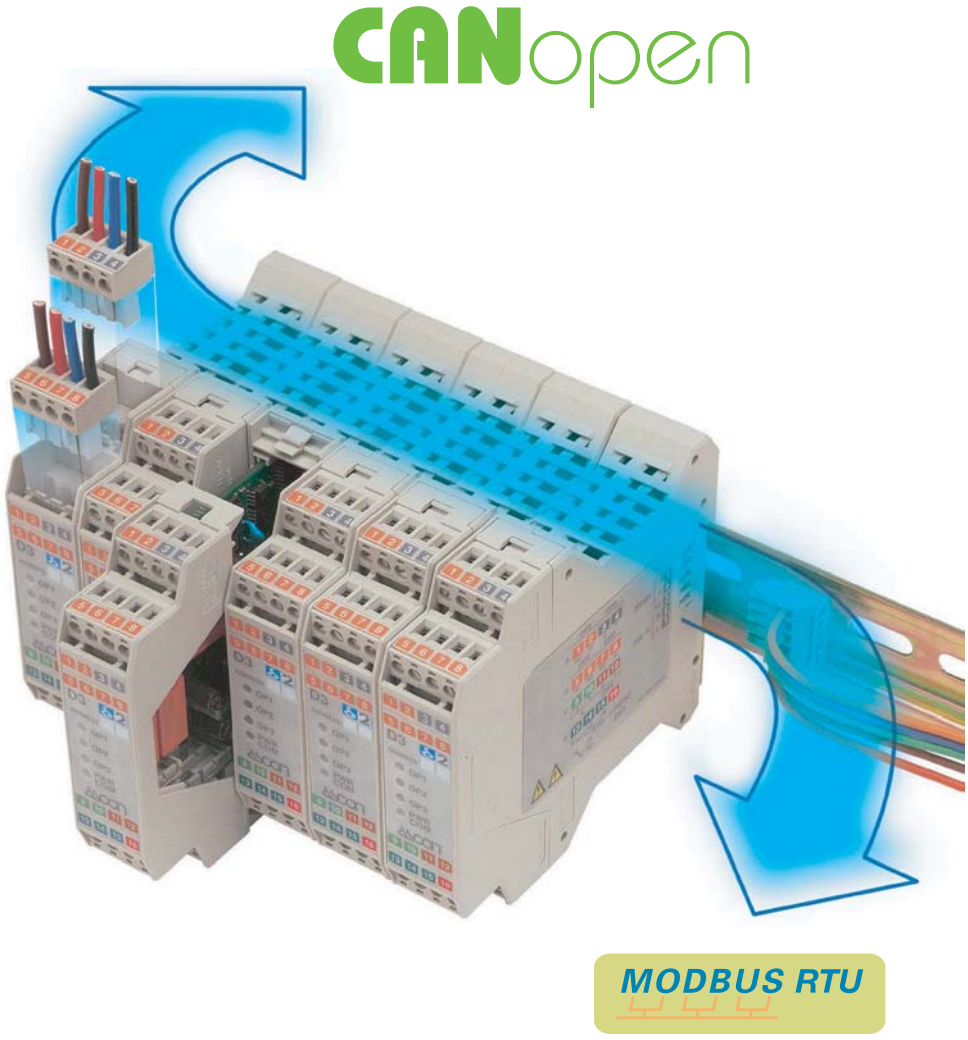
deltadue **DY-5030 Protocol Converter**
® **CANopen/Modbus Master**

S E R I E S

User Manual
M.U. DY-5030-5/10.11
Cod. J30 - 478 - 1ADY E



Copyright © 2008, 2010 ASCON spa

All rights reserved

No part of this document may be stored in a retrieval system, or transmitted in any form, electronic or mechanical, without prior written permission of ASCON Spa.

ASCON has exercised care in preparing this book and believes the information contained within to be accurate. The ASCON Products are subjected to continuous improvement in the pursuit the technological leadership. These improvements could require changes to the information contained within this book. ASCON reserves the right to change such information without notice. ASCON makes no warranty of any kind, expressed or implied, with regard to the documentation contained in this book. ASCON shall not be liable for any incidental and consequential damages in connection with, or arising out of the use of this book.

sigma**due**®, gamma**due**® and delta**due**® are trademarks of ASCON spa.

All the other tradenames or product name are trademark or registered trademarks.

ASCON spa
Via Falzarego 9/11
20021 Baranzate (MI) Italy
Fax ++39-02-3504243
http://www.ascon.it
support@ascon.it

INDEX

Prerequisites

Updated documentation

It is advised that you always check our Internet site (www.ascon.it) for the most current updates. Access the English version by clicking on the English flag and choosing English site.

Select: *Download/Documentation*, and fill the table with:

- Typology **Manual**,
- Type: **User**,
- Language: **EN • English**
- Code: **DY**

Click “SEARCH” and

- Download the file: **Ascon_MIU_DY-5030_EN.zip.**

Revision list

Revision	Date	Author	Chapter	Description
1.00	07/2006	ASCON	All	First release version
2.00	03/2007	ASCON	All	Second release version
3.00	09/2007	ASCON	All	Third release version
4.00	11/2008	ASCON	All	Fouth release version
5.00	11/2010	ASCON	All	Fifth release version

Using this manual

Specifications within the text of this manual are given in the International System of Units (SI), with non SI equivalents in parentheses.

Fully Capitalized words within the text indicate markings found on the equipment.

Words **in bold** style within the text indicate markings found in the Configuration Tools.

Warnings, Cautions and Notes are used to emphasize critical instructions:

DANGER!

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

WARNING
Indicates a potentially hazardous situation which, if not avoided, could result in death or serious injury.

Caution
Indicates a potentially hazardous situation which, if not avoided, may result in minor or moderate injury, or property damage.

Note: Highlights important information about an operating procedure or the equipment.

Current Documentation on the Internet

Make sure you are always working with the latest version of this document.

ASCON spa reserves the right to make changes to its products in the name of technological advancement. New manual revisions, when published, and can be found online at:

http://www.ascon.it

Chapter 1
Characteristics

The CANopen/Modbus master Protocol Converter is an electronic device that can be installed on DIN rail, it allows the following:

- bi-directional information between networks CANopen and Modbus.
- Electrical isolation between two BUSes
- SDO, PDO, Nodeguarding
- RS485 Serial Communications

To configure the Protocol Converter, use the Compositor software that runs under the control of Windows Operating System. It is downloadable from the site www.ascon.it in section *download/software*. Entered the download area, find the program:
`Ascon_SW_DY-`**`5030`**`_Compositor.zip`

Note: The first time you enter the Software Download area, you need to register yourself, by clicking on the register button.

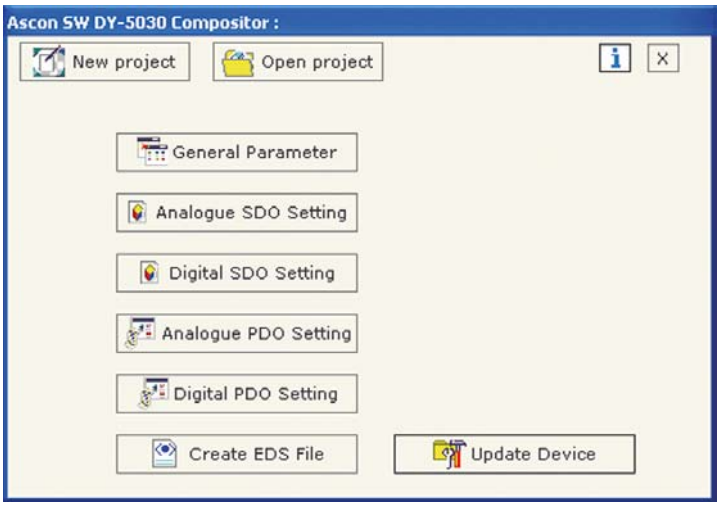
The Protocol Converter can be configured up to a maximum of 1000 SDO, 5 TPDO and 5 RPDO.

Chapter 2
Using the DY-5030 Conmpositor Software

2-1 Introduction

The CANopen/Modbus Protocol Converter, allows a CANopen network to communicate with a Modbus network that we simply call CANopen and ModBUS in this manual.

When launching it the following window appears:



The "New Project" button creates the folder which contains all the project files:

- The project is the set of files that defines a particular configurations of the CANopen/Modbus Protocol Converter. This file can also be imported and exported.
- To clone configurations of a CANopen/Modbus Protocol Converter in order to configure another device in the same manner, it is necessary to maintain the folder and all its contents.
- To clone a project in order to obtain a different version of the project, it is enough to duplicate the project folder with another name and open the new folder with the button "Open Project".

When the project is created or opened, it is possible to access the various configuration sections:

- **General Parameter;**
- **Analogue SDO Setting;**
- **Digital SDO Setting;**

- **Analogue PDO Setting;**
- **Digital PDO Setting;**
- **Create EDS file.**

2-2 General Parameter

This section defines the main communication parameters of two BUSses where the Protocol Converter is inserted.

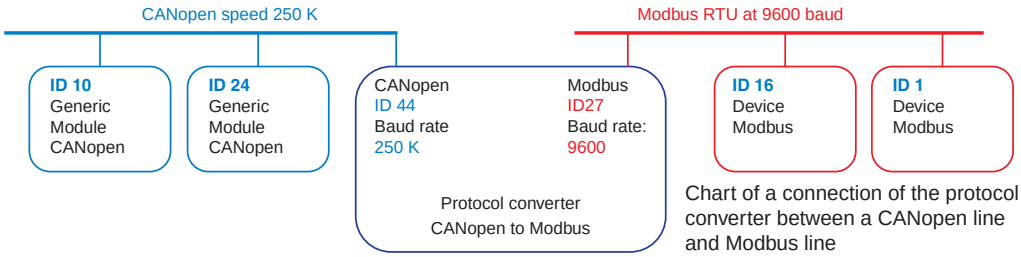
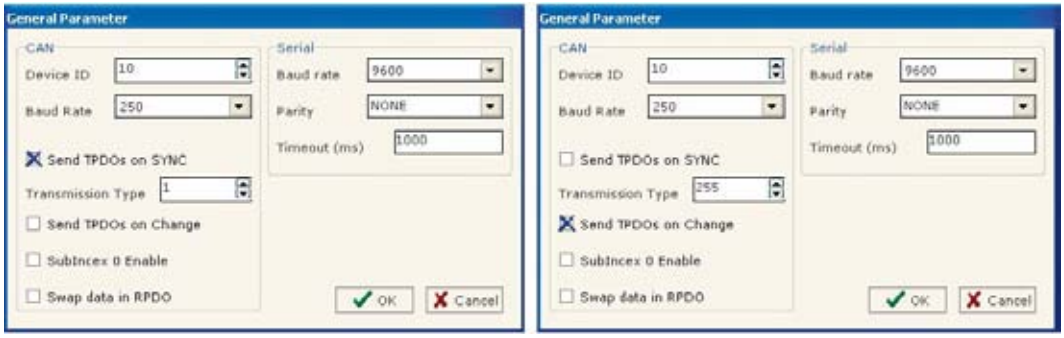


Chart of a connection of the protocol converter between a CANopen line and Modbus line



By pressing the "*General Parameter*" button, the previous window appears in which can be set the parameters of both the BUSses.

- In the field **Device ID**, the Gateway address of CANopen device can be inserted.
- In the fields **Baud Rate**, can be inserted the baud rate of the two BUSses.
- The parameter **Timeout** is expressed in milliseconds. Timeout is the maximum waiting time for Serial response.
- **Send TPDOs on SYNC**. When this option is enabled, the TPDOs transmission is started after a SYNC command is received.
- The **Transmission Type** field is connected to the SYNC field. If SYNC is enabled, the Transmission Type changes automatically the value to 1; when SYNC is disabled the field is set to 255 (to allow the Remote Request and Event management). For more information about the Transmission Type values meaning, consult the table that follows.
- **Subindex 0 Enable**. When there is only one register (Quantity = 1), if this box is checked, the value will be on Subindex 0 instead of Subindex 1;
- **Swap data in RPDO**. When this option is enabled, the 2 bytes of each word are swapped (**only for RPDO**).

- **Send TPDOs on Change** Of Status (COS). To allow TPDOs automatic transmission when there is a change of the TPDOs variable (send on Event).

Value	Type of Transmission
0	Setting this value, when a SYNC command is received, only **Changed** TPDOs (if COS enabled) and those TPDOs that have been requested with a **Remote Request** are transmitted.
1…240	Setting a value between 1…240 means that every 1…240 SYNC commands all the TPDOs are transmitted.
241…251	Reserved values
252	Setting this value, when a SYNC command is received, those TPDOs that have been requested with a **Remote Request** are transmitted
253…255	Setting these values means that **Changed** TPDOs (if COS enabled) and those TPDOs that have been requested with a **Remote Request** are transmitted immediately, independently from SYNC.

2-3 Analogue SDO setting

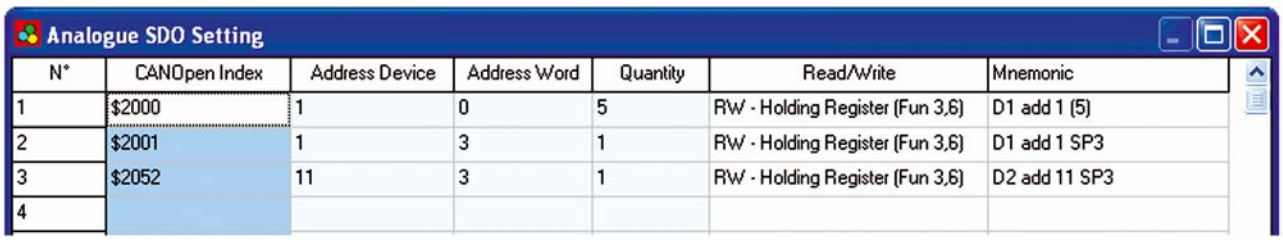
The Manufacturer area of Object Dictionary can be configured for representing data of serial line.

The following objects can be defined within the section as analogue SDO Setting:

- The CANopen SDO gives access to Modbus registers.

Example: If I want to read analogue data from Modbus network but I'm in a CANopen network :
I'll define an SDO index ($2000) and this will be associated to an analogue area inside Modbus network (device 1, Modbus address 0).
For reading word 0 on device at address 1 you have to read SDO index $2000 subindex 1.
For reading word 1 on device at address 1 you have to read SDO index $2000 subindex 2
Etc.

Sub-index 0 contains the number of PDO sub-index (if “Subindex 0 Enable” box is not checked).
Field Quantity means number of consecutive location can be read.
Read/Write = RO-Holding Register (FUN 3)
To force reading only and avoid writing the holding registers (uses Modbus function 3).
Read/Write = RW-Holding Register (FUN 3, 6)
To read and write holding registers (uses Modbus functions 3 and 6).
Read/Write = RO-Input Register (FUN 4)
To read input registers (uses Modbus function 4).



N°	CANOpen Index	Address Device	Address Word	Quantity	Read/Write	Mnemonic
1	$2000	1	0	5	RW - Holding Register (Fun 3,6)	D1 add 1 (5)
2	$2001	1	3	1	RW - Holding Register (Fun 3,6)	D1 add 1 SP3
3	$2052	11	3	1	RW - Holding Register (Fun 3,6)	D2 add 11 SP3
4						

The object $2100 to $2111 and $2200 to $220F are reserved and cannot be used.

All data will be retrieved on event read of specific SDO. The master CANopen need to have a timeout higher than 500 ms. The reply time is the time for a serial inquiry plus the serial response.

2-4 Digital SDO setting

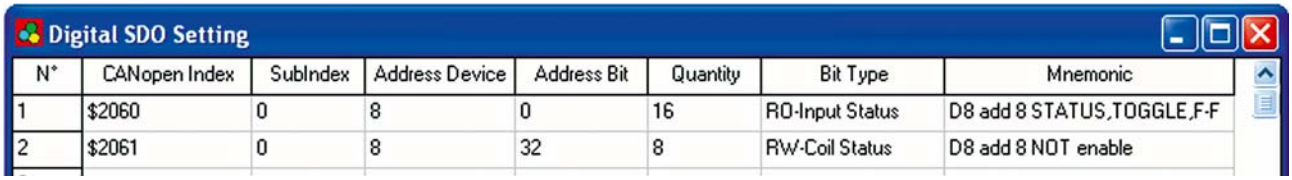
The following objects can be defined within the section as digital SDO Setting:

- The CANopen SDO gives access to Modbus bits.

Example: If I want to read data from Modbus network but I'm in a CANopen network :
I'll define an SDO index ($2060) and this will be associated to a digital area inside Modbus network (device 8, starting Modbus address 0, number of reading bits 16).
For reading all the 16 bits on device at address 8 you have to read SDO index $2060 subindex 0.
For reading all the 8 bits on device at address 8 you have to read SDO index $2061 subindex 0.

Etc.

Field Quantity means number of consecutive bits can be read;
Subindex field is suggested to always be 0;
Bit type = RO-Input Status (FUN 2);
To read input bits (uses Modbus function 2).
Bit type = RO-Coil Status (FUN 1);
To force reading only and avoid writing the output bits (coils) (uses Modbus function 1).
Bit type = RW-Coil Status (FUN 1, 5, and 15);
To read and write output bits (coils) (uses Modbus uunctions 1, 5, and 15).



N°	CANopen Index	SubIndex	Address Device	Address Bit	Quantity	Bit Type	Mnemonic
1	$2060	0	8	0	16	RO-Input Status	D8 add 8 STATUS,TOGGLE,F-F
2	$2061	0	8	32	8	RW-Coil Status	D8 add 8 NOT enable

For each SDO, can be read 16 bits max. (4 bytes: the last 2 bytes are always set to 0).
For each SDO can be written 16 bits max. (4 bytes: 2 data bytes + 2 mask bytes).

Data	0001	0001	0011	1111
Mask	0011	0110	1111	1111
Written bits	xx01	x00x	0011	1111

(bits indicated by x are not written).

The object $2100 to $2111 and $2200 to $220F are reserved and cannot be used.

All data will be retrieved on event read of specific SDO. The master CANopen need to have a timeout higher than 500 ms. The reply time is the time for a serial inquiry plus the serial response.

2-5 Analogue PDO setting

The gateway permits to use 4 analogue Receive PDO and 4 analogue Transmit PDO. All PDO are mapped to specific object; for TPDO1 ($2100,$2101,$2102,$2103) (consult the tables that follow for further details). The PDO may have a maximum length of 8 bytes divided in 4 words. Each word is linked to a word in serial bus.

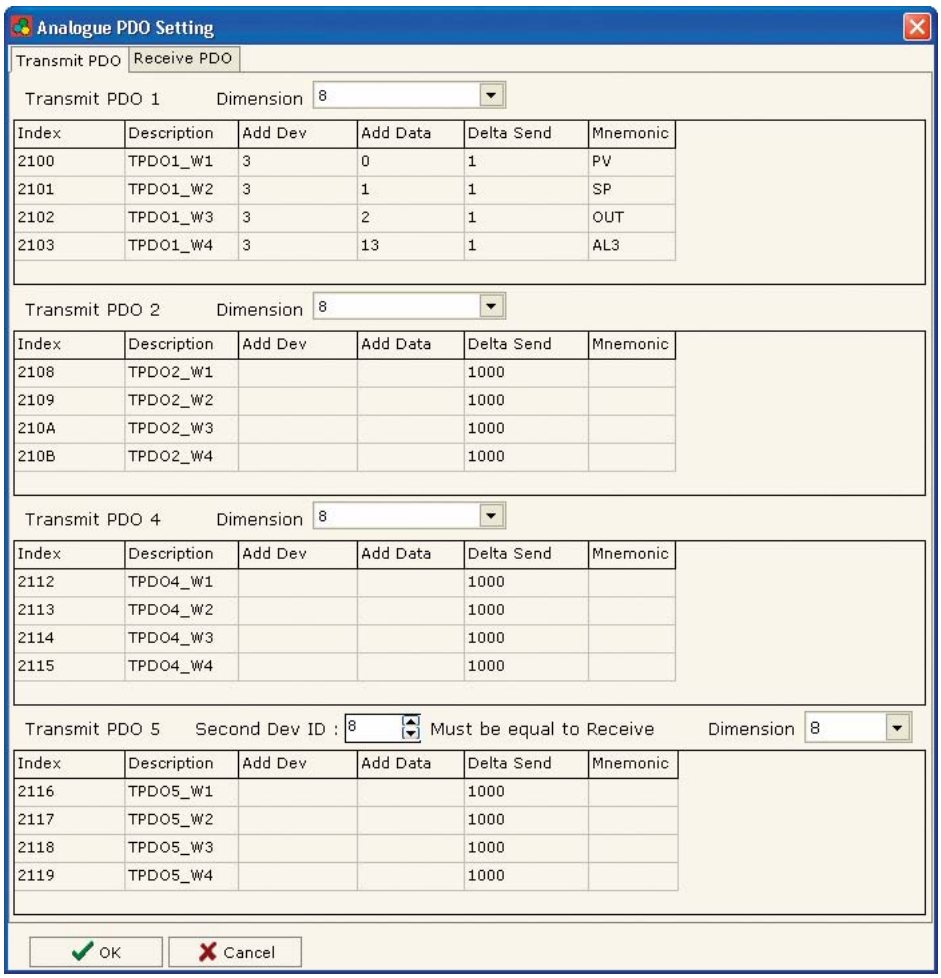
In order to use the Receive PDO5 and Transit PDO5, the user must configure a second device ID, different from the one configured in the General Parameter mask. When the device that has the device ID set in the General Parameter mask is set to Operational Mode, automatically also the device that has the second device ID is set to Operational Mode; in the same way the second device returns

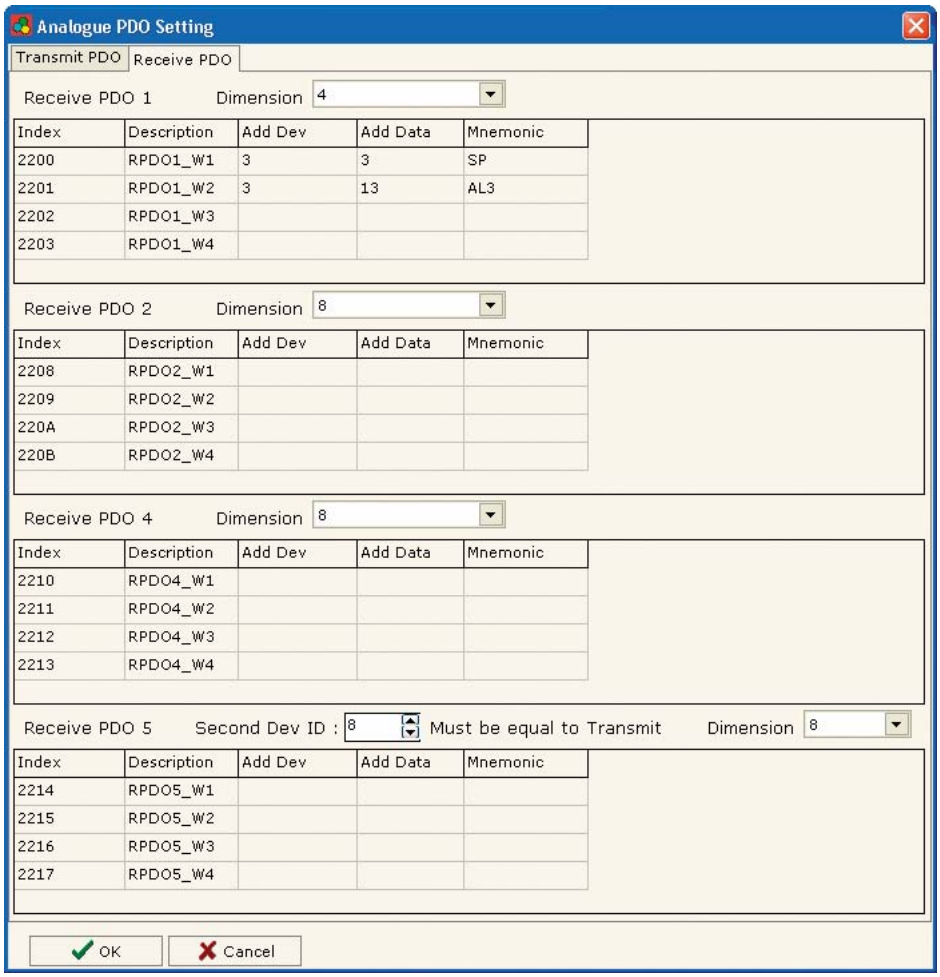
to the Pre Operational Mode when the device that has the device ID set in the Set Parameter Mask is removed from the Operational Mode.

Setting the second device in Operational or Pre Operational mode has no effect.

When all the Modbus devices are connected and working:

- writing an RPDO, the data will be written into serial device of specific addresses;
- requesting a TPDO this PDO will contain the data read from serial bus, from a specific device and addresses.





For every desired PDO it is necessary to specify:

Add Dev Modbus Slave Address of the device which contains the parameter that is to be read or write;

Add Data Address of the Modbus data register that is to be read or write.

Delta Send Only for TPDO. The smallest change of a Modbus variable that triggers the transmission of a PDO on the CAN network. When at least one Modbus word exceeds ±Delta Send range, the correspondent transmit TPDO will be automatically transmitted.

WARNING

Delta Send = 0 means that the TPDO will be continuously transmitted.

Mnemonic Optional additional comment;

Dimension Please specify the right PDO size in bytes (each index in the table uses 2 bytes). It has no effect on the unused PDOs.

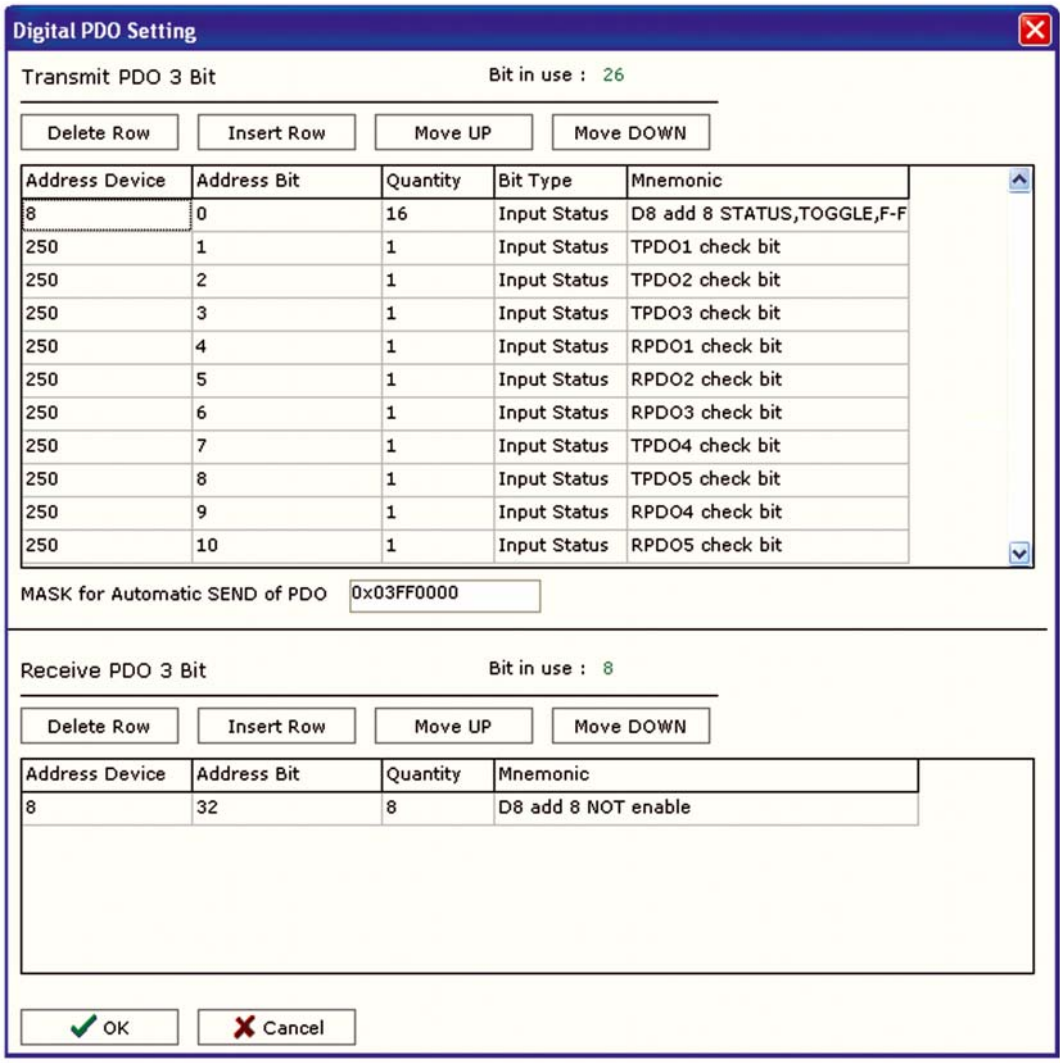
2-6 Digital PDO setting

The gateway permits to use 1 digital Receive PDO and 1 digital Transmit PDO. TPDO3 is mapped at $2110. RPDO3 is mapped at $2111. A PDO has 8 bytes length.

When all the Modbus devices are connected and working:

- Writing an RPDO, the data will be written into serial device of specific addresses;
- Requesting a TPDO this PDO will contain the data read from serial bus, from a specific device and address.

When at least one Modbus bit changes, according with the "MASK for Automatic send of PDO", the transmit TPDO will be automatically transmitted. In the "MASK for Automatic send of PDO" the less significant bits are referred to the first bits configured in the digital PDO setting page.



In the form above it is possible to configure the four parts of each PDO compiling the exchange table.

For the TPDO, can be read from the protocol converter 64 bits max. (8 data bytes). For the RPDO can be written to protocol converter 32 bits max. (8 bytes: 4 data bytes + 4 mask bytes).

Data	0001	0001	0011	1111	0001	0001	0011	1111
Mask	0011	0110	1111	1111	0011	1110	1110	1111
Written bits	xx01	x00x	0011	1111	xx01	000x	001x	1111

(bits indicated by x are not written).

2-7 Sent and received data coherence

Writing and reading using the SDOs, the protocol converter asks directly to the Modbus device. If the Modbus device does not answer, the protocol converter generates an "SDO general error" (00 00 00 08).

Writing and reading using the PDOs, the protocol converter uses the information of the internal database. If the Modbus device does not answer, the protocol converter uses the last valid information stored in the internal database. In order to point out that the PDOs information are not updated, 10 check bits (5 for the TPDO and 5 for the RPDO) are available. When the check bit is set to 1, at least one of the variables inserted in the correspondent PDO is not updated.

Inserting these check bits in the digital TPDO and configuring the "MASK for Automatic send of PDO" for automatic transmitting of the TPDO when the bits change, the TPDO can be used to know the information coherence in all the PDOs. As illustrated in the last 10 lines of the "Digital PDO setting" display, the 10 check bits conventionally are at Modbus "Address Device" 250 and Modbus "Address Bit" as described in the table that follows:

TPDO1 (analogue)	1
TPDO2 (analogue)	2
TPDO3 (digital)	3
RPDO1 (analogue)	4
RPDO2 (analogue)	5
RPDO3 (digital)	6
TPDO4 (analogue)	7
TPDO5 (analogue)	8
RPDO4 (analogue)	9
RPDO5 (analogue)	10

2-8 Create EDS file

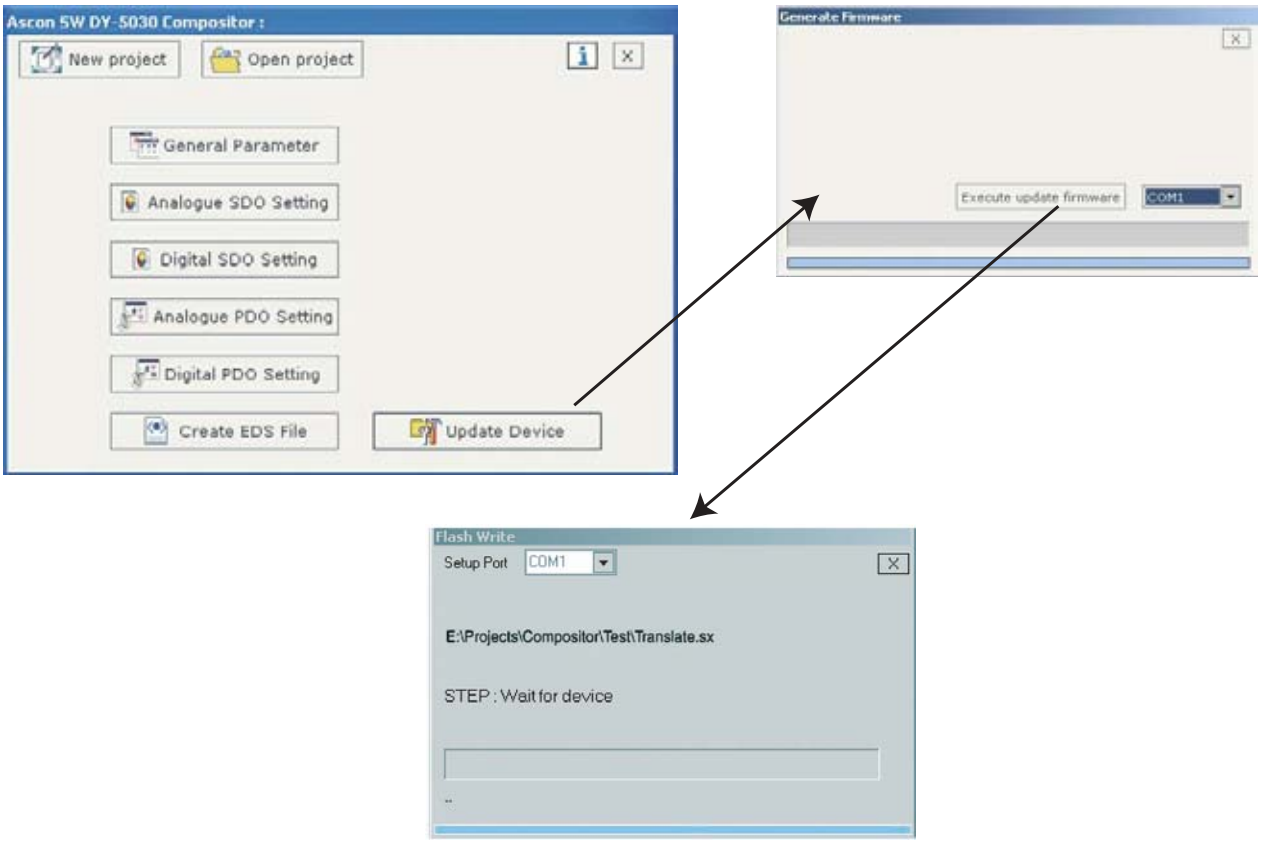
By clicking on this button the user can create a valid EDS file.

WARNING

In order to obtain a valid file that describes all the configured SDOs and PDOs, create the EDS file only after setting the SDOs and the PDOs.

2-9 Device update

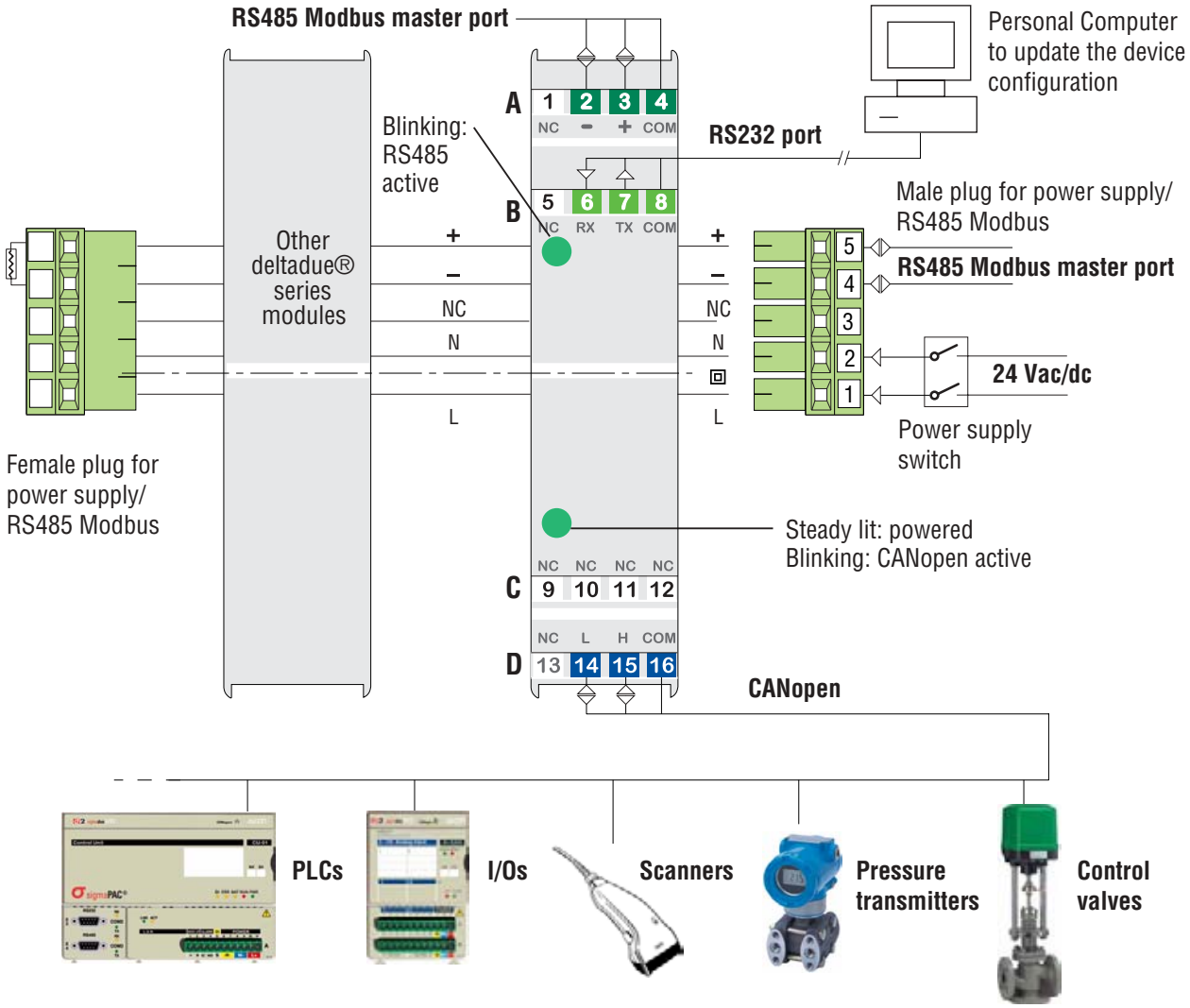
In order to download the parameters, you must click the button "*Update device*" on the main window.



1. At this point, you must boot the Protocol Converter with the provided jumper. See the "Boot jumper" paragraph in "Jumper setting" chapter.
2. Select the serial port which performs the update. Click on "*Execute update firmware*".
3. Wait for the running bar to finish.
4. Remove the jumper and reboot the protocol converter.

Chapter A
Connections and Jumpers

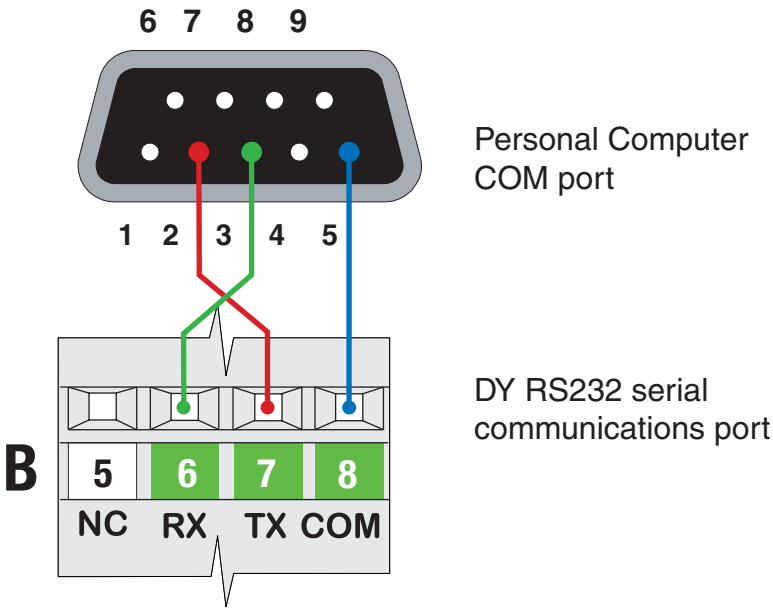
A-1 Connection Scheme



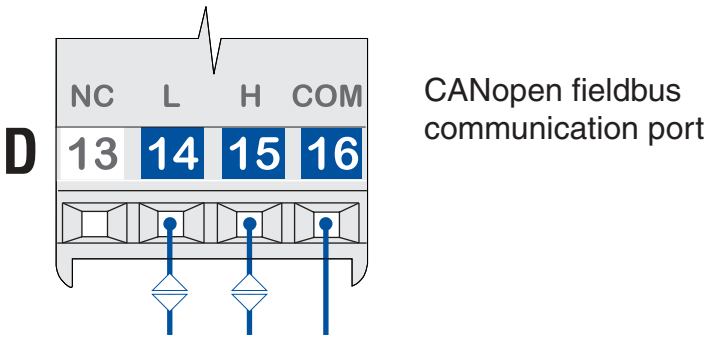
A-2 Cable Characteristics

A-2-1 RS232 Cable

The connection from RS232 terminal to a serial port (example one from a personal computer), must be made with a cable similar to the one in the following drawing. It is recommended that the RS232 cable does not exceed 15 meters.



A-2-2 CANBUS Cable Characteristics



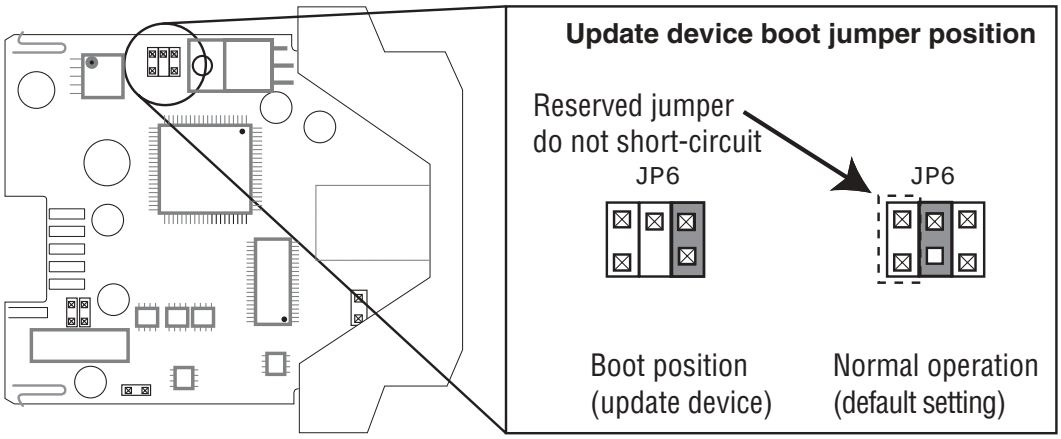
DC parameter:	Impedance	70.0 mΩ/m
AC parameters:	Impedance	120 mΩ/m
	Delay	5 ns/m
Length	**Baud Rate**	**Length max.**
	10 kbps	5000 m
	20 kbps	2500 m
	50 kbps	1000 m
	100 kbps	650 m
	125 kbps	500 m
	250 kbps	250 m
	500 kbps	100 m
	800 kbps	50 m
	1000 kbps	25 m

A-3 Jumpers Setting

On the printed circuit board of the Protocol Converter are present 5 groups of jumpers that are to be correctly set to let the Protocol Converter to function and communicate.

A-3-1 Boot Jumpers

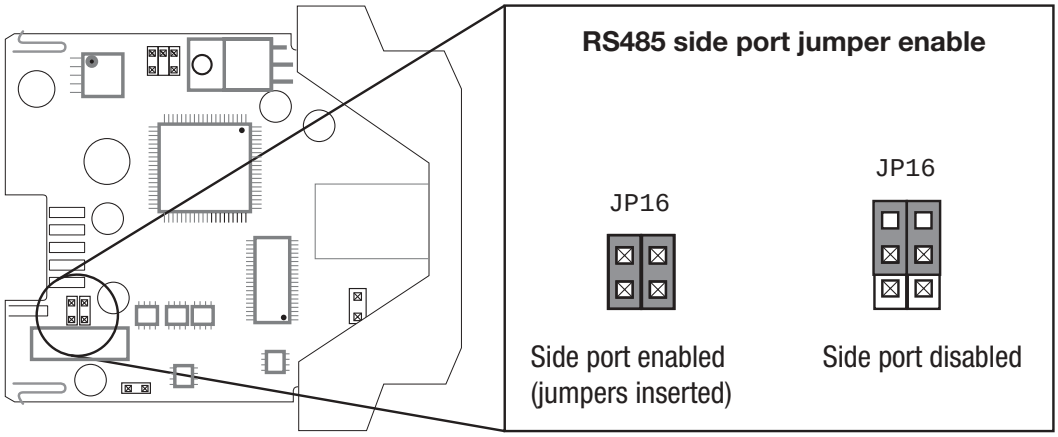
To update the device configuration, the boot jumper present on the PCB of the protocol converter must be set as follows.



A-3-2 RS485 Ports

The protocol converter has two RS485 lines that are routed on 2 different ports:

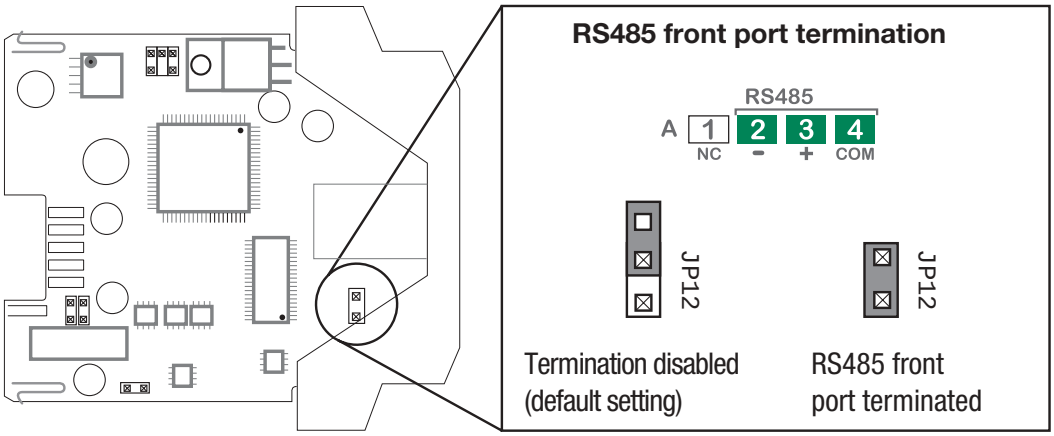
1. The Front RS485 port which can be found at terminal block A (always enabled).
2. The side RS485 port present on the connectors on the side of the converter case.



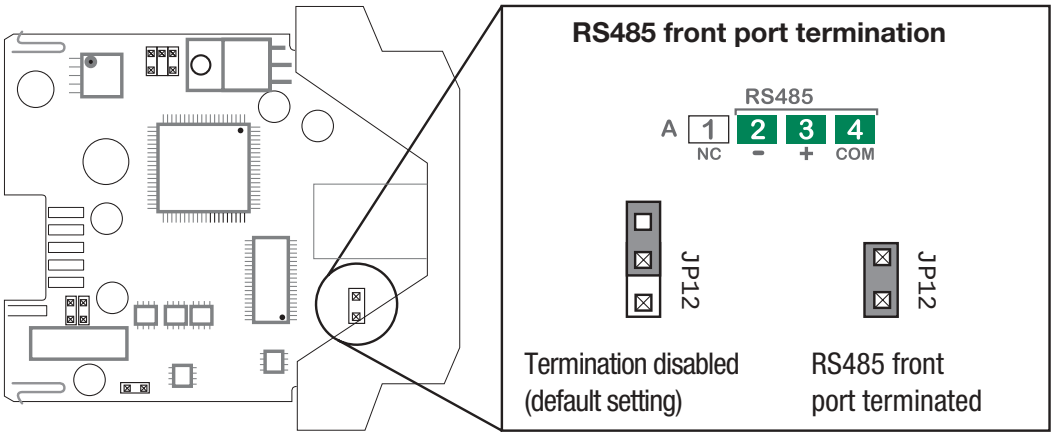
WARNING

As the protocol converter works as Master in the Fieldbus network, if another Master module (i.e.: DX) is connected to the RS485 side port, the RS485 side port of the protocol converter must be disabled.

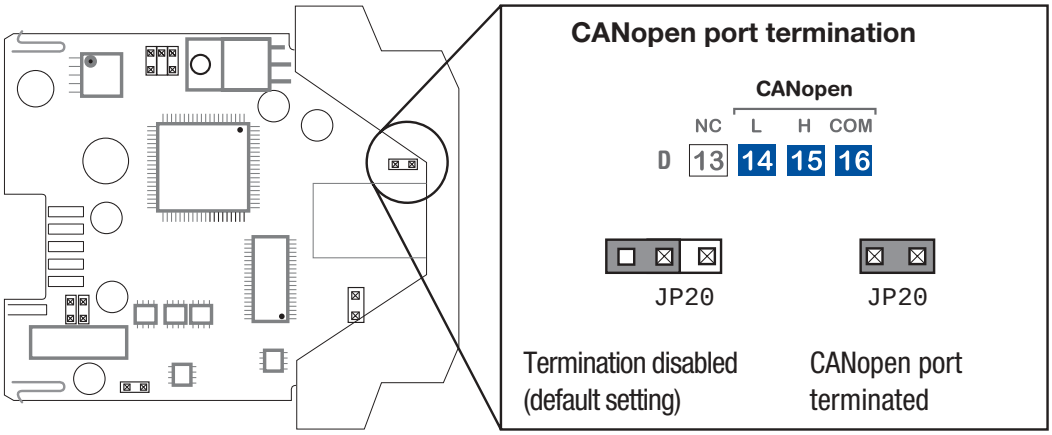
Terminating the front RS485 Port



Terminating the side RS485 Port

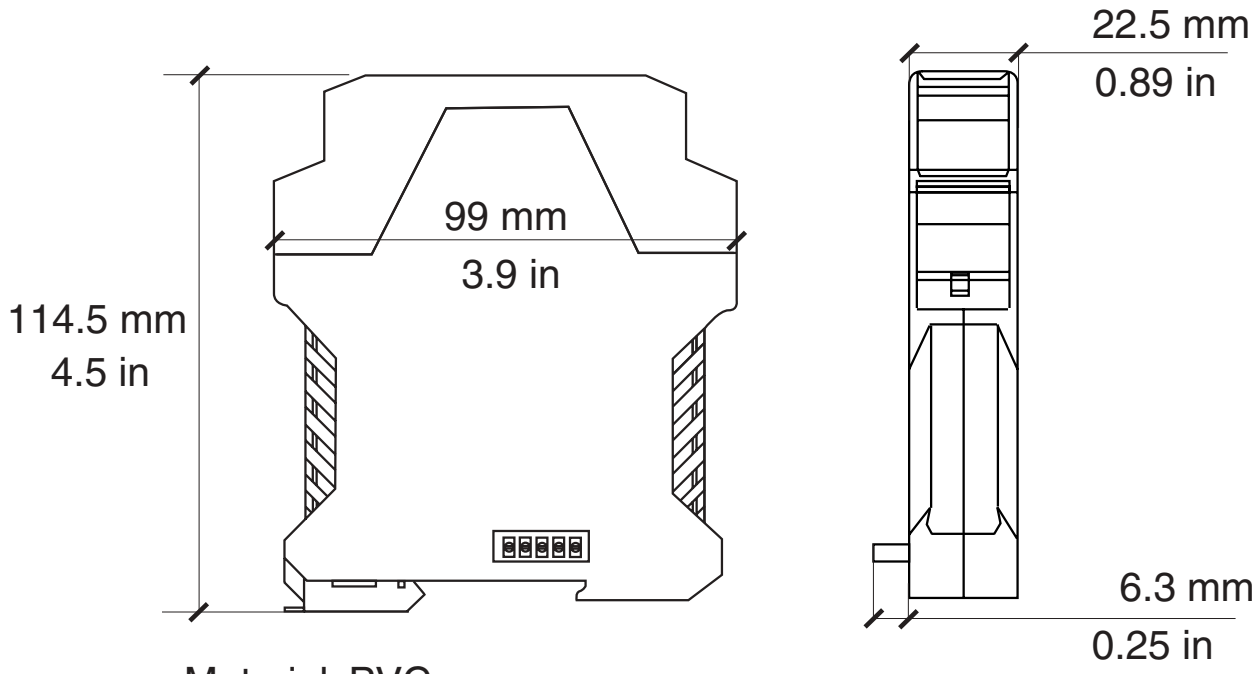


A-3-3 Terminating the CANopen Port



Appendix B
Mechanical Characteristics

B-1 Dimensions



Material: PVC
Weight: 200 g approx.