.

Managing the sub-indexes:

- Each channel uses 1 sub-index (1 per channel)
- The expansion modules and their sub-indexes are numbered from left to right and from top to bottom.

NOTE: This rule applies to all objects relating to analog outputs.

Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of sub-indexes (Number of analog channels)	UNSIGNED8	n	ro	no	no
1	Writes the value of the channel 0 analog output on the first analog module	INTEGER16	0	rw	yes	no
...						
n	Writes the value of the last analog output on the last analog module	INTEGER16	0	rw	yes	no

Object 6421H: Analog Input Interrupt Trigger Selection

Description

This object defines the event that triggers PDO transmission if object 6423H is TRUE.

Characteristics

The characteristics of this object are outlined in the table below:

Sub-Index	Description	Data Type	Default Value	Access	PDO Mapping	Saved
0	Number of sub-indexes (number of analog inputs)	UNSIGNED8	n	ro	no	yes
1	Trigger event for the channel 0 analog input on the first analog module	UNSIGNED8	7	rw	no	yes
...						
n	Trigger event for the last analog input on the last analog module	UNSIGNED8	7	rw	no	yes

Bit assignment

Bit 0: Value greater than the upper threshold

Bit 1: Value smaller than the lower threshold

Bit 2: Delta exceeded

Object 6422H: Analog Input Interrupt Source

Description

This object contains the number of the channel that generated the PDO send.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	1	ro	no	no
1	Number of channel that generated the PDO send.	UNSIGNED32	-	ro	yes	no

Object 6423H: Analog Input Global Interrupt Enable

Description

This object authorizes the trigger events of analog inputs. If the value is 1, a PDO will be sent according to the trigger mode set in object 6421H.

Properties

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
-	-	BOOLEAN	0	rw	no	no

NOTE: If the value is 0, no analog PDO will be sent in transmission mode 254 and 255.

Object 6424H: Analog Input Interrupt Upper Limit Integer

Description

This object defines the high threshold that can trigger the transmission of a PDO depending on the configuration of objects 6421H and 6423H.

Properties

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-indexes (number of analog inputs)	UNSIGNED8	n	ro	no	yes
1	High threshold of channel 0	INTEGER32	0	rw	no	yes
...						
n	High threshold of last channel	INTEGER32	0	rw	no	yes

Object 6425H: Analog Input Interrupt Lower Limit

Description

This object defines the low threshold that can trigger the transmission of a PDO depending on the configuration of objects 6421H and 6423H.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-indexes (number of analog inputs)	UNSIGNED8	n	ro	no	yes
1	Low threshold of channel 0	INTEGER32	0	rw	no	yes
...						
n	Low threshold of last channel	INTEGER32	0	rw	no	yes

Object 6426H: Analog Input Interrupt Delta Value

Description

This object defines the Delta value that can trigger the transmission of a PDO depending on the configuration of objects 6421H and 6423H.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number n of analog channels	UNSIGNED8	n	ro	no	yes
1	Delta value of channel 0 of first module	UNSIGNED32	0	rw	no	yes
...						
n	Delta value of last channel of last module	UNSIGNED32	0	rw	no	yes

Object 6443H: Analog Output Fallback Mode

Description

This object commands the fallback mode adopted by outputs in the event of an internal fault or a communication fault.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Number of sub-indexes (number of analog inputs)	UNSIGNED8	n	ro	no	yes
1	Fallback mode for channel 0	UNSIGNED8	1	rw	no	yes
...						
n	Fallback mode for last channel	UNSIGNED8	1	rw	no	yes

Fallback mode

Status	Description
0	Maintain value
1	Fallback value (defined in object 6444H)

Object 6444H: Analog Output Fallback Value

Description

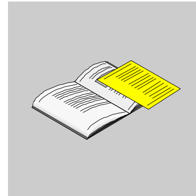
The object indicates the fallback value adopted by outputs in the event of an internal fault or a communication fault if the corresponding sub-index in the object 6443H is at 1.

Characteristics

The characteristics of this object are outlined in the following table:

Sub-index	Description	Data type	Default value	Access	PDO Mapping	Backed up
0	Sub-index number	UNSIGNED8	n	ro	no	yes
1	Fallback value for channel 0	INTEGER32	0	rw	no	yes
...						
n	Fallback value for last channel	INTEGER32	0	rw	no	yes

Appendices



Introduction

This appendix provides information on common IEC symbols used in this manual.

What's in this Appendix?

The appendix contains the following chapters:

IEC Symbols

A

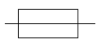
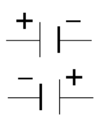
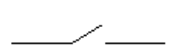
Glossary of Symbols

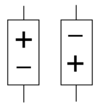
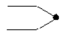
Introduction

This section contains illustrations and definitions of common IEC symbols used in describing wiring schematics.

Symbols

Common IEC symbols are illustrated and defined in the table below:

	Fuse
	Load
	AC power
	DC power
	Digital sensor/input, for example, contact, switch, initiator, light barrier, and so on.
	Earth ground

	2-wire sensor
	Thermocouple element

COB-ID

B

List of COB-IDs

Introduction

This table shows the COB-IDs reserved for PDOs on the CANopen network. There is no default COB-ID after the fifth PDO of a CANopen node. To assign a COB-ID for these PDOs, simply select a COB-ID that is not used by the network. This can be a COB-ID that corresponds to an address of a free node, or one that corresponds to a PDO that is not used by an existing node. In addition to the abovementioned COB-IDs, you can use those in the 0680H to 06DFH range.

NOTE: You must check that the COB-IDs are unique on the network.

List of COB-IDs (RxPDO)

The COB-IDs for RxPDOs on the CANopen network are listed in the following table:

Address	PDO 1400	PDO 1401	PDO 1402	PDO 1403	PDO 1404	PDO 1405	PDO 1406	PDO 1407
0	512	768	1024	1280				
1	513	769	1025	1281				
2	514	770	1026	1282				
3	515	771	1027	1283				
4	516	772	1028	1284				
5	517	773	1029	1285				
6	518	774	1030	1286				
7	519	775	1031	1287				
8	520	776	1032	1288				
9	521	777	1033	1289				
10	522	778	1034	1290				
11	523	779	1035	1291				
12	524	780	1036	1292				
13	525	781	1037	1293				

Address	PDO 1400	PDO 1401	PDO 1402	PDO 1403	PDO 1404	PDO 1405	PDO 1406	PDO 1407
14	526	782	1038	1294				
15	527	783	1039	1295				
16	528	784	1040	1296				
17	529	785	1041	1297				
18	530	786	1042	1298				
19	531	787	1043	1299				
20	532	788	1044	1300				
21	533	789	1045	1301				
22	534	790	1046	1302				
23	535	791	1047	1303				
24	536	792	1048	1304				
25	537	793	1049	1305				
26	538	794	1050	1306				
27	539	795	1051	1307				
28	540	796	1052	1308				
29	541	797	1053	1309				
30	542	798	1054	1310				
31	543	799	1055	1311				
32	544	800	1056	1312				
33	545	801	1057	1313				
34	546	802	1058	1314				
35	547	803	1059	1315				
36	548	804	1060	1316				
37	549	805	1061	1317				
38	550	806	1062	1318				
39	551	807	1063	1319				
40	552	808	1064	1320				
41	553	809	1065	1321				
42	554	810	1066	1322				
43	555	811	1067	1323				
44	556	812	1068	1324				
45	557	813	1069	1325				
46	558	814	1070	1326				
47	559	815	1071	1327				

Address	PDO 1400	PDO 1401	PDO 1402	PDO 1403	PDO 1404	PDO 1405	PDO 1406	PDO 1407
48	560	816	1072	1328				
49	561	817	1073	1329				
50	562	818	1074	1330				
51	563	819	1075	1331				
52	564	820	1076	1332				
53	565	821	1077	1333				
54	566	822	1078	1334				
55	567	823	1079	1335				
56	568	824	1080	1336				
57	569	825	1081	1337				
58	570	826	1082	1338				
59	571	827	1083	1339				
60	572	828	1084	1340				
61	573	829	1085	1341				
62	574	830	1086	1342				
63	575	831	1087	1343				
64	576	832	1088	1344				
65	577	833	1089	1345				
66	578	834	1090	1346				
67	579	835	1091	1347				
68	580	836	1092	1348				
69	581	837	1093	1349				
70	582	838	1094	1350				
71	583	839	1095	1351				
72	584	840	1096	1352				
73	585	841	1097	1353				
74	586	842	1098	1354				
75	587	843	1099	1355				
76	588	844	1100	1356				
77	589	845	1101	1357				
78	590	846	1102	1358				
79	591	847	1103	1359				
80	592	848	1104	1360				
81	593	849	1105	1361				

Address	PDO 1400	PDO 1401	PDO 1402	PDO 1403	PDO 1404	PDO 1405	PDO 1406	PDO 1407
82	594	850	1106	1362				
83	595	851	1107	1363				
84	596	852	1108	1364				
85	597	853	1109	1365				
86	598	854	1110	1366				
87	599	855	1111	1367				
88	600	856	1112	1368				
89	601	857	1113	1369				
90	602	858	1114	1370				
91	603	859	1115	1371				
92	604	860	1116	1372				
93	605	861	1117	1373				
94	606	862	1118	1374				
95	607	863	1119	1375				
96	608	864	1120	1376				
97	609	865	1121	1377				
98	610	866	1122	1378				
99	611	867	1123	1379				
100	612	868	1124	1380				
101	613	869	1125	1381				
102	614	870	1126	1382				
103	615	871	1127	1383				
104	616	872	1128	1384				
105	617	873	1129	1385				
106	618	874	1130	1386				
107	619	875	1131	1387				
108	620	876	1132	1388				
109	621	877	1133	1389				
110	622	878	1134	1390				
111	623	879	1135	1391				
112	624	880	1136	1392				
113	625	881	1137	1393				
114	626	882	1138	1394				
115	627	883	1139	1395				

Address	PDO 1400	PDO 1401	PDO 1402	PDO 1403	PDO 1404	PDO 1405	PDO 1406	PDO 1407
116	628	884	1140	1396				
117	629	885	1141	1397				
118	630	886	1142	1398				
119	631	887	1143	1399				
120	632	888	1144	1400				
121	633	889	1145	1401				
122	634	890	1146	1402				
123	635	891	1147	1403				
124	636	892	1148	1404				
125	637	893	1149	1405				
126	638	894	1150	1406				
127	639	895	1151	1407				

List of COB-IDs TxPDO

The COB-IDs for TxPDOs on the CANopen network are listed in the following table:

Address	PDO 1800	PDO 1801	PDO 1802	PDO 1803	PDO 1804	PDO 1405	PDO 1406	PDO 1407
0	384	640	896	1152				
1	385	641	897	1153				
2	386	642	898	1154				
3	387	643	899	1155				
4	388	644	900	1156				
5	389	645	901	1157				
6	390	646	902	1158				
7	391	647	903	1159				
8	392	648	904	1160				
9	393	649	905	1161				
10	394	650	906	1162				
11	395	651	907	1163				
12	396	652	908	1164				
13	397	653	909	1165				
14	398	654	910	1166				
15	399	655	911	1167				
16	400	656	912	1168				
17	401	657	913	1169				

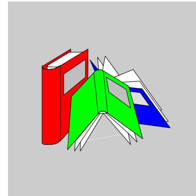
Address	PDO 1800	PDO 1801	PDO 1802	PDO 1803	PDO 1804	PDO 1405	PDO 1406	PDO 1407
18	402	658	914	1170				
19	403	659	915	1171				
20	404	660	916	1172				
21	405	661	917	1173				
22	406	662	918	1174				
23	407	663	919	1175				
24	408	664	920	1176				
25	409	665	921	1177				
26	410	666	922	1178				
27	411	667	923	1179				
28	412	668	924	1180				
29	413	669	925	1181				
30	414	670	926	1182				
31	415	671	927	1183				
32	416	672	928	1184				
33	417	673	929	1185				
34	418	674	930	1186				
35	419	675	931	1187				
36	420	676	932	1188				
37	421	677	933	1189				
38	422	678	934	1190				
39	423	679	935	1191				
40	424	680	936	1192				
41	425	681	937	1193				
42	426	682	938	1194				
43	427	683	939	1195				
44	428	684	940	1196				
45	429	685	941	1197				
46	430	686	942	1198				
47	431	687	943	1199				
48	432	688	944	1200				
49	433	689	945	1201				
50	434	690	946	1202				
51	435	691	947	1203				

Address	PDO 1800	PDO 1801	PDO 1802	PDO 1803	PDO 1804	PDO 1405	PDO 1406	PDO 1407
52	436	692	948	1204				
53	437	693	949	1205				
54	438	694	950	1206				
55	439	695	951	1207				
56	440	696	952	1208				
57	441	697	953	1209				
58	442	698	954	1210				
59	443	699	955	1211				
60	444	700	956	1212				
61	445	701	957	1213				
62	446	702	958	1214				
63	447	703	959	1215				
64	448	704	960	1216				
65	449	705	961	1217				
66	450	706	962	1218				
67	451	707	963	1219				
68	452	708	964	1220				
69	453	709	965	1221				
70	454	710	966	1222				
71	455	711	967	1223				
72	456	712	968	1224				
73	457	713	969	1225				
74	458	714	970	1226				
75	459	715	971	1227				
76	460	716	972	1228				
77	461	717	973	1229				
78	462	718	974	1230				
79	463	719	975	1231				
80	464	720	976	1232				
81	465	721	977	1233				
82	466	722	978	1234				
83	467	723	979	1235				
84	468	724	980	1236				
85	469	725	981	1237				

Address	PDO 1800	PDO 1801	PDO 1802	PDO 1803	PDO 1804	PDO 1405	PDO 1406	PDO 1407
86	470	726	982	1238				
87	471	727	983	1239				
88	472	728	984	1240				
89	473	729	985	1241				
90	474	730	986	1242				
91	475	731	987	1243				
92	476	732	988	1244				
93	477	733	989	1245				
94	478	734	990	1246				
95	479	735	991	1247				
96	480	736	992	1248				
97	481	737	993	1249				
98	482	738	994	1250				
99	483	739	995	1251				
100	484	740	996	1252				
101	485	741	997	1253				
102	486	742	998	1254				
103	487	743	999	1255				
104	488	744	1000	1256				
105	489	745	1001	1257				
106	490	746	1002	1258				
107	491	747	1003	1259				
108	492	748	1004	1260				
109	493	749	1005	1261				
110	494	750	1006	1262				
111	495	751	1007	1263				
112	496	752	1008	1264				
113	497	753	1009	1265				
114	498	754	1010	1266				
115	499	755	1011	1267				
116	500	756	1012	1268				
117	501	757	1013	1269				
118	502	758	1014	1270				
119	503	759	1015	1271				

Address	PDO 1800	PDO 1801	PDO 1802	PDO 1803	PDO 1804	PDO 1405	PDO 1406	PDO 1407
120	504	760	1016	1272				
121	505	761	1017	1273				
122	506	762	1018	1274				
123	507	763	1019	1275				
124	508	764	1020	1276				
125	509	765	1021	1277				
126	510	766	1022	1278				
127	511	767	1023	1279				

Glossary



A

Analog Input

A module containing circuits that enable analog DC (direct current) input signals to be converted into digital values that can be handled by the processor. This implies that the analog inputs are generally direct values — in other words: a value in the data table is a direct reflection of the analog signal value.

Analog output

A module containing circuits that transmit a DC (direct current) analog signal proportional to a digital input to the processor module. This implies that the analog outputs are generally direct values — in other words: a value in the data table directly governs the analog signal value.

Auto-baud

Automatic detection of the bus transmission speed, when the thumbwheel is on position 8.

C

CAN

Controller Area Network. The CAN protocol (ISO 11898) for serial bus networks was designed for interconnecting a series of intelligent devices (produced by numerous manufacturers) in intelligent systems for industrial applications in real time. CAN systems with multiple masters ensure a high level of data integrity, due to the implementation of message broadcast mechanisms and a strict error control procedure. Initially developed for the automotive industry, the CAN protocol is now used in a wide range of control system monitoring environments.

CANopen, protocol

A standard industrial protocol used on the internal communication bus. This protocol can be used to connect any standard CANopen device to the island bus.

CiA

CAN in Automation. The acronym CiA designates a non-profit-making association of manufacturers and users who are keen to promote and develop the use of higher-level protocols, based on CAN.

COB-ID

Communication Object. A communication object is a unit for transporting (a message) on a CAN network. Communication objects indicate a device's specific functionality. They are specified in the CANopen communication profile.

D

Digital input/output

Another expression used is discrete input/output. Designates an input or output with a connection by individual circuit to the module corresponding directly to a data table bit or word storing the value of the signal in this I/O circuit. A digital I/O allows the control logic discrete access to the I/O values.

DIN

"Deutsch Industrie Norm". German standards organization that defines dimensional and engineering standards. These standards are currently recognized worldwide.

E

EDS

Electronic Data Sheet. The EDS is a standard ASCII file containing information about a network device communication function and the content of its object dictionary. The EDS also defines objects specific to the device and the manufacturer.

EIA

Electronic Industries Association. Organization that draws up data and electrical/electronic communication standards.

EMC

Electro-Magnetic Compatibility. Devices that comply with EMC requirements are capable of error-free operation within the specified electro-magnetic limits of the system.

F

Fallback mode

A secure mode to which any Advantys I/O module can revert should the communication connection fail.

Fallback value

The value adopted by a device when it enters the fallback state. Generally, the fallback value is either configured, or is the device's last stored value.

I

IEC

International Electrotechnical Commission. Commission officially founded in 1906 and devoted to the advancement of theory and practice in the following sciences: electrical engineering, electronic engineering, information technology and computer engineering. The IEC1131 standard covers industrial automation equipment.

IEC 1 type input

Type 1 digital inputs support sensor signals from mechanical switching devices such as relay contacts and push-buttons operating under normal climatic conditions.

Input filtering

The period during which a sensor must keep its signal activated/deactivated before the input module detects a change of state.

Input polarity

The polarity of an input channel determines when the input module sends a 1 (one) and when it sends a 0 (zero) to the master controller. If the polarity is *normal*, an input channel will send a 1 (one) to the controller as soon as its fieldbus sensor is activated. If the polarity is *reversed*, an input channel will send a 0 (zero) to the controller as soon as its fieldbus sensor is activated.

L

LSB

Least Significant Byte. The part of a number, an address or a field that is written as the value furthest to the right in conventional hexadecimal or binary notation.

LSb

Least Significant bit. The part of a number, an address or a field that is written as the value furthest to the right in conventional hexadecimal or binary notation.

M

MSB

Most Significant Byte. The part of a number, an address or a field that is written as the value furthest to the left in conventional hexadecimal or binary notation.

MSb

Most Significant bit. The part of a number, an address or a field that is written as the value furthest to the left in conventional hexadecimal or binary notation.

N

NEMA

National Electrical Manufacturers Association.

NMT

*Network management.*NMT protocols offer services for network initialization, monitoring errors and monitoring the device status.

NO contact

Normally open contact. Also called make contacts. A pair of contact relays that is open when the relay coil is low and closed when it is energized.

O

Object dictionary

This element of the CANopen device model constitutes the internal structure plan for CANopen devices (according to the CANopen DS-401 profile). A given device's object dictionary is a conversion table which describes the types of data, the communication objects and the application objects used by the device. By accessing the object dictionary structure for a specific device via the CANopen fieldbus, you can predict how it will behave on the network and thus design a distributed application capable of implementing it.

Output polarity

The polarity of an output channel determines when the output module activates its fieldbus actuator and when it deactivates it. If the polarity is *normal*, an output channel will activate its actuator as soon as the master controller sends it the value 1. If the polarity is *reversed*, an output channel will activate its actuator as soon as the master controller sends it the value 0.

P

PDO

Process Data Object. On networks based on the CAN technology, PDOs (process data objects) are transmitted as unconfirmed broadcast messages or sent from a producer device to a consumer device. The transmit PDO object (TxPDO) from the producer device has a specific identifier corresponding to the receive PDO object (RxPDO) on client devices.

Positive logic (source)

Designates an output which, when powered up, receives DC current from its load.

Producer/consumer model

On networks adhering to the producer/consumer model, the data packets are identified according to their data content rather than their physical position. All the nodes listen to the network and consume data packets with identifiers corresponding to their functionality.

R

Repeater

An interconnection device that extends the permitted bus length.

RMS

Root Mean Square. The rms value of an alternating current, corresponding to the DC value which produces the same thermal effect. The RMS value is calculated by taking the square root of the mean of the squares of the instantaneous amplitude of a given full cycle. For a sinusoidal wave, the rms value corresponds to 0.707 of the peak value.

RTR

Remote Transmission Request. Exchange mechanism initiated by the CANopen master.

Rx

Reception. On a CAN network, for example, a PDO object is described as being an RxPDO of the slave device that receives it.

S

SDO

Service Data Object. On networks based on CAN, the fieldbus master (CANopen) uses the SDO messages to access (in read/write mode) the object dictionaries of the network nodes.

T

Thermocouple

A TC (thermocouple) consists of a bi-metal temperature transducer that gives a temperature value by measuring the difference in potential caused by the joining of two different metals, at different temperatures.

Tx

Transmission. On a CAN network, for example, a PDO object is described as being an TxPDO of the slave device that transmits it.

Index