Fieldbus Option

Profibus DP, DeviceNet, CANopen, Modbus/TCP,
EtherCAT®, Profinet IO, EtherNet/IP

For Emotron VFX/FDU 2.1 AC drive (from sw V5.13)
Emotron TSA softstarter (from sw V1.31)
Emotron AFR/AFG/DCU units

DeviceNet

Profibus DP-V1

Modbus/TCP

Modbus/TCP 2-port

EtherCAT®

Profinet IO 1-port

Profinet IO 2-port

Ethernet/IP 2-port

CANopen

Instruction manual
English

emotron
DEDICATED DRIVE
A CG Product

Fieldbus Option

Profibus DP, DeviceNet, CANopen, Modbus/TCP, EtherCAT®, Profinet IO, EtherNet/IP

For Emotron VFX/FDU 2.1 AC drive (from sw V5.13)
Emotron TSA softstarter (from sw V1.31)
Emotron AFR/AFG/DCU units

Instruction manual – English

Document number: 01-3698-01
Edition: r13
Date of release: 2024-03-18

Safety

Instruction manual

Read this instruction manual first!

This option is a supplementary part of the “main product” and the user must be acquainted with the original instruction manual of the main product. All safety instructions, warnings, etc. as mentioned in this instruction manual must be known to the user.

Safety instructions

Read the safety instructions in the instruction manual for the main product.

Installation

Installation, commissioning, dismounting, making measurements, etc. on the main product may only be carried out by personnel who are technically qualified for the task. Installation must also be carried out in accordance with the local standards. Ensure that all necessary safety measures are taken.

WARNING!
Take all necessary safety precautions during installation and commissioning to prevent personal injuries, e.g. by an uncontrolled load.

Opening the main product

WARNING!
Always switch off the mains supply before opening the main product.
For AC drives, wait at least 7 minutes to allow the buffer capacitors to discharge.

Always take adequate precautions before opening the main product, even though the connections for the control signals and jumpers are isolated from the mains voltage.

Contents

1. Introduction

This instruction manual describes the installation and operation of the Fieldbus option board 2.0, which can be built into the products
Emotron VFX/FDU 2.1 AC drive (from sw V5.13), Emotron TSA softstarter (from sw V1.31) and Emotron AFR/AFG/DCU units.
Further on in this manual we refer to the main product.

The Fieldbus option allows external equipment to exchange data and to control the products described in this manual. The Fieldbus option acts as a slave in a master-slave configuration or alternatively as a participant in a producer-consumer network.

The data and illustrations found in this document are not binding. CG Drives & Automation reserves the right to modify our products in line with our policy of continuous product development. The information in this document is subject to change without notice and should not be considered as a commitment by CG Drives & Automation.

There are many applications for this product. Those responsible for the use of this device must ensure that all necessary steps have been taken to verify that the application meets all performance and safety requirements including any applicable laws, regulations, codes and standards.

Because this option is a supplementary part of the AC drive and soft starter, the user must be familiar with the original instruction manual of the main product. All safety instructions, warnings, etc. as mentioned in these instruction manuals are to be known to the user.

The following indications can appear in this manual. Always read these first and be aware of their content before continuing:

NOTE: Additional information to avoid problems.

CAUTION!
Failure to follow these instructions can result in malfunction or damage to the main product.

WARNING!
Failure to follow these instructions can result in serious injury to the user in addition to serious damage to the main product.

1.1 Users

In general, it is assumed that the user has basic knowledge of communication systems. This instruction manual is intended for installation, maintenance, service engineers and programmers.

1.2 Before starting

Besides as this manual, have the manual for the main product ready since it contains additional information not covered here. These product manuals are referred to at various points through this manual.

This manual contains information on general network troubleshooting for your convenience. Please note that CG Drives & Automation cannot support all general questions regarding your specific network issues/installations if they are not related to the main product or fieldbus option.

1.2.1 How to use this manual

This instruction manual shall be used in combination with the main products instruction manual, which contains “communication info” for all parameters/menus.

This instruction manual contains certain chapters/parts which are general for all fieldbus /industrial Ethernet protocols.
These are:
2.1.1 State of the fieldbus module,
2.1.4 Troubleshooting using menu system information,
4. Fieldbus menus
5. Cyclic/Acyclic data.
There is also a FAQ list available in chapter 7.

Following chapters/parts contain information dedicated for specific networks:
2.3 Profibus
2.4 DeviceNet
2.5 CANopen
3.3 Modbus/TCP
3.4 EtherCAT®
3.5 Profinet
3.6 EtherNet/IP

1.3 Delivery and unpacking

Check for any visible signs of damage. Inform your supplier immediately if any damage is found. Do not install the option board if damage is found.

If the option board is moved from a cold storage room to the room where it is to be installed, condensation can form on it. Allow the option board to become fully acclimatised and wait until any visible condensation has evaporated before installing it.

1.4 Checking of contents

Table 1

Pieces	Component
1	Fieldbus module, see picture below (in AC drive case mounted on separate circuit board). On Emotron TSA mounted directly on the control board
1	Flat cables, only for use on AC drive.
1	Mounting material (AC drive).

Fieldbus and Ethernet modules

DeviceNet

Profibus DP-V1

Modbus/TCP

Modbus/TCP 2-port

EtherCAT®

Profinet IO 1-port

Profinet IO 2-port

Ethernet/IP 2-port

CANopen

1.5 Abbreviations

Table 2

Abbrevation	Description
AC drive	Frequency inverter
Acyclic data	Information which is sent to and from a node at irregular time intervals. Typically this information includes configuration data, e.g. like motor data parameters. There is no need to send this type of information at cyclic intervals since it will not change.
Arbitration	In case of transmission collision on CANbus, arbitration is a hardware based method to allow the frame with smallest COB-ID to be transmitted first with priority.
CiA	CAN in Automation international users and manufacturers group.
CIP	Common Industrial Protocol. Real-time application layer protocol implemented to create and open interface for industrial automation. Common application layer protocol for Ethernet/IP (using Ethernet), ControlNet (dedicated network) & DeviceNet (using CAN bus). Supported by ODVA
CoE	CANopen over EtherCAT
COB	Communication Object. A unit of transportation in a CAN Network. Data must be sent across a network inside a COB.
COB-ID	COB identifier. Unique identity number of a COB which is based on the node address. Lower COB-ID gets higher priority. Sometimes denoted as CAN-ID.
Cyclic data	Information which is sent to and from a node at regular time intervals. Typically this information includes control & status information like run/stop, speed reference and status of the drive, i.e. information which needs to be updated at all times. Cyclic data is sometimes denoted as process data.
DAP	Device Access Point. A module which all Profinet interfaces must have. Located in the first slot (0). Specifies the physical interface, e.g. possible cyclic times, communication methods, synchronization etc.
DCP	Discovery and Configuration Protocol. Used within Profinet to discover Profinet devices & set device/station name and IP address.

Table 2

Abbrevation	Description
Determinism	The ability to ensure that a packet is sent and received in a specific period of time, i.e. a predictable response time. Deterministic communication is needed for process control.
DHCP	Dynamic Host Configuration Protocol, i.e. automatic configuration of IP-address and network specific settings
DNS	Domain Name System, translation of host name into IP address
EDS	Configuration file for e.g. CANopen and Devicenet. Electronic Data Sheet (EDS) describes the properties of the node.
EMI	ElectroMagnetic Interference
ESI	EtherCAT Slave Information (device description in XML format)
Ethernet	A physical and data link layer technology (ref OSI model) for local area networks (LANs). Standardised in IEEE 802.3.
Ethernet/IP	Ethernet Industrial Protocol is originally developed by Rockewell Automation, currently managed by Open DeviceNet Vendors Association (ODVA). Utilizes lighter/shorter messages through UDP protocol in combination with prioritized messages with QoS and managed switches. Supports multicast messages by producer consumer method to utilize bandwidth in combination with I/O data being sent only when changed or at a settable interval. The multicast message goes only to the listed consumers via VLAN (virtual LAN) to avoid unnecessary traffic to other network participants. Switches supporting IGMP Snooping can handle the multicast messages so that they are forwarded only to the necessary ports and not all
Gateway	A gateway (router) handles traffic between the local subnet and the 'outside'. I.e. if a node wants to communicate with another node which does not belong to the same subnet, the communication goes over the Gateway.
GSD	General station description. Used for Profibus & ProfiNet network devices. A file which contains a description of the communication features of the device. Installed in PLC system and used when configuring a network. Denoted GSDML when written in XML-language.
GSD file	Profibus-specific configuration files in ASCII format containing information about the node characteristics, i.e. supported baudrate and other features. GSD=General Station Description

Table 2

Abbrevation	Description
Heartbeat	A hearbeat protocol is used in CANopen networks to monitor the nodes and verify that they are alive. Based on periodically sent messages.
IE	Industrial Ethernet
LLDP	Link Layer Discovery Protocol. Used for topology detection (neighbour discovery) and diagnostic purposes presented in a user friendly graphical format. A device supporting LLDP sends out a periodical multicast with information about itself to the link and receives & stores the same type of information from the port neighbour (in LLDP-MIB accessible via SNMP). The multicast is terminated when it reaches a switch (not transmitted any further). Currently supported by Profinet IO 2-port version.
LSB	Least Significant Byte
LSS	Layer Setting Services (LSS) is a service within CANopen to set the baudrate and node id via the CAN bus. This is currently not supported.
MAC	Media Access Control address. A fixed, unique 6 byte identifier for network interfaces assigned by IEEE. Source and destination MAC are contained in all Ethernet frames.
Master	Central unit which controls the system, usually a PLC. In a profibus network, a master is also often called “active station”.
MIB	Management Information Base. The data information sent over SNMP from a managed device to a Network management system (trap manager). Can e.g. contain information about location, contact person, device description, communication status etc. Currently supported by Profinet IO 2-port version for the standardised LLDP-MIB and MIB II.
MRP	Media Redundancy Protocol, speficied in IEC 62439. Used for redundant network ring topology, which can be reorganized automatically and fast into a line structure in the case of cable break. Note! The external MRP switches must be directly connected with each other w/o any 'non MRP' units in between. The Profinet IO 1- and 2-port option of generation M30 does currently NOT support MRP. The 2-port option of generation M40 supports MRP.
MSB	Most Significant Byte

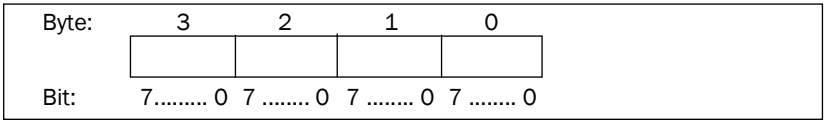
Table 2

Abbrevation	Description
NMT	Network management protocol is used by the NMT master to set the state of the devices. The CAN-frame COB-ID is always 0 for this protocol, i.e. given the highest priority.
Nodeguarding	Nodeguarding is an outdated method of checking whether a guarded CANopen device is still working in the correct network state. No longer recommended by CiA.
PCB	Printed Circuit Board
PDO	Process Data Object (cyclic I/O data)
PE	Protective Earth. Installation point which is connected to the protective earth point of the mains supply.
PLC	Programmable Logic Controller
Port mirroring	Feature in managed switches which can be used to monitor & analyze data traffic on specific switch ports. Very useable feature when analyzing network problems
Profinet IO	IE Protocol for 'real time'. Comes in two different variants; Profinet RT (Real Time) for fast IO communication in the range of ~10-15 ms reaction time and IRT (Isochronous Real Time) for very fast communication, e.g. with servo drives in the range of ≤ 1 ms reaction time.
RPI	Requested packet interval. Used in Ethernet/IP to set at which rate a node should generate process data telegrams.
SDO	Service Data Object (acyclic parameters)
SNMP	Simple Network Management Protocol. Internet protocol used to supervise/monitor devices diagnostics & network load. An NMS (Network management system) supervises a number of managed devices. Each managed device has a software agent running. In the case of fault, the agent reports back to the NMS via SNMP protocol. The retrieved information is contained in a MIB. Currently supported by Profinet IO 2-port versions (generation M30 and M40).
Station name	A unique text string/device name within Profinet which usually describes the function of the device.
Subnet	A subnet mask is used to define which part of the IP address that should be interpreted as network class respectively node address. It allows the network administrator to segment a large network into a finer mesh of subnetworks.

Table 2

Abbrevation	Description
Termination	Method of attenuating the signal at cable ends to avoid reflections. This is usually done with a simple resistor connection matching the impedance of the cable.
UDINT	Unsigned Double Integer (32-bit value)
USINT	Unsigned Short Integer (8-bit value)
VLAN	Virtual Local Area Network. Used to segment a large physical network into smaller 'logical' networks. Not to be confused with WLAN (wireless LAN).
XML	eXtended Markup Language

1.6 Byte/Bit numbering



1.7 Control board Software and fieldbus module compability

The fieldbus options are supported from the control board software versions listed inTable 3 below.

Table 3 Fieldbus options valid from software versions.

Fieldbus option	Supported from software version		
	Emotron FDU/VFX 2.0 AC drives	Emotron FDU/VFX 2.1 AC drives	Emotron TSA softstarter
Profibus	4.0	5.03	-
DeviceNet	4.0	5.03	-
Modbus TCP	4.11	5.03	1.0X
EtherCAT	4.32	5.03	1.26
Profinet IO	4.32	5.03	1.0X
Profinet IO + S2	-	5.13	1.31
EtherNet IP	4.36	5.03	1.25
CANopen	4.42	5.03	1.30

1.8 Further information

Further information about specific fieldbus system can be found on the Internet.

Profibus: www.profibus.com
DeviceNet: www.odva.org
CANopen: www.can-cia.org
Modbus/TCP: www.modbus-ida.org
EtherCAT: www.ethercat.org
Profinet: www.profinet.com
EtherNet/IP: www.odva.org

Specific information about CG Drives & Automation options/products and configuration files can be found at: www.emotron.com or www.cgglobal.com

2. Fieldbus networks

2.1 General description

All fieldbus options available through CG Drives & Automation are based on an open serial communication standard that enables data exchange between the main products and a network controller (PLC master).

Advantages of using a serial communication network:

- Geographically distributed systems can be controlled from a central unit (master)
- Production data can be collected and analyzed
- Shorter cable lengths (when compared to using analogue cables for every node)
- Centralized supervision/diagnostics
- Exchangeability of units when using a network standard
- Modular design
- Easy to extend network when necessary
- Easy installation (less time needed)
- Documentation of network easy (when compared to analogue signals)
- More robust against interference when using digital signals (detection of error and retransmitting message)

The network controller/master (usually a PLC = programmable logic controller) polls the slave nodes (e.g. AC drive/softstarter), which respond with a telegram over the network and perform commands sent by the master unit.

2.1.1 State of the fieldbus module

At any given time, the module is considered to be operating in one out of seven states. The current state of the fieldbus module is presented in menu [2693] on the main product.

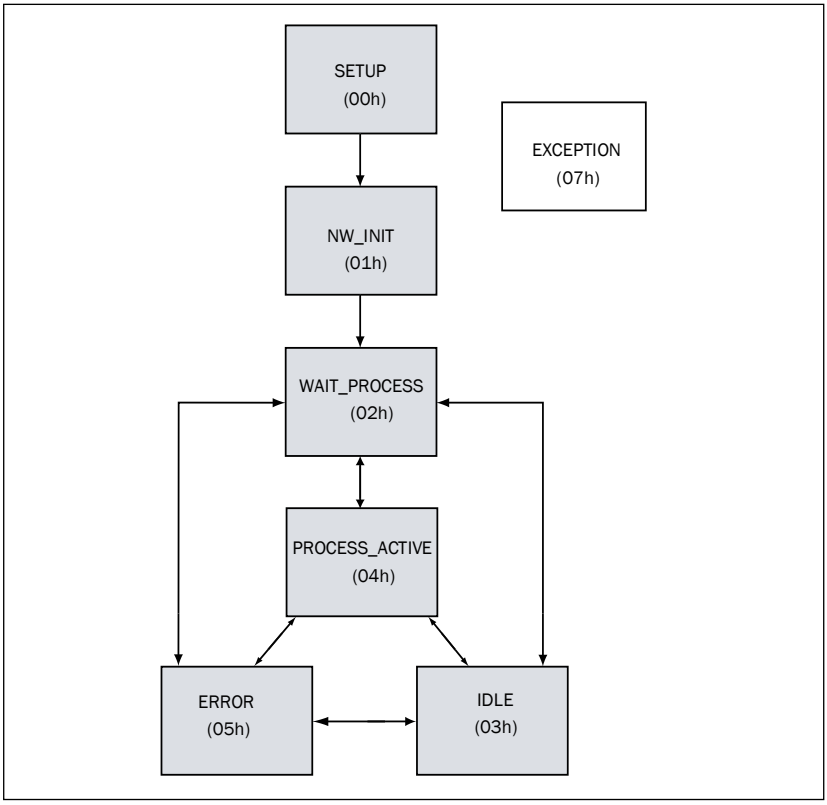


Table 4 Seven states

State no.	State name	Description
00	SETUP	The module is not yet initialised. No exchange of process data.
01	NW_INIT	The module is currently performing network related initialisation tasks. No exchange of process data (ignored).
02	WAIT_PROCESS	The network process data is temporarily inactive (ignored).This state will trigger a trip/warning if menu [2641] is not Off. If a Profibus or DeviceNet bus line is cut off the node will enter this state.
03	IDLE	The network is in idle mode (network specific behaviour).
04	PROCESS_ACTIVE	The network process data channel is active, i.e. process data is exchanged.
05	ERROR	There is at least one serious network error. Process data will be ignored.
07	EXCEPTION	Network communication is halted due to: - Invalid network configuration parameter - Timeout - etc. This status is not recoverable, which means that the module must be restarted (power off - power on) to be able to exchange network data again.

2.1.2 Common fieldbus network problems

This section describes how to locate network-related problems in a Profibus network. However, the hints given apply to most standard fieldbus systems.

1. A and B wires crossed somewhere in the network (applies to RS485 physical interface fieldbus systems).
2. Network not terminated in the ends (of each segment).
3. Incorrect nodes (non-ending nodes) terminated.
4. Cable cut off.

A common method of finding network-related problems is to reduce the network step-by-step until the problem disappears. This can be done by discon-

necting the outgoing profibus cable from a specific node and terminating at this point.

Furthermore, a simple multimeter can come in handy for measuring the resistance of the cable (checking connection).

The PLC-master system may also provide additional diagnostic information.

Has the correct *.gsd file been installed?

There are also more advanced tools available on the market, so called “network analysers” which can be used when encountering problems of a more difficult nature.

2.1.3 Troubleshooting using a multimeter

This section describes how to do simple measurements on a Profibus network. The technique is basically the same as for other fieldbusses but with different resistance values in cable/termination (please refer to your standard fieldbus documentation).

NOTE: The resistance measurements must be performed while the network is powered down.

- Check that the resistance between A&B wires (pins 3 and 8 in the D-sub connector) is approximately 100 ohm. The measured value depends on correct termination, cable type and total cable length.
- A very low value indicates a short circuit somewhere in the network.
- A Profibus cable has a resistance of approx. 50 ohm/km.
- Check if there is a short circuit between the shield and either the A or B wire. The resistance should be equal or higher than 10 kohm (but is likely to be much higher).
- Check if the shield is connected to the chassis and/or PE-din rail. Normally the resistance should be less than or equal to 10 ohm.
- Check that the A and B wires are not crossed.

2.1.4 Troubleshooting using menu system information

2.1.4.1 Checking internal communication

It is important to check that the internal communication between the control board and the fieldbus option is working. Consider the following issues:

1. Go through the page 119 in chapter 4.Fieldbus menus to make sure all settings are correct.
2. Check menu [2696]-[2698] — is any counter continuously increasing? In this case it indicates a problem with the internal communication between the control board and the fieldbus module or that no process data has been configured.
3. Check that the flat cable between the control board and the fieldbus board is properly connected and polarized.

If all the above is normal, then it is likely that the problem lies outside the AC drive. Continue with the next section, chapter 2.1.4.2 .

2.1.4.2 Checking external communication

1. Check LED status of the fieldbus module — error indication? power on?
2. Check menu [2693] — state of fieldbus module
3. Check menu [2692] — SUP-bit — is the slave node supervised by a master?

NOTE: Watchdog must be enabled on PLC.

NOTE: The main product must be restarted before the new address setting will become active, see next section, 2.1.4.3, for information.

2.1.4.3 Restart module with new settings

The main product must be restarted before the new setting(s) will be used. There are two ways of doing so:

1. Soft restart by changing menu [261] to RS232/485(AC drive) or RS232(TSA) and press Enter. Then change the setting to Fieldbus and press Enter again. (For AC drives this is only possible from control board software version 4.11 and later, see menu [922].)
   The main product will reboot during approximately 5 seconds. For safety reasons it is only possible to change menu [261] when the main product is stopped.

2. Hard restart by switching off the power, wait for the Control Panel to become unlit, then switch the power back on.

2.1.4.4 Solutions to common problems

Table 5 Solutions

Problem	Solution
The main product does not accept any cyclic data commands.	Check that the main product menu [2633] is set to RW (read/write). Check the PLC diagnostic messages. Check menu [2632] and [2634] — cyclic data size. Is the same mapping used for the PLC master?
The main product ignores the reference value sent by the master.	Check menu [2699] — incoming Basic process data. For AC dives also check menu [214] — Ref Control.
The main product does not start when given start command.	Check menu [2699] — incoming Basic process data and menu [215] on AC drive or [2151] on Emotron TSA — Run/Stp Ctrl.

2.2 Recommendations for fieldbus network installation

This section gives general installation recommendations.

- Always build your network so that it is possible to use the highest available baudrate for the network standard currently in use. This will make your installation robust and you can easily increase the baudrate in the future without replacing cables etc.
- Use the recommended cable according to each network standard. Take environmental issues into account, such as cables resistant to oil, water, grease, temperature, UV radiation, etc.
- Avoid drop cables (valid for Profibus networks).
- Always use shielded cables.
- A general recommendation is to connect the cable shield to the PE at every cable end see Fig. 1. Note that this might require an equipotential bonding system.

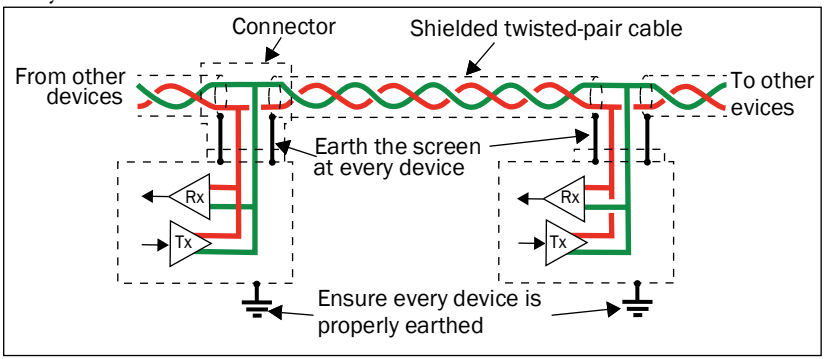


Fig. 1 Ensure that every device is properly earthed.

- Place the network cable as far away from power cables as possible. A distance of at least 20 cm is recommended.
- When crossing power cables, try doing this at a 90-degree angle. Avoid laying the network cable in parallel to the power cable.
- Do not twist, stretch or crimp any cables.
- Protect the cable from being damaged, e.g. by using cable channels/ grounded metallic cable racks.

2.3 Profibus

The profibus protocol is specified by the EN 50170 standard. There are three main communication profiles within Profibus:

1. Profibus-FMS (Fieldbus Message Specification)
2. Profibus-DP (Decentralised Periphery)
3. Profibus-PA (Process Automation)

The module supplied from CG Drives & Automation supports Profibus-DP (version 1).

The physical transmission medium of the bus is a twisted pair of cables (according to the RS-485 standard). Maximum length is 100-1200 metres (depending on the baudrate).

Up to 32 nodes can be connected in the same network segment without using repeaters. With repeaters it is possible to connect up to 127 nodes (including repeaters and master stations) in the network.

NOTE: Repeaters do not have a station address, but they count towards the maximum number of stations in each segment.

When adding a new node to the network the master needs to be informed of the network change. This is done by installing a configuration file (*.gsd). This file contains information about the new slave node's available functions.

2.3.1 Profibus fieldbus interface and LED indication

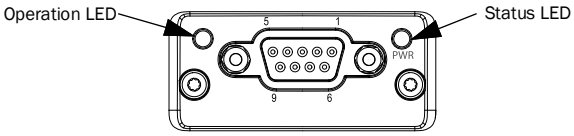


Fig. 2 Module front view

2.3.1.1 Operation LED description

Table 6

Status	Indication	Comment
Off	Not on-line/No power	Check if menu [261] Com type is set to Fieldbus.
Green	On-line, data exchange	All ok.
Flashing green	On-line, clear	–
Flashing red (1 flash)	Set-Prm service error	Master and slave configuration does not match. E.g. master is setup to use DpV2 functionality when the slave only supports DpV1.
Flashing red (2 flashes)	Chk_Cfg service error	Master and slave amount/ type of data does not match.

2.3.1.2 Status LED description

Table 7

Status	Indication	Comment
Off	No power or not initialised	Anybus state=SETUP or NW_INIT
Green	Initialised	Module has left the NW_INIT state
Flashing green	Initialised, diagnostic event(s) present	Extended diagnostic bit is set
Red	Exception error	Anybus status=EXCEPTION

2.3.1.3 Profibus connector

This connector is galvanically isolated. See Fig. 2.

Table 8

Pin	Signal	Description
1		
2		
3	B line	Positive RxD/TxD, RS485 level
4	RTS	Request to send
5	GND Bus	Isolated ground
6	5 V Bus output	Isolated +5 V. Max 10 mA output for fixed installation in industrial environment.
7		
8	A line	Negative RxD/TxD, RS485 level
9	Shield	

+5V and GND Bus are used for termination purposes.

NOTE: A profibus plug with integrated micro controller/LEDs for network diagnostics may require more than 10 mA output. In such case use a standard D-sub connector instead and use other external tools like e.g. Profitrace to analyze network quality.

2.3.2 Cable types

There are a number of different Profibus cable types for different applications (e.g. food industry, cable between buildings etc.). More information about these can be found in the IEC 1158-2 standard. The most commonly used Profibus cable is cable type A, which is described below:

2.3.2.1 Cable type A

Table 9

Cable design	shielded twisted pair
Surge Impedance	135-165 ohm
Loop Resistance	110 ohm/km
Core Diameter	0.64 mm
Core Cross Section	> 0.34 mm^2
Capacitance per unit length	< 30 pF/m

The length of cable allowed depends on the baudrate used:

Table 10

Baudrate [kbit/s]	9.6	19.2	93.75	187.5	500	1500	12000
Line length cable type A[m]	1200	1200	1200	1000	400	200	100

NOTE: The length may be extended by using repeaters.

The twisted pair consists of data lines “A” and “B”. It is important not to cross these cables.

According to the standard:

A = Green wire

B = Red wire

2.3.3 Bus termination

Termination of the network is important to avoid signal reflections. The first and the last unit (in every segment) must be terminated in a Profibus network.

2.3.3.1 Example

A profibus segment with 32 units (repeater is calculated as 1 unit even if it does not have an address). Termination is represented by an "X" at both ends in Fig. 3.

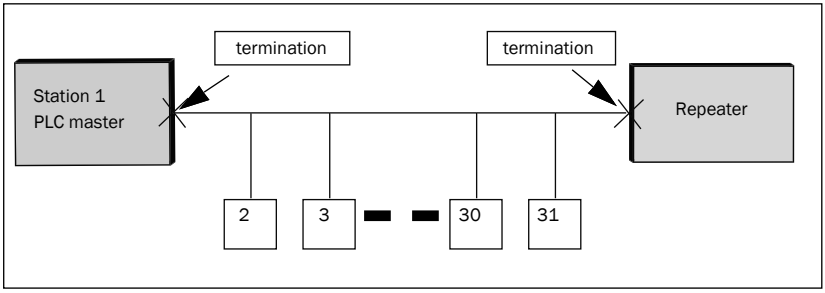


Fig. 3 Termination

NOTE: There is no termination switch on the fieldbus module itself. Therefore a standard Profibus D-sub with a built-in termination switch must be used.

NOTE: Do not terminate any nodes in the middle of the network line. This will cause the data signals to attenuate and with the worst case scenario being the network not working at all.

2.3.4 Address setting

The module must be assigned to a unique node address (a.k.a device address) in order to communicate on the Profibus network. The valid setting range is from 0 to 125. The node address is set in menu [2631].

NOTE: The main product must be restarted before the new setting(s) will become active. See chapter 2.1.4.3 for more information.

2.3.5 Baudrate

The baudrate is automatically detected by the module.

2.3.6 Profibus Sync/Freeze functionality

The Profibus option from CG Drives & Automation supports both sync and freeze which enables the user to synchronize the inputs (Freeze mode) or the outputs (Sync mode) of a group of nodes.

The FREEZE control freezes the physical input data present on one or more slaves simultaneously, like taking a snap shot. The selected slave(s) will stay frozen until an UNFREEZE command is issued.

The SYNC control works much in the same way, it will lock the physical output data present on one or more slaves simultaneously. This data will remain static until an UNSYNC command or new SYNC command has been issued.

2.3.7 Profibus Fail Safe operation

When setting the PLC to stop (clear), the fieldbus option will jump to state 3 (idle), see menu [2693]. In some cases it is desireable to stop the motor in this case. This is done by setting menu [561] VIO 1 Dest to “Enable” and menu [562] VIO 1 Source to “Com Active”

2.4 DeviceNet

DeviceNet is a low-cost communications network that can be connected to different industrial devices. Up to 64 nodes can be connected in the same network segment. DeviceNet is a standard open network — the specification, protocol and other useful information are available to anyone from ODVA's web page www.odva.org. The bus topology used is linear (trunkline/dropline); power and signal are integrated on the same network cable.

When adding a new node to your network you also need to inform the master of the network change. This is done by installing a configuration file (*.eds). This file contains information about the new slave node's available functions and settings.

2.4.1 DeviceNet fieldbus interface and LED indication

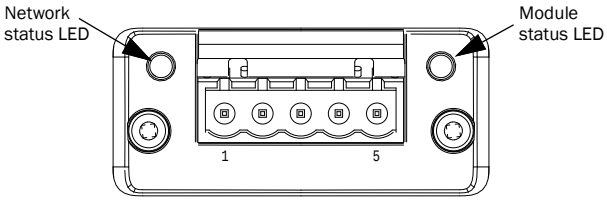


Fig. 4 Module front view

2.4.1.1 Network status LED description

Table 11

Status	Indication	Comment
Off	Not on-line/No power	Check if menu [261] Com type is set to Fieldbus. Check baudrate.
Green	On-line, one or more connections are established.	All Ok.
Flashing green (1 Hz)	On-line, no connections established.	Node has no master.
Red	Critical link failure.	Check for duplicate MAC ID and network power.

Table 11

Status	Indication	Comment
Flashing red (1 Hz)	One or more connections timed-out.	Check poll time of master.

2.4.1.2 Module status LED description

Table 12

Status	Indication
Off	No power
Green	Operating in normal condition.
Flashing green (1 Hz)	Missing or incomplete configuration, device needs commissioning.
Red	Unrecoverable Fault(s).
Flashing red (1 Hz)	Recoverable Fault(s).

2.4.1.3 DeviceNet connector

Table 13

Pin	Signal	Colour code	Description
1	V-	Black	Negative bus supply voltage
2	CAN_L	Blue	CAN low bus line
3	SHIELD	Bare	
4	CAN_H	White	CAN high bus line
5	V+	Red	Positive bus supply voltage

NOTE: The supply voltage should be 24 V DC ±10%.

2.4.2 Cable type

Only use DeviceNet cables that meet or exceed ODVA specifications. For detailed information about DeviceNet installation, please see ODVA's cable recommendation documents on www.odva.org.

Table 14

Baudrate [kbit/s]	125	250	500
Thick trunk length, [m (ft)]	500 (1,640)	250 (820)	100 (328)
Thin trunk length, [m (ft)]	100 (328)	100 (328)	100 (328)
Flat trunk length, [m (ft)]	420 (1,378)	200 (656)	75 (246)
Maximum drop length, [m (ft)]	6 (20)	6 (20)	6 (20)
Cumulative drop length, [m (ft)]	156 (512)	78 (256)	39 (128)

2.4.3 Bus termination

You must terminate the trunk line at both ends with 121 Ohms, 1%, 1/4 W terminating resistors.

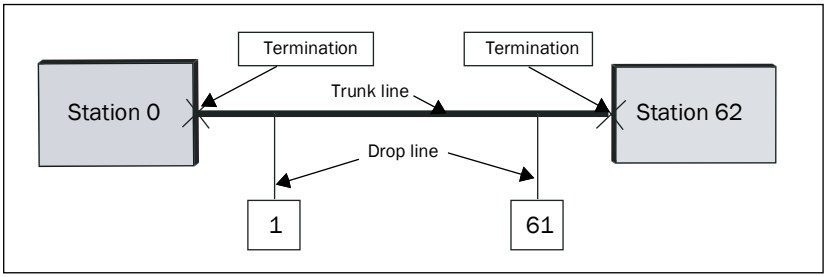


Fig. 5 Bus termination for DeviceNet

2.4.4 Address setting

The module must be assigned to a unique node address (a.k.a device address) in order to communicate on the DeviceNet network. The valid setting range is from 0 to 62 (default value 62).

NOTE: The master PLC and a computer interface node will also allocate addresses.

NOTE: The main product must be restarted before the new setting(s) will become active. See chapter 2.1.4.3 for more information.

2.4.5 Baudrate

The baudrate is automatically detected by the module.

2.4.6 Supported CIP Objects

The DeviceNet module has a generic profile (ProdType=0). It supports the following CIP objects:

Mandatory objects according to the DeviceNet standard:

Identity Object (01h)

Message Router (02h)

DeviceNet Object (03h)

Assembly Object (04h)

Connection Object (05h)

Parameter Object (OFh)

Acknowledge Handler Object (2Bh)

Vendor specific (valid for AC drives):

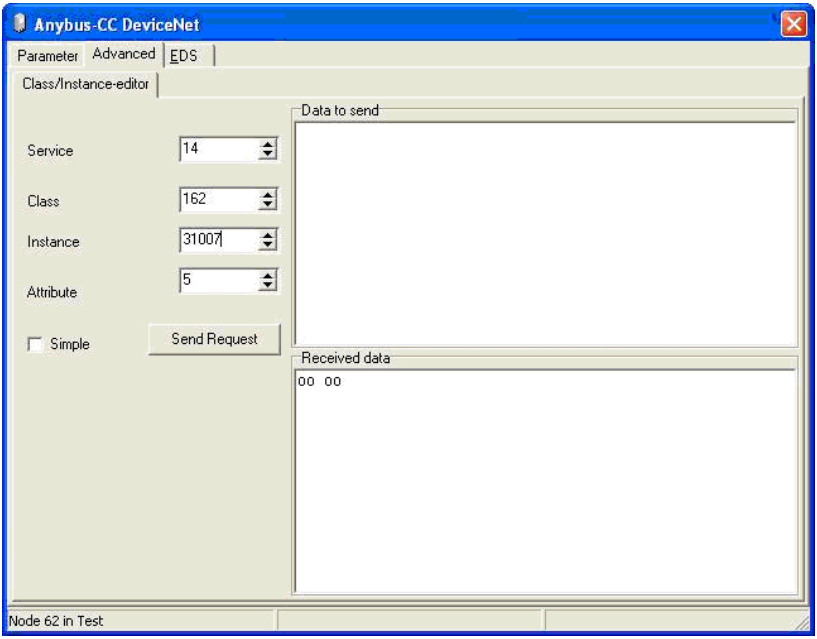
ADI Object (A2h) to access product parameters.

To read/write a certain parameter use service 16dec=SET or 14dec=GET.

All parameter values are reached by using class 162dec (A2h) and the instance number according to the AC drive manual. Attribute 5dec = VALUE

2.4.6.1 Example reading product parameter (valid for AC drive)

We want to read the Current from the main product. In the main product manual we find the Devicenet instance number to be 31007 (Current). We use service 14dec=GET. Class 162dec is used together with attribute 5dec for DATA.



NOTE: Currently there is no support for ADI object attribute 8, default value. Use main product menu [243] Default>Set instead to perform factory default

2.5 CANopen

CAN in Automation (CiA) is the international users and manufacturers organization that develops and supports CAN-based protocols. See: https://www.can-cia.org/canopen/ .
CANopen is a higher communication protocol layer on top of a data link and physical layer. The physical layer for CANopen usually is Controller Area Network (CAN).
Each telegram sent on the network is identified and prioritized by its 11-bit communication object identifier, more known as COB-ID (sometimes denoted CAN-ID). The COB-id usually consist of a 4 bit function code + a 7 bit node id. The 7-bit node id allows the node address to be set between 1-127. The CAN frame with the lowest COB-ID on the bus gets priority and is thus used for time critical functions.
Since every node is allowed to send on the bus, transmission collisions must be avoided. CAN uses a hardware method called arbitration to solve this where an electrical 0 is driven and where an electrical 1 has a passive pull-up solution. Since all nodes listen to their own transmission they can also detect the situation where a node with higher priority (lower COB-id) transmits and can then fall back into listening mode.
As an example, the NMT master controls the state of the entire network and sends its commands with a fixed COB-ID of all zeros.
CANopen uses multiple communication models, e.g. master/slave for the NMT protocol, client/server for SDO protocol and also a producer/consumer model which enables e.g. inter-slave data exchange without involvement of master and without specific data requests.
CANopen also supports the heartbeat protocol which is used to verify that specific nodes on the network are alive. The principle of this is that a heartbeat producer, which is usually a slave device, periodically sends an “I’m alive” message on the CAN bus. A corresponding heartbeat consumer receives this message and thereby knows that the slave is in operation. If no message is received within a specific time limit, the consumer can take action for this.
Process data can be exchanged e.g. cyclically as Process Data Objects (PDOs) on the bus. Data is transmitted with small overhead in a broadcast message.
Acyclic device parameter read/write requests are exchanged using Service Data Object (SDOs). Manufacturer specific CANobjects can be reached between range 2000h to 5FFFh.
This CANopen module is compliant with profile DS301 V4.02. It has galvanically isolated bus electronics and supports all standard baud rates (automatic baudrate detection is also available). Layer setting services (LSS) is NOT sup-

ported (otherwise used for setting node id and baud rate). CAN standard frames with 11-bit identifier field is supported, whereas 29-bit identifier field is NOT allowed.

2.5.1 CANopen fieldbus interface and LED indication

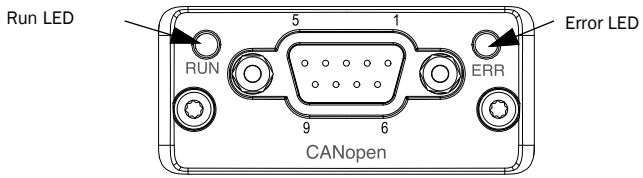


Fig. 6 Module front view.

2.5.1.1 LED indication

According to CiA-303 , part 3, indicator specification the different LED states are defined as:

Table 15

LED state	Description
LED on	The LED shall be constantly on
LED off	The LED shall be constantly off
LED flickering	Shall indicate the iso-phase on and off with a frequency of approximately 10 Hz: on for approximately 50 ms and off for approximately 50 ms.
LED blinking	Shall indicate the iso-phase on and off with a frequency of approximately 2.5 Hz: on for approximately 200 ms and off for approximately 200 ms.
LED single flash	Shall indicate one short flash (approximately 200 ms), followed by a long off phase (approximately 1000 ms).
LED double flash	Shall indicate a sequence of two short flashes (approximately 200 ms), separated by an off phase (approximately 200 ms). The sequence is finished by a long off phase (approximately 1000 ms).
LED triple flash	Shall indicate a sequence of three short flashes (approximately 200 ms), separated by an off phase (approximately 200 ms). The sequence is finished by a long off phase (approximately 1000 ms).
LED quad-rople flash	Shall indicate a sequence of four short flashes (approximately 200 ms), separated by an off phase (approximately 200 ms). The sequence is finished by a long off phase (approximately 1000 ms).

2.5.1.2 Run LED

This LED reflects the status of the CANopen communication

Table 16

LED status	Indication	Description
Off	-	No power
Green on	Operational	Module in state Operational
Green, blinking	Pre-Operational	Module in state Pre-Operational
Green, single flash	Stopped	Module in state Stopped
Green, flickering	Autobaud	Baud rate detection in progress
Red on	Exception state (fatal event)	Module has shifted to state Exception

If both LEDS turn red, this indicates a fatal event; the bus interface is shifted into a physical passive state. Contact CG Drives & Automation support in such case.

2.5.1.3 Error LED

This LED indicates CANopen related errors

Table 17

LED status	Indication	Description
Off	-	No power or the device is in working condition.
Red, single flash	Warning limit reached	A bus error counter reached or exceeded its warning level
Red, flickering	NA/ LSS (not supported)	LSS service error. Note that LSS is not supported
Red, double flash	Error Control Event	A guard- (NMT-slave or NMT-master) or heartbeat event (Heartbeat consumer) has occurred.
Red on	Bus off (Fatal Event)	Bus off

If both LEDS turn red, this indicates a fatal event; the bus interface is shifted into a physical passive state. Contact CG Drives & Automation support in such case.

2.5.1.4 CANopen d-sub interface

Table 18

Pin	Signal
1	-
2	CAN_L
3	CAN_GND
4	-
5	CAN_SHLD
6	-
7	CAN_H
8	-
9	-
Housing	CAN_SHIELD

2.5.2 CANopen cable type

See CiA's document DR 303-1 V1.3 CANopen cabling and connector pin assignment for detailed information. It is strongly recommended to always use shielded / twisted pair cable type to avoid EMI from power cables and other sources.
The characteristic impedance of the line used should be about 120 ohms.

For drop cables a wire cross-section of 0.25 to 0.34 mm^2 may be an appropriate choice in many cases.

2.5.3 CANopen bus termination

The bus cable shall be terminated at both ends by termination resistors.

The cables, connectors and termination resistors used in CANopen networks shall meet the requirements defined in ISO11898-2.

The table below shows general guidelines for CANopen networks with less than 6 nodes:

Table 19 General guidelines for CANopen networks with less than 6 nodes

Bus length [m]	Bus cable (1)		Termination resistance [o]	Baudrate [kbit/s]
	Length-related resistance [mO/m]	Cross-section [mm^2]		
0 to 40	70	0.25 to 0.34	124	1000 at 40 m
40 to 300	<60	0.34 to 0.6	150 to 300	>500 at 100 m
300 to 600	<40	0.5 to 0.6	150 to 300	>100 at 500 m
600 to 1000	<26	0.75 to 0.8	150 to 300	>50 at 1 km

2.5.4 Address setting

The module must be assigned to a unique node address in order to communicate on the CANopen network. The valid address range is from 1-127 (0 is reserved for broadcasting). The node address is set in menu [2631] Address.

NOTE: The main product must be restarted before the new setting(s) will become active. See chapter 2.1.4.3 for more information.

2.5.5 Baudrate setting

The baudrate for the CANopen module can be set in menu [2635] CAN-Baudrate. Valid range is 0-9 as per table below. Default setting is 8 = 1 Mbps.

Table 20

Value	Baudrate
0	10 kbps
1	20 kbps
2	50 kbps
3	Reserved (Fieldbus module Firmware versions 3.xxx) 100 kbps (Fieldbus module Firmware versions previous to 3.0)
4	125kbps
5	250 kbps
6	500kbps
7	800 kbps
8	1 Mbps
9	Auto *)

*) Under normal traffic conditions, i.e. with cyclic bus traffic above 2 Hz, the baud rate should be detected within 5 seconds.

Note! The automatic baud rate detection will NOT work if there is no traffic on the network.

NOTE: The main product must be restarted before the new setting(s) will become active. See chapter 2.1.4.3 for more information.

2.5.6 PDO COB-id setup

The serial link between the control board and CANopen module has a forced configuration to always exchange data for all 9 T/RPDOs which consist of one 32-bit unsigned BASIC data PDO + 8 additional process data values also of type 32-bit unsigned. According to DS302, only T/RPDO 1 to 4 are enabled by default, see table 21 below. The number of PDOs can easily be expanded by setting up transmit/receive PDO mapping via CANobjects 16XXh and 1AXXh and then enabling the corresponding COB-id via CANobjects 14XXh and 18XXh (see Table 23). Available PDO-mappable CANobjects are listed in chapter 2.5.7 .

Table 21 RPDO default COB-id

RPDO No	Default COB ID	Default transmission type	Description
1	200h + Node ID	254	Default enabled according to DS301
2	300h + Node ID	254	Default enabled according to DS301
3	400h + Node ID	254	Default enabled according to DS301
4	500h + Node ID	254	Default enabled according to DS301

Table 22 TPDO default COB-id

TPDO No	Default COB ID	Default transmission type	Description
1	180h + Node ID	254	Default enabled according to DS301
2	280h + Node ID	254	Default enabled according to DS301
3	380h + Node ID	254	Default enabled according to DS301
4	480h + Node ID	254	Default enabled according to DS301

RPDO 5..9 and TPDO 5..9 are disabled by default. This is indicated by MSB of the 32-bit COB ID set to one. The EDS-file therefore has a default value of 0x8XXX XXXX (MSB set) for these CAN objects.

The following CAN-objects are related to PDO COB-id allocation:

Table 23

Index	Object name	Sub-Index	Description	Type	Acess	Notes
1400h ... 141Fh	Receive PDO parameter	00h	Largest sub-index supported	U8	RO	02h
		01h	COB ID used by PDO	U32	RW	-
		02h	Transmission type	U8	RW	-
1800h ... 181Fh	Transmit PDO parameter	00h	Largest sub-index supported	U8	RO	05h
		01h	COB ID used by PDO	U32	RW	-
		02h	Transmission type	U8	RW	-
		03h	Inhibit time	U16	RW	-
		05h	Event Timer (ms)	U16	RW	-

2.5.6.1 PDO COB-id transmission types

PDO transmission triggering can be set up as follows:

Event driven

Message transmission triggered by:

Table 24

Transmission Type	Description	Notes
254/255	COS	When Process data has been changed. (The performance will be dependent on the number of PDO´s using COS)
1...240	Cyclic Synchronous	For synchronous this is the expiration of the specified transmission period, synchronized by the reception of the SYNC object. The data will be synchronized only to the module (current process data in buffer) and not all the way down to the application.
0	Acyclic Synchronous	A transmission type of zero means that the message shall be transmitted synchronously with the SYNC object but not periodically. Only on when COS is fullfilled (SYNC & COS).

Timer driven

Message transmission is either triggered by the occurrence of a device-specific event (COS) or if a specified has elapsed without the occurrence of the event.

Table 25

Transmission Type	Description	Notes
254/255	COS/Timer	Message transmission is either triggered by the occurrence of a device-specific event (COS) or if a specified time has elapsed without occurrence of the event.

2.5.7 PDO mapping

For CANopen menu [2632] PrData mode is forced to Basic and menu [2634] Add PrValues is forced to 8. This means that the serial link between the control board and the CANopen module exchanges 9 process data in/out, each with datatype 32 bit unsigned integer, i.e. BASIC (32-bit unsigned int) + 8 additional process data (32-bit unsigned int) in each direction. This means that up to a maximum of 9 RPDOs and 9 TPDOs can be configured for data exchange over the CANopen network. Note however that according to DS301, only four TPDOs and four RPDOs are enabled by default (see chapter 2.5.6).

Any of the available process data in table below can be mapped to any PDO (1-9). Each mappable CANobject is of datatype 32-bit unsigned int.

Each process data corresponding to fieldbus signals can in turn be mapped to any parameter in the main product by configuring the corresponding Modbus parameter address in menu [2661..266G]. The modbus parameter address is listed in the main product manual. An alternative to change the mapping of menu [2661..266G] via PPU menu system is to use the corresponding CANobjects listed in Table 26 and Table 27.

Table 26 Receive PDO default mapping

Available Process data	CANobject (32 bit uint) PDO mappable	Configure mapped Modbus register by PPU menu	Configure mapped Modbus register by CANobject	Default RPDO configuration	Comment
32-bit basic	5e00	N/A	N/A	RPDO1 (CANobj 1600h sub 1 mapped as 5e00 0020)	See chapter 5.2.2.1 ,Table 63, control message to AC drive or chapter 5.2.2.2 , Table 67, control message to softstarter.
FB signal 1	5f00	[2661]	0x4af1	RPDO2 (CANobj 1601h sub 1 mapped as 5f00 0020)	See chapter 5.2.2.4 page 148, Default mapping = 0 (disabled)
FB signal 2	5f01	[2662]	0x4af2	RPDO3 (CANobj 1602h sub 1 mapped as 5f01 0020)	
FB signal 3	5f02	[2663]	0x4af3	RPDO4 (CANobj 1603h sub 1 mapped as 5f02 0020)	
FB signal 4	5f03	[2664]	0x4af4	---	See chapter 5.2.2.4 page 148, Default mapping = 0 (disabled)
FB signal 5	5f04	[2665]	0x4af5	---	
FB signal 6	5f05	[2666]	0x4af6	---	
FB signal 7	5f06	[2667]	0x4af7	---	
FB signal 8	5f07	[2668]	0x4af8	---	

Table 27 transmit PDO mapping

Available Process data	CANobject (32 bit uint) PDO mappable	Configure mapped Modbus register by PPU menu	Configure mapped Modbus register by CANobject	Default RPDO configuration	Comment
32-bit basic	5c00	N/A	N/A	TPDO1 (CANobj 1A00h sub 1 mapped as 5c00 0020)	See chapter 5.2.2.1 ,Table 65, control message from AC drive or chapter 5.2.2.2 Table 69 Status message from softstarter.
FB signal 9	5d00	[2669]	0x4af9	TPDO2 (CANobj 1A01h sub 1 mapped as 5d00 0020)	See chapter 5.2.2.4 page 148, Default VFX/FDU mapping = 31002 (speed) + Default TSA mapping = 31004 (torque)
FB signal 10	5d01	[266A]	0x4afa	TPDO3 (CANobj 1A02h sub 1 mapped as 5d01 0020)	See chapter 5.2.2.4 page 148, Default VFX/FDU mapping = 31004 (torque) + Default TSA mapping = 31006 (shaft power)
FB signal 11	5d02	[266B]	0x4afb	TPDO4 (CANobj 1A03h sub 1 mapped as 5d02 0020)	See chapter 5.2.2.4 page 148, Default VFX/FDU mapping = 31005 (shaft power) + Default TSA mapping = 31008 (rms current)

Table 27 transmit PDO mapping

Available Process data	CANobject (32 bit uint) PDO mappable	Configure mapped Modbus register by PPU menu	Configure mapped Modbus register by CANobject	Default RPDO configuration	Comment
FB signal 12	5d03	[266C]	0x4afc	---	See chapter 5.2.2.4 page 148, Default VFX/FDU mapping = 31007 (current) + Default TSA mapping = 31009 (main voltage)
FB signal 13	5d04	[266D]	0x4afd	---	See chapter 5.2.2.4 page 148, Default VFX/FDU mapping = 31008 (output volt) + Default TSA mapping = 31010 (heat-sink temp)
FB signal 14	5d05	[266E]	0x4afe	---	See chapter 5.2.2.4 page 148, Default VFX/FDU mapping = 31009 (frequency) + Default TSA mapping = 31024 (used thermal cap)
FB signal 15	5d06	[266F]	0x4aff	---	See chapter 5.2.2.4 page 148, Default VFX/FDU mapping = 31010 (dc voltage) + Default TSA mapping = 31025 (TSA Status)

Table 27 transmit PDO mapping

Available Process data	CANobject (32 bit uint) PDO mappable	Configure mapped Modbus register by PPU menu	Configure mapped Modbus register by CANobject	Default RPDO configuration	Comment
FB signal 16	5d07	[266G]	0x4b00	---	See chapter 5.2.2.4 page 148, Default VFX/FDU mapping = 31011 (igbt temp) + Default TSA mapping = 31027 (DigIn status)

Note! TPDOs has a default Modbus mapping according to column "Comment" in Table 27 above. This can of course be changed to any other mapping suiting the application by altering menu [2661..266G] or alternatively writing to CANobject according to Table 26 and Table 27.

2.5.7.1 PDO mapping example

By default (according to DS301) four TPDOs and RPDOs are enabled. The PDOS are default configured like this:

Table 28 Output from the drive to the PLC (TPDOs)

TPDO1	TPDO2	TPDO3	TPDO4
32-bit Basic data status word	32-bit process value by default mapped to Modbus register 31002 speed	32-bit process value by default mapped to Modbus register 31004 torque	32-bit process value by default mapped to Modbus register 31005 shaft power.
Default Mapped CANobject 5c00	Default Mapped CAN-object 5d00	Default Mapped CANobject 5d01	Default Mapped CANobject 5d02

Table 29 Input to the drive from the PLC (RPDOs)

RPDO1	RPDO2	RPDO3	RPDO4
32-bit Basic data CONTROL word.	32-bit process value. By default not mapped to any Modbus reg (0).	32-bit process value. By default not mapped to any Modbus reg (0).	32-bit process value. By default not mapped to any Modbus reg (0).
Default Mapped CANobject 5e00	Default Mapped CANobject 5f00	Default Mapped CANobject 5f01	Default Mapped CANobject 5f02

Example 1: Map current to TPDO2

It is very easy to change the mapping of any PDO. Let us for instance assume that we would like to change mapping of TPDO2 from speed (default) to e.g. current which has Modbus register 31007 (ref: main product manual). From Table 27, page 44 we can then either change menu [2669] to 31007 or we can alternatively write 31007dec via SDO channel to CANobject 0x4af9 (42809-FB Signal 9). Both ways will result in that TPDO2 now points to parameter current instead of speed. A third way of doing the same is to change tha mapped CANobject of TPDO2. This is done by changing CANobject 1A01 sub 01 from pointing to 5d00 (FB signal 9) to instead point to 5d03 (FB signal 12) which by default is mapped to Modbus parameter current.

Example 2: Map Relay 1 function to RPDO2

In the same manner we can change the mapping of RPDO2. Assume that we would like to map the function of Relay1 to this PDO. From the main product manual we can see that the Modbus instance no for Relay 1, menu [551] is 43273. We then simply configure ppu menu [2661] FB signal 1 to 43273 or alternatively we can also configure this via SDO channel write by sending

43273 dec to CANobject 0x4af1 (42801-FB Signal 1). With RPDO2 we can now set the function of Relay 1.

Example 3: Add on/enable more PDOs

Further, it is very simple to add on even more PDO's when necessary. Simply map the wanted CANobject from Table 26 or Table 27, page 44 to the selected PDO via 16XXh (receive PDO mapping, see chapter 2.5.8) or 1AXXh (transmit PDO mapping, see chapter 2.5.8) and remember to also set & enable the PDO's corresponding COB-id. Specify which Modbus address to map to this new PDO by configuring the corresponding Fieldbus signal via PPU or via SDO command as described in Table 26 or Table 27. The Modbus parameter number is listed under each menu in the main product manual under 'Communication information' and 'modbus instance no'. Up to 9 TPDOs and 9 RPDOs can be configured.

Example 4: The scale / unit of each PDO

The scale and unit of each mapped Modbus parameter is found in the main product manual under the corresponding menu - section 'Communication information' and 'Fieldbus format'.

Lets for instance look at FB signal 12 of Table 27. This is per default mapped to Modbus register 31007 current. If we look in the main product manual under menu [716] Current we find the section 'Communication information' and in that also 'Fieldbus format' which states Long, 1=0.1A. Thereby we know that data of e.g. 1234 dec from this PDO is equal to a current of 123.4 Amps.

2.5.8 CANopen Object Dictionary

The standard object dictionary is implemented according to the DS302 specification (v4.02) from CiA (CAN in Automation).

Table 30

Index	Object Name	Sub Index	Description	Type	Acess	Notes
0005h	Dummy Object	00h	Dummy Object	U8	WO	-
0006h	Dummy Object	00h	Dummy Object	U16	WO	-
0007h	Dummy Object	00h	Dummy Object	U32	WO	-
1000h	Device Type	00h	Device Type	U32	RO	0000 0000h (No profile)
1001h	Error register	00h	Error register	U8	RO	-
1003h	Pre-defined error field	00h	Number of errors	U8	RW	
		01h... 06h	Error field	U32	RO	
1005h	COB-ID Sync	00h	COB-ID Sync	U32	RW	Default value is 0000 0080h. The Anybus CompactCom 30 CANopen module does not have Sync producer support.
1008h	Manufacturer device name	00h	Manufacturer device name	Visible string	RO	-
1009h	Manufacturer hardware version	00h	Manufacturer hardware version	Visible string	RO	
100Ah	Manufacturer software version	00h	Manufacturer software version	Visible string	RO	
100Ch	Guard time	00h	Guard time	U16	RW	-
100Dh	Life time factor	00h	Life time factor	U8	RW	-

Table 30

Index	Object Name	Sub Index	Description	Type	Acess	Notes
1010h	Store Parameters	00h	Largest sub index supported	U8	RO	02h
		01h	Store all parameters	U32	RW	Baud rate and Node ID cannot be stored using this command. Relevant only for communication parameters.
		02h	Store Communication parameters	U32	RW	
1011h	Restore parameters	00h	Largest sub index supported	U8	RO	04h
		01h	Restore all default parameters	U32	RW	-
		02h	Restore communication default parameters	U32	RW	-
		04h	Restore manufacturer parameters to Default.	U32	RW	-
1014h	COB ID EMCY	00h	COB ID EMCY	U32	RO	-
1015h	Inhibit Time EMCY	00h	Inhibit Time EMCY	U16	RW	Default value is 0000h
1016h	Consumer Heartbeat Time	00h	Number of entries	U8	RO	01h
		01h	Consumer Heartbeat Time	U32	RW	Node ID + Heartbeat Time. Value must be a multiple of 1 ms.
1017h	Producer Heartbeat Time	00h	Producer Heartbeat Time	U16	RW	-
1018h	Identity object	00h	Number of entries	U8	RO	04h
		01h	Vendor ID	U32	RO	-
		02h	Product Code	U32	RO	
		03h	Revision Number	U32	RO	
		04h	Serial Number	U32	RO	

Table 30

Index	Object Name	Sub Index	Description	Type	Acess	Notes
1400h ... 141Fh	Receive PDO parameter	00h	Largest subindex supported	U8	RO	02h
		01h	COB ID used by PDO	U32	RW	-
		02h	Transmission type.	U8	RW	-
1600h ... 161Fh	Receive PDO mapping	00h	No. of mapped application objects in PDO	U8	RW	-
		01h	Mapped object #1	U32	RW	-
		02h	Mapped object #2	U32	RW	-
		03h	Mapped object #3	U32	RW	-
		04h	Mapped object #4	U32	RW	-
		05h	Mapped object #5	U32	RW	-
		06h	Mapped object #6	U32	RW	-
		07h	Mapped object #7	U32	RW	-
		08h	Mapped object #8	U32	RW	-
1800h ... 181Fh	Transmit PDO parameter	00h	Largest subindex supported	U8	RO	05h
		01h	COB ID used by PDO	U32	RW	-
		02h	Transmission type	U8	RW	-
		03h	Inhibit time	U16	RW	-
		05h	Event Timer (ms)	U16	RW	-

Table 30

Index	Object Name	Sub Index	Description	Type	Acess	Notes
1A00h ... 1A1Fh	Transmit PDO mapping	00h	No. of mapped application objects in PDO	U8	RW	-
		01h	Mapped object #1	U32	RW	-
		02h	Mapped object #2	U32	RW	-
		03h	Mapped object #3	U32	RW	-
		04h	Mapped object #4	U32	RW	-
		05h	Mapped object #5	U32	RW	-
		06h	Mapped object #6	U32	RW	-
		07h	Mapped object #7	U32	RW	-
		08h	Mapped object #8	U32	RW	-

2.5.9 Example reading service data object (SDO)

The following example is valid for AC drive but the principles are the same regardless of used main product.

We want to read the Current [716] from the main product. In the main product manual we find the EtherCAT / CANopen index stated to be 0x23ef.
The fieldbus format of this parameter is data-type Long, 1 = 0.1A.

By sending an SDO-read object request to Object number 0x23ef with subindex 0 we receive a result back which is 4 bytes (data-type long) and where the result value is e.g. 125. This means that the actual current value is 12.5 Amps.

3. Industrial Ethernet networks

3.1 General description

3.1.1 History

Ethernet was invented more than 30 years ago, May 22 1973, by Mr. Robert “Bob” Metcalfe and David Boggs. Robert developed a mechanism of interconnecting a Xerox printer and several computers via one communication media. The name Ethernet came from the term “Ether” which scientists in the beginning believed was the medium for propagation of electromagnetic waves.

3.1.2 The difference between office Ethernet and Industrial Ethernet

The reason for not using standard office Ethernet for industrial applications is mainly the non deterministic response time. Standard Ethernet utilizes Carrier-Sense Multiple Access with Collision Detection (CSMA/CD) to manage data transmissions. Simply explained this means that two nodes may start to transmit at the same time and if a 'collision' is detected they both back-off a random time. The definition of determinism is the ability to get a response back within a consistent and predictable time.

With office Ethernet we do not know if the response comes after e.g. 100 ms or 600 ms. I.e. it is not really suitable for real time applications. Given the risk of collisions during data transmissions we get a non deterministic behavior. This is solved in many ways for the different Industrial Ethernet (IE) protocols, but a common factor is to use switches to avoid communication collisions. A switch works as a single node-to-node link and can buffer meesags. In this way collisions can be avoided. IE protocols may also use lighter protocols with less overhead, reserved time slots use priority on telegrams, multicast etc. to solve the real time demands.

Another difference v.s. office Ethernet is an industrial environment with much more EMI, higher temperature and humidity, vibrations etc. This requires much more robust cables, contacts and switches (supporting priority, network management). The office network and the production network are normally separated from each other through firewalls, allowing data exchange but in a secure way.

3.1.3 General Ethernet menus

In menu [265] Ethernet you will find all general Ethernet settings, see Table 58, page 120. If any Ethernet parameter is altered, all submenus of menu [265] Ethernet will be flashing to indicate the change. Altered parameters will only be updated after power cycling (see second note below).

All Ethernet related settings are stored in the fieldbus module itself, which means that you may switch it over to a new replacement main product if you would like to keep all communication settings and the unique MAC address.

The IP address is a 4 byte node address (default 0.0.0.0) which may be manually configured in menu [2651]. Alternatively, it may be automatically set by a DHCP server (Dynamic Host Configuration Protocol) if you turn menu [2655] DHCP to setting On.

NOTE: If DHCP is enabled (ON) parameters in menu [2651]-[2654] become read only and may only become altered by the DHCP server.

You may also use the IPconfig PC program (download from www.emotron.com) to change Ethernet settings, see § 3.1.4. In general, it is recommended that you contact your network administrator to get a proper installation. Note that overlapping IP addresses will cause conflicts on the network.

In menu [2652] the MAC address is presented (MAC=Media Access Control Address). This is a unique world-wide 6 byte node address. Access is read only. The MAC address is also printed on a label on the bottom side of the module (see § 8.1.2 on how to insert/remove the module from the circuit board).

The subnet mask is configured in menu [2653]. It is used to segment nodes belonging together (e.g. a production cell) into a common group called subnet. The subnet mask defines which portion of the IP address should be seen as a Net & Subnet ID (defining network group) and which should be seen as a Host ID (defining a node within the network group).

Gateway IP address is defined in menu [2654]. A gateway is the device which controls the access between networks, whereas switches take care of access within a single network.

Menu [2655], DHCP may be set to ON or OFF. When set to ON, the IP address of the node is automatically configured by a DHCP server. When set to OFF, the IP address needs to be configured manually.

NOTE: The main product must be restarted before the new setting(s) will become active**. See chapter 2.1.4.3 for more information.**

Table 31 Ethernet settings – configuration methods

Setting / Set by	Control Panel	IPconfig PC program	Web interface
IP address	X	X	X
Subnet mask	X	X	X
Gateway address	X	X	X
DHCP setting	X	X	X
Host name		X	X
Domain name			X
Primary DNS		X	
Secondary DNS		X	

3.1.4 Anybus IPconfig

You may use the PC program Anybus IPconfig (download it from CG Drives & Automation's web page) to see all Industrial Ethernet options installed nodes in your network (valid for modbus/TCP). With this program you can also re-configure each nodes network specific parameters.

NOTE: Some switches may filter out broadcasts sent by the IPconfig program. An alternative in such case is to plug your laptop with IPconfig directly to the fieldbus module. This program uses port 3250.

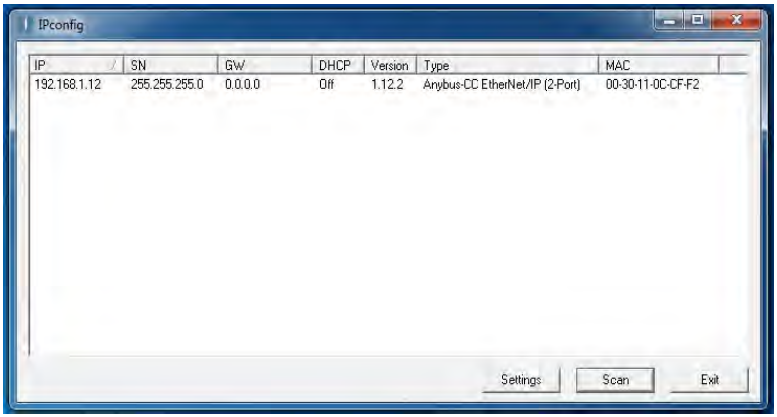


Fig. 7 The IPconfig tool

By double clicking on a line (e.g. the IP number) the following configuration window will be shown (example):

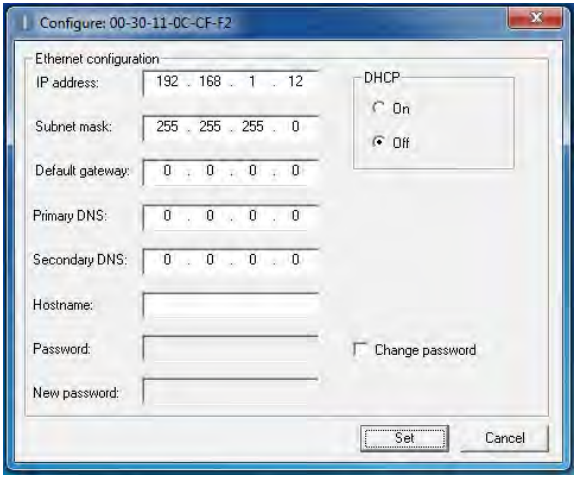


Fig. 8 Network specific configuration using IPconfig

NOTE: The password protection is currently not supported.

You may now manually change the settings of the node.

Press Set to upload the settings.

3.1.5 Web server interface

The CG Drives & Automation Industrial Ethernet options (Modbus /TCP, ProfinetIO, EtherNet/IP) has a user friendly, built in web server.

This web server may be accessed through any PC with e.g. internet explorer installed. Simply enter the IP address of the Emotron node you wish to reach in the Address field of Internet Explorer.

At any time you may use the Refresh button in your Web browser (normally function key F5 with Internet Explorer) to update the web page with the most recent values.

In the main menu you will be able to select between the links Network Interface and Parameter data.

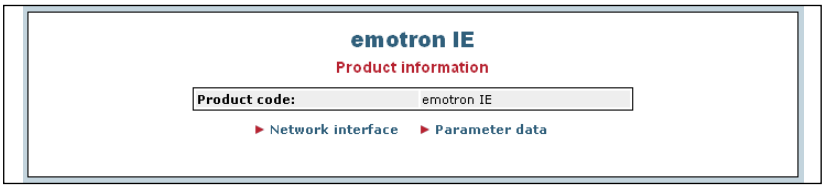


Fig. 9 Main menu

NOTE! Menu [2657] WEB Server can be used to enable access to WEB server. The default setting of menu [2657] is Off for cyber security reasons. Valid from VSD sw 5.12. See Table 58.

In the Network Interface web page you will be able to see the following information:

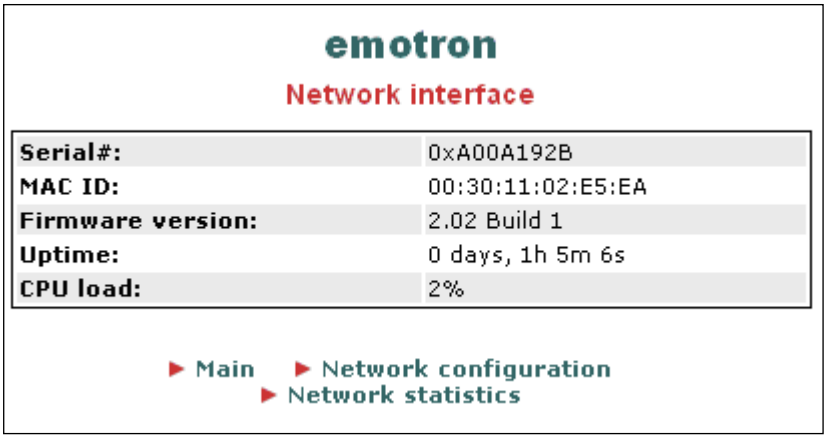


Fig. 10 Web page Network Interface

The Fieldbus modules serial number & firmware version, MAC ID which is a 6 byte globally unique ID of the node. The MAC ID is used by network switches to route the telegram to the correct port.

From this web page you may select another two links, Network configuration and Network statistics (general information about the network node/link status). Under Network configuration you will be able to do a user friendly configuration of the same parameters as described in § 3.1.3, page 54.

This page may also contain IE-protocol specific settings, like the Modbus/TCP time-out settings in the example on next page. Protocol specific settings are described further on in this manual.

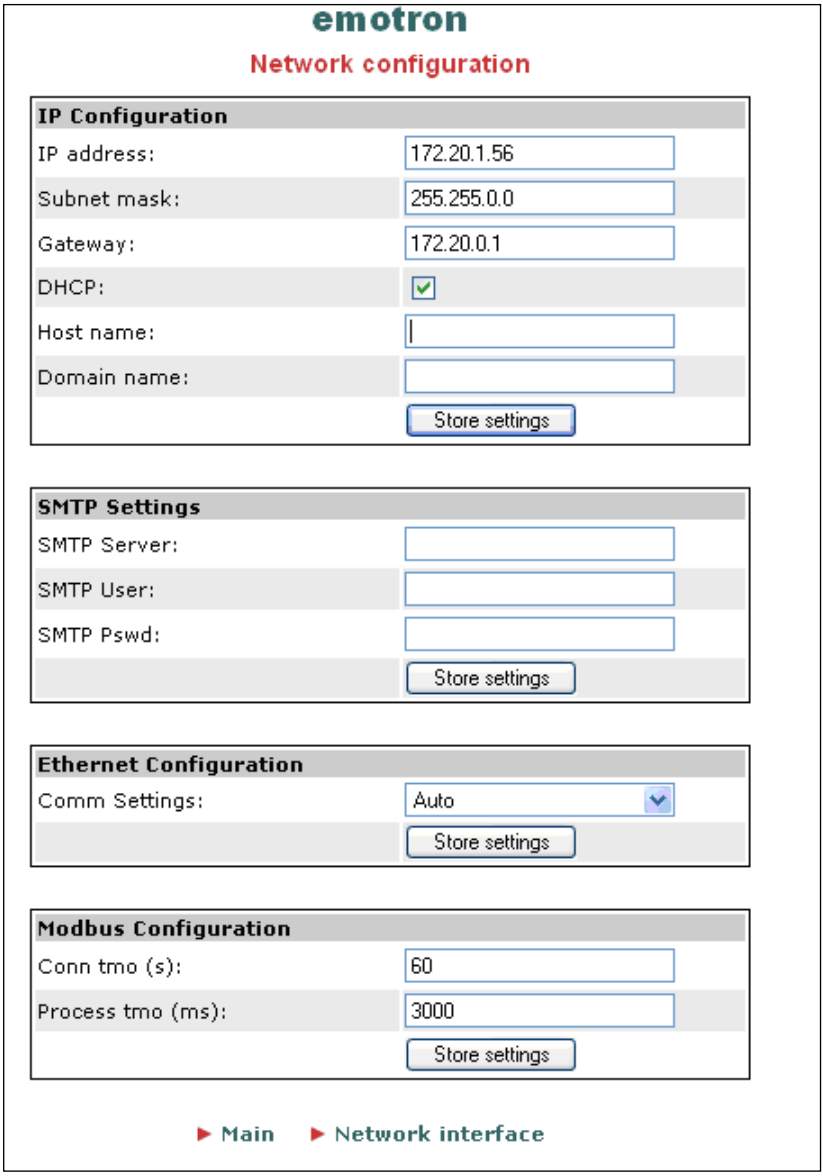


Fig. 11 Example: Network configuration page of a Modbus/TCP module

DNS (Domain Name Server) is supported. This means that you can make a logical connection between a name/text string and an IP address. Ask your network administrator for a Host and Domain name and fill it in on the web page (only possible through web interface). Further, the address to the DNS server(s) should be filled in by using the IPConfig PC program, see section 3.1.4 on page 55. In your web browser you may then type in the Host name, e.g. AC drive03 and use this name in the address field of internet explorer instead of an IP address like e.g. 172.20.1.56 to connect to the web server.

SMTP settings (Simple Mail Transfer Protocol to handle e-mail) are currently not in use.

Under Comm Settings you may select "Auto" baudrate (default) or 10/100 Mbit/s in Half (HDX) or full (FDX) duplex.

Modbus/TCP specific settings:

Conn tmo (s): Default value 60(s). If there has been no Modbus/TCP connection for a certain amount of time, e.g. because of a broken network cable, the module will kill the unused connection to make sure that new connections/ resources are available (avoiding lock up). Function may be disabled by setting the value to 0(s).

Process tmo (ms): Default value 0 (ms). After a Process Active Timeout (i.e. no Modbus/TCP connection for a certain time), the Network status LED will start to flash with a red colour. Further, SUP bit will go to 0 and the state machine in control panel menu [2693] will go to state 2, WAIT_PROCESS.

NOTE: SUP-bit indication in menu [2692] and time-out status LED indication are only active if a Process tmo value separate from zero is used.

If you go back to the Main menu you will also be able to reach the Parameter data link, which contains an overview of all the parameters in the AC drive see Fig. 12.

All user parameters in the AC drive are listed by ordinal from one to the total amount of parameters. The ordinal is listed in column denoted "#".

NOTE: The ordinal does not have any connection to the Modbus Instance no/Devicenet no listed in the main product manual or "Communication settings" list on www.emotron.com/ www.cgglobal.com.

Under the second column “Parameter” the name identifying a specific parameter is presented. The following denotation is used:

ParSet//Triplog list//Fieldbus Signal – Modbus number – Menu name

If the parameter belongs to a parameter set {A-D} or a trip log list {AL1-AL9} it is listed in the first field. If it is a fieldbus signal, it is indicated as {S1-S16}. If it does not belong to a parameter set nor a trip log list or is a fieldbus signal, the field is left blank.

In the second field the Modbus Instance no is listed to identify a unique parameter. All parameters listed in the main product manual has such a number.

If the parameter has an existing PPU menu name this is also listed here to make identification easier.

The right most column “Value” allows you to read or write to a certain parameter. The range (min-max) is described in the main product manual, look for the corresponding Modbus Instance no.

NOTE: All values presented on the web server use the Fieldbus format specified in the manual for the main product (The E-int format is never used here to make the human interpretation easier).

The page is not automatically updated when e.g. a value changes. To update the page, press the refresh button in Internet Explorer.

Further, you may choose how many parameter you would like to display in one page by entering “Number of parameters per page” in the top of the page and pressing Set.

It is also possible to scroll one page back or forward by pressing the labels “<< Previous” or “Next >>”, or simply by a mouse click on the block of parameters you wish to view, e.g. “1-10” in the figure below.

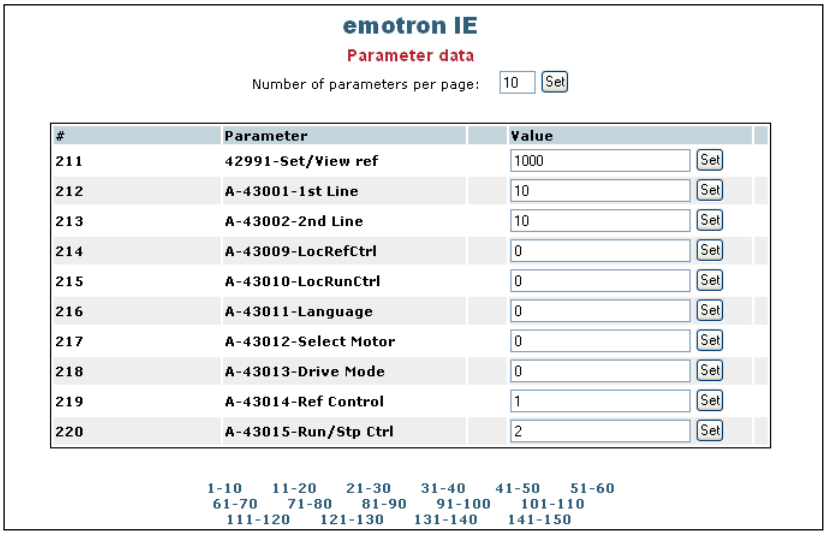


Fig. 12 Parameter web page interface

HINT: Store or print a list of all the main product parameters as a backup by setting “number of parameters per page” to the total amount of parameter in the main product. Note that this procedure may take several minutes. This list is also useful as a look-up table (see below).

Table 32 Look-up table - ordinal v.s. Modbus instance number for Emotron AC drives

# (ordinal)	Modbus number	Comment
14	30052	Start of 3x parameters
92	31101	Trip log list 1 “AL1”
126	42301	Start of 4x parameters
157	42801	Start of fieldbus signals
212	43001	Start of parset A
585	31151	Trip log list 2 “AL2”
619	31201	Trip log list 3 “AL3”
653	31251	Trip log list 4 “AL4”
687	31301	Trip log list 5 “AL5”
721	31351	Trip log list 6 “AL6”
755	31401	Trip log list 7 “AL7”
777	31451	Trip log list 8 “AL8”
811	31501	Trip log list 9 “AL9”
845	44001	Start of parset B
1212	45001	Start of parset C
1579	46001	Start of parset D

Table 33 Look-up table - ordinal v.s. Modbus instance number for Emotron TSA softstarter

# (ordinal)	Modbus number	Comment
8	30052	Start of 3x parameters
83	31101	Trip log list 1 “AL1”
136	42300	Start of 4x parameters
227	42801	Start of fieldbus signals
291	43001	Start of parset A
742	31201	Trip log list 2 “AL2”
794	31301	Trip log list 3 “AL3”
846	31401	Trip log list 4 “AL4”
898	31501	Trip log list 5 “AL5”
950	31601	Trip log list 6 “AL6”
1002	31701	Trip log list 7 “AL7”
1054	31801	Trip log list 8 “AL8”
1106	31901	Trip log list 9 “AL9”
1158	44001	Start of parset B
1607	45001	Start of parset C
2056	46001	Start of parset D

NOTE: Ordinal may vary depending on software revision.

NOTE: All parameter data format/scaling/range information is found in the main product manual for each modbus register. Note that the “modbus format” is used. For some parameters the specific Emotron format “eint” may be used. This is also described in the main product manual for each modbus register or in the “Communication settings” list on www.emotron.com/ www.cgglobal.com.

3.2 Recommendations for Industrial Ethernet - IE network installation

In general, the same hints as given in § 2.2 can be re-used.

- Most IE protocols have excellent installation guidelines. Use them!
- History tells us that most network problems arise from bad cable installations & contacts. Make sure that you use robust components in your network. Do not use products for office environment.
- When bending cables: Use edge protectors (for sharp edges) and clamps of suitable size to keep the bending radius fixed.
- Avoid torsion and squeezing.
- Install data and power supply cables separately (different cable routes)
- Use robust, preferably managed network switches.
- Never look at open fibre ends directly. Use protection caps on loose fibre ends.
- Redundant cables should be laid in separate path compare to the original cable, to avoid simultaneous damage to both cables.
- Keep the industrial network traffic separated from the office network.
- Use CAT5e STP cable or better.
- Protocol vendors have different installation guidelines to follow. In some cases like e.g. ProfiNET there is a recommendation to connect the shield of the network cable directly to PE/FE (functional earth). Whereas it in other cases is recommended to leave it and instead allow the integrated RC-filter of the ABCC module connect the shield towards PE/FE internally. Follow the recommendations of the protocol vendor in this case.

3.3 Modbus/TCP

Modbus/TCP is an open industrial protocol created by a Schneider Electric & Modbus/IDA, which is a no-profit, independent, member-based organization.

It is a client/server protocol, where a client sends a request to a server using TCP port 502. The server in turn processes the message and sends a response back to the client. A client may be a PLC master or a PC whereas the server may be an CG Drives & Automation Emotron FDU 2.0, VFX 2.0. or TSA

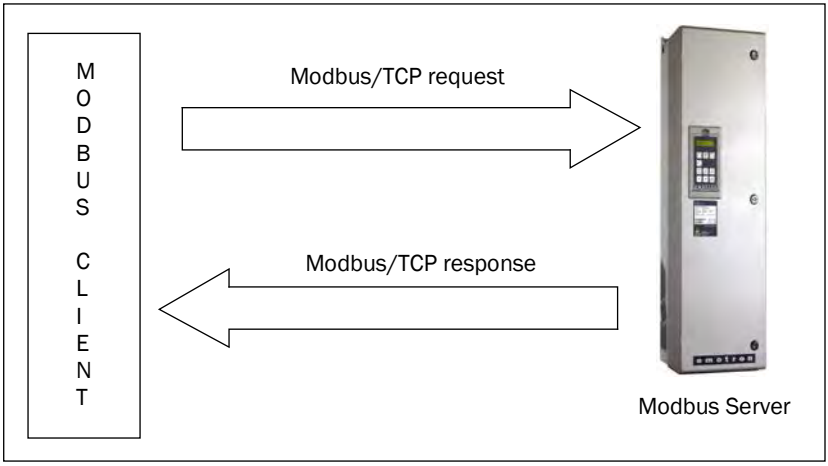


Fig. 13 Modbus/TCP Client/Server relation

3.3.1 Modbus/TCP interface and LED indication

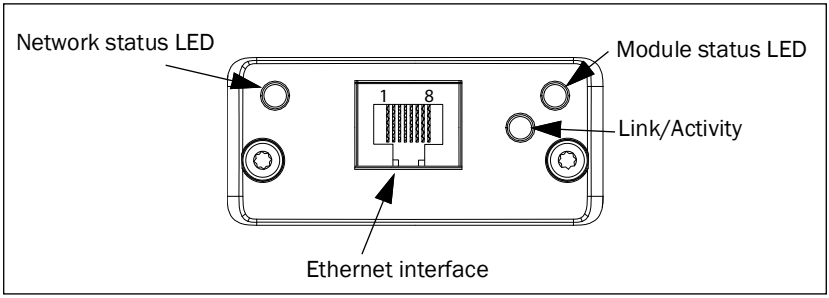


Fig. 14 LED indication for RJ45 connector

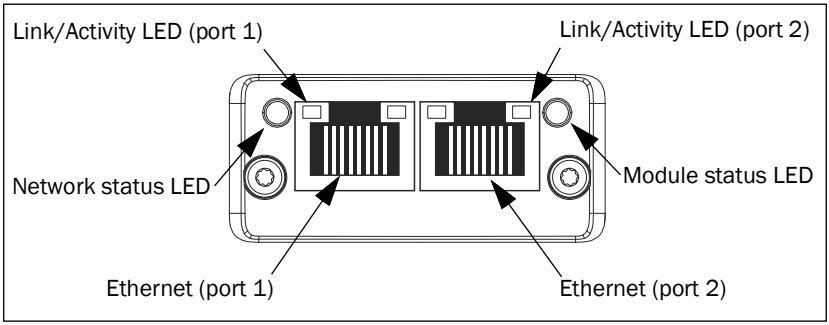


Fig. 15 LED indication for 2-port RJ45 connector

3.3.1.1 Network status LED description

NOTE: A test sequence is performed on this LED during startup

Table 34

Status	Description
Off	No power or no IP address
Green	Module is in Process Active or Idle state
Flashing green	Waiting for connection
Red	Duplicate IP address or FATAL event
Flashing red	Process active time-out

3.3.1.2 Module status LED description

NOTE: A test sequence is performed on this LED during startup

Table 35

Status	Comment
Off	No power
Green	Normal operation
Red	Major fault, module is in state EXCEPTION (or FATAL event)
Flashing red	Minor fault

3.3.1.3 Link/activity LED description

NOTE: A test sequence is performed on this LED during startup

Table 36

Status	Comment
Off	No link, no activity
Green	Link established
Green, flickering	Activity, 100Mbit/s
Yellow	Link established, 10Mbit/s
Yellow, flickering	Activity, 10Mbit/s

3.3.1.4 Modbus/TCP connector description

This module uses a “RJ45” (8P8C connector).

Table 37 RJ45 connector description

Pin	Signal	Description
1	Tx+	Pos. Transmit twisted pair
2	Tx-	Neg. Transmit twisted pair
3	Rx+	Pos. Receive twisted pair
4		
5		
6	Rx-	Neg. Receive twisted pair
7		
8		

3.3.2 Cable type

Recommendation is to use at least a CAT-5e (enhanced) straight cable. STP (Shielded Twisted Pair) should be used in environments with strong electromagnetic fields. ANSI recommends using CAT6 for new installations. Operating temperature is normally -10 to +60°C for a CAT-5e cable (check with cable manufacturer). CAT-5e is fully backward compatible to CAT-5 cables.

Maximum CAT-5e cable length 100 m (328 feet) including patch cables.

Table 38 Typical performance comparison of Cat-5, Cat-5e and Cat-6 cables

Cable Description	Cat-5 TIA/EIA-568-A Superseded	Cat-5e TIA/EIA-568-B Class D	Cat-6 TIA/EIA-568-B.2 Class E
Typical Band-width	100 MHz	350 MHz	550 MHz
Intersertion loss @100 MHz [dB]			
Cable	22.0	22.0	21.3
Connector	0.4	0.4	0.2
Channel	24.0	24.0	21.3
Next (Near End Cross Talk) @100 MHz [dB]			
Cable	N/S*	35.3	39.9
Connector	N/S*	43.0	54.0
Channel	N/S*	30.1	39.9
Elfext (Equal Level Far-End Crosstalk) @100 MHz [dB]			
Cable	N/S*	23.8	27.8
Connector	N/S*	35.1	43.1
Channel	N/S*	17.4	23.3
Return Loss @100 MHz [dB]			
Cable	16.0	20.1	20.1
Connector	14.0	20.0	24.0
Channel	8.0	10.0	12.0

* N/S = Not specified

3.3.3 Baudrate

The Modbus/TCP option supports 10/100 Mbit/s, auto negotiation is default, but full or half duplex operation is selectable. For further information see section 3.1.5 on page 57.

3.3.4 Modbus/TCP protocol description

Modbus/TCP is basically a Modbus RTU telegram (well known industry standard) encapsulated within the data part of a TCP frame on Ethernet. A Modbus/TCP frame does however not contain any CRC field (Cyclic Redundancy Check) or Address because since this is taken care of by other lower layers in the OSI model (IP address is used).

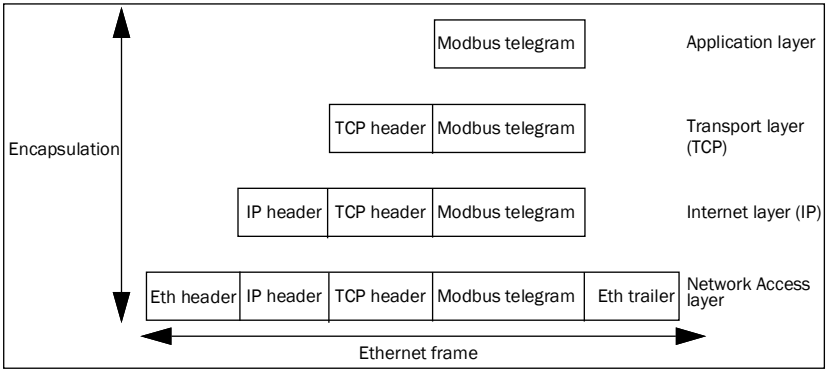


Fig. 16 Encapsulation of a Modbus frame using the OSI model

NOTE: PORT 502 must be used for Modbus/TCP telegrams sent to the Fieldbus option. This option can handle up to four simultaneous Modbus/TCP connections.

A general Modbus RTU frame consists of an Address field, Function code (with some additional information like starting parameter address, number of registers/bytes etc), Data field & a Checksum.

As mentioned before the Address field and the Checksum is removed in Modbus/TCP frames because it is handled by lower layers. The two fields left (Function code and Data) forms what is called a Protocol Data Unit (PDU).

NOTE: All parameter data format/scaling/range information is found in the main product manual for each modbus register. Note that the "modbus format" is used. For some parameters the specific Emotron format "eint" may be used. This is also described in the main product manual.

A few other fields are added to the PDU in a Modbus/TCP frame:

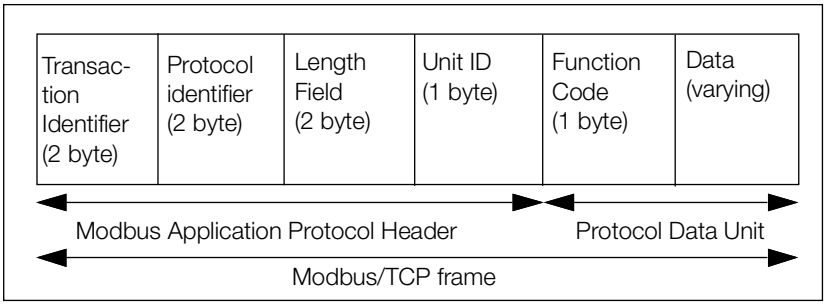


Fig. 17 Contents of a Modbus/TCP frame

In the MPAB header, the transaction identifier is set by the client to identify each unique request since the responses may not be received in the same order as the requests.

The Protocol identifier is always 0 for Modbus services. Other values are reserved for future use.

The Length field is used as a byte counter for the rest of the telegram, so the receiver can make sure that it has received the correct amount of bytes before starting to process the information.

The unit identifier is only used if there are several nodes behind one IP address. This is not used for the Emotron FDU 2.0, Emotron VFX 2.0 and can be left unimplemented / set to 0.

The Modbus/TCP module supports the following function codes:

Table 39 Function codes for Modbus/TCP

Function name	Function code	Modbus registers
Read coil status	1 (01h)	1-9, 38-43
Read Holding Registers	3 (03h)	40001-49999
Read Input Registers	4 (04h)	30001-39999
Force single coil	5 (05h)	1-9, 38-43
Write Single Register	6 (06h)	40001-49999
Force multiple coils	15 (0Fh)	1-9, 38-43
Write Multiple Registers	16 (10h)	40001-49999
Read/Write Multiple Registers	23 (17h)	30001-39999, 40001-49999

3.3.5 How to convert modbus numbers into starting addresses

For input registers (modbus numbers 30001-39999) the starting addresses are calculated by subtracting 30001 from the modbus number in the table, e.g. modbus number 30011 has a starting address of 30011-30001=10.

For holding registers (modbus numbers 40001 and up) the starting addresses are calculated by subtracting 40001 from the Modbus number in the table, e.g. modbus number 41137 has the starting address 1136.
The modbus address in the following examples (chapter 3.3.6 to chapter 3.3.15) are valid for AC drives.

3.3.6 Read Coil Status

Reads the status of changeable digital parameters.

Example

Requesting the RUN status. Result is that the AC drive is stopped.

RUN status: Modbus no = 2 (02h), start address 1 (01h)

Data: Stopped = 0

1 byte of data: Byte count = 01

Request message

Field name	Hex value
Function	01
Start address HI	00
Start address LO	01
Number of Coils HI	00
Number of Coils LO	01

Response message

Field name	Hex value
Function	01
Byte count	01
Data	00

3.3.7 Read Holding Registers

Example

Reading the currently selected language, modbus number 43011 with starting address 0BC2h. The result is that the language is set to 1 (Swedish).

Request message

Field name	Hex value
Function	03
Start address HI	0B
Start address LO	C2
Number of Registers HI	00
Number of Registers LO	01

Response message

Field name	Hex value
Function	03
Byte count	02
Data HI first register	00
Data LO first register	01

3.3.8 Read Input Registers

Example

Reading modbus register 31002, the output speed, with corresponding starting address 03E9h. The result is that the motor is stopped (zero speed).

If you wish to read register 31003, output torque, at the same time, just increase the number of registers from 01 to 02 in the request message. The response message will then have a byte count of 04 and also contain the data information (2 bytes) of register 31003.

Request message

Field name	Hex value
Function	04
Start address HI	03
Start address LO	E9
Number of Registers HI	00
Number of Registers LO	01

Response message

Field name	Hex value
Function	04
Byte count	02
Data HI first register	00
Data LO first register	00

3.3.9 Force Single Coil

Sets the status of one changeable digital parameter.

Example

In the example below we set modbus register 2 (run). Note that Data HI should be set to 0xFF and Data LO to 0x00 to force the coil high.

Request message

Field name	Hex value
Function	05
Start address HI	00
Start address LO	01
Data HI	FF
Data LO	00

Response message

Field name	Hex value
Function	05
Start address HI	00
Start address LO	01
Data HI	FF
Data LO	00

3.3.10 Write Single Register

Example

Set parameter with modbus number 43020, Level/Edge, to Edge = 1. The corresponding starting address is 0BCBh.

Request message

Field name	Hex value
Function	06
Start address HI	0B
Start address LO	CB
Data HI	00
Data LO	01

Response message

Field name	Hex value
Function	06
Start address HI	0B
Start address LO	CB
Data HI	00
Data LO	01

3.3.11 Force Multiple Coil

Sets the status of multiple changeable digital parameters.

Example

In the example below a random modbus number has been used (may not exist in the main product).

Request message

Field name	Hex value
Slave address	01
Function	0F
Start address HI	00
Start address LO	00
Number of Coils HI	00
Number of Coils LO	02
Byte count	01
Coil no. 0-1 status (0000 0011B)	03

Response message

Field name	Hex value
Slave address	01
Function	0F
Start address HI	00
Start address LO	00
Number of Coils HI	00
Number of Coils LO	02

3.3.12 Write Multiple Register

Example

In the example below a random modbus number has been used (may not exist in the AC drive).

Request message

Field name	Hex value
Function	10
Start address HI	00
Start address LO	11
Number of Registers HI	00
Number of Registers LO	02
Byte count	04
Data HI first register	00
Data LO first register	FA
Data HI second register	00
Data LO second register	37

Response message

Field name	Hex value
Function	10
Start address HI	00
Start address LO	11
Number of Registers HI	00
Number of Registers LO	02

3.3.13 Write/Read Multiple Register

Sets and reads the contents of multiple changeable parameters in the same message.

Example

Set modbus parameter 43064, thermal protection, to PTC=1 and also set next parameter 43065, Motor class, to Class F=5. The corresponding starting address for modbus parameter 43064 will be 0BF7h.

At the same time we will read the contents of modbus numbers 43035 and 43036 which are fieldbus settings for process data mode and R/W settings. The result will be 4 = 4 bytes process data and 0 = R/W allowed. The corresponding starting address for modbus number 43035 will be 0BDAh.

Request message

Field name	Hex value
Function	17
Start read address HI	0B
Start read address LO	DA
Number of read registers HI	00
Number of read registers LO	02
Start write address HI	0B
Start write address LO	F7
Number of write registers HI	00
Number of write registers LO	02
Byte count	04
Data HI first register	00
Data LO first register	01
Data HI second register	00
Data LO second register	05

Response message

Field name	Hex value
Function	17
Byte count	04
Data HI first register	00
Data LO, first register	04
Data HI second register	00
Data LO second register	00

3.3.14 Exception codes

Exc. code	Name	Description
01	Illegal function	This unit doesn't support the function code.
02	Illegal data address	The data address is not within its boundaries.
03	Illegal data value	The data value is not within its boundaries.

3.3.15 Regrouping of registers

If you would like to read several different modbus registers with gaps in between their parameter numbers, it would normally require sending several e.g. "Read Holding Register" commands. To avoid this it is possible to group the registers of interest to a common area and read all at once with a single "Read Holding Register" command. This will make the programming easier and you will utilise the bandwidth of your network better.

You may configure up to 16 registers which are mapped to a common area. It is possible to configure all sorts of combinations where all 16 registers are "read registers", "write register" or a combination. Note that when using a combination of "read & write" registers, it is suitable to group all "read registers" in one block and all "write registers" in the next.

3.3.15.1 How to regroup registers

There are three ways to configure the grouping of parameters:

a) With the Control Panel by setting menu [2661] - [266G]

b) Configuration from web server (e.g. using Internet explorer)

c) Configuration by Modbus telegram

Example: We want to regroup modbus parameter 31002 (Speed - see AC drive manual), 31007 (Current), 31011 (Heatsink Tmp) as “read registers” and 43271 (DigOut1) and 43273 (Relay 1) as “write registers”.

Using configuration method a), simply alter the control panel settings as follows:

Table 40

Control panel menu	Set to
[2661] FB Signal 1	31002 (Speed)
[2662] FB Signal 2	31007 (Current)
[2663] FB Signal 3	31011 (Heatsink Tmp)
[2664] FB Signal 4	43271 (DigOut 1)
[2665] FB Signal 5	43273 (Relay 1)

Table 1 Regrouping of registers - control panel configuring.

NOTE: It can also be suitable to leave a gap (for spare parameters) between the block of read- and write registers if more parameters are to be added in the future. Modbus parameter 0 is used to indicate a spare parameter.

The same configuration using method b):

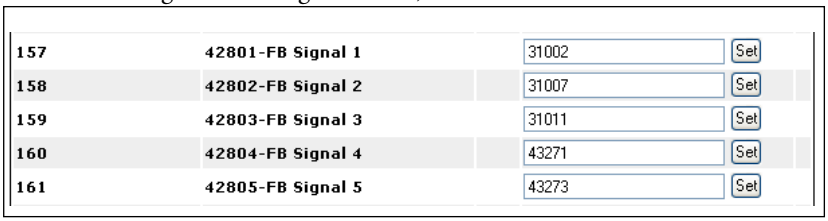


Fig. 18 Regrouping of registers - web server configuring

The 16 configurable registers are named FB Signal 1-16 and has the modbus addresses 42801-42816.

It is also possible to configure this regrouping using the modbus protocol (alternative c).

Modbus register 42801-42816 corresponds to the FB Signal 1-16.

According to the example above, sending a “Write Multiple Register” command with start address corresponding to modbus register 42801 and with data 31002,31007,31011,43271 & 43272 will result in the same configuration as in a) and b).

HINT: Control and Status registers (see § 3.3.13) with modbus registers 49971 and 49972 can also be mapped to this group since they are often read/written.

3.3.15.2 How to access the DATA of the regrouped registers

The corresponding DATA to the 16 configurable registers are accessed by modbus register 42821-42836 (DATA of the registers mapped onto modbus register 42801-42816).

When configuration of the regrouped registers are finished we can send a single telegram “Write/Read Multiple Registers” to access all parameters.

Start read address should be set to the modbus address that corresponds to the first “read register”, in the example above modbus register 42821. Number of read registers should be set to 3. Start write address should be set to the modbus address that corresponds to the first “write register”, in the example above modbus register 43824. Number of write registers should be set to 2. Data should be set to the function of DigOut1 and Relay1.

The result of the configuration example may also be seen in the web server:

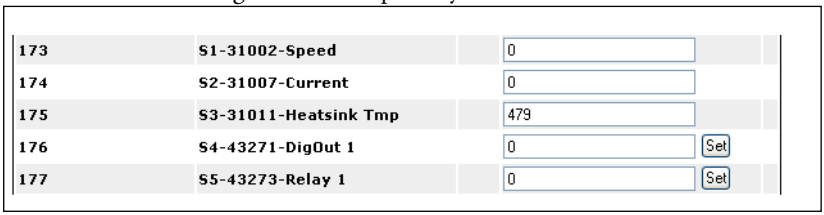


Fig. 19 Regrouping of registers - DATA of mirrored registers

HINT: Regrouping may alternatively be used to present up to 16 parameters of interest in a common block on the web server.

In Fig. 19 above, the five registers (according to the example above) are indicated with S1-S5 and also presented with modbus number and name.

3.3.16 Control & Status register for Modbus/TCP

For the Modbus/TCP protocol there exists no true process data since all communication is running in an acyclic manner. The menus [2632] PrDataMode and [2634] AddPrValues, described in chapter 5., are not used for Modbus/TCP.

However, the control and status data described in Table 63 can still be used by accessing modbus parameter numbers 49971 and 49972. Note that reading/writing to these registers are handled in a special way as follows.

When writing register 49971 and 49972 it will be interpreted as the Control message to AC drive, see Table 63, page 133 or for Emotron TSA see Table 65, page 135.

When reading register 49971 and 49972 it will be interpreted as the Status message from AC drive, see Table 65, page 135 or for Emotron TSA see Table 69, page 140.

NOTE: modbus register 49971 is mapped to byte 2 & 3 of process data and modbus register 49972 is mapped to byte 0 & 1. See data contents in Table 63 and Table 65.

Table 41 Modbus registers for process data

Modbus register	MSB/LSB part	Compare to basic process data
49972	LSB	Byte 0
	MSB	Byte 1
49971	LSB	Byte 2
	MSB	Byte 3

3.4 EtherCAT®

EtherCAT® is a registered trademark and a patented technology, licensed by Beckhoff Automation GmbH, Germany. EtherCAT, Ethernet for Control Automation Technology, is an open industrial protocol which has been optimised for time critical applications. However, the protocol is equally suitable for general I/O-data transfer and can be used throughout the entire automation network from a small temperature sensor to a fast servo system with very high response time demands.
The protocol rights were in year 2004 passed to the EtherCAT Technology Group (ETG) to continue promoting the protocol. ETG is a user and vendor organization based in Nürnberg, Germany with at the time of writing over 1740 members.

EtherCAT is part of the IEC standard 61158 as well as IEC 61784-2.

3.4.1 EtherCAT® interface and LED indication

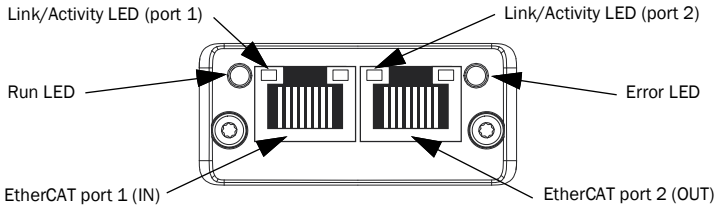


Fig. 20

Following indicator states are defined according to DR303-3 (CiA).

LED on	The LED is constantly on
LED off	The LED is constantly off
LED flashing 10 Hz	Indicates the iso-phase on and off with a frequency of 10 Hz; on for 50 ms and off for 50 ms.
LED flashing 2.5 Hz	Indicates the iso-phase on and off with a frequency of 2.5 Hz; on for 200 ms and off for 200 ms.
LED single flash	One short flash (200 ms) followed by a longer off phase (1000 ms).
LED double flash	Sequence of two short flashes (200 ms), separated by an off phase (200 ms). The sequence ends with a longer off phase (1000 ms)

3.4.1.1 Run LED

This LED reflects the status of the CoE (CANopen over EtherCAT) communication.

Table 42

Status	Indication	Description
Off	INIT	CoE device in INIT-state (or no power)
Green	OPERATIONAL	CoE device in Operational state
Green flashing	PRE-OPERA-TIONAL	CoE device in Pre-operational state
Green single flash	SAFE-OPERA-TIONAL	CoE device in Safe-operational state
Red **	Fatal event)	Error

**) If Run LED and Error LED turn red, this indicates a fatal event, forcing the bus interface to a physical passive state. Contact CG Drives & Automation support.

3.4.1.2 Error LED

This LED indicates EtherCAT communication errors etc.

Table 43

Status	Indication	Description
Off	No error	No error (or no power)
Red flashing	Invalid configuration	State change received from master is not possible due to invalid register or object setting.
Red, double flash	Application watchdog timeout	Sync manager watchdog timeout
Red **	Application controller failure	Anybus module in Exception

**) If Run LED and Error LED turn red, this indicates a fatal event, forcing the bus interface to a physical passive state. Contact CG Drives & Automation support.

3.4.1.3 Link/Activity LED

The left of the two LEDs that are integrated in each port indicates the EtherCAT link status.

Table 44

Status	Indication	Description
Off	No link, no activity	Link not sensed (or no power)
Green	Link established	Link sensed, no traffic detected
Green, flashing	Link sensed, activity detected	Link sensed, traffic detected

3.4.1.4 Ethernet Connector (RJ45)

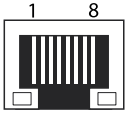


Table 45

Pin	Signal	Notes
1	Tx+	
2	Tx-	
3	Rx+	
4	-	Normally left unused, to secure signal integrity, these pins are tied together and terminated to PE via a filter circuit in the module.
5	-	
6	Rx-	
7	-	Normally left unused, to secure signal integrity, these pins are tied together and terminated to PE via a filter circuit in the module.
8	-	

3.4.2 EtherCAT technology description

3.4.2.1 How to utilise the bandwidth

In general for automation networks, the process data length for most nodes is short compared to the complete Ethernet frame. It would therefore lead to low utilisation of the available bandwidth to send a Ethernet frame to each and every node, like it is done in a traditional master-slave network. Instead, EtherCAT utilises one single Ethernet frame which is then shared by all nodes on the network. All nodes on the network have a dedicated segment/part of the data contained inside this 'global' Ethernet frame, where it reads its inputs and adds its outputs. Thus, resulting in a good bandwidth utilisation.

3.4.2.2 Fast processing of I/O data 'on the fly' handled by hardware ASIC

The Ethernet packet is transferred from node to node in a logical ring. Each EtherCAT slave reads/writes the data 'on the fly' while the telegram passes through the device to the next slave node. This procedure is handled by hardware/ASIC which causes only a minimum delay of a fraction of a microsecond for each device. This can be compared to traditional handling of a Ethernet software stack handled by a microcontroller which generally leads to significantly longer delays.

Handling the process image transfer with an ASIC allows EtherCAT to achieve very short cycle times, since the host microprocessor which forwards the I/O data to the application does not affect the performance of the bus. The Ethercat Slave Controller (ESC, Asic) simply stores the data in a DPRAM which then can be read by the host microprocessor at its own pace without affecting bus performance. The host microprocessor in turn forwards the data to the host application (inverter control board) through e.g. a serial channel.

3.4.2.3 Physical interface and virtual ring topology

EtherCAT uses a standard Ethernet physical interface & cables, running on 100 Mbit/s (using e.g. 100 BASE-TX or 100 BASE-FX). Both pairs in a cat 5e cable is used which enables full duplex communication. See detailed cable recommendations in chapter 3.3.2. In reality one pair is used for the outgoing EtherCAT telegram and the other pair for the returning frame, creating a virtual ring for all nodes. If a slave device does not detect a downstream device it automatically 'short circuits' the incoming and outgoing line, so the last device in the line actually forms the virtual ring. An EtherCAT network may also use a physical

ring topology if the PLC master is equipped with two ports. Networks which use a ring topology provide a measure of redundancy since the master can reach all nodes from two paths in the case of cable break.

3.4.2.4 Synchonization with distributed clocks

Some EtherCAT slaves also support the concept of distributed clocks (DC), where a 64-bit 1ns counter is synchronized for all nodes and master. EtherCAT even takes the propagation delay of the Ethernet telegram into account when synchronizing events, e.g. robotic movement. The system time can also be synchronized by a Grand Master Clock based e.g. on a GPS received signal to get a coordinated universal time (UTC). The clock synchronization and propagation delay measurement is done periodically to minimise the error.

3.4.2.5 EtherCAT to Ethernet

With EtherCAT it is not necessary to use any switches since all slave nodes are equipped with both an incoming- and outgoing RJ45 port, enabling daisy-chaining. Avoiding switches will keep the cost of the network down as well as keeping the delays to a minimum (no queuing of telegrams in switches). Note however that special switches can be used to connect standard Ethernet equipment to a EtherCAT network, e.g. for browsing a EtherCAT device with a built in webserver. This is denoted as 'Ethernet over EtherCAT', EOE.

Since the EtherCAT telegram is encapsulated into a standard Ethernet frame it is also possible to exchange EtherCAT information over any standard Ethernet network (using UDp/ip). It should however be noted that the deterministic properties of EtherCAT is lost while doing so.

3.4.2.6 EtherCAT gateway to other protocols

To conserve earlier infrastructure investments it is possible to connect EtherCAT to almost any other classic fieldbus protocol by e.g. using gateways.

3.4.2.7 EtherCAT network topology - it's your choice

Even though it is natural to build a EtherCAT network in a Line//Daisy chain topology because of the two-port implementation, it is also possible build e.g. a tree/star topology or to use drop lines.

3.4.2.8 General description of the protocol

EtherCAT supports up to 65535 connected devices within a network. Each connected node is configured from a EtherCAT configuration tool by the use of a EtherCAT Slave Information (ESI) file. The ESI file contains information about the nodes features, i.e. what functions it is supporting or not and also contains a list of the acyclic parameters (called SDO's - Service Data Objects) as well as the cyclic I/O data configuration (called PDO's - Process Data Objects), Vendor ID etc. The ESI file is implemented in eXtensive Markup Language (XML) and is easily readable through any XML-viewer or even a browser. When they system boots up, the PLC master checks that the information it has, matches with what is setup on the slave side. If OK, the network goes up and starts to exchange data.

EtherCAT uses a standard Ethernet Frame with a maximum of 1514 byte, containing an Ethernet Header (6 byte destination, 6 byte source, EtherType = 0x88A4 for EtherCAT telegram), EtherCAT specific data and finally a frame check sequence (FCS).

The EtherCAT data consists of a 2 byte header and 48-1498 bytes of Datagrams to different EtherCAT slave nodes. The header itself contains information about the length of the following datagrams as well as the Type of data sent.

The datagrams in byte 48-1498 does in turn consist of a datagram header, data and a working counter (WKC). The header contains e.g. a 8 bit command and a 32 bit address, plus length field etc. The 32 bit address field can be used for different addressing methods, e.g. position/node/logical addressing. The working counter is used by the master to keep track of that all slaves have executed read & write properly.

The position address points out which slave (order in the virtual ring) should receive the datagram and to which local memory address (16 bit allows you to address up to max 64 kByte of memory). The node address works in a similar way but where a 16-bit node address points out a specific slave. Finally, the logical addressing can access a 4Gb master memory space which all slaves are mapped to. This mapping is handled by the master (max 65535 slaves, each with max 64 kByte memory).

Each EtherCAT slave (ESC) has a Fieldbus Memory Management Unit (FMMU) to translate a global address space into a local memory address. The SYNC managers handle the DP-ram access for cyclic and acyclic data to achieve consistent data between the ESC and the microcontroller (host) unit.

3.4.3 Functionality and settings of the EtherCAT option

This chapter describes the most important features and settings of the EtherCAT option to allow you to get a quick start. There is also a PLC example project for Beckhoffs Twincat system available on www.emotron.com along with documentation.

3.4.3.1 Supported features

The module support four Sync Managers, 0-3, to support Mailbox read/write and RxPDOs and TxPDOs. There are eight available FMMUs. The EtherCAT master can uses these freely for any purpose.

For Process data exchange one TPDO (transmit PDO) and one RPDO (receive PDO) is available.

For acyclic parameter exchange of service data objects (SDOs) there is support for CANopen over EtherCAT (CoE).

The following addressing modes are supported:

- Position addressing (physical position in logical ring - automatically assigned by master)
- Node addressing (unique, dedicated slave address)
- Logical addressing (dedicated part of the total 32bit, 4Gb address space)

Node addressing can be used when a static unit address is needed. The station alias / node address can be read from menu [2631] Address (read-only). Setting is done via the network. In TwinCAT this is done via EtherCAT tab for the node - Advanced Settings - ESC Access - E2PROM - Configured Station Alias.

The following EtherCAT services are supported: APRD, APWR, APRW, FPRD, FPWR, FPRW, BRD, BWR, LRD, LWR, LRW, ARMW, FRMW.

NOTE: Currently there is no support for distributed clocks (DC) or web server functionality.

The module supports PDI watchdog, which monitors the data exchange between the ESC and the local MPU inside the option. There is also support for SM watchdog, which monitors the PDO communication between the PLC master and the EtherCAT option.

NOTE: The SM watchdog needs to be enabled to allow menu [264] Com Fault and menu [2692] SUP-bit to function.

The device has passed conformance testing.

The ESI-file downloadable from www.emotron.com contains all inverter parameters (SDOs) identified by a unique EtherCAT index which can be used to find the corresponding parameter description in the main product manual (see section 'communication information' for each menu in the main product manual).

Further the ESI-file contains a default PDO setup with 4 byte basic data (see chapter 5.2.2.1) for the most commonly used control/status data exchange. Note that the PDO setup can be expanded with up to 8 additional process values in/out (see chapter 5.2.2.4). This is explained in more detail in a example below as well as in a PLC project example available on www.emotron.com.

3.4.3.2 PDO setup

The ESI-file contains a default setup of 4 byte basic data (see chapter 5.2.2.1).This can optionally be expanded with up to 8 additional process values in/out (denoted FB Signals) if more data is needed in the application (see chapter 5.2.2.4 including example). The FB signals can be configured both from the AC drive panel and over CoE from the PLC system.

The TxPDO and RxPDO mapping are always identical in size and data types. PDO setup is explained in detail in chapter 5.2. Further, PLC example project & documentation 'Process Data Object' is available on www.emotron.com.

Outputs/RxPDOs to AC drive (process data from PLC to AC drive)

Table 46

Type	PDO Index (hex)	Data type	Description	Configuration by AC drive menu	Configuration by CoE index (hex)
BASIC (default)	0x5E00	USINT	See Table 63 control message, byte 0 Enabled if menu [2632] set to Basic	-	-
BASIC (default)	0x5E01	USINT	See Table 63 control message, byte 1 Enabled if menu [2632] set to Basic	-	-
BASIC (default)	0x5E02	USINT	See Table 63 control message, byte 2 Enabled if menu [2632] set to Basic	-	-
BASIC (default)	0x5E03	USINT	See Table 63 control message, byte 3 Enabled if menu [2632] set to Basic	-	-
FB signal 1	0x5F00	UDINT	See chapter“Mapping of additional process values” on page 148 , Enabled if menu [2634] AddPrValues is >=1	[2661] FB Signal 1	0x4AF1
FB signal 2	0x5F01	UDINT	See chapter“Mapping of additional process values” on page 148 Enabled if menu [2634] AddPrValues is >=2	[2662] FB Signal 2	0x4AF2
FB signal 3	0x5F02	UDINT	See chapter“Mapping of additional process values” on page 148 Enabled if menu [2634] AddPrValues is >=3	[2663] FB Signal 3	0x4AF3

Table 46

Type	PDO Index (hex)	Data type	Description	Configuration by AC drive menu	Configuration by CoE index (hex)
FB signal 4	0x5F03	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPr-Values is >=4	[2664] FB Signal 4	0x4AF4
FB signal 5	0x5F04	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPr-Values is >=5	[2665] FB Signal 5	0x4AF5
FB signal 6	0x5F05	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPr-Values is >=6	[2666] FB Signal 6	0x4AF6
FB signal 7	0x5F06	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPr-Values is >=7	[2667] FB Signal 7	0x4AF7
FB signal 8	0x5F07	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPr-Values is >=8	[2668] FB Signal 8	0x4AF8

NOTE: Set unused FB signals to 0.

AC drive Inputs/TxPDOs (process data from AC drive to PLC)

Table 47

Type	PDO Index (hex)	Data type	Description	Configuration by AC drive menu	Configuration by CoE index (hex)
BASIC (default)	0x5C00	USINT	See Table 65 status message, byte 0 Enabled if menu [2632] set to Basic	-	-
BASIC (default)	0x5C01	USINT	See Table 65 status message, byte 1 Enabled if menu [2632] set to Basic	-	-
BASIC (default)	0x5C02	USINT	See Table 65 status message, byte 2 Enabled if menu [2632] set to Basic	-	-
BASIC (default)	0x5C03	USINT	See Table 65 status message, byte 3 Enabled if menu [2632] set to Basic	-	-
FB signal 9	0x5D00	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPrValues is >=1	[2669] FB Signal 9	0x4AF9
FB signal 10	0x5D01	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPrValues is >=2	[266A] FB Signal 10	0x4AFA
FB signal 11	0x5D02	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPrValues is >=3	[266B] FB Signal 11	0x4AFB

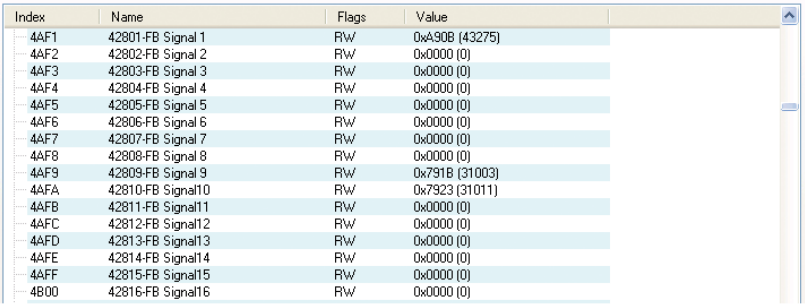
Table 47

Type	PDO Index (hex)	Data type	Description	Configuration by AC drive menu	Configuration by CoE index (hex)
FB signal 12	0x5D03	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPrValues is >=4	[266C] FB Signal 12	0x4AFC
FB signal 13	0x5D04	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPrValues is >=5	[266D] FB Signal 13	0x4AFD
FB signal 14	0x5D05	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPrValues is >=6	[266E] FB Signal 14	0x4AFE
FB signal 15	0x5D06	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPrValues is >=7	[266F] FB Signal 15	0x4AFF
FB signal 16	0x5D07	UDINT	See chapter“Mapping of additional process values” on page 148, Enabled if menu [2634] AddPrValues is >=8	[266G] FB Signal 16	0x4B00

NOTE: Set unused FB signals to 0.

Example CoE online configuration

How to setup CoE Online according to the example in chapter 5.2.2.3:



Index	Name	Flags	Value
4AF1	42801-FB Signal 1	RW	0xA90B (43275)
4AF2	42802-FB Signal 2	RW	0x0000 (0)
4AF3	42803-FB Signal 3	RW	0x0000 (0)
4AF4	42804-FB Signal 4	RW	0x0000 (0)
4AF5	42805-FB Signal 5	RW	0x0000 (0)
4AF6	42806-FB Signal 6	RW	0x0000 (0)
4AF7	42807-FB Signal 7	RW	0x0000 (0)
4AF8	42808-FB Signal 8	RW	0x0000 (0)
4AF9	42809-FB Signal 9	RW	0x791B (31003)
4AFA	42810-FB Signal10	RW	0x7923 (31011)
4AFB	42811-FB Signal11	RW	0x0000 (0)
4AFC	42812-FB Signal12	RW	0x0000 (0)
4AFD	42813-FB Signal13	RW	0x0000 (0)
4AFE	42814-FB Signal14	RW	0x0000 (0)
4AFF	42815-FB Signal15	RW	0x0000 (0)
4B00	42816-FB Signal16	RW	0x0000 (0)

Fig. 21 CoE online configuration example: Screenshot from Beckhoffs TwinCAT system manager.

3.5 Profinet

Profinet is a collection name of an open industrial Ethernet standard of Profibus & Profinet International (PI) for automation. PI is an international umbrella organization which started 1995. Today (at the time of writing) there are more than 25 regional Profibus/Profinet unions and more than 1700 members world wide.

Profinet consists of two different major versions, Profinet CBA (component based automation) and Profinet IO. Profinet IO can in turn be split up into two subversions, Profinet RT (real-time) and IRT (isochronous real time). Each version is explained briefly below.

Profinet CBA is an object oriented approach where the idea is to use standardized 'building blocks' in the PLC system to achieve a fast and unified implementation. Profinet CBA uses 'slow' TCP/IP communication with response times in the range of ~ 100 ms.

Profinet RT is a fast version for real-time date exchange in the range of ~ 10 ms cycle time. The protocol is based on a prioritized Ethernet telegram, using QoS and VLAN to allow the managed switches to prioritize the RT-telegrams ahead of normal TCP/IP traffic. The protocol stack is also optimized in the sense that OSI layers 3 (network) and 4 (transport) are skipped to minimized the delay. Altogether this enables the implementation of a fast industrial Ethernet network.

Profinet IRT is a very fast version (data exchange cycle time ≤ 1 ms) of Profinet IO which is typically used for servo systems / motion control. The protocol is based on a dedicated time slot for real time traffic and special IRT switches.

It is allowed to mix RT- and IRT-nodes in a network. It is however not allowed to daisy chain an IRT-node after an RT-node as IRT traffic can only be passed through Switches with IRT support.

Profinet is part of the IEC standard 61158 and IEC 61784.

3.5.1 Profinet IO interface and LED indication

The Profinet IO option is available in three different versions:

- 1-port version Profinet IO (conformance class A). Generation M30. Article number 01-5954-XX
- 2-port version Profinet IO (conformance class B). Generation M30. Article number 01-5956-XX
- 2-port version Profinet IO including S2 system redundancy (conformance class B and C). Generation M40. Article number 01-8174-50.

The 2-port versions have a built-in switch which allows you to daisy chain your devices in a simple manner. Further, the 2-port version has additional support for SNMP & LLDP (available MIBs LLDP & MIB-II) which the 1-port version does not support. Both the 1-port and the 2-port modules supports Identification & Maintenance with I&M0.

NOTE:
For AC drives:
When using the 2-port version it is recommended to have a standby supply option installed, which allows the built in switch (and LAN) to continue to operate even though the mains to the AC drive is off.

For Emotron TSA Softstarters:
Make sure that the control supply voltage remains on when the mains is switched off, to allow the switch to continue to operate at all times

---

NOTE: All three Profinet IO version uses different GSD-files! These can be downloaded from our website www.emotron.com.

The Ethernet interface operates at 100 Mbit/s, full duplex, as required by Profinet.

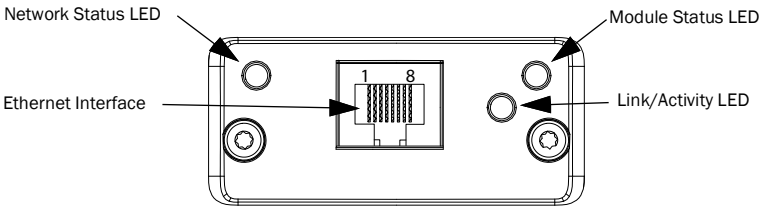


Fig. 22 Profinet IO 1-port version.

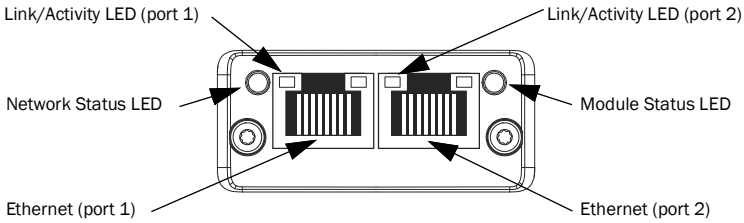


Fig. 23 Profinet IO 2-port version.

3.5.1.1 Network Status LED

NOTE: A test sequence is performed on this LED during startup.

Table 48 Valid for M30 generation

LED State	Description	Comments
Off	Offline	-No power -No connection with IO Controller
Green	Online (RUN)	-Connection with IO Controller established -IO Controller in RUN state
Green, flashing	Online (STOP)	-Connection with IO Controller established -IO Controller in STOP state

Table 49 valid for M40 generation

LED State	Description	Comments
Off	Offline	-No power -No connection with IO Controller
Green	Online (RUN)	-Connection with IO Controller established -IO Controller in RUN state
Green, flashing	Online (STOP)	-Connection with IO Controller established -IO Controller in STOP state or IO data bad -IRT synchronization not finished
Green, blinking	Blink	Used by engineering tools to identify the node on the network
Red	Fatal event	Major internal error (this indication is combined with a red module status LED)
Red, 1 flash	Station Name error	Station Name not set
Red, 2 flashes	IP adress error	IP address not set
Red, 3 flashes	Configuration error	Expected Identification differs from Real Identification

3.5.1.2 Module Status LED

NOTE: A test sequence is performed on this LED during startup.

Table 50 Valid for M30 generation

LED State	Description	Comments
Off	Not initialized	No power - or - Module in “SETUP” or “NW_INIT” state
Green	Normal operation	Module has shifted from the “NW_INIT” state
Green, 1 flash	Diagnostic Event(s)	Diagnostic event(s) present
Green, 2 flashes	Blink	Used by engineering tools to identify the node on the network
Red	Exception Error	Module state in “EXCEPTION”
Red, 1 flash	Configuration Error	Expected Identification differs from Real Identification
Red, 2 flashes	IP Address Error	IP address not set
Red, 3 flashes	Station Name Error	Station Name not set
Red, 4 flashes	Internal Error	Module has encountered a major internal error

The LED signal pattern is made up of a sequence of short flashes, each flash separated by a 250 ms long off phase. The sequence ends with a longer off phase (750 ms).

Table 51 valid for M40 generation

LED State	Description	Comments
Off	Not Initialized	No power OR Module in SETUP or NW_INIT state
Green	Normal Operation	Module has shifted from the NW_INIT state
Green, flash	Diagnostic Event(s)	Diagnostic event(s) present
Red	Exception error	Device in state EXCEPTION
	Fatal event	Major internal error (this indication is combined with a red network status LED)
Alternation Red/ Greed	Firmware update	Do NOT power off the module. Turning the module off during this phase could cause permanent damage

3.5.1.3 Link/Activity LED

The left of the two LEDs that are integrated in each port indicates the Ethernet link status.

Table 52

LED State	Description	Comments
Off	No Link	No link, no communication present
Green	Link	Ethernet link established, no communication present
Green, flickering	Activity	Ethernet link established, communication present

3.5.2 Profinet RT technology description

Profinet IO follows the producer/consumer model for data exchange. This means that a node can send information to multiple nodes by multicasting which saves bandwidth in comparison to sending one dedicated telegram to each and every recipient from the master.

Profinet IO is based on a switched network where either managed switches or special IRT switches (in the case for Profinet IRT) are used. Normal switches used for IT may be used in theory, but it is NOT recommended due to that they are not fit for Industrial environments. Profinet traffic can be mixed with non-realtime IT-traffic (like HTTP), but is limited to max ~ 50% of the bandwidth. This means that Profinet traffic can be mixed with any other type of Ethernet protocol. To get the cyclic RT telegram prioritised by the managed switches, the VLAN with QoS according to IEEE 802.3Q is used in combination with a dedicated EtherType telegram (0x8892). This allows fast channeling through the switches. The switch handles the Profinet frames as cut-through (passing them on directly to the outport) to minimize the delay.

Further, the Profinet protocol stack is optimized for Profinet telegrams where e.g. layer 3 & 4 of the OSI-model is bypassed to achieve minimum stack delay. The received realtime Profinet telegrams are identified by reading the EtherType field in the frame.

The addressing of a Profinet IO device is based on a symbolic station name together with a unique MAC address. The station name can in turn be assigned to a specific IP-address. The Discovery and Control Protocol (DCP) is used for configuration of address settings. Alternatively the IP setting can be performed by DHCP. Manual IP address setting directly on the main product is no longer necessary. Within a Profinet network, the device is addressed by its device MAC address only. Between subnets the Profinet information is sent via UDP/IP (using IP-addressing).

The cyclic telegram (Ethernet telegram) is sent unacknowledged between the producer and consumer. The reason for being unacknowledged is that there is no point in re-sending 'old' information. It is then better to transmit the new, 'latest known' process data.

Acyclic data is requests are handled by the UDP/IP protocol.

Proxy servers or gateways can be used to interconnect to other protocols, e.g. Profibus. The proxy server may for instance work as a Profibus master on one side and as a Profinet slave node on the other. Whereas the gateway work as a slave on both sides.

In Profinet, any topology may be used, but a ring has the advantage of redundancy. The MRP (Media Redundancy Protocol) or variants of it (e.g. MRRT or (R)STP) is used to transform a ring into a line when a transmission path is not working. Note that the all MRP switches handling redundant information paths must be connected directly to each other w/o any interconnected 'non MRP' switch. Only the Profinet IO generation M40 module supports MRP directly (integrated MRP switch). MRP is still possible to achive with Profinet IO generation M30 modules, but means that external switches with MRP support must be used.

Profinet RT can also be transmitted wirelessly using Bluetooth or Wifi technology.

3.5.3 Profinet IRT technology description

In real-time applications it is important to achieve a deterministic behavior. Deterministic communication is a system where the data transmissions are done in a predictable and controlled manner and within strict time limits.

Latency is a predictable delay between a transmitter and a receiver, e.g. due to transmission time and delays in e.g. switches & protocol stacks over the network path. In the case for Profinet IRT, the protocol stack latency is minimized by omitting layer 3–6 of the OSI model and also by using a strict scheduling mechanism (Time Division Multiple Acccess, TDMA) to allocate specific timeslots for the transmission of real-time data (bandwidth reservation). Both of these techniques will help to minimize delay and jitter. Switches with IRT support is used to cut-through IRT traffic with minimum delay. IRT stands for Isochronous Real Time and isochronous means that the real time data is passed periodically at an 'equal' (iso) time (chronous).

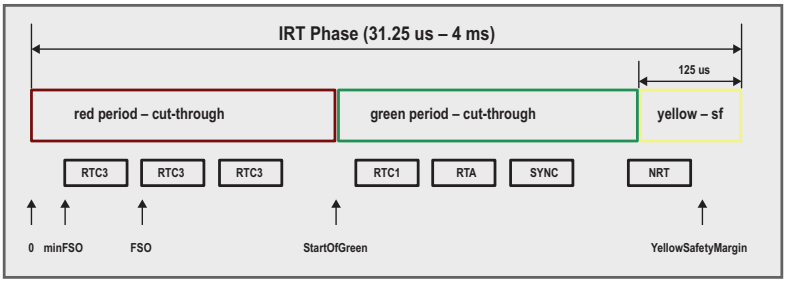


Fig. 24 Profinet IRT communication cycle

In the figure above we can see a typical communication cycle for Profinet IRT. The red period is the deterministic part where prioritized IRT packets are transmitted. The second green period is uses for clock synchronization telegrams (PTCP sync), RT-traffic as well as any other open standard Ethernet telegram / protocol. There is also a safety margin (yellow period) allocated to secure that the next red period with IRT traffic starts its transmission in time. Frames that cannot be forwarded within the remaining time available must be stored temporarily (switch changes to store-and-forward mode) and are instead sent the next green period. The red period is the synchronized bus cycle which is used to transfer the process data with benefits of determinism and minimal jitter. All slave nodes receive and transmit during the red phase. TIA portal secures of that the time is sufficient for all slaves to exchange process data. This allocated time for the red phase is based on the information about the network latency via the PTP protocol, which is described more in detail below. Each slave have a dedicated part / timeslot in the red phase to respond back to the master.

Jitter is the definition of variance in latency from one packet to the next. For example: If a telegram arrives to the recipient 20ms after it was transmitted the first time and 30ms the next time, on can state that this connection has a rather high jitter.

Both lateny and jitter is a problem for high performance motion control applications like e.g. moving robotic arms, paper printing presses and more.

Profinet IRT utilizes the IEEE 1588 Precision Time Protocol (PTP) both to share a common real-time clock across the network, but also to compute and compensate for the delays inherent in network switches and cabling between them. IRT devices are capable of computing these delays down to nanosecond accuracy. When all IRT nodes are synchronized to the same clock source (often the PLC masters clock) they know exactly when the IRT cycle in figure 24

above starts. This allows all IRT data / process data to be exchanged in a deterministic manner with very low jitter of about 10-100 microseconds and at a known regular (isochronous) time interval. During the time which the IRT phase takes place, normal TCP/IP or RT traffic is buffered within a switch. As TCP/IP traffic is not transferred during the IRT phase it won't affect the deterministic behavior. In theory one could for instance watch a movie on a Profinet IRT network without disturbing the real time applications.

The clock synchronization allows the data exchange to happen at the exact same time across the entire network (each Profinet IRT node has a HW synch pulse to indicate the start or IRT phase and all nodes are synchronized with this pulse). The Profinet IRT module allows a cycle time down to 250 microseconds. The cycle time can be set to a specific interval from TIA portal. The PTP protocol is described more in detail below.

The PTP protocol measures the frame residence time within each bridge (def. no end node 2-port switch). The residence time is the time required for receiving, processing queuing and transmission of timing information from the ingress (incoming) to egress (outgoing) port. Further, the PTP protocol also considers link latency of each hop (propagation delay between two adjacent bridges / network nodes).

A simplified description of PTP follows:

a) A “grand master” clock is selected, often the PLC master.

b) The designated master transmits its clock information at a regular time interval up to 10 times per second (called sync messages) to all slaves, marking the exact time the message was sent.

c) Each slave then sends a ‘delay request message’ back to the master clock which timestamps the exact time it receives this message.

d) The grand master receives the message from the slave (round trip time is now known since the master stored the time it sent the message and can compute how much time has elapsed since it received the response back). The master now sends back a ‘delay response message’ to the slave with the timestamp from the ‘delay request message’.

e) The slave can now adjust its time to match the master clock based on the information it has from both the first ‘sync message’ as well as the just received ‘delay response message’. A clock offset is used as compensation. The difference between the local sync clock and the time coming from the sync frame from the PLC (grand master) must stay within 1µs. Otherwise the slave is considered not in sync.

Syncronized movement of e.g. applications using servo drives may also be triggered based on the common clock as reference. All servo drives may for instance trigger a movement a specific delta-time after the red phase has taken place (all nodes are then updated with the new reference at the same time). The additional delta-time delay is to allow the application processor to finish handling the internal communication buffers and interrupts before executing the command.

All together this allows a Profinet IRT network to achieve a very high level of accuracy for real time data with low jitter as the network can compensate even for small propagation delays in Ethernet cables between different network IOs.

The expected jitter within a Profinet IRT networks is down to 1 microsecond.

Within an IRT-domain all switches must be IRT-capable to maintain deterministic communication.

3.5.4 Functionality and settings of the Profinet IO option

This chapter describes the most important features and settings of the Profinet IO option to allow you get a quick start. There are also a Profinet PLC example project for Siemens Simatic S7 and TIA portal available on our website www.emotron.com.

3.5.4.1 Supported features

- LLDP for graphical topology overview and visual diagnostics (supported by 2-port module only)
- SNMP for diagnostics with MIB-II and LLDP-MIB (supported by 2-port module only)
  - MIB-II contains read-only information about com status etc.
- Integrated web server with com statistics and possibility to set/view inverter parameters
- Full duplex, 100 Mbit/s, transformer isolated interface
- Identification & Maintenance, support for I&M0.
- Supporting API 0 (application process instance)
- Address setting via DCP and DHCP
- Available both in 1- and 2-port version (the later with built in switch for simplified daisy chaining)

- The 1-port module supports conformance class A and the 2-port version conformance of generation M30 supports class B. The 2-port Profinet IO + S2 module supports both class B and C.
- For M30 generation modules GSDML-files are available for PLC masters with both DAP version 1.x (older) and 2.x.
- M40 generation supports S2 system redundancy
  - System redundancy relies on the concept of multiple connections to a device or controller to maintain a system in the event of a critical failure.
  - S2 means that two (2) independent PLC's (Application Relations, ARs) can be connected to a single (S) network interface slave node (single Network Access Point, NAP).
  - The PLCs can be connected to the slave node via either one or optionally two Ethernet cables (dual port option). The redundancy in the S2 case consists of PLC recovery. If the primary PLC fails, the secondary PLC immediately takes over the control.

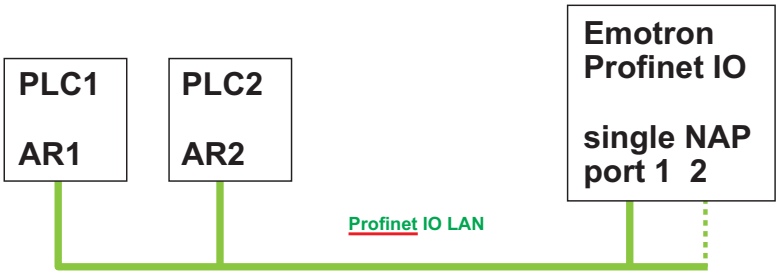


Fig. 25 S2 system redundancy example

- M40 generation also support Media Redundancy Protocol (MRP). MRP allows you to create a very robust LAN with redundant paths (like a ring topology) between the PLC and your device. This allows the traffic to continue flowing even though a network cable have been damaged. Note that all the switches used to form the redundant ring must also support MRP. MRP can detect a ring break and form a line topology within ~ 300ms.

3.5.4.2 Quickstart information Profinet IO

The device name (station name) & IP address of the Profinet IO node is setup from the PLC configuration software. This is described more in detail in the PLC code example available on www.emotron.com (example for AC drive but in general applicable for Emotron TSA as well). The allocated IP address can be viewed (read only) on the AC drive in menu [2651] IP address. The station name is stored in non-volatile memory in the Profinet option.

The setup of the Profinet IO modules are identical to Profibus when it comes to process data configuration from the AC drive side. By default the AC drive is setup to use 'Basic mode', see chapter 5.2.2.1. This can be expanded with up to 8 additional process values in/out, see chapter 5.2.2.4.

The web server interface is described in chapter 3.1.5.

3.6 EtherNet/IP

Ethernet Industrial Protocol (EtherNet/IP) is an open industrial network standard. It has been developed by Rockwell Automation but is now managed by ODVA (Open DeviceNet Vendors Association). It is designed for use in process control and industrial automation applications. EtherNet/IP™ was introduced in 2001 and today is one of the most developed, widely spread , proven and complete industrial Ethernet network solutions.

EtherNet/IP is a member of a family of networks that implements the Common Industrial Protocol (CIP™) at its upper layers (other CIP family protocols are DeviceNet, CompoNet and ControlNet). As a truly media-independent protocol that is supported by hundreds of vendors around the world, CIP provides users with a unified communication architecture throughout the enterprise.

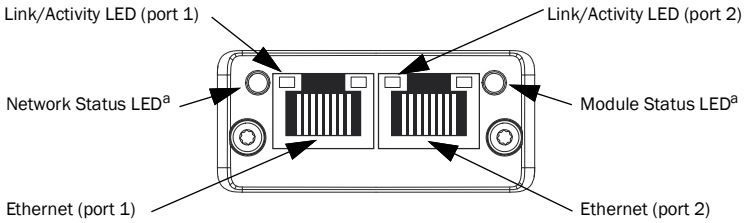
3.6.1 EtherNet/IP interface and LED indications

The EtherNet/IP option is currently available in following variant.

- 2-port version EtherNet/IP (Announce-based redundancy protocol)

The 2-port versions has a built-in switch which allows you to daisy chain your devices in a simple manner.

NOTE: When using the 2-port version it is recommended to have a standby supply option installed, which allows the built in switch (and LAN) to continue to work even though the mains to the product is switched off.



[a]. Test sequences are performed on the Network and Module status LED during startup

Fig. 26 EtherNet/IP 2-port version.

3.6.1.1 Network status LED

NOTE: A test sequence is performed on this LED during startup.

Table 53 Network status LED

LED State	Description
Off	No power - or - no IP address
Green	Online, one or more connections established (CIP Class 1 or 3)
Green, flashing	Online, no connections established
Red	Duplicate IP address, FATAL Error
Red, flashing	One or more connections timed out (CIP Class 1 or 3)

3.6.1.2 Module status LED

NOTE: A test sequence is performed on this LED during startup.

Table 54 Module status LED

LED State	Description
Off	No power
Green	Controlled by a scanner in Run state
Green, flashing	Not configured or Scanner in Idle state
Red	Major fault (ECXCEPTION-state, FATAL Error etc.)
Red, flashing	Recoverable fault(s)

3.6.1.3 Link/Activity LEDs

Table 55 Link/Activity LEDs

LED State	Description
Off	No link, no activity
Green	Link (100 Mbit/s) established
Green, flickering	Activity (100 Mbit/s)

Table 55 Link/Activity LEDs

LED State	Description
Red	Link (10 Mbit/s) established
Red, flickering	Activity (10 Mbit/s)

3.6.2 EtherNet/IP technology description

To achieve good performance for real-time process data (implicit, time-critical data), EtherNet/IP utilizes the lighter UDP protocol along with multicasting and Qos (Quality of Service). UDP telegrams are connectionless (no receive confirmation and no establishment of data link) and makes no guarantee that the data will get from one device to another, however UDP messages are smaller and can be process more quickly than TCP/IP messages. Quality of Service (QoS) can also be used to give priority to more important cyclic I/O data over normal TCP/IP traffic. Standard acyclic data (explicit, non time critical) are exchanged via the connection oriented well known protocols.

Process data transactions take place by use of the Producer-Consumer model scheme. This means that any node can send (produce) data to a number of recipients (consumers) at the same time by utilizing multicast and VLAN technology.

For this reason the used switches must be able to handle multicast telegrams, i.e. only managed switches should be used. Otherwise, in the case with an unmanaged switch, a multicast message would flood to all ports. Further, managed switches have better support for redundancy and error message handling / statistics / port mirroring features and more.

EtherNet/IP also supports redundant network paths e.g. by the announce based or beacon based protocol. This allows a ring topology network to continue to run even though the cable is cut off.
Note! The redundant topology must be formed by a DLR capable switch (ring supervisor).

3.6.3 Functionality and settings of the EtherNet/IP option

This chapter describes the most important features and settings of the EtherNet/IP option to allow you to get a quick start. There are also PLC example projects for Rockwells controlLogix series available on www.emotron.com along with documentation.

3.6.3.1 Supported features

- 2 x Ethernet ports
- Device level Ring (DLR) and Linear topology
- Galvanically isolated bus interface
- Integrated web server
- CIP parameter object support
- 10/100 Mbit with full/half duplex
- DHCP

3.6.3.2 Quick start information EtherNet/IP

Detailed quick start guide for this EtherNet/IP module is available on www.emotron.com.

The setup of process data for EtherNet/IP on the main product is the same as for e.g. DeviceNet. By default the main product is setup to use 'Basic mode', see chapter 5.2.1 page 127. This can be expanded with up to 8 additional process values in/out, see chapter 5.2.2.5 page 150.

The web server interface is described in chapter 3.1.5 page 57.

3.6.3.3 Supported CIP Objects EtherNet/IP

Mandatory objects according to the EtherNet/IP standard:

- Identity Object (01h)
- Message Router (02h)
- Assembly Object (04h)
- Connection Manager (06h)
- Parameter Object (0Fh)

- DLR Object (47h)
- QoS Object (48h)
- Port Object (F4h)
- TCP/IP Interface Object (F5h)
- Ethernet Link Object (F6h)

Vendor Specific Objects:

- ADI Object (A2h) for acyclic access of product parameters.

4. Fieldbus menus

All menus are described in the instruction manual for the AC drive. The following two sections give an overview of parameters used for communication.

4.1 General communication menus

Table 56 General operation menus on Emotron AC drives

Menu number		Parameter settings
210 Operation		
	214 Reference control	Remote, Keyboard, Com, Option
	215 Run/Stp Ctrl	Remote, Keyboard, Com, Option
	216 Reset Ctrl	Remote, Keyboard, Com, Remote+Keyb, Com+Keyb, Rem+Keyb+Com, Option

Table 57 General operation menus on Emotron TSA softstarters

Menu number		Parameter settings
210 Operation setup		
	215 Action control	
	2151 Run/Stp Ctrl	Remote, Int + Ext Keyboard, Com, Option, Int keyboard, Ext keyboard
	2152 Jog Ctrl	Remote, Int + Ext Keyboard, Com, Option, Int keyboard, Ext keyboard
	216 Reset Ctrl	Remote, Keyboard, Com, Remote+Keyb, Com+Keyb, Rem+Keyb+Com, Option

For information about the general parameters, please see the instruction manual for the main product.

Table 58 General communication menus

Menu number			Parameter settings
260 Serial Comm			
	261 Com Type		Fieldbus RS232/485 (only AC drive) RS232 (only Softstarter) USB/RS485/BT (only Softstarter) Select Fieldbus
	263 Fieldbus		
		2631 Address	Node address For EtherCAT: Read-only station alias/node address.
		2632 PrData Mode	None, Basic, Extended
		2633 Read/Write	RW/R R=read/log only RW=read/write
		2634 AddPrValues	0-8
		2635 CANBaudrate	CANopen baudrate. For details see chapter 2.5.5 page 38
	264 Com Fault		
		AC drive 2641 ComFlt Mode	Off, Trip, Warning
		Softstarter 2641 ComFltAA	No action, Hard trip, Soft trip, Warning
		2642 ComFlt Time	0.1-15.0 s Default: 0.5 s
	265 Ethernet		
		2651 IP-address	4-byte internet address Default 0.0.0.0
		2652 MAC address	6 byte (read only)
		2653 Subnet mask	4 byte Default 0.0.0.0
		2654 Gateway	4 byte Default 0.0.0.0

Table 58 General communication menus

	2655 DHCP	On/Off Default On
	2656 FTP Server	On/Off Default Off Note: Available for VSD sw V4.42/ V5.12 and newer. The FTP server is per default disabled for Cyber Security reasons
	2657 WEB Server	On/Off Default Off Note: Available for VSD sw V4.42/ V5.12 and newer. The WEB server is per default disabled for Cyber Security reasons.
266 FB Signal		
	2661 FB Signal1	
	. .	
	266G FB Signal16	

4.2 Status menus

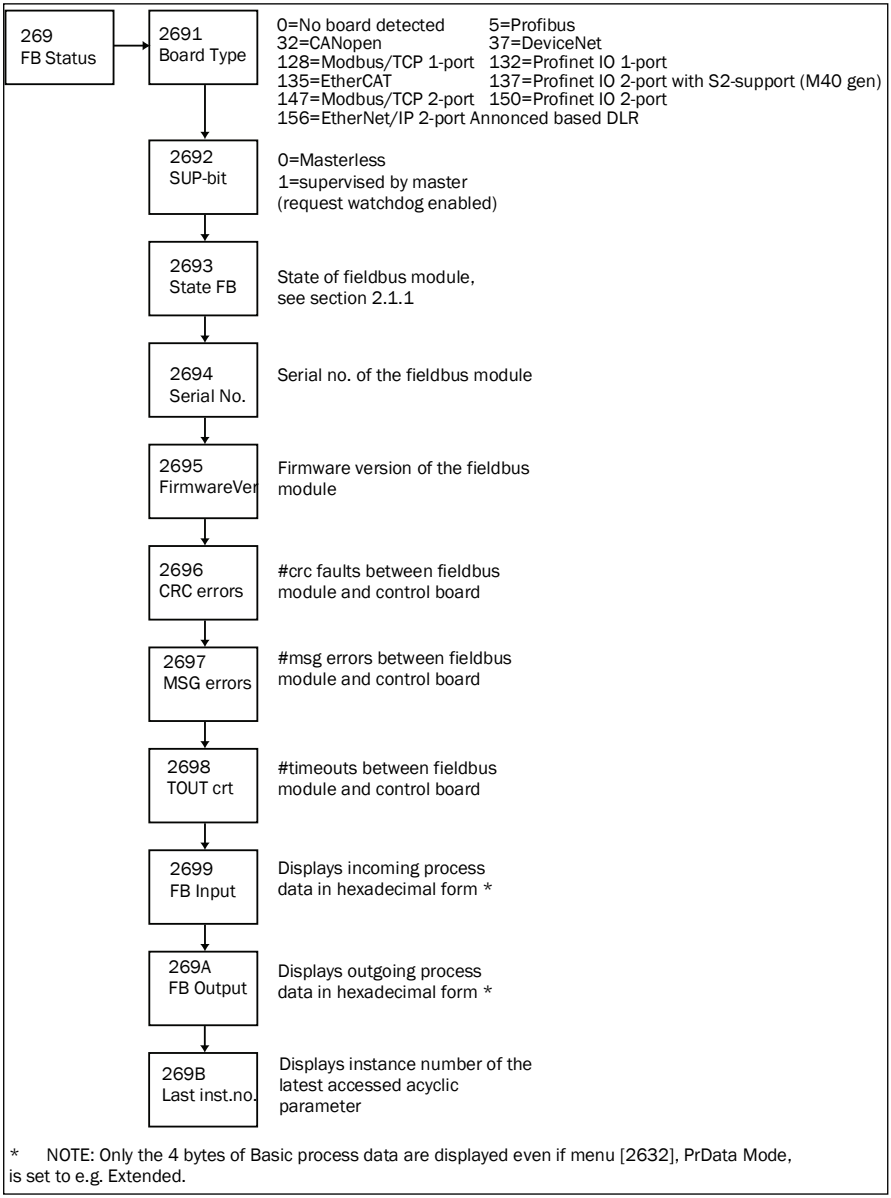


Fig. 27 Status menus

4.3 Quick Setup

The following flowcharts are step by step guides for advanced users.

NOTE: For Industrial Ethernet options additional parameters according to § 3.1.3 must be configured.

Emotron AC drive

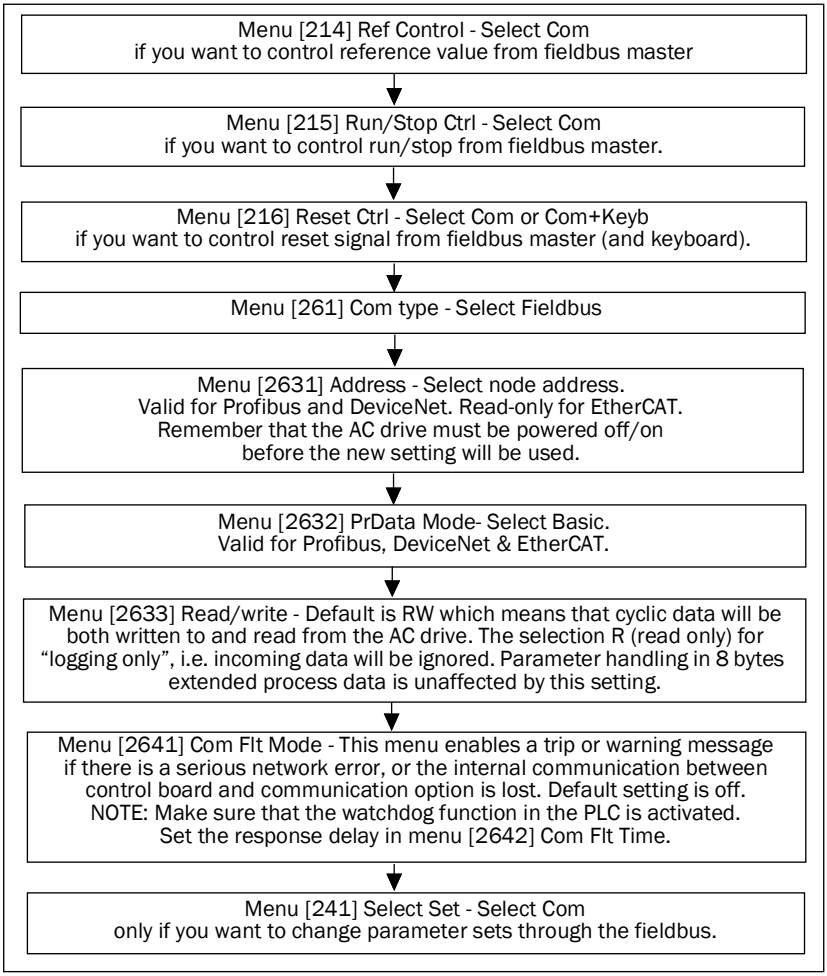


Fig. 28 Step by step guide for AC drive

Emotron TSA Softstarter

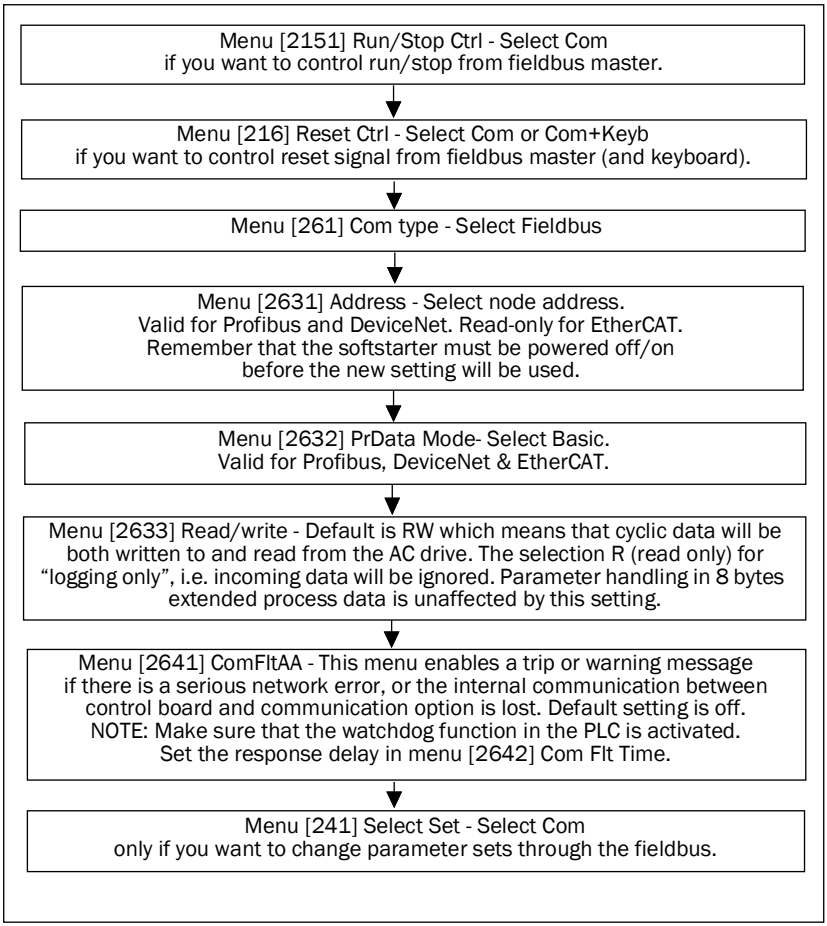


Fig. 29 Step by step guide for Emotron TSA softstarters.

NOTE: The main product must be restarted before the new setting(s) will become active. See chapter 2.1.4.3 for more information.

5. Cyclic/Acyclic data

This chapter is valid for Profibus, DeviceNet, EtherCAT, Profinet IO and EtherNet/IP options.

All of CG Drives & Automation's current fieldbus and IE options have been implemented in such a way that the scaling of all parameters (and cyclic data) are identical regardless of the type of fieldbus selected.

The difference lies within the fieldbus-specific method of retrieving parameter data. This is described in this chapter.

5.1 Byte swapping

Different fieldbus systems may use different data formats called "Little endian" or "Big endian". Big endian means that the most significant byte comes first in a word.

Table 59 Example data format network comparison list

Network	Data format
Profibus	MSB first
EtherNet/IP	LSB first
Profinet IO	MSB first
DeviceNet	LSB first
CANopen	LSB first
Modbus RTU	LSB first
Modbus/TCP	LSB first
EtherCAT	LSB first

5.1.1 Example

The decimal number 23041 is equal to 5A01 hex. It is stored as 5A01 in the big endian format which Profibus uses i.e, 5A on the lowest memory address position and 01 on the highest.

Table 60 Value 5A01 hex represented in big and little endian format.

	Memory address			
	Big endian		Little endian	
Address	0	1	0	1
Data	5A	01	01	5A

In a little endian system it would be stored as 015A.

The CG Drives & Automation fieldbus option takes care of the necessary byte swapping needed to be compatible with each network standard. Note, however, that the PLC master may have a different data format than the fieldbus network, i.e. this may require you to perform byte swapping in the PLC.

5.2 Cyclic data (process data)

5.2.1 PLC configuration of modules/cyclic data

The module mapping in the PLC should be implemented according to settings in menu [2632] PrData Mode (process data mode) and [2634] AddPrValues (additional process values). The module configuration for process data is setup according to the combination of the two settings. This is described in examples further on in this chapter.

You may for instance use Basic mode only, or combine it with up to 8 additional process values.

You could use Extended mode only, or combine it with up to 8 additional process values.

Or you could simply set [2632] PrData Mode to None and use pure process values only, i.e. [2634] AddPrValues set to a value separate from zero.

NOTE: For Emotron FDU/VFX 2.0, additional process values are only available from control board software version 4.11 see menu [922].

Table 61

Menu	[2632] PrData Mode	[2634] AddPrValues
Selection	None/Basic/Extended	0-8
Default value:	Basic	0

In menu [2632], PrData Mode, the following can be selected:

Table 62

None	Use only additional process values [2634].
Basic	Use 4 byte process data control/status information, see § 5.2.2
Extended	Use 4 byte process data (same as Basic setting) + additional proprietary protocol for advanced users, see § 5.2.2 – Mapping of the extended mode.

Basic uses the standard functionality, e.g. like start/stop, changing direction and reference value is controlled. This is the recommended setting (default).

Advanced users who wish to access more than additional 8 extra process values in/out (selectable in menu [2634]) may use the setting Extended.

The setting Extended uses the exact same 4 bytes of process data for control/status as for setting Basic, but also an additional 32-bit module used for a proprietary overlayed protocol similar to the industrial standard Modbus RTU. With this overlayed protocol any process value may be accessed, but only one at a time.

In menu [2634], additional process values, the number of extra 32-bit process values you wish to read and/or write is selected. Additional write process values are easily selectable from control panel menu [2661-2668], denoted FB Signal 1-8 in the control panel menu. In a similar manner, read values are selectable in menu [2669-266G], denoted FB Signal 9-16 in the control panel menu. This is available with all process data modes selected in menu [2632].

The complete PLC module mapping is dependant on the two menus [2632] and [2634].

If menu [2634] is set to 0 (no extra process values used), the module mapping must be done exactly as described in:

– § 5.2.1.1 if menu [2632] is set to Basic
– § 5.2.1.2 if menu [2632] is set to Extended

If menu [2634] is set to a value separate from 0 (extra process values are used) then map additional 32-bit process data values in the same manner as in the example in § 5.2.1.3.

NOTE: It is important to configure the PLC with the same order of in/out information and size type (e.g. word/byte) as the slave node has been set up with. If the node is not properly set up a configuration fault will occur.

NOTE: If you would like to use Profibus acyclic DPv1 read/write functionality it is important that the configuration modules are mapped exactly as below in a consecutive block without any gaps and that the E-address for the first input module and the first output module are equal, see example Fig. 31.

5.2.1.1 Module configuration Basic mode

Selected configuration in menu [2632], PrData Mode = Basic.

Module configuration in PLC as Input 4 x 1 byte, Output 4 x 1 byte.

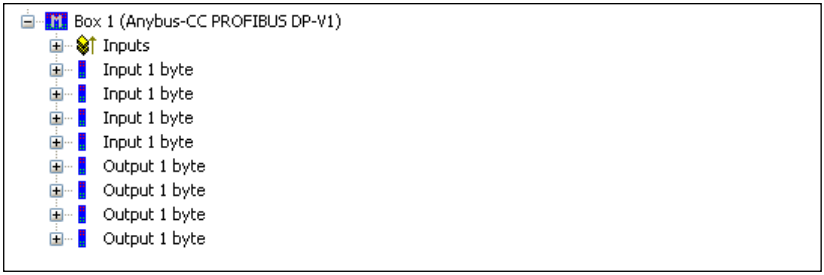


Fig. 30 Module configuration Basic mode

Example: In Basic mode it is important to map the modules in the correct order. The first 4 modules in the PLC should be mapped as “input” and of “byte” type. The last 4 modules should be mapped as “output” and be of “byte” type as well.

5.2.1.2 Module configuration Extended mode

NOTE: Not recommended for new designs. Use additional process value instead, see chapter 2.5.7 page 42.

Selected configuration in menu [2632], PrData Mode = Extended.

Module configuration in PLC as Input 4 x “1 byte” + 1 x “2 words”, Output 4 x “1 byte” + 1 x “2 words”.

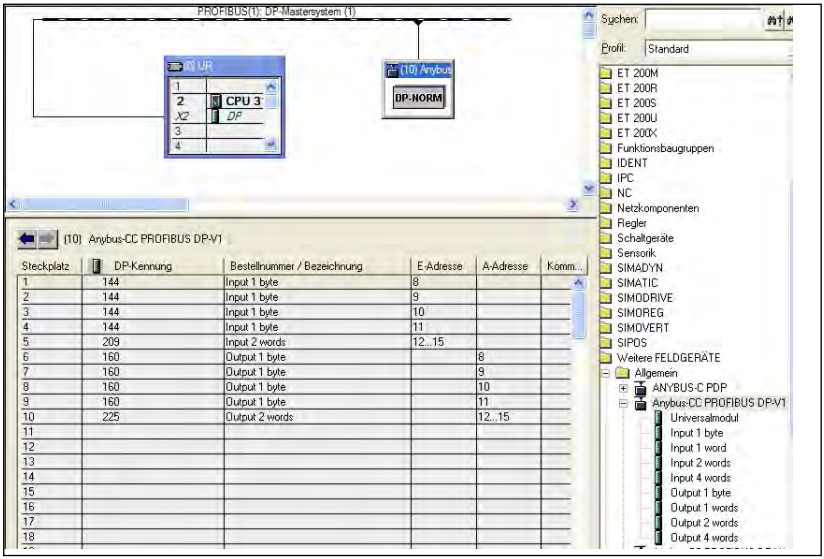


Fig. 31 Module configuration Extended mode (example from Simatic hardware configuration)

5.2.1.3 Additional configuration of extra process values

Example A: Menu [2632] set to Basic and Menu [2634] set to 2. Use module configuration:

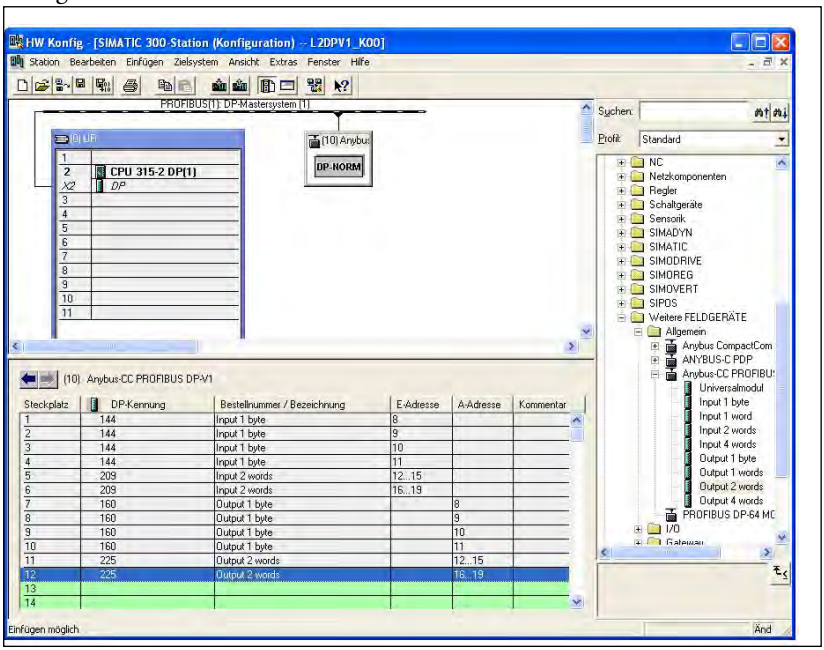


Fig. 32 Module configuration Basic mode with two additional process values

Example B: Menu [2632] set to Basic and menu [2634] set to 4. Use module configuration:

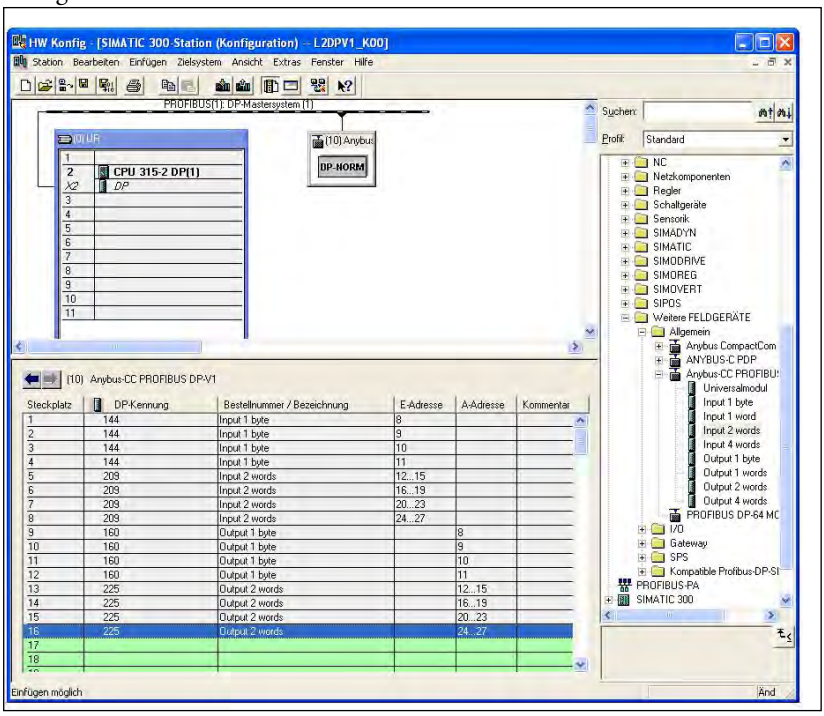


Fig. 33 Module configuration Basic mode with four additional process values

The other possible configurations are module mapped in similar manner. Input/ Output module mapping are always of the same size and type. Note that the order of input/output and data type is important.

5.2.2 Module data mapping

This chapter will first describe the mapping of the 4 byte Basic mode (menu[2632]=Basic) and after that also the mapping of Extended mode (menu[2632]=Extended).

These two modes (Basic, Extended) can also be combined with extra process data values (set in menu [2634]), which is described in the end of this chapter, together with examples.

5.2.2.1 Mapping of Basic mode for AC drives

Control message to AC drive

Table 63

Byte	Bit	Information type	Values	Comment
0	0	Run/stop	0=Stop 1=Run	Only active when [215] Run/Stop = Com
	1-2	Run direc-tion	Bit2/Bit1 00 = stop 01 = run right 10 = run left 11 = Bipolar mode 1)	Only active when [215] Run/Stop = Com
	3	Reset	Reset = 0->1 (edge)	Only active when [216] Reset Ctrl = Com
	4-7	Spare		
1	0-1	Drive mode	Bit1/Bit0 00 = Speed 01 = Torque 10 = V/Hz 11 = Value ignored	The two drive mode bits in the fieldbus control messages do no longer change the drive mode. They are ignored. It is however recommended to use setting 11 (value ignored) to achieve a backward compatible setting with older drive firmware.

Table 63

Byte	Bit	Information type	Values	Comment
	2-3	Parameter set	Bit3/Bit2 00 = Parset A 01 = Parset B 10 = Parset C 11 = Parset D	Only active when [241] Select set = Com
	4-7	Spare		
2	0-7	Reference signal (continued next byte)	10 bit unsigned value (bit 1 byte 3 most significant) = 0-1023d = 0-3FFh = 0-MaxSpeed [343] or 0-MaxTorque [351] 2)	Only active when [214] Ref Control = Com
3	0-1	Reference signal		
	2-7	Spare		

1. With Fieldbus signal set to 42905, the run direction is generated by the sign of this reference value. With control board software version 4.20 and earlier setting ”11” equals stop.
2. When run direction =11 (Bipolar mode) use 42905 as a reference value mapped to a fieldbus signal.

Example

Table 64

Byte 3 (MSB)	Byte 2	Byte 1	Byte 0 (LSB)
0x02	0x80	0x02	0x03

This means that the AC drive is set to run right, no reset active, V/Hz mode, parameter set A used, reference value set to 0x280 (0x280=280Hex).

Status message from AC drive

Table 65

Byte	Bit	Information type	Values	Comment
0	0	Run/Stop status	0 = Stopped 1 = Running	
	1-2	Run dir status	Bit2/Bit1 00 = stopped 01 = run right 10 = run left 11 = not valid	
	3	Relay 1	0 = X2:32-X2:31 closed 1 = X2:32-X2:33 closed	Depending on function selected for Relay1, [551]
	4	Relay 2	0 = X2:42-X2:41 closed 1 = X2:42-X2:43 closed	Depending on function selected for Relay2, [552]
	5	DigOut 1	0 = 0V 1 = active	Depending on function selected for DigOut1, [541]
	6	DigOut 2	0 = 0V 1 = active	Depending on function selected for DigOut2, [542]
	7	Run Ctrl	0=run/stp ctrl via remote/keyboard/option 1=run/stp ctrl via com	
1	0-1	Drive mode	Bit1/Bit0 00 = Speed 01 = Torque 10 = V/Hz (11 = not valid)	The two drive-mode bits in the fieldbus response message are set equal to the two bits received in the control message (mirrored to be backward compatible with older drive firmware). They do not reflect the actually used drive mode. Recommend not using these bits (don't care).

Table 65

Byte	Bit	Information type	Values	Comment
	2-6	Trip/ Warning status	5 bit value (bit 5 most significant) 0 = No trip/warning or ExtTrip2** 1 = Motor I^2t or AnIn<Offset** 2 = PTC or ExtTrip3** 3 = Motor lost or ExtTrip4** 4 = Locked rotor 5 = External trip 6 = Mon MaxAlarm 7 = Mon MinAlarm 8 = Com 9 = PT_100 10 = Deviation 11 = Pump 12 = Ext. MotTemp 13 = LC Level 14 = Brake * 15 = Option 16 = Over temp trip (Mains) 17 = Over Curr Fast 18 = Over Volt Dec 19 = Over Volt Gen 20 = Over Volt (Mains) 21 = Over Speed 22 = Under voltage 23 = Power fault 24 = Desat 25 = DC-link error 26 = Internal Error 27 = Ovolt m cut 28 = Over volt warning 29 = STO Active 30 = Crane Comm 31 = Encoder	See bit 7 byte 1 if trip or warning. * AC drive sw 4.21 and later ** Due to expansion of trip types for later sw revisions the number of trips exceeds the limitation of the 5-bit representation in the process message. For this reason some of the values in the table will have double representation. The complete and uniquely identifiable list is given for menu [722] Warning in the main product manual. By mapping the Modbus parameter 31016 to any additional process value (see chapter 5.2.2.4), each and every trip & warning can be uniquely identified as per the table described for menu [722] in the main product manual.

Table 65

Byte	Bit	Information type	Values	Comment
	7	Trip	0=No trip 1=Trip	Valid from control board software 4.10
2	0-7	AnOut 1	8 bit value (bit 7 most significant) 0-10 V=0-255d 4-20 mA=50-255d	Speed is default. See [530] and submenus.
3	0-7	AnOut 2		Torque is default. See [530] and submenus.

Example

Table 66

Byte 3 (MSB)	Byte 2	Byte 1	Byte 0 (LSB)
0xFF	0x00	0x02	0x2B

This means that the AC drive is set to run right, relay 1 X2:32 - X2:33 is closed, relay 2 X2:42 - X2:41 is closed, DigOut 1 is high, DigOut is low, run control is set to run/stp by remote/keyboard/option, drive mode is V/Hz, no errors or warnings, AnOut 1 is at minimum level and AnOut 2 is at maximum level.

5.2.2.2 Mapping of Basic mode for Emotron TSA Softstarter

Control message to softstarter

Table 67

Byte	Bit	Information type	Values	Comment
0	0	Run/stop	0=Stop 1=Run	Only active when [2151] Run/Stop ctrl= Com
	1-2	Run direction	00 = stopped 01 = Run Forward 10 = Run Reverse 11 = Not valid (causing stop)	Only active when [2151] Run/Stop ctrl= Com **External contactor must be used in case with two rotation directions!**
	3	Reset	0 ->1 = Reset	Only active when [216] Reset Ctrl = Com
	4 - 6	Start method (Operation mode)	000 = Use setting in menu [331] 001 = Linear torque 010 = Square torque 011 = Voltage 100 = DOL (101, 110 and 111 not valid (causes no change))	Selection 000 will allow user to ignore setting if not of interest. Only active when [2151] Run/Stop ctrl= Com Start method menu [331]
	7	Not used		Spare
1	0-1	Jog direction	00 = Jog Stop 01 = Jog Forward 10 = Jog Reverse 11 = not valid (causing stop)	Jog functionality only enabled if run = 0 (i.e. bit 0 byte 0 set to 0) and menu [2152] Jog ctrl set to “Com”
	2-3	Parameter set request	00 = Parset A 01 = Parset B 10 = Parset C 11 = Parset D	Only active when [241] Select set = Com
	4-7	Not used		Spare

Table 67

Byte	Bit	Information type	Values	Comment
2	0-7	Jog speed forward reference	0 = Don't overwrite parameter setting >0 = Overwrite parameter setting Unit [%] of nominal speed, max 30%	Menu [2152] needs to be set to Com. Limited to 30 % Menu [351]
3	0-7	Jog speed reverse reference	0 = Don't overwrite parameter setting >0 = Overwrite parameter setting Unit [%] of nominal speed, max 30%	Set menu [2152] to Com Limited to 30 % . Menu [352]

Example

Table 68

Byte 3 (MSB)	Byte 2	Byte 1	Byte 0 (LSB)
0x00	0x00	0x00	0x03

This means that the softstarter is set to run forward, no reset active. Start method set in menu [331] via control panel is used. Jog not active. Parameterset A is selected. No jog speed forward and reverse set since jog not active (Otherwise writing value 0x00 also means that value set by control panel will be used, i.e. setting of menu [351] and [352]).

Status message from softstarter

Table 69

Byte	Bit	Information type	Values	Comment
0	0	Run, Jog/ Stop	0 = Stopped/Coasting 1 = Running or Jogging	1 = Acc, Dec, Run, Brake, Jog
	1-2	Run/Jog dir status	00 = stop 01 = run Forward / Jog Forward 10 = run Reverse / Jog Reverse 11 = not valid	01 = Acc, Run, Jog forward 10 = Acc, Run, Jog reverse
	3	Relay 1	0 = not activated 1 = activated	Depending on function selected for Relay1, [551]
	4	Relay 2	0 = not activated 1 = activated	Depending on function selected for Relay2, [552]
	5	Relay 3	0 = not activated 1 = activated	Depending on function selected for Relay3, [553]
	6	Bypass status	0 = Bypass inactive 1 = Bypass active	1 = Ramp ready and bypass contactor closed
	7	Com Run Ctrl	0=run/stp via remote/keyboard/option 1=run/stp via com	
1	0-1	Active parameter set	00 = Parset A 01 = Parset B 10 = Parset C 11 = Parset D	

Table 69

Byte	Bit	Information type	Values	Comment
	2 - 6	Trip/Warning status	6 bit value (bit 7 most significant) 0 = No trip/warning 1 = Motor I2t 2 = PTC (Motor PTC) 3 = Locked rotor 4 = External trip 5 = External trip 2 6 = Current limit 7 = Start limit 8 = Communication error 9 = PT100 (Motor PT100) 10 = Bypass open 11 = Bypass closed 12 = Monitor MaxAlarm 13 = Monitor MaxPreA 14 = Monitor MinPreA 15 = Monitor MinAlarm 16 = Over temp (TSA over temp) 17 = Thyristor open 18 = Thyristor short 19 = Phase loss (single phase) 20 = M Phase loss (multiple) 21 = Under voltage 22 = Motor termOpen 23 = Current unbalance 24 = CtrlSupFault (contr voltage) 25 = Spare25 (for future use) 26 = Internal Error 1 27 = Phase reversal 28 = Spare28 (for future use) 29 = Internal Error 2 30 = Over voltage 31 = Voltage unbalance	See bit 7 byte 1 if trip or warning.
	7	Trip	0 = No Trip / 1 = Trip	

Table 69

Byte	Bit	Information type	Values	Comment
2	0 - 7	AnOut 1	8 bit value (bit 7 most significant) 0-10 V=0-255d 4-20 mA=50-255d	Current is default. See [530] and sub-menus.
3	0 - 7	Motor current [%]	Motor current in % of rated motor current (menu [224]), 0 - 255%	0 - 255 % of rated motor current. If peak values during start are of interest - use additional process value & map current.

Example

Assume that we have set analogue out value to be 0-10V in menu [532], corresponding to 0-255d over fieldbus and that Analogue out function in menu [531] is set to Used thermal capacity indicating if it is possible to do more repeated starts or not.

Table 70

Byte 3 (MSB)	Byte 2	Byte 1	Byte 0 (LSB)
0x7F	0x7F	0x00	0xDB

This means that the Emotron TSA is in operation, running forward. Relay 1 is active whereas Relay2 and 3 is not active. Bypass is enabled which means that full speed is reached. Com Run Ctrl is high which means that run/stop commands are controlled via fieldbus. Active parameterset is A. Since byte 1 is 0x00, this means that there is no trip or warning present (in the case of trip bit 7 byte 1 would have been set, indicating trip present and the cause would then be decoded by bit 2-6 in byte 1). Byte 2 is 0x7F which is equal to 127dec. 127/255=50% which means that the Used thermal capacity is 50%. The motor current in byte 3 is 0x7F =127 dec [%]. This means that we are running at a motor current which is 127% of rated motor current of menu [224]. If the actual current is greater than 255% the value will be clamped to 255 dec.

5.2.2.3 Mapping of the extended mode, only valid for AC drives

NOTE: Not recommended for new designs. Use additional process value instead, see chapter 2.5.7 page 42.

This mode is recommended for advanced users only. For most applications it is recommended to use additional process values instead, see chapter 5.2.2.4. The first four bytes in/out of the extended mode will be exactly the same as for the Basic mode. With the extended mode you can read/write to any parameter in the main product with a protocol similar to Modbus-RTU. All Modbus parameter addresses & scaling information can be found in the main product manual.

How to read a parameter

If one wants to read a parameter, the first word in the message to the main product should be set to 0xFFFF and the second word should be set to the Modbus address of what parameter to read.

In the return message from the main product, the first word will always return the Modbus address of the accessed parameter. Since the Profibus network is much faster than the internal serial link it is required to make sure that they stay synchronized, e.g. when the PLC master logs several parameters in a round-robin fashion. With this solution the PLC master will always know which parameter the data in the response message belongs to.

The second word will always return the Data of the accessed parameter. For parameters which are of type long, the eint format should be used (see description in the AC drive manual). Only exception is error messages, see error handling according to Table 75 and Table 76.

Example: How to read parameter 43044, motor current=54.5 A ($545_D=221_H$):

Message from PLC to AC drive:

Table 71

0xA824 (=modbus addr)		0xFFFF (=read param)		4 byte Basic process data			
byte 7 (MSB) 0xA8	byte 6 (LSB) 0x24	byte 5 (MSB) 0xFF	byte 4 (LSB) 0xFF	byte 3	byte 2	byte 1	byte 0

Message from AC drive to PLC:

Table 72

Data (=motor current)		0xA824 (=modbus addr)		4 byte Basic process data			
byte 7 (MSB) 0x02	byte 6 (LSB) 0x21	byte 5 (MSB) 0xA8	byte 4 (LSB) 0x24	byte 3	byte 2	byte 1	byte 0

NOTE: Since the Profibus network can be much faster compared to the internal serial link between the control board and the fieldbus module, the first response process data messages returned after a new request may be a response to a previously requested modbus parameter. This is why the modbus address is always returned in the response message (data and address will always be synchronized). The PLC programmer needs to make sure that he gets a proper response back before continuing to read (or write) a new modbus parameter.

How to write a parameter

If one wants to write a parameter, the first word should be set to the Modbus address of the parameter and the second word should contain the data to write.

In the return message from the AC drive, the first word will always return the Modbus address of the accessed parameter. Since the Profibus network is much faster than the internal serial link it is required to make sure that they stay synchronized, e.g. when the PLC master logs several parameters in a round-robin fashion. With this solution the PLC master will always know which parameter the data in the response message belongs to.

The second word will always return the Data of the accessed parameter. For parameters which are of type long, the eint format should be used (see description in the AC drive manual). Only exception is error messages, see error handling according to Table 75 and Table 76.

Example: How to set the function of parameter 43274, relay 2, to on (see available functions in the instruction manual for VFX 2.0 or FDU 2.0):

Message from PLC to AC drive:

Table 73

0x0001 (=set relay to on)		0xA90A (=modbus addr)		4 byte Basic process data			
byte 7 (MSB) 0x00	byte 6 (LSB) 0x01	byte 5 (MSB) 0xA9	byte 4 (LSB) 0x0A	byte 3	byte 2	byte 1	byte 0

Message from AC drive to PLC:

Table 74

0x0001 (=relay set to on)		0xA90A (=modbus addr)		4 byte Basic process data			
byte 7 (MSB) 0x00	byte 6 (LSB) 0x01	byte 5 (MSB) 0xA9	byte 4 (LSB) 0x0A	byte 3	byte 2	byte 1	byte 0

See note above regarding synchronized data/modbus address!

How to handle response errors when reading or writing to a certain parameter:

The protocol sends exception codes back when it is not possible to read or write to a specific parameter. The exception codes are:

Table 75

Exc. code	Name	Description
0xFF01	Illegal function	This unit doesn't support the function code. It could also indicate that the server (or slave) is in the wrong state to process a request of this type. Will also be used for parameters that are not allowed to be changed when the AC drive is running.
0xFF02	Illegal data address	The data address is not within its boundaries.
0xFF03	Illegal data value	The data value is not within its boundaries (or when trying to write to read-only parameter)

To distinguish a successful response from a faulty response (after a read or write request) you may check if the most significant byte of the first word is equal to 0xFF.

E.g. If the first word contains a 0xFFXX value, the second word will always contain the modbus address of the failed request message. The PLC programmer should handle potential error messages and decide what action to take.

Error Response process data (from AC drive to PLC):

Table 76

0xA824 (=modbus addr with exception)		0xFFxx (exception code) xx=01 "not at run" xx=02 illegal address xx=03 illegal data (oor / R only)		4 byte Basic process data			
byte 7 (MSB) 0xA8	byte 6 (LSB) 0x24	byte 5 (MSB) 0xFF	byte 4 (LSB) 0xXX	byte 3	byte 2	byte 1	byte 0

5.2.2.4 Mapping of additional process values

Select the maximum number of read (or write) 32-bit process values to be used in menu [2634], use whichever is highest out of number of read // number of write process values. Both the request and the response process data messages will then reserve space for this additional information.

Configure which process values (FB signals) to write in menu [2661-2668].

Configure which process values (FB signals) to read in menu [2669-266G].

This is further described in an example below.

E.g. if you wish to use 3 extra process values for reading, but only 1 extra process value for writing, select [2634] to 3. Select the process value you wish to write in menu [2661] and set menu [2662-2668] to 0 (not used). Select the three process values you wish to read in menu [2669-266B] and set menu [266C-266G] to 0 (not used).

NOTE: The parameter number (FB signal number) for each process value can be found in the main product manual or "Communication settings" list on www.emotron.com/www.cgglobal.com, as the listed Modbus instance number. Unused process values must be set to 0. All process values are represented with a consistent, standard signed 32-bit long format.

Example for AC drive

TSA Softstarter uses same principle of configuration as the AC drive but may vary in parameter numbering.
Besides controlling start/stop etc. with 4 bytes Basic process data, we also want to read process values Torque & Heatsink temperature and write the process value Relay 3.

Select Basic in menu [2632].

Select 2 in menu [2634] (the maximum number when comparing number of read values (2) and number of write values (1)).

Select menu [2661] to e.g. 43275 for Relay3 function, this is the write process value

Select menu [2662-2668] to 0 (not used, since only one write process value is used)

Select menu [2669] to e.g. 31003 for Torque, this is the first read process value

Select menu [266A] to e.g. 31011 for Heatsink temperature, this is the second process value

Select menu [266B-266G] to 0 (not used, since only two read process values is used)

Message from PLC to AC drive:

Table 77

2'nd write process data value				1'st write process data value				4 byte Basic process data			
Byte 11 (LSB)	Byte 10	Byte 9	Byte 8 (MSB)	Byte 7 (LSB)	Byte 6	Byte 5	Byte 4 (MSB)	byte 3	byte 2	byte 1	byte 0

The process data value for Relay3 is contained in byte 4-7. The second write process data is currently not used, menu [2662] is set to 0 for this reason. However, If you would like to use the second write process value later on, simply change menu [2662] from 0 to the corresponding Modbus instance number.

For Profibus which uses the big endian format (MSB first), byte 4 represents MSB and byte 7 LSB for the first process value.

Message from AC drive to PLC:

Table 78

2'nd read process data value				1'st read process data value				4 byte Basic process data			
Byte 11 (LSB)	Byte 10	Byte 9	Byte 8 (MSB)	Byte 7 (LSB)	Byte 6	Byte 5	Byte 4 (MSB)	byte 3	byte 2	byte 1	byte 0

The process data value for Torque is contained in byte 4-7. The second read process data value Heatsink temperature is contained in byte 8-11.

For Profibus which uses the big endian format (MSB first), byte 4 represents MSB and byte 7 LSB. For the second process data value, byte 8 represents MSB and byte 11 LSB.

5.2.2.5 Other comments

The setting in menu [2633] Read/Write (for logging purposes) will only affect the process data corresponding to 4 byte Basic mode.

Only the 4 bytes corresponding to contents of Basic mode will be displayed in menu [2699]-[269A] to avoid too much information in the control panel window.

5.2.3 Replace an old fieldbus option

If you want to replace an AC drive from Emotron containing the old fieldbus option ABIO-64 with a new one you are able to do so quite easily.

Do as follows:

1. Change the configuration file for the slave node.
2. Compare the process data content of ABIO-64 to the content in § 5.2.2, page 133. Make the necessary changes in the PLC program by comparing the process data contents.

5.3 Acyclic data (parameters)

Profibus parameters can be set or read through the read/write services in the DPv1 protocol. Each parameter can be accessed through a unique slot/index. All available parameters are listed in the manual for the AC drive or in the case with Emotron TSA in a “Communication settings” list on www.emotron.com/ www.cgglobal.com. Note that there might be interdependencies between two or more parameters. For example, a change to one parameter may affect the value and the max/min limits of another parameter. For further information, refer to the AC drive manual.

DeviceNet parameter handling is described in § 2.4.6, page 31.

All parameters with DeviceNet / Ethernet/IP instance number 4XXXX are readable/writable. All parameters with DeviceNet / Ethernet/IP instance number 3XXXX are read only.

On www.emotron.com there are PLC code examples for e.g. Profibus (Simatic S7), EtherCAT (TwinCAT), ProfiNet and also Ethernet/IP (Logix designer).

5.4 EmoSoftCom – a help tool

EmoSoftCom is a PC program which can be used to upload parameters from the AC drive or Emoton TSA. The application parameters can then be stored in a file system as a backup. Simply install the program on your laptop or PC and connect the serial port to the product.

You will recognize all parameter settings in an expandable menu tree.

For AC drive; the RS232-cable can either be connected to the product using a serial communication option (isolated version available) or alternatively through a non-isolated 9-pole D-sub behind PPU (applies to VFX 2.0 and FDU 2.0).

For Emotron TSA softstarter; either one can connect to the non-isolated on-board RS232 port or alternatively install a USB communication option (isolated) on the control board for faster communication.

EmoSoftCom also allows you to log and display information from the AC drive or Emoton TSA. It may be used as a simple oscilloscope to examine signals. The parameter settings may be stored to your computer’s hard drive, which may be re-used when replacing a unit or programming several units with the same setup.

6. Parameter sets and Trip log lists

Table 79 Trip log list for AC drive

Trip log list	DeviceNet and Ethernet/IP Instance number	Profibus Slot/Index	EtherCAT index (hex)	Profinet IO index
1	31101 to 31150	121/245 to 122/39	244d - 247e	1101-1150
2	31151 to 31200	122/40 to 122/89	247f - 24b0	1151-1200
3	31201 to 31250	122/90 to 122/139	24b1 - 24e2	1201-1250
4	31251 to 31300	122/140 to 122/189	24e3 - 2514	1251-1300
5	31301 to 31350	122/190 to 122/239	2515 - 2546	1301-1350
6	31351 to 31400	122/240 to 123/34	2547 - 2578	1351-1400
7	31401 to 31450	123/35 to 123/84	2579 - 25aa	1401-1450
8	31451 to 31500	123/85 to 123/134	25ab - 25dc	1451-1500
9	31501 to 31550	123/135 to 123/184	25dd - 260e	1501-1550

Table 80 Trip log list for Emotron TSA softstarter

Trip log list	DeviceNet and Ethernet/IP Instance number	Profibus Slot/Index	EtherCAT index (hex)	Profinet IO index
1	31101 to 31154	121/245 to 122/43	244d - 2482	1101 - 1154
2	31201 to 31254	122/90 to 122/143	24b1 - 24e6	1201 - 1254
3	31301 to 31354	122/190 to 122/243	2515 - 254a	1301 - 1354
4	31401 to 31454	123/35 to 123/88	2579 - 25ae	1401 - 1454
5	31501 to 31554	123/135 to 123/188	25dd - 2612	1501 - 1554
6	31601 to 31654	123/235 to 124/33	2641 - 2676	1601 - 1654
7	31701 to 31754	124/80 to 124/133	26a5 - 26da	1701 - 1754
8	31801 to 31854	124/180 to 124/233	2709 - 273e	1801 - 1854
9	31901 to 31954	125/25 to 125/78	276d - 27a2	1901 - 1954

Table 81 Parameter set list, valid for both AC drive and Emotron TSA

Parameter sets	DeviceNet and Ethernet/IP Instance number	Profibus Slot/Index	EtherCAT index (hex)	Profinet IO index
A	43001 to 43899	168/160 to 172/38	4bb9 - 4f3b	19385-20283
B	44001 to 44899	172/140 to 176/18	4fa1 - 5323	20385-21283
C	45001 to 45899	176/120 to 179/253	5389 - 5706	21385-22283
D	46001 to 46899	180/100 to 183/233	5771 - 5af3	22385-23283

7. Frequently asked questions

This chapter contains frequently asked questions (FAQ's).

Table 82

Question	Answer
Can I swap between e.g. keyboard control and control via com?	Yes, use e.g. parameter set A for keyboard control & parameter set B for COM control. For AC drive. Set e.g. DigOut1 to OFF in parameter set A and DigOut1 to ON in parameter set B. In this manner it is possible to see the actual parameter set from the fieldbus side by checking bit 5, byte 0 (DigOut1) in the response process data message. For Emotron TSA softstarter: The active parameter set is returned in the response message.
My Step7 PLC using Profibus goes into STOP mode for some reason. What could be the problem?	It could be caused by e.g. diagnostic events. When an event occurs it calls the corresponding OB in your PLC program. If the OB does not exist a fault is triggered and the PLC will STOP. Make sure that OB82 and OB83 exist even though they are empty (containing no code). Another alternative would be to change the DP Interrupt mode. This is further described in your Step7 documentation.
What is the response time for the cyclic process data when only considering the internal communication between the AC drive and the fieldbus module?	Typical 10 ms, Worst case 15 ms.
I am new to using Profibus//Simatic PLC. Does CG Drives & Automation supply any example function blocks available to speed up the implementation?	Yes, Simatic S7 PLC function blocks for the AC drive can be downloaded from www.emotron.com.

Table 82

Question	Answer
I am using a Profibus module. Which *.GSD file should I use, the HMS_1811.gsd or the HMSB1811?	For all new Profibus delivered year 2008 and on, the HMSB1811.gsd file should be used. If you are uncertain which firmware version you have - check menu [2695]. Note if it is of type 2.xx or 1.xx. For all 2.xx modules use the HMSB1811.gsd and for all 1.xx modules use HMS_1811.gsd.
I am using a DeviceNet option and I get a revision mismatch with the EDS-file I am using. What should I do?	First find out the firmware version of the fieldbus module installed. You will see the version in menu [2695]. Note if it is of type 2.xx or 1.xx. Open your *.EDS file with a text editor e.g. Notepad. Update the revision information (to the revision used in the module) in the following three rows located in the beginning of the file: Example: Revision = 2.1; MajRev = 2; MinRev = 1;
I do not see any Parameter Data on the web server. Why?	This can happen if you power up the Ethernet module after power up of the main product. Solution is to power cycle the AC drive together with the module connected to the control board. Also check setting of menu [2657] Web server setting.

Table 82

Question	Answer
How do I interpret the process data of control panel menu 2699-269A?	This is an example for AC drives, but the same principle of interpretation is used for Emotron TSA softstarters. However, the TSA Softstarter uses process data mapping according to Table 67 and Table 69, page 140. The values are presented in hexadecimal form & should be interpreted according to § 5.2.2.1. Example: Menu [2699] displays 1100003. This means: Byte 1 = 0x03 = Run and Run to the right Byte 2 = 0x00 = Drive mode = speed, and parameter set A, Byte 3 = 0x10 = Low byte of reference value (total 10 bit) Byte 4 = 0x01 = Two most significant bits of reference value (total 10 bit) Menu [269A] displays 3232008B. This means: Byte 1 = 0x8B = 1000 1011b = Running, run left, Relay1 on, Run Ctrl=1=run/stp by com. Byte 2 = 0x00 = Drive mode speed, no warning/error status. Byte 3 = 0x32 AnOut1 value Byte 4 = 0x32 AnOut2 value
I do not recognize menu [2632] PrData Mode	The name has been changed in control board revision 4.11. It is completely backward compatible to old setting in menu “SizeOfData”. PrDataMode=Basic, equal to SizeOfData=4 PrDataMode=Extended, equal to SizeOfData=8
I do not recognize menu [264] Com Fault	For AC drive the name has been changed in control board revision 4.11. The submenu [2641] Com Flt Mode is backward compatible to old setting i menu “Interrupt”. In menu [2642] Com Flt Time the response time is entered.
How can I easily/ quickly view the address of the main product unit in the PPU display?	You can add e.g menu [2631] Address or menu [2651] IP Address to the toggle list (see instruction in the main product manual). By a simple button push on the PPU you can then jump automatically to this menu.

8. Installation, AC drives

Installation in Emotron FDU/VFX 2.1 AC drives and Emotron AFR/AFG/DCU units

8.1 Installation in type IP54, IP20/21 and IP23

This chapter describes how to mount option boards in the AC drive.

8.1.1 Polarization of flat cables

The flat cable is marked with a colour on one side and has a pin on the micromatch male contact. This side must be matched to the female micromatch contact on the control board and option board respectively, where a small hole in the board is located.

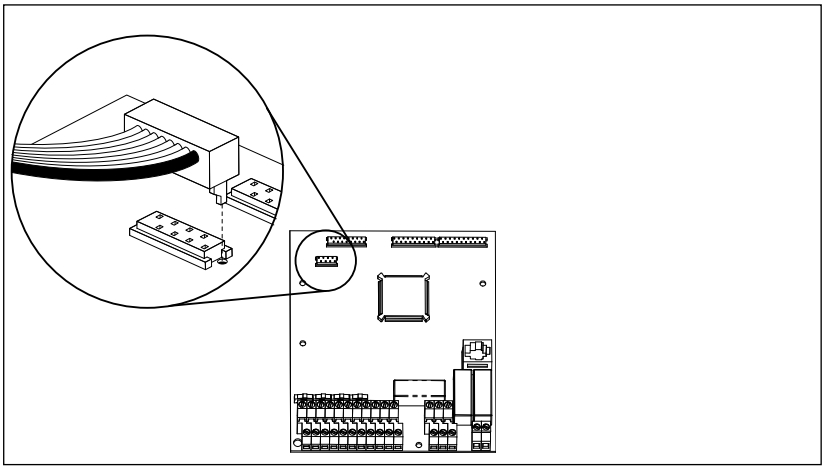


Fig. 34 Polarization of flat cables.

CAUTION!
Incorrect connection might cause damage to both the option and to the control board/external equipment.

8.1.2 Mechanical mounting

Make sure that the AC drive has been switched off for at least seven minutes to ensure that the capacitor bank is discharged before continuing with installation! Also make sure that no external equipment connected to the drive's interface is powered on.

NOTE: Correct installation is essential for fulfilling the EMC requirements and for proper operation of the module.

8.1.2.1 Mounting the option board

The fieldbus option board is always mounted on the slot marked C on the mounting plate. In this example we assume that no other option board is installed.

Delivered with the option board kit

- Option board and four screws, M3 x 6.
- 8-pole flat cable for connection between the fieldbus board and the control board.

Mounting

1. Connect the 8-pole flat cable to the X4 connector on the control board with the cable downwards as in Fig. 35.

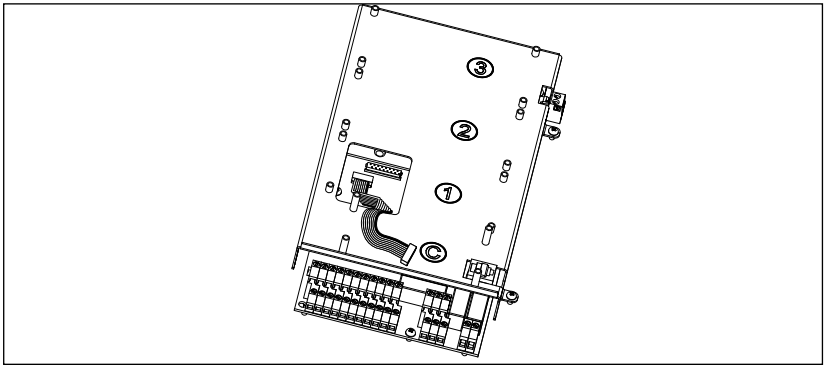


Fig. 35 Flat cable connected to the control board.

NOTE: For polarization of the flat cable, see section 8.1.1 on page 160.

2. Put the option board on the spacers on the slot marked C. Fasten the board using the four screws.

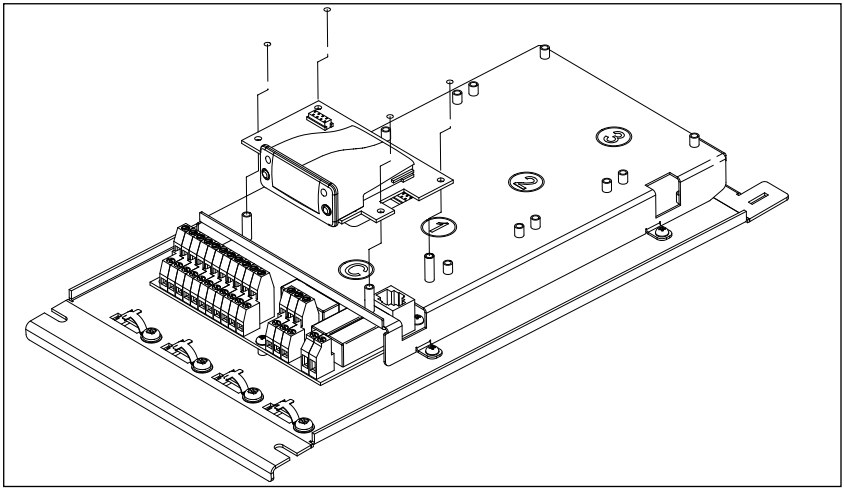


Fig. 36 Mounting the Fieldbus option board

3. Connect the other end of the 8-pole flat cable to the X2 connector on the option board. Make sure that the polarization is correct as in section 8.1.1 on page 160.

NOTE: Connect the micro match male contact to the option in the same manner as on the control board, i.e. the pin on the micro match contact must be fitted into the hole in the PCB.

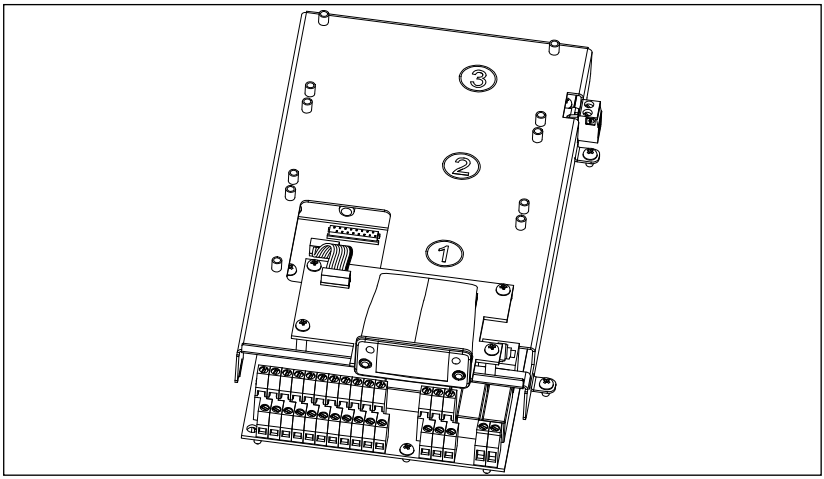


Fig. 37 Flat cable connected to the option board

9. Installation, Emotron TSA softstarters

This chapter describes how to mount fieldbus module directly to the control-board.

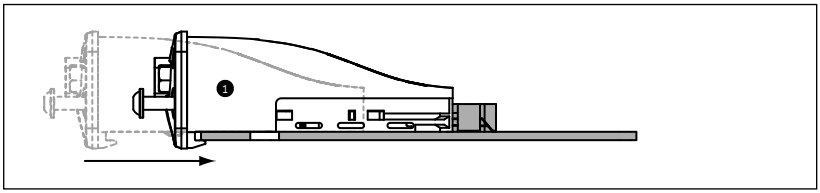
On Emotron TSA softstarters up to two option boards and one fieldbus/communication module can be mounted.

9.1 Mechanical mounting

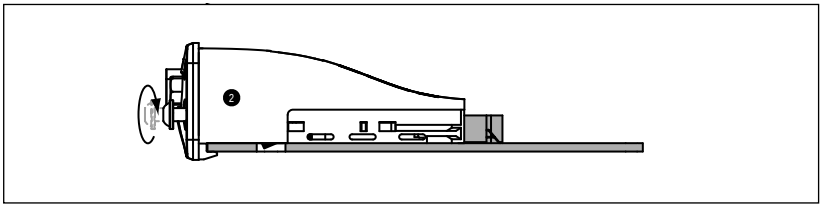
Make sure that all power supplies (i.e. both Main and control supply) is switched off before continuing the installation! Also make sure that no external equipment connected to the softstarters interface is switched on

NOTE: Correct installation is essential for fulfilling the EMC requirements and for proper operation of the module.

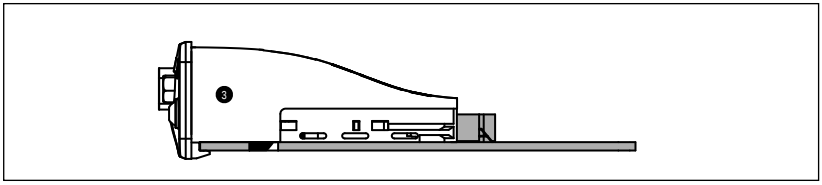
9.1.1 Mounting a fieldbus module



1. Gently slide the fieldbus module into place. Make sure that the module slides on to the guides on the board and into the 50-pin Compact Flash contact. Do NOT use any force!



2. Make sure that the fieldbus module goes all the way in. Then tighten the two screws on the front of the module. The module is now locked to the board through fastening holes in the PCB .



10. Shielding

Emotron AC drive and Emotron TSA softstarter

This chapter is valid for both AC drives and Emotron TSA softstarter.
The fieldbus connector case is grounded via fieldbus modules interface and does not require any additional screen connection. I.e. the cable should go directly to RJ45 contact without any additional screen connection.

On DeviceNet however it is important to connect the network cable shield firmly (low ohmic connection) to the mounting plate (ground) as in the picture below, this screen connection example is valid for non Ethernet cables.

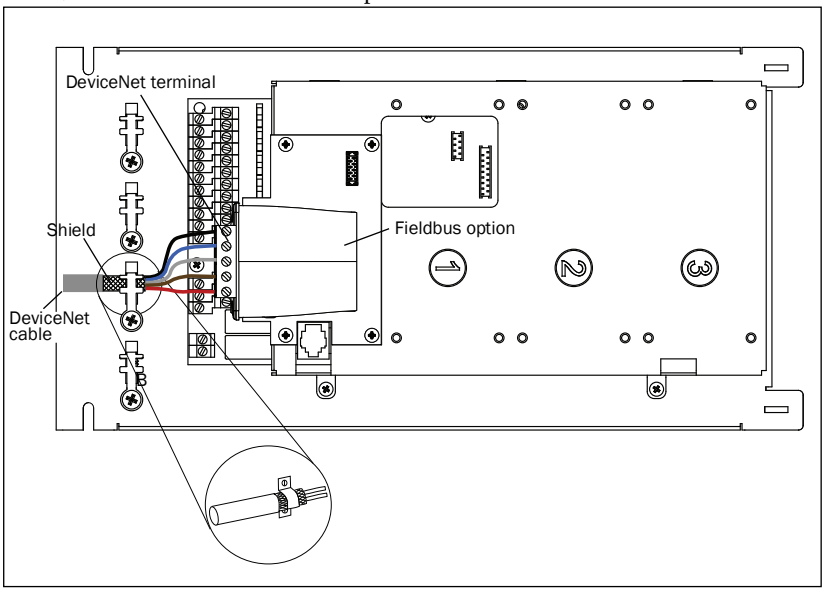


Fig. 38 Example: Emotron AC drive installed with DeviceNet.

The shield must end at the clamp. Only the wires should continue into the DeviceNet terminals.

In most cases it is recommended that the shield be connected to PE at both ends see also Fig. 39.
Note that this might require an equipotential bonding system.

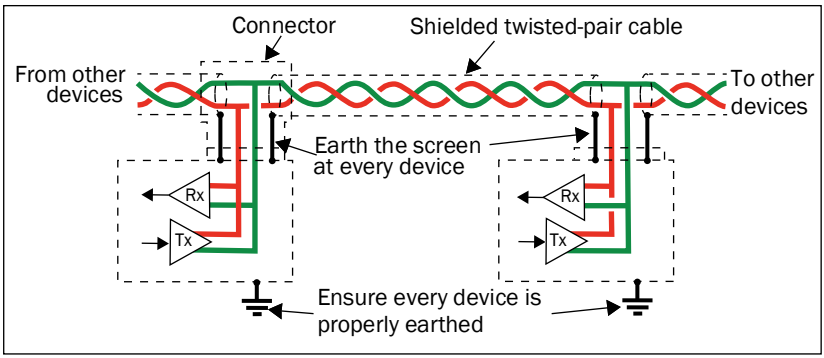


Fig. 39 Ensure that every device is properly earthed.

When the variable speed drive is installed in a cabinet, we recommend that the shield also be connected to the external shield DIN-rail (the DIN-rail is connected to a PE-rail). Note that the shield of the network cable should be kept whole. Connection should cover the largest area possible.

In environments with severe interference, optical signal transmission is the safest and cheapest solution. Moreover, this is usually the recommended media when communicating between buildings. Optical solutions have the advantages of being electrically insulated, lightweight and enable cable lengths of several kilometres to be used. They are also not susceptible to electromagnetic interference, which means that they can be installed along power cables. Please note that CG Drives & Automation does not manufacture or sell fibre optical transmitters/solutions. These are, however, widely available on the market and may be used together with our products.

An equipotential bonding system should be used to level out voltage differences between nodes. The shield in cables must not be used for this purpose.

Main equipotential bonding conductors should be as short as possible. Their resistance from bonded part to the Main Earth Terminal (MET) should, in general, not exceed 0.05 ohm.

CG Drives & Automation Sweden AB
Mörsaregatan 12
Box 222 25
SE-250 24 Helsingborg
Sweden
T +46 42 16 99 00
F +46 42 16 99 49
www.cgglobal.com / www.emotron.com

CG Drives & Automation 01-3698-01r13, 2024-03-18