

**CANOPEN OPTION BOARD OPTE6
USER MANUAL**

TABLE OF CONTENTS

Document: DPD01091D

Release date : 30082018

| | |
|---|-----------|
| 1. Safety | 6 |
| 1.1 Danger | 6 |
| 1.2 Warnings | 7 |
| 1.3 Earthing and earth fault protection | 8 |
| 2. CANopen option board OPTE6 - General | 9 |
| 2.1 Overview | 9 |
| 2.2 Software | 9 |
| 2.2.1 CAN | 9 |
| 2.2.2 CANopen | 9 |
| 3. CANopen protocol description | 10 |
| 3.1 NMT | 10 |
| 3.2 Node control protocols | 11 |
| 3.3 Error control protocols | 12 |
| 3.3.1 Heartbeat protocol | 12 |
| 3.3.2 Node guarding protocol | 13 |
| 3.3.3 EMCY object | 13 |
| 3.4 SDO protocol | 15 |
| 3.5 PDO protocol | 17 |
| 3.5.1 PDO communication parameter record | 18 |
| 3.5.2 COB ID | 18 |
| 3.5.3 Transmission type | 19 |
| 3.5.4 PDO parameter mapping record | 20 |
| 3.6 SYNC protocol | 21 |
| 3.6.1 SYNC with counter | 21 |
| 3.7 Communication objects | 23 |
| 3.7.1 0X1000 - Device Type | 23 |
| 3.7.2 0X1001 - Error Register | 23 |
| 3.7.3 0X1003 - Pre-defined Error Field | 24 |
| 3.7.4 0X1005 - COB ID SYNC | 24 |
| 3.7.5 0X100C - Guard Time | 24 |
| 3.7.6 0X100D - Life Time Factor | 24 |
| 3.7.7 0X1014 - COB ID EMCY | 25 |
| 3.7.8 0X1016 - Heartbeat Consumer Entries | 25 |
| 3.7.9 0X1017 - Producer Heartbeat Time | 26 |
| 3.7.10 0X1018 - Identify Object | 26 |
| 3.7.11 0X1019 - Synchronous counter overflow value | 26 |
| 3.7.12 0X1029 - Error behaviour | 27 |
| 3.8 Saving and restoring the object dictionary | 27 |
| 3.8.1 0X1010 Store parameter field | 27 |
| 3.8.2 0X1011 Restore default parameters | 27 |
| 4. CANopen option board OPTE6 - technical data | 29 |
| 4.1 General | 29 |
| 4.2 New features | 30 |
| 4.3 CAN cable | 30 |
| 4.3.1 Isolated ground connection | 31 |
| 4.3.2 Recommended cable | 32 |
| 5. OPTE6 layout and connections | 33 |
| 5.1 Layout and connections | 33 |
| 5.2 LED Indications | 35 |
| 5.3 Jumpers | 36 |

| | | |
|------------|---|------------|
| 6. | Installation..... | 38 |
| 6.1 | Installation in VACON® NX..... | 38 |
| 6.2 | Installation in VACON® 100..... | 40 |
| 6.3 | Installation in VACON® 20..... | 45 |
| 6.3.1 | Frames MI1, MI2, MI3 | 45 |
| 6.3.2 | Frames MI4, MI5 | 48 |
| 6.4 | Installation in VACON® 20 X and 20 CP | 52 |
| 6.5 | Installation in VACON® 100 X (Frames MM4-MM6)..... | 54 |
| 7. | Commissioning | 58 |
| 7.1 | OPTE6 panel parameters..... | 58 |
| 7.1.1 | OPTE6 additional panel parameters..... | 59 |
| 7.1.2 | Panel parameter change reaction..... | 62 |
| 7.1.3 | Replacing option board and parameter restore..... | 64 |
| 7.2 | OPTE6 Panel Monitor Values | 65 |
| 7.3 | VACON PC-tools | 66 |
| 7.3.1 | PC tool support | 66 |
| 7.3.2 | OPTE6 option board firmware update with VACON® Loader | 66 |
| 7.3.3 | PC Tools for VACON® NXP: NCDriver | 69 |
| 7.3.4 | PC Tools for VACON® 100 and VACON® 20: VACON Live | 71 |
| 7.4 | Quick instructions for controlling the motor..... | 74 |
| 8. | CANopen option board interface | 75 |
| 8.1 | Supported drive modes | 75 |
| 8.2 | Velocity mode | 76 |
| 8.2.1 | PDS State machine | 76 |
| 8.2.2 | CiA-402 objects | 79 |
| 8.2.3 | PDO configuration | 82 |
| 8.3 | Bypass mode | 82 |
| 8.3.1 | PDO configuration | 83 |
| 8.4 | Default process data application mapping..... | 83 |
| 8.4.1 | FB Control Word | 84 |
| 8.4.2 | FB Control Word Extension (general control word)..... | 84 |
| 8.4.3 | FB Speed Reference | 85 |
| 8.4.4 | FB Process data Input 1...16 | 85 |
| 8.4.5 | FB Processdata Input mapping in application | 85 |
| 8.4.6 | FB Status Word | 86 |
| 8.4.7 | FB Status Word Extension (general status word) | 86 |
| 8.4.8 | FB Actual Speed..... | 86 |
| 8.4.9 | FB Processdata Output 1...16 | 87 |
| 8.4.10 | FB Processdata Output mapping in application | 87 |
| 8.5 | VACON anyparameter service | 87 |
| 8.5.1 | Error responses | 87 |
| 8.5.2 | Examples..... | 88 |
| 9. | Fault tracing..... | 89 |
| 9.1 | Typical fault conditions | 90 |
| 9.2 | Fieldbus timeout fault (F53)..... | 91 |
| 9.3 | Detailed fault code | 92 |
| 10. | APPENDIX A: OBJECT DICTIONARY | 93 |
| 10.1 | Communication segment..... | 93 |
| 10.2 | Manufacturer Segment..... | 99 |
| 10.3 | Device Profile Segment..... | 101 |
| 11. | APPENDIX B - FIEDLBUS PARAMETRIZATION..... | 102 |
| 11.1 | Fieldbus control and reference selection | 102 |
| 11.1.1 | Controlling fieldbus parameter | 103 |




| | | |
|------------|---|------------|
| 11.2 | Response to fieldbus fault | 104 |
| 12. | APPENDIX C - FIELDBUS PROCESS DATA MAPPING AND SCALING | 105 |
| 13. | APPENDIX D - CONTROL AND STATUS WORD | 108 |
| 13.1 | Control Word bit description | 108 |
| 13.2 | Status Word Descriptions | 110 |
| 13.3 | Control word bit support in drives..... | 111 |
| 13.4 | Status word bit support in drives..... | 112 |
| 14. | APPENDIX E - FIELDBUS OPTION BOARD COMMUNICATION | 113 |
| 14.1 | Requirements for communication modes..... | 113 |
| 14.2 | Fiedlbus communication mode features and limitations | 114 |
| 14.3 | Normal fieldbus communication | 115 |
| 14.4 | Fast fieldbus communication | 116 |
| 14.5 | Normal Extended Mode | 117 |
| 15. | APPENDIX F - PARAMETERS FOR APPLICATION DEVELOPERS | 118 |

1. SAFETY

This manual contains clearly marked cautions and warnings that are intended for your personal safety and to avoid any unintentional damage to the product or connected appliances.

Please read the information included in cautions and warnings carefully.

The cautions and warnings are marked as follows:

| | |
|---|------------------------------------|
|  | = DANGER! Dangerous voltage |
|  | = WARNING or CAUTION |
|  | = Caution! Hot surface |

1.1 DANGER



The **components of the power unit are live** when the drive is connected to mains potential. Coming into contact with this voltage is **extremely dangerous** and may cause death or severe injury.



The **motor terminals U, V, W and the brake resistor terminals are live** when the AC drive is connected to mains, even if the motor is not running.



After disconnecting the AC drive from the mains, **wait** until the indicators on the keypad go out (if no keypad is attached, see the indicators on the cover). Wait 5 more minutes before doing any work on the connections of the drive. Do not open the cover before this time has expired. After expiration of this time, use a measuring equipment to absolutely ensure that no voltage is present. **Always ensure absence of voltage before starting any electrical work!**



The control I/O-terminals are isolated from the mains potential. However, the **relay outputs and other I/O-terminals may have a dangerous control voltage** present even when the AC drive is disconnected from mains.



Before connecting the AC drive to mains make sure that the front and cable covers of the drive are closed.



During a ramp stop (see the Application Manual), the motor is still generating voltage to the drive. Therefore, do not touch the components of the AC drive before the motor has completely stopped. Wait until the indicators on the keypad go out (if no keypad is attached, see the indicators on the cover). Wait additional 5 minutes before starting any work on the drive.

1.2 WARNINGS



The AC drive is meant for **fixed installations only**.



Do not perform any measurements when the AC drive is connected to the mains.



The **earth leakage current** of the AC drives exceeds 3.5mA AC. According to standard EN61800-5-1, a **reinforced protective ground connection** must be ensured. See Chapter 1.3.



If the AC drive is used as a part of a machine, the **machine manufacturer is responsible** for providing the machine with a **supply disconnecting device** (EN 60204-1).



Only **spare parts** delivered by the manufacturer can be used.



At power-up, power break or fault reset **the motor will start immediately** if the start signal is active, unless the pulse control for Start/Stop logic has been selected. Furthermore, the I/O functionalities (including start inputs) may change if parameters, applications or software are changed. Disconnect, therefore, the motor if an unexpected start can cause danger.



The **motor starts automatically** after automatic fault reset if the auto restart function is activated. See the Application Manual for more detailed information.



Prior to measurements on the motor or the motor cable, disconnect the motor cable from the AC drive.



Do not touch the components on the circuit boards. Static voltage discharge may damage the components.



Check that the **EMC level** of the AC drive corresponds to the requirements of your supply network.



Wear protective gloves when you do mounting, cabling or maintenance operations. There can be sharp edges in the AC drive that can cause cuts.

1.3 EARTHING AND EARTH FAULT PROTECTION



CAUTION!

The AC drive must always be earthed with an earthing conductor connected to the earthing terminal marked with

The earth leakage current of the drive exceeds 3.5mA AC. According to EN61800-5-1, one or more of the following conditions for the associated protective circuit must be satisfied:

- 0) The protective conductor must have a cross-sectional area of at least 10 mm² Cu or 16 mm² Al, through its total run.
- a) Where the protective conductor has a cross-sectional area of less than 10 mm² Cu or 16 mm² Al, a second protective conductor of at least the same cross-sectional area must be provided up to a point where the protective conductor has a cross-sectional area not less than 10 mm² Cu or 16 mm² Al.
- b) Automatic disconnection of the supply in case of loss of continuity of the protective conductor.

The cross-sectional area of every protective earthing conductor which does not form part of the supply cable or cable enclosure must, in any case, be not less than:

- 2.5mm² if mechanical protection is provided or
- 4mm² if mechanical protection is not provided.

The earth fault protection inside the AC drive protects only the drive itself against earth faults in the motor or the motor cable. It is not intended for personal safety.

Due to the high capacitive currents present in the AC drive, fault current protective switches may not function properly.



Do not perform any voltage withstand tests on any part of the AC drive. There is a certain procedure according to which the tests must be performed. Ignoring this procedure can cause damage to the product.

NOTE! You can download the English and French product manuals with applicable safety, warning and caution information from <https://www.danfoss.com/en/service-and-support/>.

REMARQUE Vous pouvez télécharger les versions anglaise et française des manuels produit contenant l'ensemble des informations de sécurité, avertissements et mises en garde applicables sur le site <https://www.danfoss.com/en/service-and-support/>.

2. CANOPEN OPTION BOARD OPTE6 - GENERAL

2.1 OVERVIEW

OPTE6 is a CANopen adapter board for VACON® AC drives. The board allows the AC drive to be controlled by using the CANopen protocol. The board implements the AC drive profile with the velocity mode.

The option board firmware implements the following protocol specifications:

- **CiA-301** CANopen communication specification version 4.2
- **CiA-402** CANopen Profile for Drives and Motion Controller version 3.2
 - Device: AC drive
 - Operation mode: velocity mode
- **CiA-303-3** CANopen indicator specification, implemented by using 2 CANopen status led indicators

2.2 SOFTWARE

2.2.1 CAN

The CAN data link layer protocol is standardised in ISO 11898. The standard describes mainly the data link layer composed of the logical link control (LLC) sub layer and the media access control (MAC) sub layer, and some aspects of the physical layer of the OSI reference model.

2.2.2 CANOPEN

CANopen is an application layer protocol on top of the CAN bus.

The protocol specification describes:

- Set of bit rates to support
- Network Management (NMT)
- Service data transmission (SDO)
- Process data transmission (PDO)
- Error message transmission (EMCY)
- Node status monitoring (heartbeat and node guarding)
- Identity information
- Parameter saving and restoring

3. CANOPEN PROTOCOL DESCRIPTION

3.1 NMT

NMT network management manages CANopen, and is a mandatory, common feature for all devices. The protocol describes several node control services and the state machine.

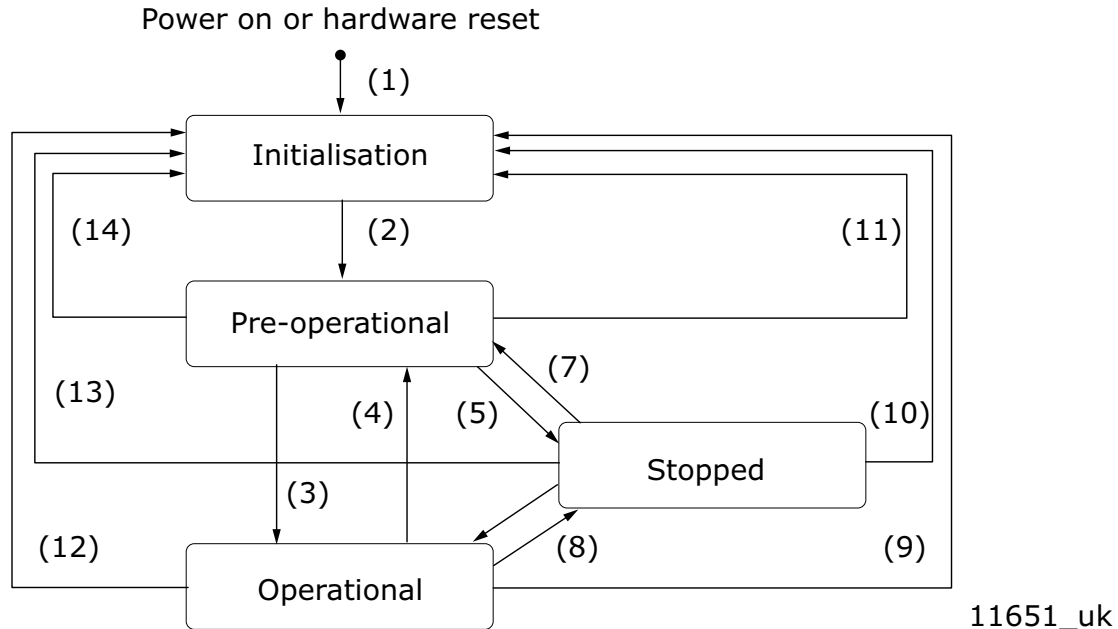


Figure 1. NMT state machine

1 = When the power is on, the NMT state is entered autonomously

2 = The NMT state initialisation is finished, the NMT pre-operational state is entered automatically

3 = NMT service starts with remote node indication or by local control

4 and 7 = NMT service enters pre-operational indication

5 and 8 = NMT service stops remote node indication

6 = NMT service starts remote node indication

9, 10 and 11 = NMT resets node indication

12, 13 and 14 = Indication of NMT service reset communication

Boot-up protocol

After a node starts, it will enter automatically into the pre-operational state. Always when this transition occurs, a boot-up message is sent into the bus.

Table 1: Boot-up message

| CAN ID | LENGTH | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|-----------------|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0x700 + Node ID | 1 | 0 | | | | | | | |

3.2 NODE CONTROL PROTOCOLS

Protocol start remote node

The start remote node message sets the node(s) into operational state. See Figure 1. NMT state machine. If the node ID in the message is set to '0', the message affects all nodes (broadcast).

Table 2: Start remote node message

| CAN ID | LENGTH | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|--------|--------|-------|---------|-------|-------|-------|-------|-------|-------|
| 0x0 | 0x2 | 0x1 | NODE ID | | | | | | |

Protocol stop remote node

The stop remote node message sets the node(s) into stopped state. See Figure 1 NMT state machine. If the node ID in the message is set to '0', the message affects all nodes (broadcast). When the node is in stopped state, it will not answer to SDO or PDO messages.

Table 3: Stop remote node message

| CAN ID | LENGTH | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|--------|--------|-------|---------|-------|-------|-------|-------|-------|-------|
| 0x0 | 0x2 | 0x2 | NODE ID | | | | | | |

Protocol enter pre-operational

The enter pre-operational message sets the node(s) into pre-operational state. See Figure 1. NMT state machine. If the node ID in the message is set to '0', the message affects all nodes (broadcast). When the node is in pre-operational state, it will not answer to PDO messages.

Table 4: Enter pre-operational message

| CAN ID | LENGTH | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|--------|--------|-------|---------|-------|-------|-------|-------|-------|-------|
| 0x0 | 0x2 | 0x80 | NODE ID | | | | | | |

Protocol reset node

The reset node message makes the node(s) apply application reset. See Figure 1. NMT state machine. Application reset sets the whole object dictionary back to the default or previously saved values. If the node ID in the message is set to '0', the message affects all nodes (broadcast). After the node has made the application reset, it will enter the pre-operational state automatically from the initialising state. This also creates a boot-up event and the boot-up message is sent after the reset.

Table 5: Reset node message

| CAN ID | LENGTH | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|--------|--------|-------|---------|-------|-------|-------|-------|-------|-------|
| 0x0 | 0x2 | 0x81 | NODE ID | | | | | | |

Protocol reset communication

The reset communication message makes the node(s) apply communication reset. See Figure 1. NMT state machine. Communication reset does not affect the object dictionary values. If the node ID in the message is set to '0', the message affects all nodes (broadcast). After the node has made the communication reset, it will enter the pre-operational state automatically from the initialising state. This also creates a boot-up event and the boot-up message is sent after the reset.

Table 6: Reset communication message

| CAN ID | LENGTH | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|--------|--------|-------|---------|-------|-------|-------|-------|-------|-------|
| 0x0 | 0x2 | 0x82 | NODE ID | | | | | | |

3.3 ERROR CONTROL PROTOCOLS

It is not allowed to use guarding protocol and heartbeat protocol on one NMT slave at the same time. If the heartbeat producer time is unequal 0, the heartbeat protocol is used.

3.3.1 HEARTBEAT PROTOCOL

Heartbeat protocol defines the producer and consumer. The producer node sends its NMT status that is then available for any consumer node. The consumer node is the receiver of heartbeat messages. The producer node has a timing parameter that indicates how often the heartbeat message should be sent. The consumer node has a relative parameter that indicates how often the heartbeat message should be received. If the consumer does not receive the heartbeat message within the time defined in the heartbeat object entry, an error event occurs.

Table 7: Node status description

| Value | Description |
|-------|-----------------|
| 0x0 | Boot-up |
| 0x4 | Stopped |
| 0x5 | Operational |
| 0x7F | Pre-operational |

Table 8: Heartbeat message

| CAN ID | LENGTH | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|-----------------|--------|--------|-------|-------|-------|-------|-------|-------|-------|
| 0x700 + Node ID | 0x1 | Status | | | | | | | |

Table 9: Heartbeat-related objects in OD

| Index | Description |
|--------|-------------------------|
| 0x1016 | Consumer heartbeat time |
| 0x1017 | Producer heartbeat time |
| 0x1029 | Error behaviour |

3.3.2 NODE GUARDING PROTOCOL

Node guarding protocol is a NMT master driver protocol, where the master sends a remote transmission request, which is answered by the slave. The slave response includes one data byte that consists of a NMT slave state, and a toggle bit that toggles every response.

NOTE! The CiA application note 802 recommends that the node guarding protocol should not be used, because of different handling of RTR frames in CAN controllers.

OPTE6 option board does not have a hardware-triggered automatic response to the RTR frame. RTR information is handled by software, and the response data always consists of updated information.

Table 10: Node guarding RTR frame (remote request)

| CAN ID | LENGTH | RTR | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|-----------------|--------|-----|-------|-------|-------|-------|-------|-------|-------|-------|
| 0x700 + Node ID | 0x0 | 1 | | | | | | | | |

Table 11: Node guarding response

| CAN ID | LENGTH | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|-----------------|--------|-------|--------|-------|-------|-------|-------|-------|-------|
| 0x700 + Node ID | 0x1 | t | Status | | | | | | |

Table 12: Node guarding slave status

| Status | |
|--------|-----------------|
| Value | Description |
| 0x4 | Stopped |
| 0x5 | Operational |
| 0x7F | Pre-operational |

Table 13: Node guarding related objects in OD

| Index | Description |
|--------|------------------|
| 0x100C | Guard time |
| 0x100D | Life time factor |

3.3.3 EMCY OBJECT

Option board works as an EMCY producer. The EMCY object is transmitted when a fault occurs in the drive or option board. To switch off the EMCY producer, disable the EMCY COB-id by writing MSB to 1 (object 0x1014).

When an error occurs, the EMCY message is transmitted with the current value of the error register and the error code is inserted into the pre-defined error field list. The newest error code is always the first sub-index on the error field list. When all active errors are cleared, an empty EMCY object is transmitted.

If a drive-internal fault occurs, the MSEF field contains the drive fault code. See the application and user manual for possible fault codes. The ER field holds a bit coded value of the error type. See object 0x1001 for more details.

Table 14: EMCY message

| CAN ID | LENGTH | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|----------------|--------|-------|-------|-------|-------|-------|-------|-------|-------|
| 0x80 + Node ID | 0x8 | EEC | | ER | MSEF | | | | |

Table 15: EMCY message data fields

| | |
|------|----------------------------------|
| EEC | Emergency error code |
| ER | Error register value |
| MSEF | Manufacturer-specific error code |

Table 16: Used EMCY error codes and description MSEF fields

| DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|--------|-------|---|-------|-------|-------|-------|-------|
| 0x0000 | ER | 3: Number of remaining error sources | | | | | |
| 0x1000 | | Drive fault codes | | | | | |
| 0x8110 | | - | | | | | |
| 0x8120 | | - | | | | | |
| 0x8130 | | 3: Heartbeat consumer subindex 4: Heartbeat consumer node-ID | | | | | |
| 0x8140 | | - | | | | | |
| 0x8210 | | - | | | | | |
| 0x8220 | | - | | | | | |
| 0x8240 | | - | | | | | |
| 0x8250 | | - | | | | | |

Table 17: Description and behavior of different error situations

| EEC | Description | Error behaviour | Err LED |
|--------|---------------------------|--|--------------|
| 0x0000 | Error Reset or No Error | If MSEF field is empty all error sources are cleared and drive fault is cleared. | - |
| 0x1000 | Generic Error | Drive fault codes have changed. | - |
| 0x8120 | CAN in Error Passive Mode | EMCY is sent after CAN driver goes back to active state. This also clears the fault. | Single flash |

Table 17: Description and behavior of different error situations

| | | | |
|--------|------------------------|---|-----------------|
| 0x8130 | Life Guard Error | Error is reset when a RTR is received or either of the life guard objects (0x100C, 0x100D) is written to zero. | Double flash |
| | Heartbeat Error | Error is reset when a HB message is received by the HB consumer, or the consumer entry is changed (either Node-ID or Heartbeat Time). | |
| 0x8140 | Recovered from Bus-Off | EMCY is sent after CAN driver goes back to active state. This also clears the fault. | On |
| 0x8250 | PDO timer expired | Error is cleared when a PDO is received (in expired PDO). | Quadruple flash |

All communication errors are reset if a reset command is given. This does not however reset drive faults if there are active error sources.

EMCYs are also created in some cases, even though a fault is not created. These are for notification only.

Table 18: Notification EMCY objects

| EEC | Description |
|--------|---------------------------------------|
| 0x8110 | CAN overrun (objects lost) |
| 0x8210 | PDO not processed due to length error |
| 0x8220 | PDO length exceeded |
| 0x8240 | Unexpected SYNC data length |

Table 19: EMCY-relate objects in OD

| Index | Description |
|--------|------------------------------|
| 0x1001 | Error register |
| 0x1003 | Pre-defined error field list |
| 0x1014 | EMCY object COB-ID |

3.4 SDO PROTOCOL

The Option board contains one SDO server. The SDO protocol provides a direct access to the object entries of the object dictionary of the CANopen device. Each message is acknowledged by the server. The protocol is mostly used to set and read parameters from the object dictionary at the pre-operational state. Some objects have limitations for SDO usage at the operational state.

Up to four bytes can be transferred by using the expedited transfer, where the data fits into one CAN message. For bigger than 4-byte object sizes, segmented transfer must be used. Optionally, block transfer is also possible with bigger data types. Block transfer is most efficient with big data sizes.

Table 20: SDO-related objects in OD

| Index | Description |
|--------|-----------------------------|
| 0x1200 | SDO server parameter object |

Some objects have limitations for SDO usage at the operational state. The following SDO abort codes can be returned by OPTE6 option board.

Table 21. SDO abort codes

| Abort code | Description |
|------------|---|
| 0504 0001h | Client/server command specifier not valid or unknown |
| 0504 0005h | Out of memory |
| 0601 0000h | Unsupported access to an object |
| 0601 0002h | Attempt to write a read only object |
| 0602 0000h | Object does not exist in the object dictionary |
| 0604 0041h | Object cannot be mapped to the PDO |
| 0604 0042h | The number and length of the objects to be mapped would exceed PDO length |
| 0604 0043h | General parameter incompatibility reason |
| 0606 0000h | Access failed due to a hardware error |
| 0607 0010h | Data type does not match, length of service parameter does not match |
| 0609 0011h | Sub-index does not exist |
| 0609 0030h | Invalid value for parameter (download only) |
| 0609 0031h | Value of parameter written too high (download only) |
| 0609 0032h | Value of parameter written too low (download only) |
| 0800 0020h | Data cannot be transferred or stored to the application |
| 0800 0022h | Data cannot be transferred or stored to the application because of the present device state |
| 0800 0024h | No data available |

3.5 PDO PROTOCOL

Process data objects PDOs are used to transmit real-time data with no protocol overhead. Each PDO has its mapping and communication parameter record.

There are two different types of PDOs. Transmit PDOs for producing data into network and Receive PDOs for consuming data from network. OPTE6 board supports 5 receive and 5 transmit PDOs.

Each of rxPDO and txPDO mapping parameter record contains four (4) process data items. For example, 0x01600 Receive PDO Mapping Parameter 1 contains following process data items:

- Mapping entry 1
- Mapping entry 2
- Mapping entry 3
- Mapping entry 4

All VACON® AC drives supports transferring of eight (8) process data items between master device and AC drive. Using of 9-16 process data items requires Normal Extended Communication or Fast Communication support from VACON® AC drive. See details of communication modes in following chapters:

- APPENDIX E - FIELD BUS OPTION BOARD COMMUNICATION
- APPENDIX F - PARAMETERS FOR APPLICATION DEVELOPERS

If master device assigns 16 process data items and VACON® AC drive supports only 8 process data items, then content of process data items 9-16 is ignored in VACON® AC drive.

Table 22: PDO-related objects in OD

| Index | Description |
|--------|--|
| 0x1400 | 1st rxPDO communication parameter record |
| 0x1401 | 2nd rxPDO communication parameter record |
| 0x1402 | 3rd rxPDO communication parameter record |
| 0x1403 | 4th rxPDO communication parameter record |
| 0x1404 | 5th rxPDO communication parameter record |
| 0x1600 | 1st rxPDO mapping parameter record |
| 0x1601 | 2nd rxPDO mapping parameter record |
| 0x1602 | 3rd rxPDO mapping parameter record |
| 0x1603 | 4th rxPDO mapping parameter record |
| 0x1604 | 5th rxPDO mapping parameter record |
| 0x1800 | 1st txPDO communication parameter record |
| 0x1801 | 2nd txPDO communication parameter record |
| 0x1802 | 3rd txPDO communication parameter record |
| 0x1803 | 4th txPDO communication parameter record |
| 0x1804 | 5th txPDO communication parameter record |
| 0x1A00 | 1st txPDO mapping parameter record |
| 0x1A01 | 2nd txPDO mapping parameter record |
| 0x1A02 | 3rd txPDO mapping parameter record |
| 0x1A03 | 4th txPDO mapping parameter record |
| 0x1A04 | 5th txPDO mapping parameter record |

3.5.1 PDO COMMUNICATION PARAMETER RECORD

PDO communication parameter record defines the COB-id, transmission type and how often the PDO is transmitted. The fields can be modified during the pre-operational state.

Table 23: PDO communication parameter record

| Indexes | Sub-index | Name | Data type | RX PDO | TX PDO |
|---------|-----------|-----------------------------|------------|---------------|--------|
| 0x1400 | 0 | Highest sub-index supported | UNSIGNED8 | ro | ro |
| 0x1401 | 1 | COB ID | UNSIGNED32 | r/w | r/w |
| 0x1402 | 2 | Transmission type | UNSIGNED8 | r/w | r/w |
| 0x1403 | 3 | Inhibit time | UNSIGNED16 | ro | r/w |
| 0x1404 | 4 | Reserved | UNSIGNED8 | ro | ro |
| 0x1800 | 5 | Event timer | UNSIGNED16 | r/w | r/w |
| 0x1801 | 6 | SYNC start value | UNSIGNED8 | Not available | r/w |
| 0x1802 | | | | | |
| 0x1803 | | | | | |
| 0x1804 | | | | | |

3.5.2 COB ID

COB ID determines whether the PDO is valid (active) and using 11-bit or 29-bit frames.

Table 24: COB ID

| 31 | 30 | 29 | 28 | 11 | 10 | 0 |
|-------|----------|-------|---------|----|----|---------------|
| Valid | Reserved | Frame | 0x00000 | | | 11-bit CAN-ID |
| | | | | | | 29-bit CAN-ID |

Table 25: COB ID data fields

| Bit(s) | Value | Description |
|---------------|-------|---|
| Valid | 0 | PDO exists / enabled |
| | 1 | PDO does not exist / disabled |
| Reserved | x | Not applicable |
| Frame | 0 | 11-bit CAN-ID valid |
| | 1 | 20-bit CAN-ID valid |
| 29-bit CAN-ID | x | 29-bit CAN-ID of the CAN extended frame |
| 11-bit CAN-ID | x | 11-bit CAN-ID of the CAN base frame |

3.5.3 TRANSMISSION TYPE

Table 26: PDO transmission types

| Value | Description | Receive PDO | Transmit PDO |
|-------|---|----------------|--------------|
| 0x00 | Synchronous (acyclic) | X | X |
| 0x01 | Synchronous (cyclic every sync) | X ₁ | X |
| 0x02 | Synchronous (cyclic every 2 nd sync) | X ₁ | X |
| 0x03 | Synchronous (cyclic every 3 rd sync) | X ₁ | X |
| 0x04 | Synchronous (cyclic every 4 th sync) | X ₁ | X |
| ... | ... | X ₁ | X |
| 0xF0 | Synchronous (cyclic every 240 th sync) | X ₁ | X |
| 0xF1 | Reserved | - | - |
| ... | ... | - | - |
| 0xFB | Reserved | - | - |
| 0xFC | RTR-only (synchronous) | - | X |
| 0xFD | RTR-only (Event-driven) | - | X |
| 0xFE | Event-driven (manufacturer-specific) | X | X |
| 0xFF | Event-driven (device and application profile) | X | X |

¹ For receive PDO, each sync transmission mode equals the same. Each sync always activates the latest received PDO value.

Synchronous means that the PDO is transmitted after the SYNC. The CANopen device starts sampling the data with the reception of the SYNC. If the transmission mode of the PDO is acyclic, the CANopen device gives an internal event, the sampling starts with the next SYNC and the PDO is transmitted afterwards. If the transmission mode is cyclic, the sampling starts with the reception of every SYNC, every second SYNC, every third SYNC etc. depending on the given value, and the PDO is transmitted afterwards.

RTR-only means that the PDO is requested via RTR. If the transmission mode of the PDO is synchronous, the CANopen device starts sampling with the reception of every SYNC and will buffer the PDO. If the mode is event-driven, the CANopen device starts the sampling with the reception of the RTR and transmits the PDO immediately.

Event-driven means that the PDO can be transmitted at any time based on the occurrence of the internal event of the CANopen device. An event that triggers the OPTE6 transmission occurs when the data mapped into the PDO is changed. Also, an event timer can be used to create transmit events.

Inhibit time

For transmit PDOs, the inhibit time defines the minimum transmission interval, when 0xFE or 0xFF transmission types are selected. For receive PDOs, the inhibit time is disabled. The inhibit time is 16bit unsigned value that is given as multiple of 100µs. Zero value means that the inhibit time is disabled.

Event timer

For a transmit PDO event, the timer defines the maximum interval between the transmissions, if the transmission type is set to 0xFE or 0xFF.

For a receive PDO event, the timer activates the deadline monitoring. The deadline monitoring is activated at the first received PDO. If the time between the after the last PDO received is longer than defined in the event timer, a fault will occur.

Event timer is 16bit unsigned value that is given as multiple of 1ms. Zero value means that the event timer is disabled.

NOTE: It is recommended to define non-zero Event timer value when using Event-driven transmission type. By default Event timer value is zero, which together with Event-driven transmission type means that OPTe6 CANopen transmits TPDO only when its data content changes. In certain situations TPDO data content rarely or never changes. For example, when running zero speed, TPDO1 data content (status word and actual speed) does not change after zero speed has been reached.

Sync start value

Sync start value gives the possibility to compensate network peak traffic in case of sync transmission mode. If the sync start value is zero, the normal sync behaviour for the PDO is used. If the sync start value is greater than zero, the PDO waits for the SYNC message that contains the counter value. When the counter value of a SYNC message equals the SYNC start value, the first SYNC message is regarded as received. The sync start value must not be changed while the PDO exists. See the SYNC message format in Table 30.

3.5.4 PDO PARAMETER MAPPING RECORD

Each PDO consists of a maximum of 8 bytes of mapped data. To data map the PDO, use a corresponding mapping record that consists of index, sub-index and the length of the mapped object.

NOTE! All VACON® AC drives supports transferring of eight (8) process data items between master device and AC drive. Read about usage of 9-16 process data items in Chapter 3.5.

Table 27: PDO mapping structure

| | | | | | |
|-------|----|-----------|---|--------|---|
| 31 | 16 | 15 | 8 | 7 | 0 |
| Index | | Sub-index | | Length | |

Table 28: PDO mapping parameter record

| Indexes | Sub-index | Name | Data type | Access |
|--|-----------|---------------------------------|------------|--------|
| 0x1600 0x1601 0x1602 0x1603 0x1604 0x1A00 0x1A01 0x1A02 0x1A03 0x1A04 | 0 | Number of mapped objects in PDO | UNSIGNED8 | r/w |
| | 1 | 1st object to be mapped | UNSIGNED32 | r/w |
| | 2 | 2nd object to be mapped | UNSIGNED32 | r/w |
| | 3 | 3rd object to be mapped | UNSIGNED32 | r/w |
| | 4 | 4th object to be mapped | UNSIGNED32 | r/w |

To data map the PDOs, first disable the related PDO COB ID in the pre-operational state. In the mapping structure, write the sub-index 0 to zero (number of mapped objects). Then write the mapping structures on the mapping parameter record, starting from the sub-index 1. When you have written all the necessary structures, write the sub-index 0 to correspond to the mapped objects.

Example on how to write a dummy object to RPDO1 4th entry (when using Bypass mode) is explained below:

Table 29. RPDO mapping example

| Transfer data (hex) | Interpretation |
|-------------------------|--|
| 23 00 14 01 01 02 00 80 | Write RPDO1 COB-ID (1400:01) to invalid (0x8000 0201) |
| 2F 00 16 00 00 00 00 00 | Write RPDO1 mapping number of entries (1600:00) to 0 |
| 23 00 16 04 10 00 06 00 | Write RPDO1 4 th entry (1600:04) to Dummy object (00060010) |
| 2F 00 16 00 04 00 00 00 | Write RPDO1 mapping number of entries (1600:00) to 4 |
| 23 00 14 01 01 02 00 00 | Write RPDO1 COB-ID (1400:01) as valid (0x201) |

3.6 SYNC PROTOCOL

Sync protocol is used by PDOs when the transmission is synchronous. The sync object that is defined by COB ID in the object 0x1005 triggers the transmission of the txPDOs, or activates the previously received data of the rxPDO. At the default sync message the CAN-ID is 0x80. The sync message is a zero-length message but optionally it can consist of an 8bit counter.

Table 30: SYNC message

| CAN ID | LENGTH |
|--------|--------|
| 0x80 | 0x0 |

Table 31: SYNC message with counter

| CAN ID | LENGTH | DATA0 |
|--------|--------|---------|
| 0x80 | 0x1 | Counter |

3.6.1 SYNC WITH COUNTER

When a counter is used in a sync message, the PDOs that have a defined sync start value compare the value against the sync message counter. The sync producer counter will overflow after it reaches the value defined in its 'synchronous counter overflow value' at the object 0x1019. Also, the sync consumer has the object 0x1019 even when the value itself is ignored. When the value of the sync consumer is greater than zero, the sync counter handling and expecting of the sync messages with counter are activated.

When the sync start value and the sync counter value match, the first sync message is regarded as received.

The following figure shows an example of SYNC messaging, when the slave is configured with:

- 0x1019 - Synchronous counter 128
- 0x1800,2 - Transmission type = 2 (Cyclic, No. of SYNCs = 2)
- 0x1800,6 - Sync start value = 4

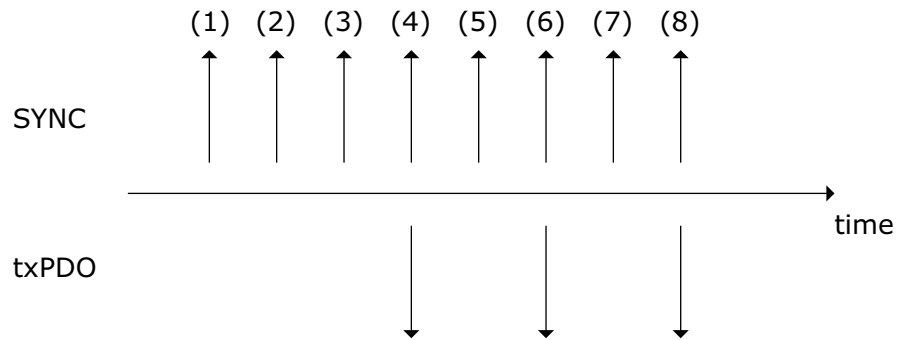


Figure 2. txPDO responses to SYNC messages

Table 32: Sync-related object in OD

| Index | Description |
|--------|--|
| 0x1005 | COB ID SYNC |
| 0x1019 | Synchronous counter |
| 0x1014 | EMCY object COB ID |
| 0x1400 | 1st rxPDO communication parameter record |
| 0x1401 | 2nd rxPDO communication parameter record |
| 0x1402 | 3rd rxPDO communication parameter record |
| 0x1403 | 4th rxPDO communication parameter record |
| 0x1404 | 5th rxPDO communication parameter record |
| 0x1800 | 1st txPDO communication parameter record |
| 0x1801 | 2nd txPDO communication parameter record |
| 0x1802 | 3rd txPDO communication parameter record |
| 0x1803 | 4th txPDO communication parameter record |
| 0x1804 | 5th txPDO communication parameter record |

3.7 COMMUNICATION OBJECTS

3.7.1 0X1000 - DEVICE TYPE

The device type object indicates basic information about the device, including the supported device profile and the profile settings.

Table 33: 0x1000 Device type

| Index | Sub-index | Value | Name | Data type | Access |
|--------|-----------|------------|-------------|------------|--------|
| 0x1000 | - | 0x00010192 | Device type | UNSIGNED32 | const |

Value description:

0x0192 = 402 (Drive profile)

0x0001 = AC drive with PDO set for a generic drive device

3.7.2 0X1001 - ERROR REGISTER

Error register indicates the active error code.

Table 34: 0x1001 Error register

| Index | Sub-index | Value | Name | Data type | Access |
|--------|-----------|-------|----------------|-----------|--------|
| 0x1001 | - | 0x0 | Error register | UNSIGNED8 | ro |

Table 35: Error register bit descriptions

| Bit | Meaning |
|-----|---|
| 0 | Generic error |
| 1 | Current |
| 2 | Voltage |
| 3 | Temperature |
| 4 | Communication error (overflow, error state) |
| 5 | Device profile-specific* |
| 6 | Reserved* |
| 7 | Manufacturer-specific* |

* Not used/supported

3.7.3 0X1003 - PRE-DEFINED ERROR FIELD

Pre-defined error field is a list of errors signaled with an EMCY object, listing the error history of up to 9 error entries. Sub-index 1 contains the latest error.

Table 36: 0x1003 Pre-defined error field

| Index | Sub-index | Value | Name | Data type | Access |
|--------|-----------|-------|----------------------|------------|--------|
| 0x1003 | 0 | 0x0 | Number of errors | UNSIGNED8 | ro |
| | 1 | 0x0 | Standard error field | UNSIGNED32 | ro |
| | .. | .. | .. | .. | .. |
| | 9 | 0x0 | Standard error field | UNSIGNED32 | ro |

3.7.4 0X1005 - COB ID SYNC

Defines the synchronisation message COB ID. Receiving the sync message causes actions in the PDOs that have a synchronous transmission mode.

Table 37: 0x1005 COB ID sync

| Index | Sub-index | Value | Name | Data type | Access |
|--------|-----------|------------|-------------|------------|--------|
| 0x1005 | - | 0x00000080 | COB ID sync | UNSIGNED32 | r/w |

3.7.5 0X100C - GUARD TIME

The object contains the guard time in milliseconds. As a default, guarding is disabled.

Table 38: 0x100C Guard time

| Index | Sub-index | Value | Name | Data type | Access |
|--------|-----------|--------|------------|------------|--------|
| 0x100C | - | 0x0000 | Guard time | UNSIGNED16 | r/w |

3.7.6 0X100D - LIFE TIME FACTOR

Life time factor is used together with guard time, which is multiplied with the life time factor.

Table 39: 0x100D Guard time

| Index | Sub-index | Value | Name | Data type | Access |
|--------|-----------|-------|------------|-----------|--------|
| 0x100D | - | 0x00 | Guard time | UNSIGNED8 | r/w |

Node life time = life time factor x guard time. If node life time is zero, guarding is disabled.

3.7.7 0X1014 - COB ID EMCY

The object defines the emergency message COB ID.

Table 40: 0x1014 COB ID EMCY

| Index | Sub-index | Value | Name | Data type | Access |
|--------|-----------|------------------------|-------------|------------|--------|
| 0x1014 | - | 0x00000080+ node id | COB ID EMCY | UNSIGNED32 | r/w |

Setting MSB (bit 31) to 1 will disable sending of EMCY messages.

3.7.8 0X1016 - HEARTBEAT CONSUMER ENTRIES

The device can act as the heartbeat consumer. Up to 8 devices can be monitored, as defined in the table below. If the heartbeat transmission delay of a defined node ID exceeds the heartbeat time, the error behaviour is activated according to the error behaviour object.

Table 41: 0x1016 Heartbeat consumer entries

| Index | Sub-index | Value | Name | Data type | Access |
|--------|-----------|-------------|----------------------------|------------|--------|
| 0x1016 | 0 | 0x08 | Number of entries | UNSIGNED8 | ro |
| | 1 | 0x0000 0000 | Consumer heart beat time 1 | UNSIGNED32 | r/w |
| | 2 | 0x0000 0000 | Consumer heart beat time 2 | UNSIGNED32 | r/w |
| | 3 | 0x0000 0000 | Consumer heart beat time 3 | UNSIGNED32 | r/w |
| | 4 | 0x0000 0000 | Consumer heart beat time 4 | UNSIGNED32 | r/w |
| | 5 | 0x0000 0000 | Consumer heart beat time 5 | UNSIGNED32 | r/w |
| | 6 | 0x0000 0000 | Consumer heart beat time 6 | UNSIGNED32 | r/w |
| | 7 | 0x0000 0000 | Consumer heart beat time 7 | UNSIGNED32 | r/w |
| | 8 | 0x0000 0000 | Consumer heart beat time 8 | UNSIGNED32 | r/w |

Table 42: Consumer heartbeat time entry

| | | | | | |
|---------------------------|-----------|-----------|-----------|----------------|----------|
| 31 | 24 | 23 | 16 | 15 | 0 |
| Not used, must be zeroes. | | Node ID | | Heartbeat time | |

3.7.9 0X1017 - PRODUCER HEARTBEAT TIME

Heartbeat producer object consists of the time in milliseconds (ms) that it takes to transmit the heartbeat message into the network. If the value is zero, the heartbeat is not used.

Table 43: 0x1017 Producer heartbeat time

| Index | Sub-index | Value | Name | Data type | Access |
|--------|-----------|--------|-------------------------|------------|--------|
| 0x1017 | - | 0x0000 | Producer heartbeat time | UNSIGNED16 | r/w |

Table 44: Heartbeat message

| | | |
|--------------|--------|------------|
| CAN ID | LENGTH | DATA0 |
| 0x700 + node | 0x1 | Node state |

3.7.10 0X1018 - IDENTIFY OBJECT

The object gives information about the option board

Table 45: 0x1018 Identify object

| Index | Sub-index | Value | Name | Data type | Access |
|--------|-----------|-------|-------------------|------------|--------|
| 0x1018 | 0 | 0x04 | Number of entries | UNSIGNED8 | ro |
| | 1 | 0x90 | Vendor ID | UNSIGNED32 | ro |
| | 2 | - | Product code | UNSIGNED32 | ro |
| | 3 | - | Revision number | UNSIGNED32 | ro |
| | 4 | - | Serial number | UNSIGNED32 | ro |

3.7.11 0X1019 - SYNCHRONOUS COUNTER OVERFLOW VALUE

The synchronous counter overflow value defines whether a counter is mapped into the SYNC message, as well as the highest value the counter can reach. 0 disables the sync counter.

Table 46: 0x1019 Synchronous counter

| Index | Sub-index | Value | Name | Data type | Access |
|--------|-----------|-------|---------------------|-----------|--------|
| 0x1019 | - | 0x00 | Synchronous counter | UNSIGNED8 | r/w |

3.7.12 0X1029 - ERROR BEHAVIOUR

Error behaviour allows a change in the default error behaviour if there is a communication error.

Table 47: 0x1029 Error behaviour

| Index | Sub-index | Value | Name | Data type | Access | Min | Max |
|--------|-----------|-------|---------------------|-----------|--------|-----|-----|
| 0x1029 | 0 | 0x01 | Number of entries | UNSIGNED8 | ro | 2 | 2 |
| | 1 | 0x00 | Communication error | UNSIGNED8 | r/w | 0 | 2 |
| | 2 | 0x01 | Internal error | UNSIGNED8 | r/w | 1 | 1 |

Table 48: Error behaviour

| Value | Description |
|--------|--------------------|
| 0 | Pre-operational |
| 1 | No change in state |
| 2 | Stopped |
| 3..127 | Reserved |

3.8 SAVING AND RESTORING THE OBJECT DICTIONARY

CANopen defines a way of restoring the values in an object dictionary to the defaults and saving the values if the modified values must be valid after the power cycle. The manufacturer-specific bypass configuration can be restored to the object dictionary.

3.8.1 0X1010 STORE PARAMETER FIELD

To save the object dictionary, use the object 0x1010 'Store Parameter Field'.

The option board only saves the parameters in the object dictionary with a command. Autonomous saving is not supported. To save the parameters in the object dictionary, write the value 0x65766173 (ASCII "save") into the sub-index by using the SDO protocol.

Table 49: 0x1010 Store parameter field

| Index | Sub-index | Name | Data type | Access |
|--------|-----------|-----------------------------|------------|--------|
| 0x1010 | 0 | Highest sub-index supported | UNSIGNED8 | ro |
| | 1 | Save all parameters | UNSIGNED32 | r/w |

3.8.2 0X1011 RESTORE DEFAULT PARAMETERS

The object values of the object dictionary are restored to defaults by using the object 0x1011. Option board supports restoring All parameters (sub index 1) and manufacturer-specific Bypass mode defaults (sub index 4).

To restore parameters, write 0x64616F6C (ASCII "load") into the sub-index by using the SDO protocol. Default object values are selected after reset. Restore all default parameter restores CiA-402 default parameters (refer to Chapter 8.2.2.2). Bypass parameter set is described in Chapter 8.3.

Table 50: 0x1011 Restore default parameters

| Index | Sub-index | Name | Data type | Access |
|--------|-----------|-----------------------------------|------------|--------|
| 0x1011 | 0 | Highest sub-index supported | UNSIGNED8 | ro |
| | 1 | Restore all default parameters | UNSIGNED32 | r/w |
| | 4 | Restore bypass parameter set* | UNSIGNED32 | r/w |
| | 5 | Restore Puller parameter set | UNSIGNED32 | r/w |
| | 6 | Restore Co-Extruder parameter set | UNSIGNED32 | r/w |

* The bypassed set disables the CiA-402 drive profile and resets the PDO mapping to the vendor specific configuration. See Chapter 8.2 and Chapter 8.3 for more information.

4. CANOPEN OPTION BOARD OPTe6 - TECHNICAL DATA

4.1 GENERAL

Table 51. Technical data of CANopen option board

| | | |
|--|--|--|
| CAN bus electrical isolation | 500 VDC | |
| Ambient temperature | As specified in drive specification (-10°C ... 40°C) | |
| Storing temperature | As specified in drive specification (-40°C ... 70°C) | |
| Humidity | 0-95%, non-condensing, corrosive | |
| Vibration and electrical safety | EN 61800-5-1 (2007) 5... 15.8 Hz 1mm (peak) 15.8 ...150 Hz 1 G | |
| Emission | C2 level, EN 61800-3 (2004) | |
| Immunity | C2 level, EN 61800-3 (2004) | |
| CAN Interface | Isolation | 2500 V rms isolation with a less than 10-ns propagation delay |
| | Protection | ±8kV ESD IEC 61000-4-2 Contact Discharge ±80V Fault Protection greater than ±12V common Mode Range |

4.2 NEW FEATURES

The following table shows new major features that are added in the OPTE6 CANOpen option board's firmware versions.

Table 52. OPTE6 CANopen firmware versions

| New features | Firmware version |
|--|------------------|
| <ul style="list-style-type: none"> Support for VACON® NXP drive. Support for Fast Communication and 16 process data when installed to NXP drive. See details in Chapter 14. APPENDIX E - FIELDBUS OPTION BOARD COMMUNICATION. | V010 |
| <ul style="list-style-type: none"> Support for modifiable CANopen parameters and monitor values via keypad in VACON® 100 product family. CANopen parameters can also be stored and restored to/from keypad or PC tool. Functionality requires the following or newer AC drive control firmware version: <ul style="list-style-type: none"> VACON® 100 INDUSTRIAL and 100 X: FW0072V029 VACON® 100 FLOW: FW0159V018 Support for CiA-420 EUROMAP extruder and puller. Note that this requires special AC drive application. | V009 |
| <ul style="list-style-type: none"> Support for new 70CVB01605 hardware. Firmware can be used also with older 70CVB01124 hardware. | V008 |
| <ul style="list-style-type: none"> Miscellaneous improvements | V007 |
| <ul style="list-style-type: none"> Miscellaneous improvements | V006 |
| <ul style="list-style-type: none"> Support for VACON® 100 INDUSTRIAL, VACON® 100 FLOW, VACON® 100 X, VACON® 20 X/CP and VACON® 20 drives. | V005 |

4.3 CAN CABLE

The recommended cables for installations are 4 wires twisted and a shielded cable with an impedance of 120 Ohm. The network topology is a 2-wire bus line that is terminated at both ends by resistors representing the characteristic impedance of the bus line. The typical CAN cable impedance is 120 Ohm, and so for the termination resistors of ~120 Ohm must be used. For long networks a higher resistor value must be used (150-300 Ohm).

Table 53. Bus parameter relation to cable length

| Cable length | Max bit rate [kbit/s] | Max cable resistance [mΩ/m] |
|--------------|-----------------------|-----------------------------|
| 0-40 m | 1000 | Max 70 |
| 100 m | 500 | <60 |
| 500 m | 100 | <40 |
| 1 km | 50 | <26 |

4.3.1 ISOLATED GROUND CONNECTION

The OPTE6 option board is galvanically isolated. In CANopen networks that are completely galvanically isolated the CAN ground signal is carried in the cable line. It is connected at only one point into common ground potential. If one CAN device with not galvanically isolated interface is connected to the network, the potential for isolated CAN ground is given. Therefore only one device with not galvanically isolated interface may be connected to the network.

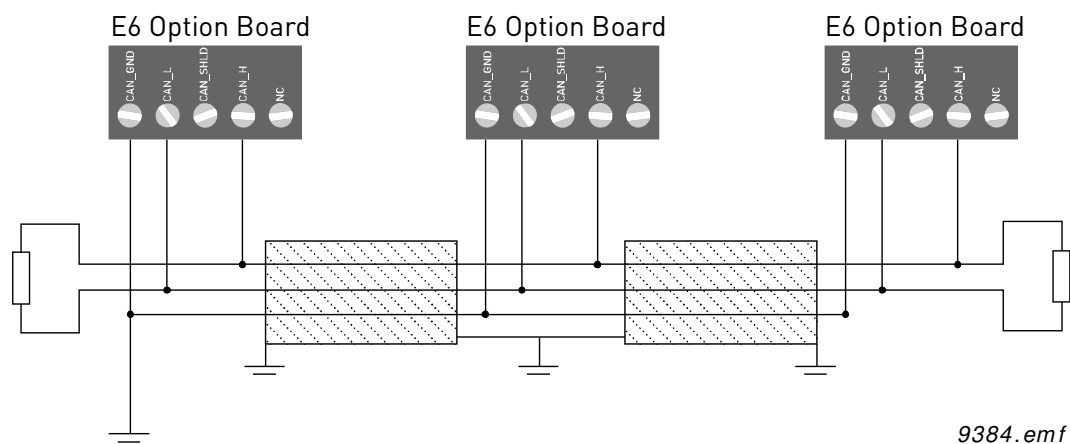


Figure 3. Completely isolated nodes

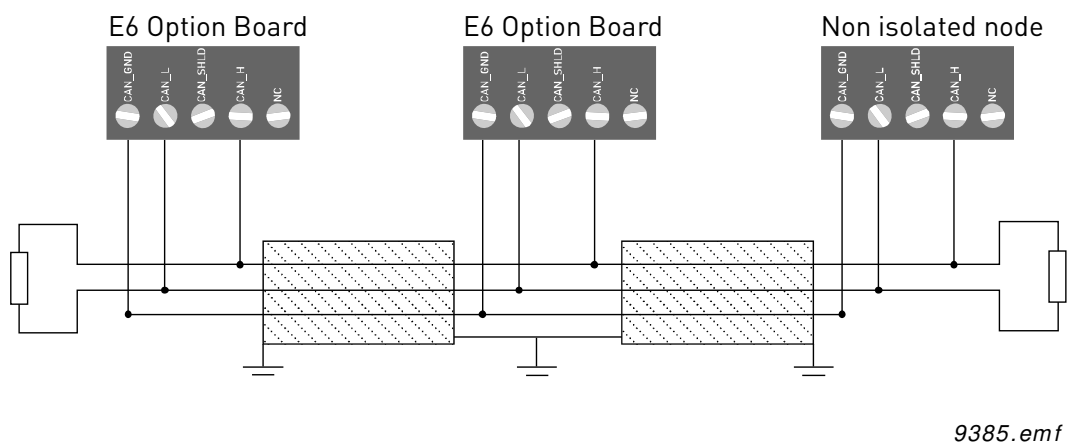


Figure 4. CAN network with one non-isolated node

4.3.2 RECOMMENDED CABLE

For all OPTe6 installations the use of 4-wire cable is recommended. 4 wires enable the connection of isolated digital grounds with nodes.

VACON® recommends the following cable:

UNITRONIC® BUS CAN FD P

Colour-coded in accordance with DIN 47100



Figure 5. Recommended cable

Table 54. Cable thickness, length and baud rate relation

| Bit rate | Min cable thickness | | | |
|---------------------|---------------------|------|-----|-----|
| 1 Mbit/s | 0.25 | | | |
| 500 kbit/s | 0.25 | 0.34 | | |
| 250 kbit/s | 0.25 | 0.34 | 0.6 | |
| 125 kbit/s | 0.25 | 0.34 | 0.6 | |
| 100 kbit/s | 0.25 | 0.34 | 0.6 | 0.6 |
| 50 kbit/s | 0.25 | 0.34 | 0.6 | 0.6 |
| Cable length | 25 | 100 | 250 | 500 |

5. OPTE6 LAYOUT AND CONNECTIONS

5.1 LAYOUT AND CONNECTIONS

OPTE6 has two different hardware revisions with slightly different layout. Layout is different in LED arrangement and termination resistor orientation.

The two hardware revisions are marked with different product codes, and this product code can be located from the sticker on the top side (location marked in Figure 6).

The two hardware revisions are named 70CVB01605 and 70CVB01124.

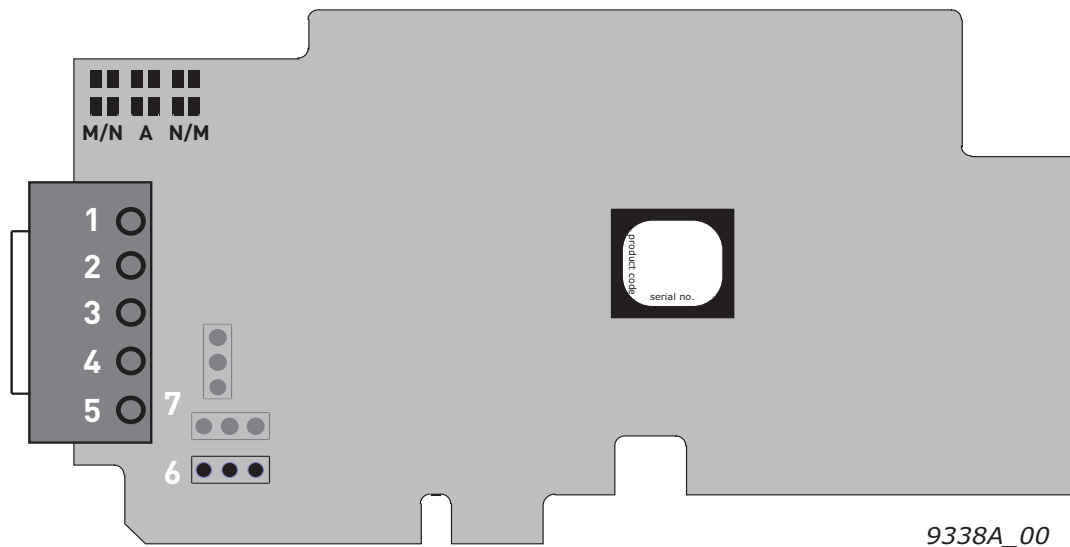


Figure 6. OPTE6 board layout

- 1 = CAN GND (isolated digital ground reference)
- 2 = CAN L
- 3 = SHIELD (shield connector)
- 4 = CAN H
- 5 = NC (No connection)
- 6 = Grounding option jumper
- 7 = Bus termination resistor

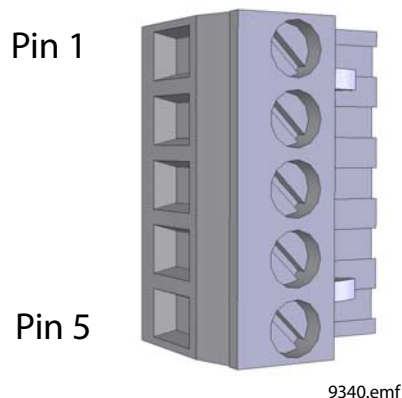


Figure 7. CAN connector

CAN connector pinout

| | |
|---|--|
| | |
| 1 | CAN GND, isolated digital ground reference |
| 2 | CAN LO |
| 3 | Shield connector |
| 4 | CAN HI |
| 5 | No connection |

5.2 LED INDICATIONS

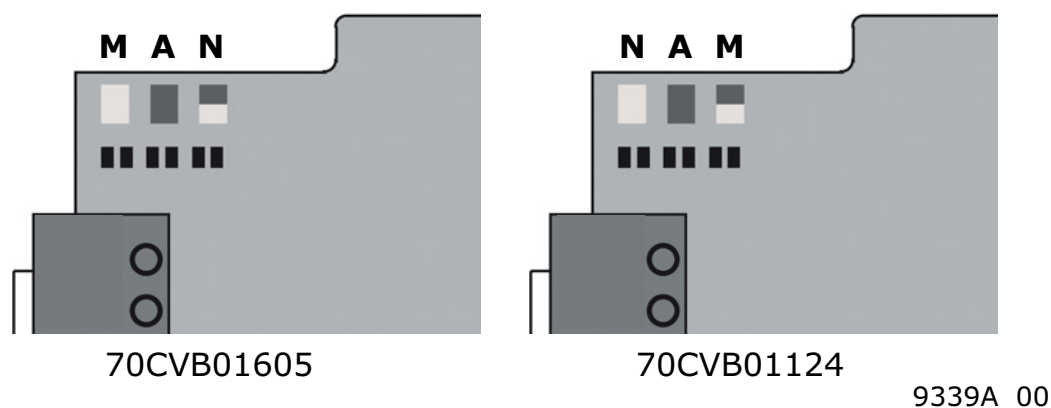


Figure 8. LED indicators

M = CANopen run led

A = CANopen err led

N = Board status

CANopen run led (green)

| | |
|--------------|---|
| | |
| Blinking | The CANopen device is in the pre-operational state. |
| Single flash | The CANopen device is in the stopped state. |
| ON | The CANopen device is in the operational state. |

CANopen err led (red)

| | |
|-----------------|---|
| OFF | No error |
| Blinking | Invalid configuration |
| Single flash | At least one of the error counters of the CAN controller has reached or exceeded the warning level (too many error frames). |
| Double flash | A guard event (NMT slave or NMT master) or a heartbeat event (heartbeat consumer) has occurred. |
| Quadruple flash | An expected PDO was not received before the event timer elapsed. |
| ON | The CAN controller is bus-off. |

Board status led (green)

| | |
|--------------------|---|
| | |
| OFF | Option board is not activated. |
| ON | Option board is in initialization state, waiting activation command from the AC drive. |
| Blinking (once/1s) | Option board is activated and in RUN state. Option board is ready for external communication. |

5.3 JUMPERS

The termination resistor jumper location differs on used hardware version. The jumper locations can be seen from figure below.

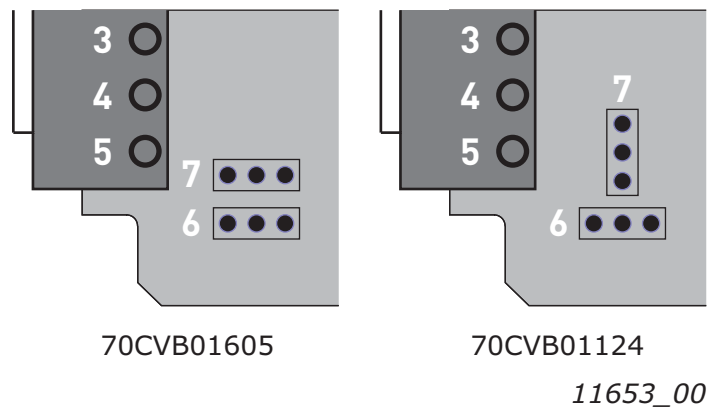


Figure 9. Jumper locations

The jumper (7) settings for the CAN bus termination resistor are shown in the figure below.

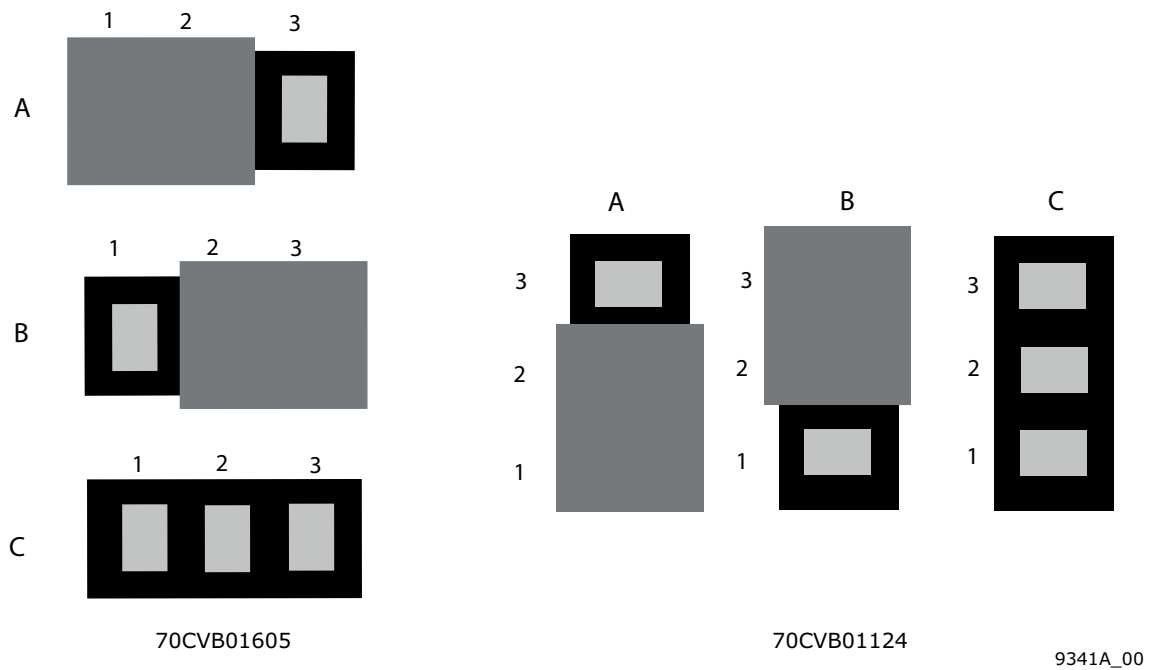


Figure 10. CAN bus termination jumper

A = Termination resistor 120 Ohm connected

B = Termination resistor is not connected to the CAN bus. (Factory default setting)

C = Termination resistor is not connected to the CAN bus

The jumper (6) settings for the CAN cable shield grounding are shown in the figure below.

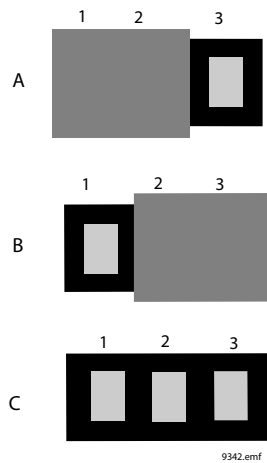


Figure 11. CAN shield grounding option

A = CAN connector pin 3 (shield) connected to the drive chassis with a high-impedance RC circuit. Recommended option when equipotential bonding is poor.

B = CAN connector pin 3 (shield) connected directly into the drive chassis. Recommended option when equipotential bonding is good. (Factory default setting)

C = CAN connector pin 3 is not connected.

6. INSTALLATION

The VACON® OPTE6 CANopen option board can be used with the following VACON® AC drives.

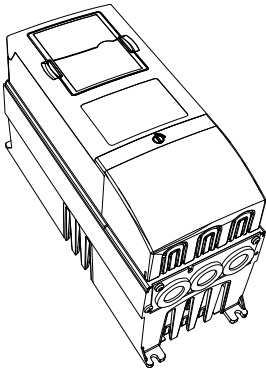
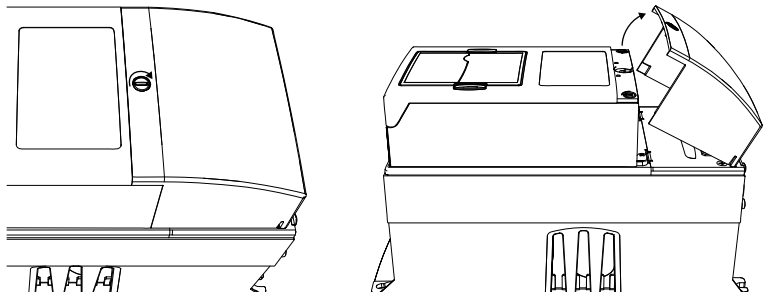
Table 55. OPTE6 usage in different VACON® AC drives

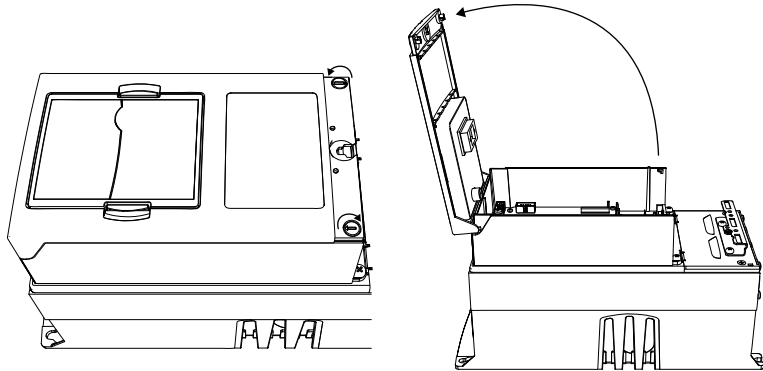
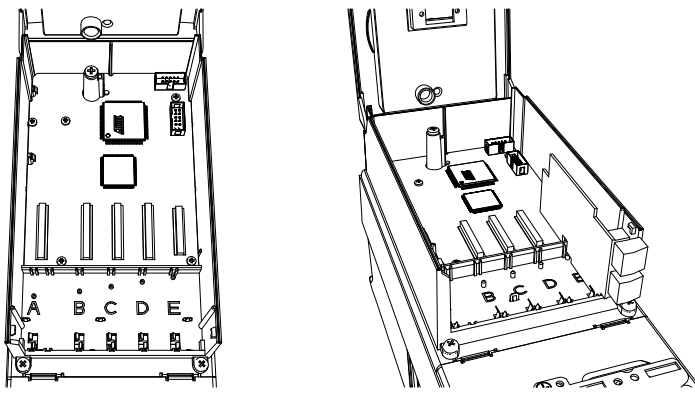
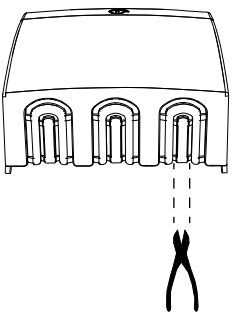
| AC drive | Slots | From AC drive SW version on | From OPTE6 SW version on |
|---------------------------------|-------|-----------------------------|--------------------------|
| VACON® NXP | D, E | NXP00002V196 | V010 |
| VACON® 100 INDUSTRIAL and 100 X | D, E | FW0072V015 | V007 |
| VACON® 100 FLOW | D, E | FW0159V011 | V007 |
| VACON® 20 | - | FW0107V011 | V007 |
| VACON® 20 X and CP | - | FW0117V007 | V007 |

6.1 INSTALLATION IN VACON® NX

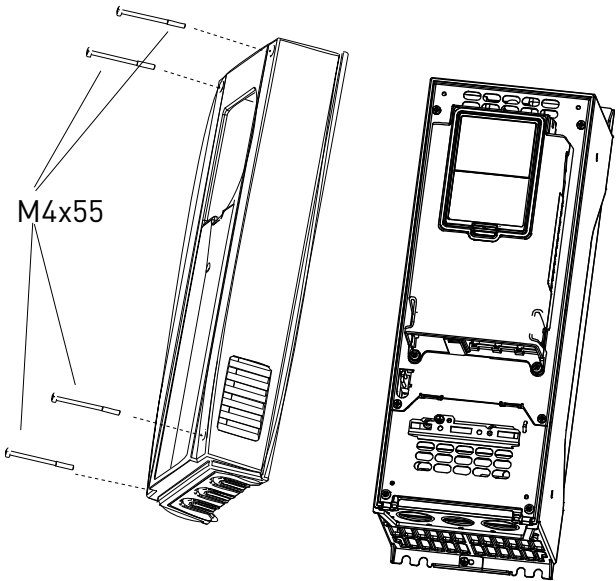



Make sure that the AC drive **is switched off** before an option or fieldbus board is changed or added!

| | |
|---|---|
| 1 | <p>VACON® NX AC drive.</p>  |
| 2 | <p>Remove the cable cover.</p>  |

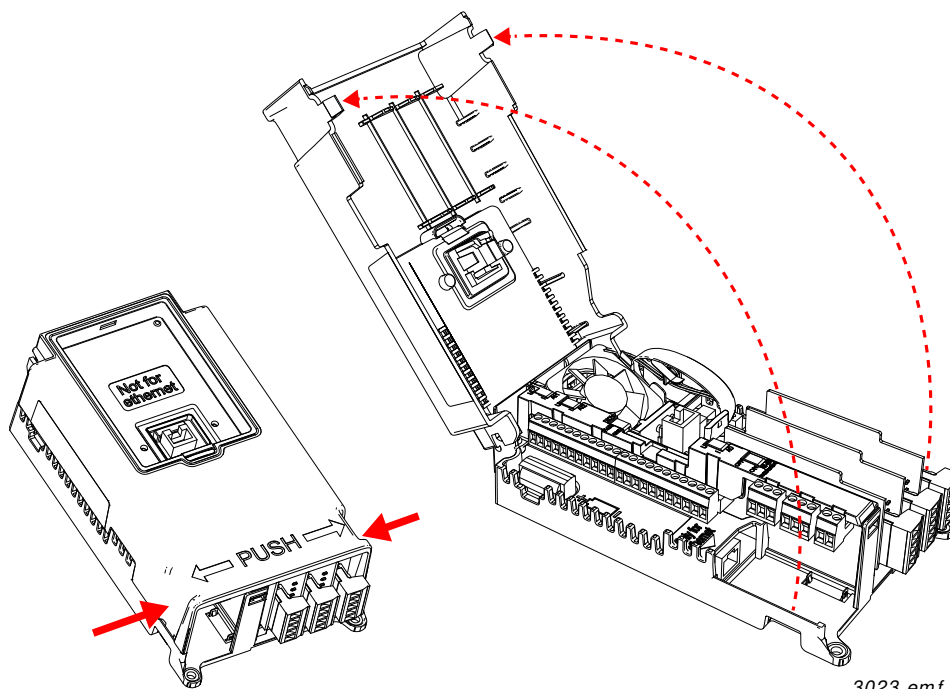
| | |
|-----------------|--|
| <p>3</p> | <p>Open the cover of the control unit.</p>  |
| <p>4</p> | <p>Install the OPTE9 Option Board in slot D or E on the control board of the AC drive. Make sure that the grounding plate fits tightly in the clamp.</p>  |
| <p>5</p> | <p>Make a sufficiently wide opening for your cable by cutting the grid as wide as necessary.</p>  |
| <p>6</p> | <p>Close the cover of the control unit and the cable cover.</p> |

6.2 INSTALLATION IN VACON® 100

| | |
|---|--|
| 1 | <p>Open the cover of the AC drive.</p>  <p style="text-align: right; font-size: small;">9174.emf</p> |
|  DANGER | <p>The relay outputs and other I/O-terminals may have a dangerous control voltage present even when AC drive is disconnected from mains.</p> |

2

Open the inner cover to reveal the option board slots **(C,D,E)**. See Figure below.

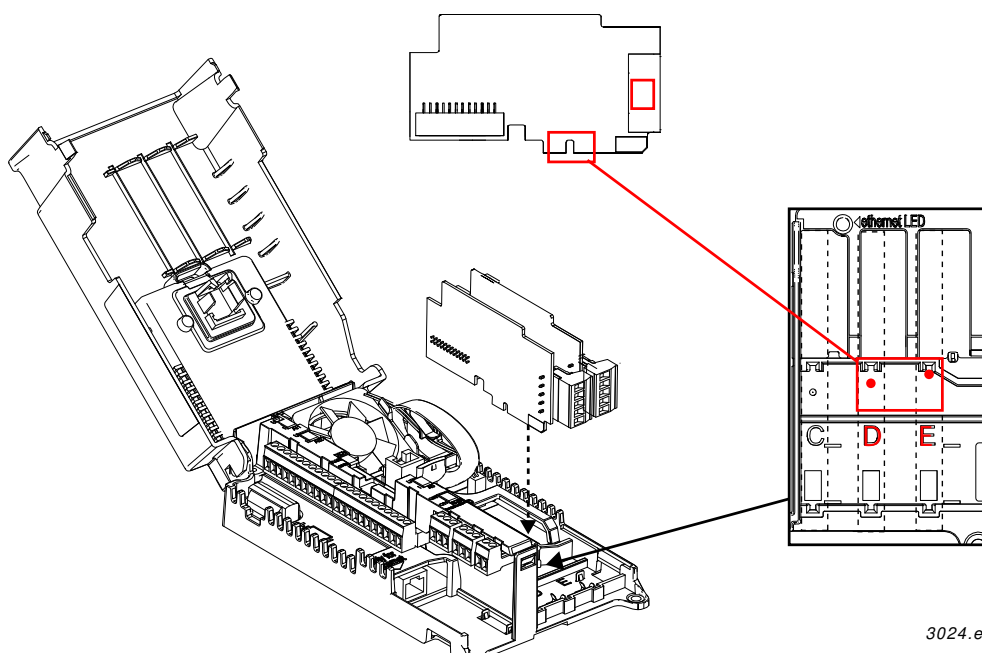


3023.emf

3

Install the fieldbus board into slot **D** or **E**. See figure below.

NOTE: Incompatible boards cannot be installed on the AC drive. Compatible boards have a slot coding that enable the placing of the board.

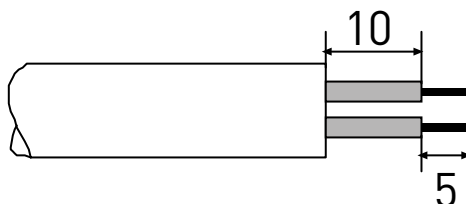


3024.emf

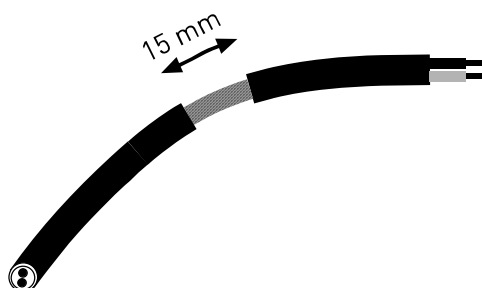
4

Strip about 15 mm of the fieldbus cable (see specification in ch. 4.3) and cut off the grey cable shield. Remember to do this for both bus cables (except for the last device).

Leave no more than 10 mm of the cable outside the terminal block and strip the cables at about 5 mm to fit in the terminals. See picture below.



Also strip the cable now at such a distance from the terminal that you can fix it to the frame with the grounding clamp. Strip the cable at a maximum length of 15 mm. **Do not strip the aluminum cable shield!**

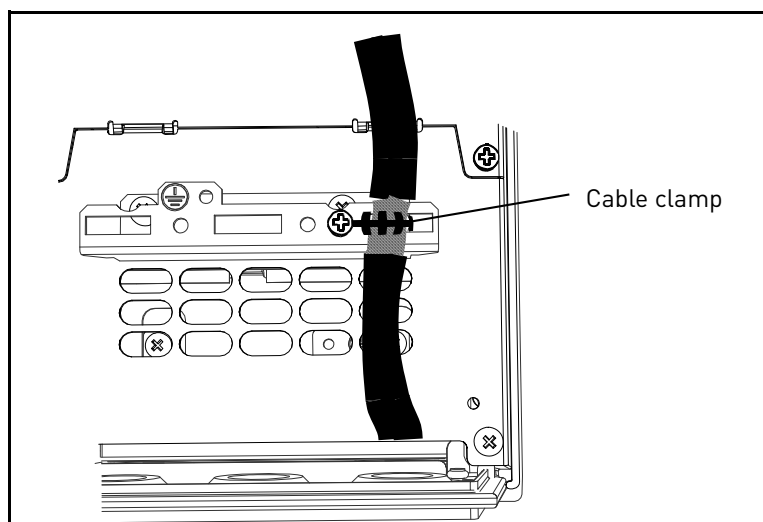
**5**

Then connect the cable to its appropriate terminals on the OPTE6 CANopen option board terminal block.

6

Using the cable clamp included in the delivery of the drive, ground the shield of the CAN cable to the frame of the AC drive.

NOTE: This can be done in all drives if there is no difference in PE potential between the drives. However, if there is PE potential difference then the shield should be connected to PE only at one point in the system. The shields of the cables shall be joint but not connected to several PE points with different potential.



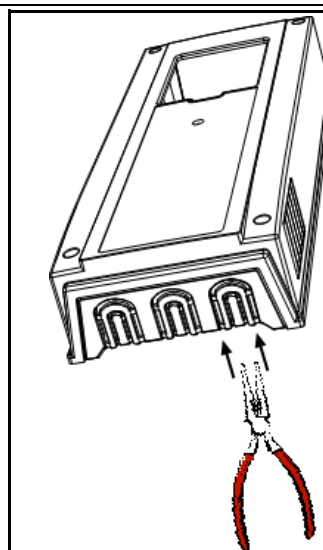
7

If the AC drive is the last device on the bus, the bus termination must be set with jumper X13 (see ch. 5.3.)

8

Unless already done for the other control cables, cut free the opening on the AC drive cover for the fieldbus cable (protection class IP21).

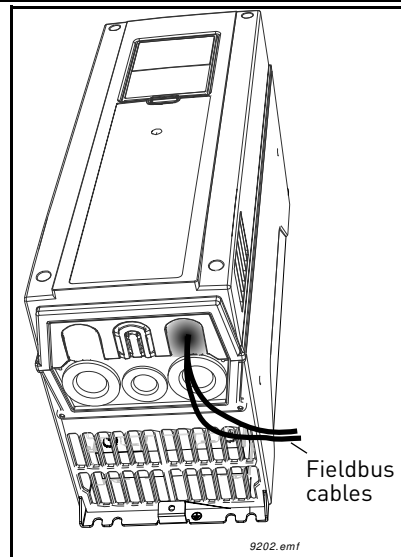
NOTE: Cut the opening on the same side you have installed the board in!



9

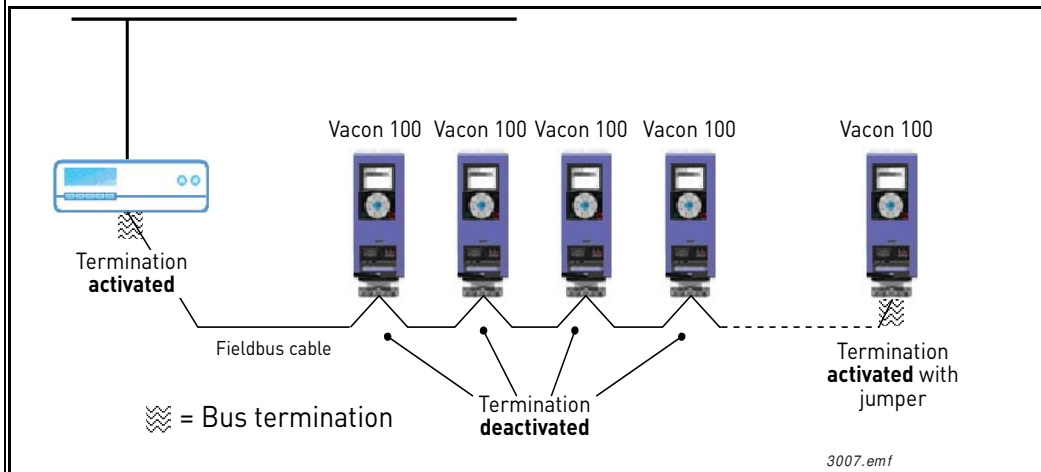
Remount the AC drive cover and run the cable as shown in picture.

NOTE: When planning the cable runs, remember to keep the distance between the fieldbus cable and the motor cable at a **minimum of 30 cm**. It is recommended to route the option board cables away from the power cables as shown in the picture.



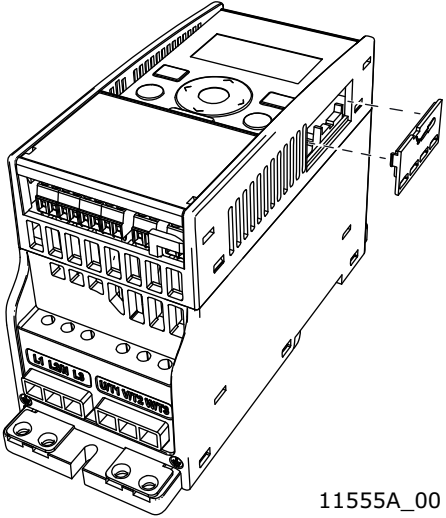
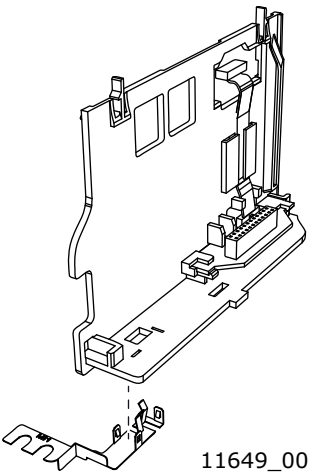
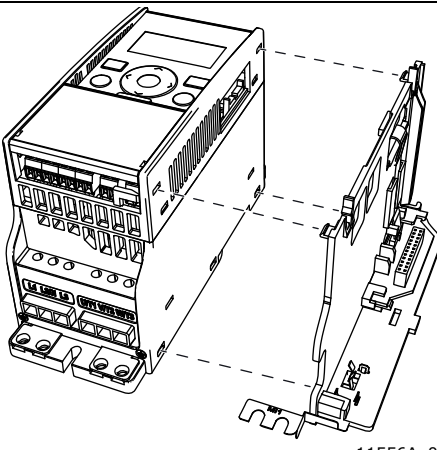
10

The bus termination must be set for the first and the last device of the fieldbus line. See picture below. See also step 7 on page 43. We recommend that the first device on the bus and, thus, terminated was the Master device.



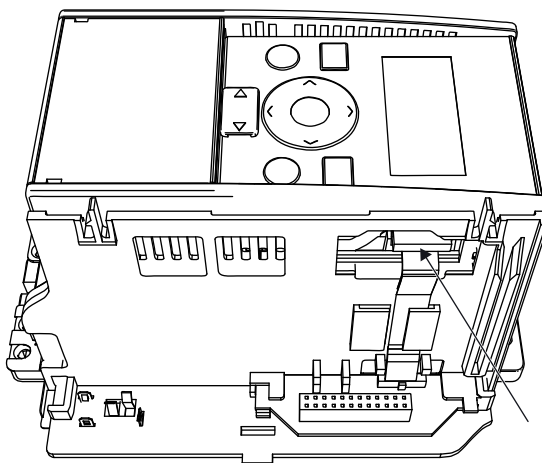
6.3 INSTALLATION IN VACON® 20

6.3.1 FRAMES MI1, MI2, MI3

| | |
|---|---|
| 1 | <p>Remove the cable connector lid from the AC drive.</p>  <p>11555A_00</p> |
| 2 | <p>Select a correct grounding plate and attach it to the option board mounting frame. The grounding plate is marked with the supported enclosure size.</p>  <p>11649_00</p> |
| 3 | <p>Attach an option board mounting frame to the AC drive.</p>  <p>11556A_0</p> |

4

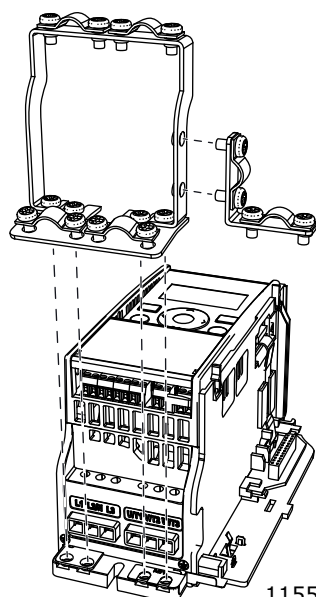
Connect the flat cable from the option board mounting frame to VACON® 20.



11557A_00

5

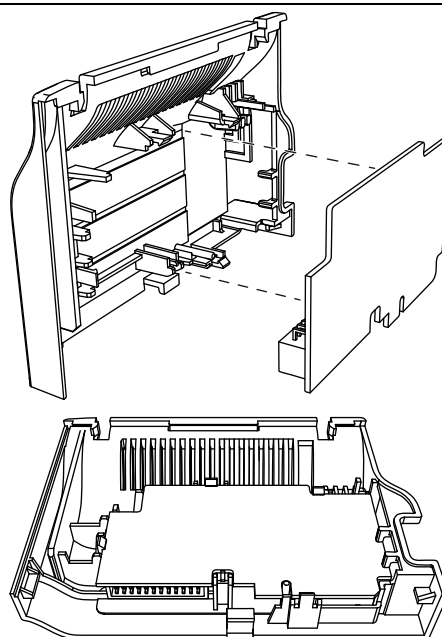
If a cable strain relief is required, attach the parts as shown in the figure.



11558A_00

6

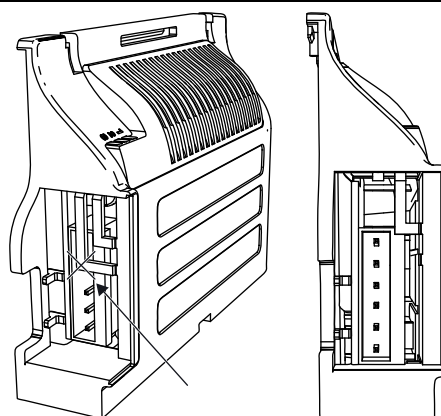
Install the option board to the option board holder. Make sure that the option board is securely fastened.



11559A_00

7

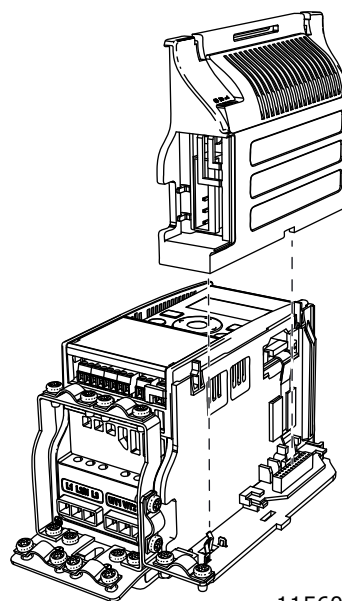
Cut free a sufficiently wide opening for the option board connector.



11650_00

8

Attach the option board cover to VACON[®] 20. Attach the strain relief cable clamp with screws if needed.



11560A_00

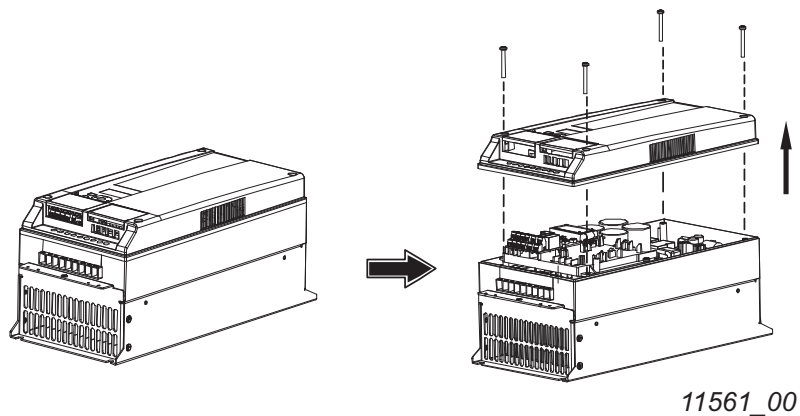
6.3.2 FRAMES MI4, MI5



Make sure power is disconnected before opening the VACON® 20 cover.

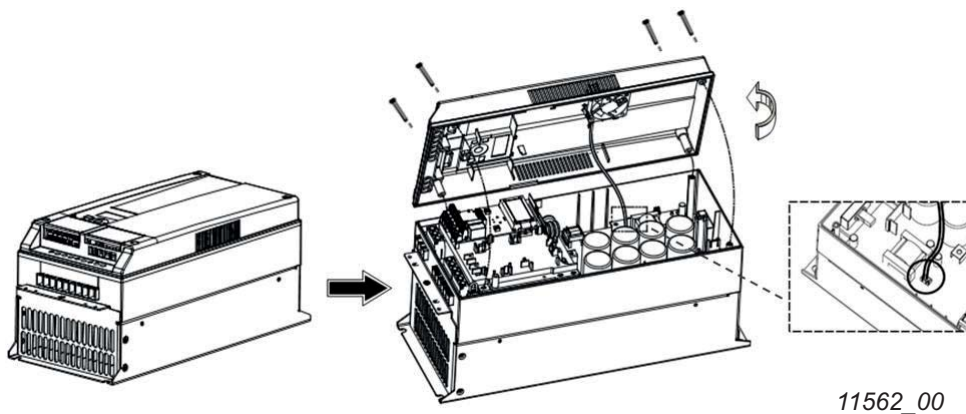
1

1a: For MI4: Open the cover.



11561_00

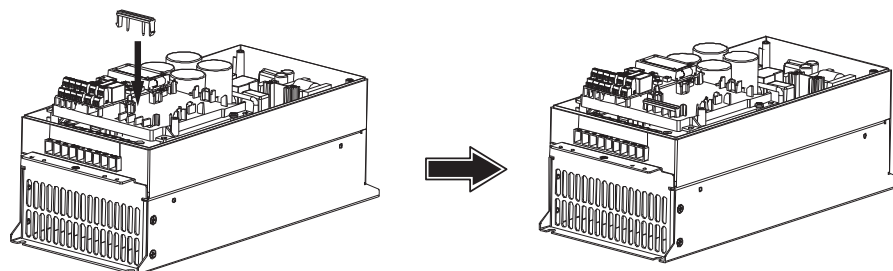
1b: For MI5: Open the cover and release the fan connector.



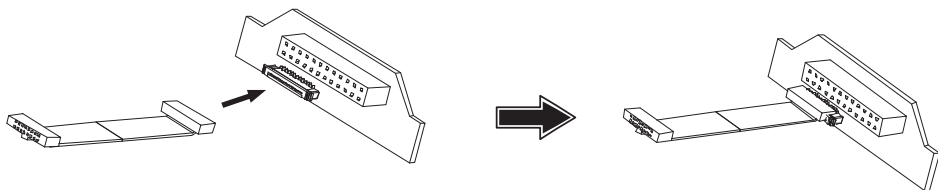
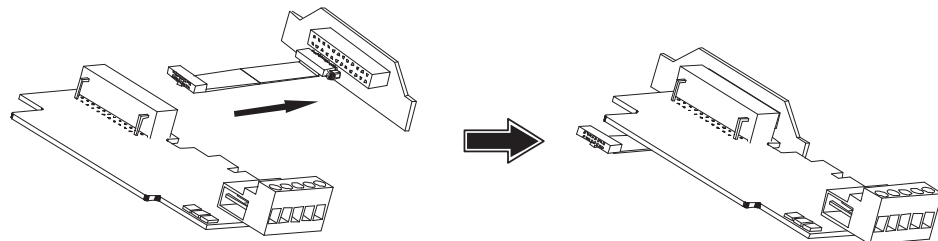
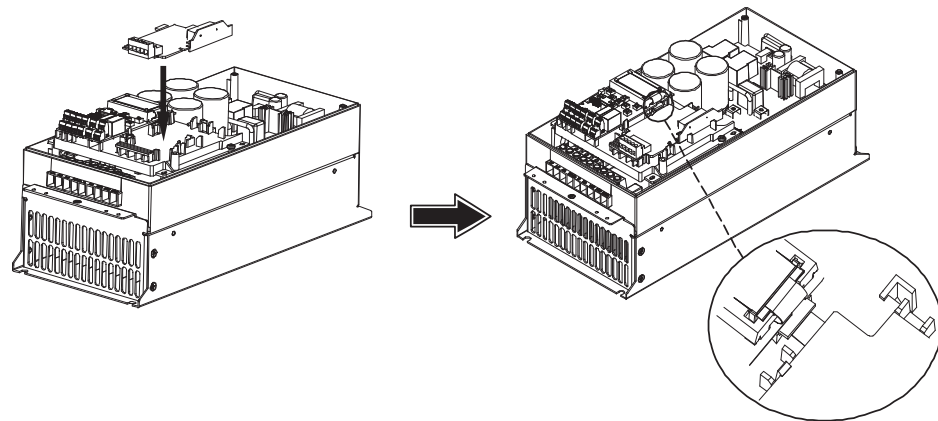
11562_00

2

Attach the option board support.

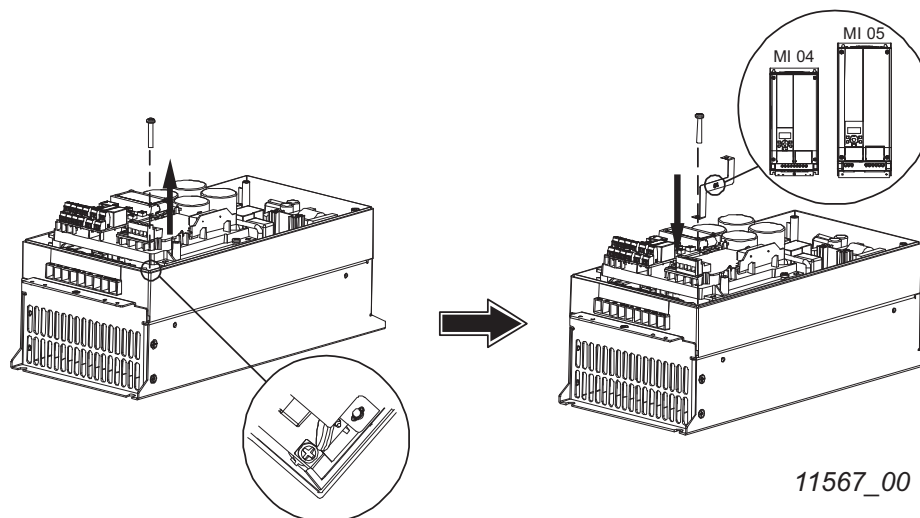


11563_00

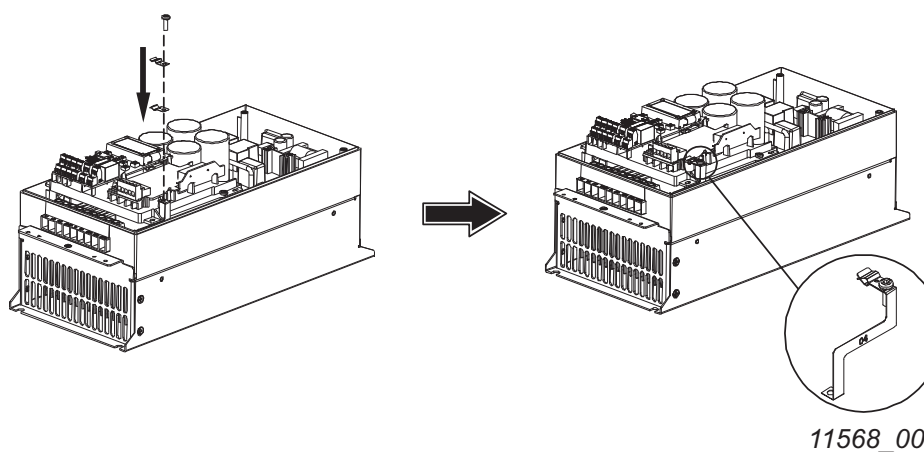
| | |
|-----------------|--|
| <p>3</p> | <p>Connect the flex cable to option board connector PCB.</p>  <p>11564_00</p> |
| <p>4</p> | <p>Connect the option board to connector PCB.</p>  <p>11565_00</p> |
| <p>5</p> | <p>Attach the option board with connector PCB to VACON® 20 and connect the flex cable.</p>  <p>11566_00</p> |

6

Attach a suitable grounding plate to VACON® 20. The grounding plate is marked with supported enclosure size.

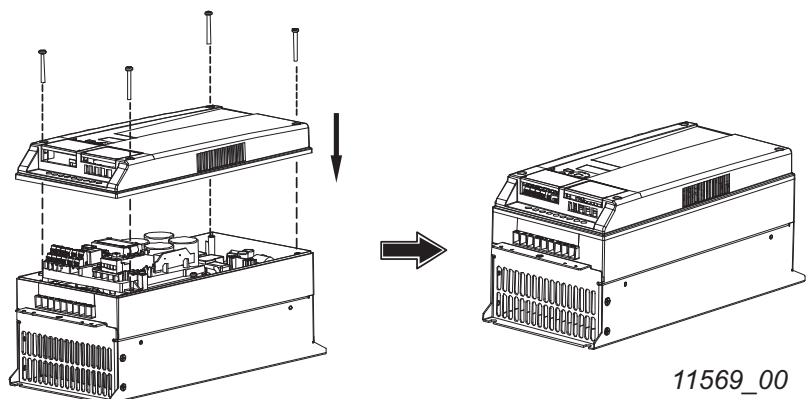
**7**

Assemble a clamp on top of the grounding plate on both sides of the option board.

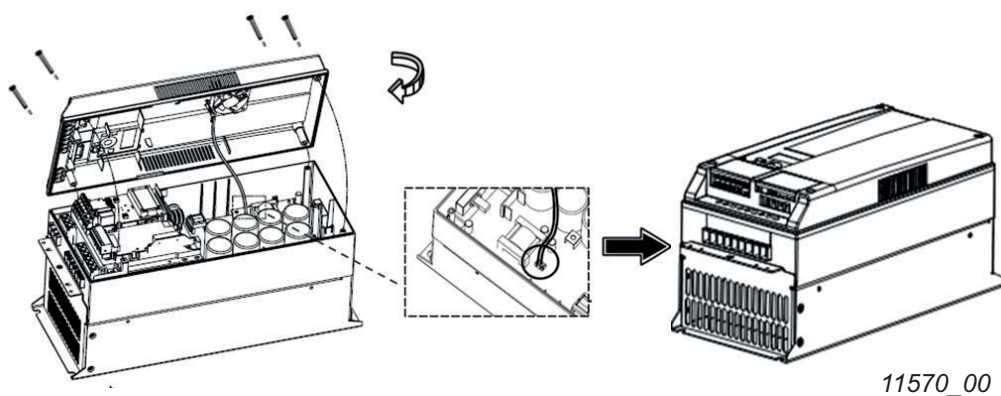


8

8a: For MI4: Close the cover.



8b: For MI5: Remount the fan connector and close the cover.



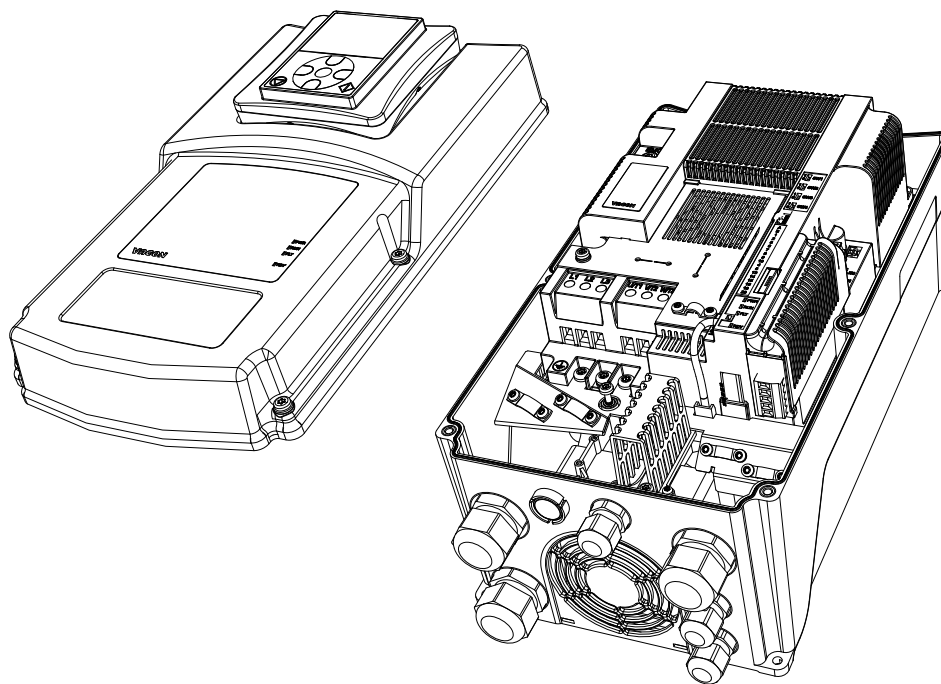
6.4 INSTALLATION IN VACON® 20 X AND 20 CP



Do not add or replace option boards or fieldbus boards on an AC drive with the power switched on. This may damage the boards.

1

Open the cover of the drive.



11643_00

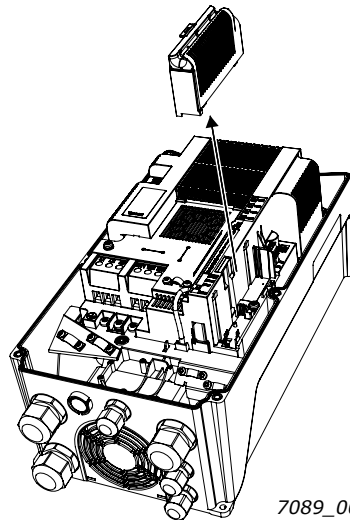
MU3 example



The relay outputs and other I/O-terminals may have a dangerous control voltage present even when the drive is disconnected from mains.

2

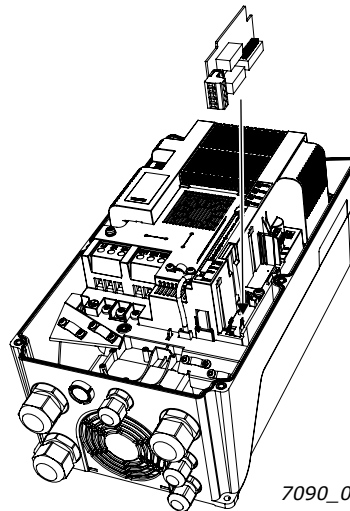
Remove the option slot cover.



7089_00

3

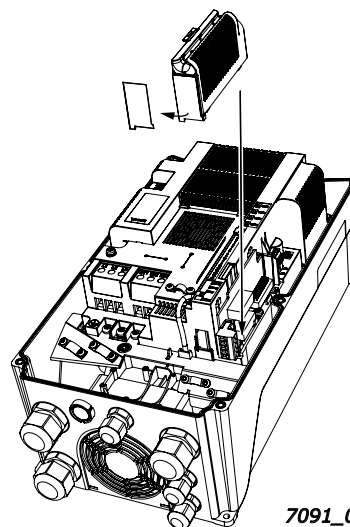
Install the option board into the slot as shown in the figure.



7090_00

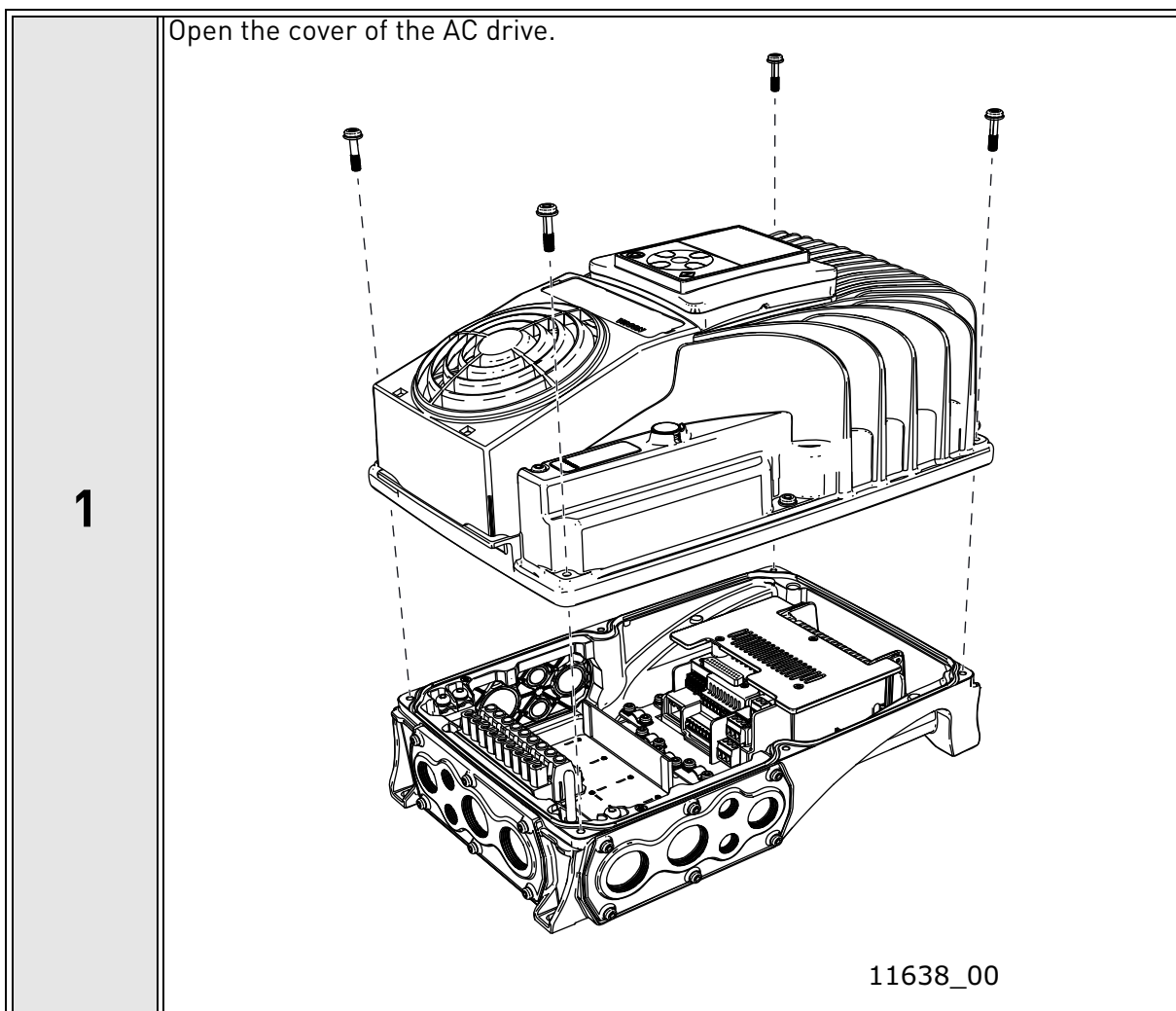
4

Mount the option slot cover. Remove the plastic opening for the option board terminals.



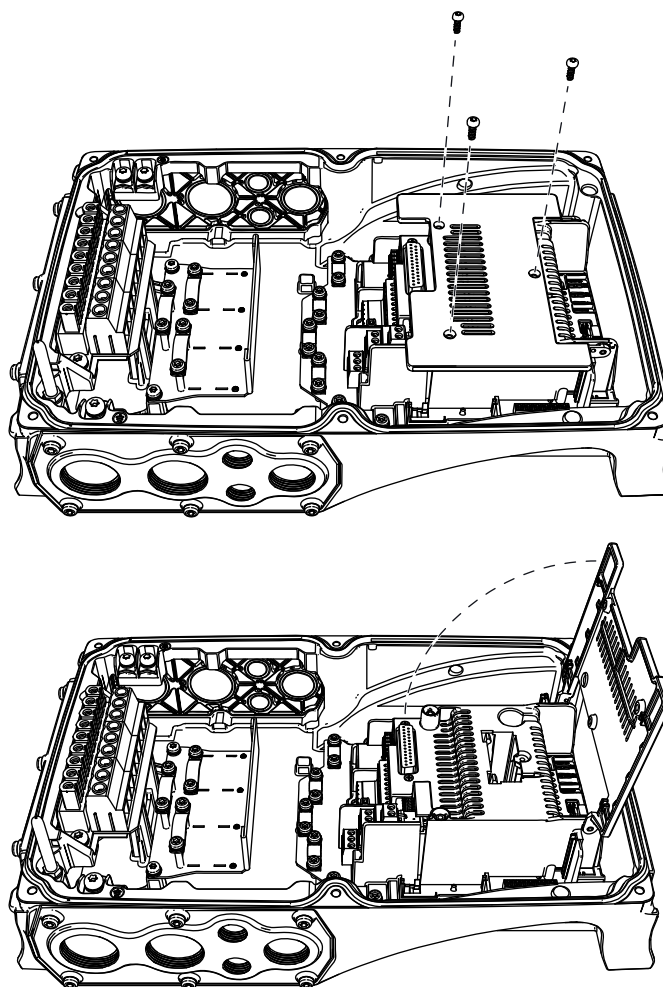
7091_00

6.5 INSTALLATION IN VACON® 100 X (FRAMES MM4-MM6)



2

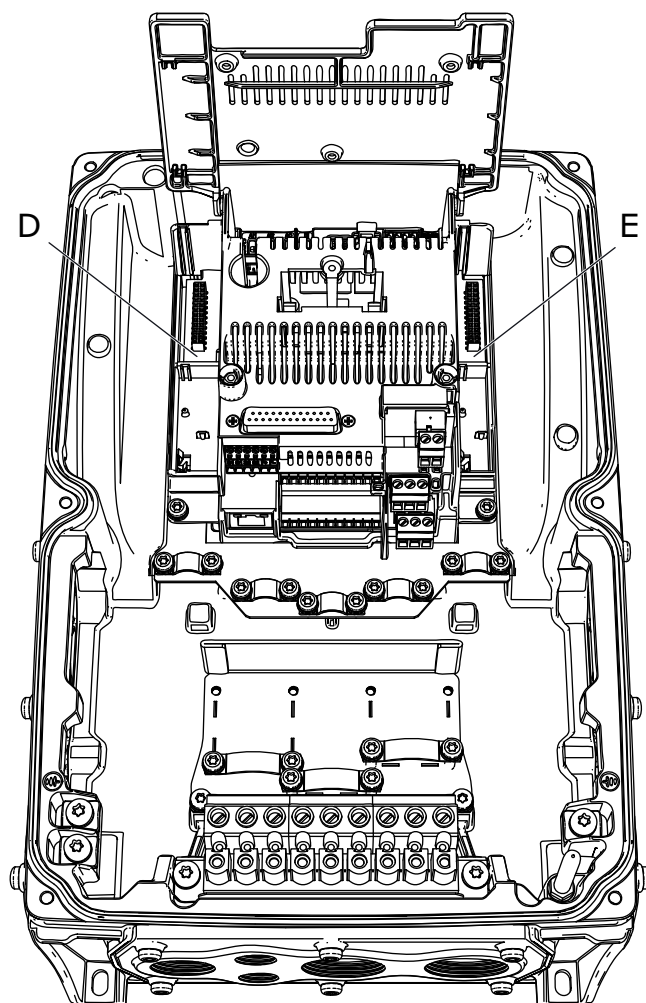
To get access to the option board slots, remove the screws and open the cover of the control unit.



11639_00

3

Install the option board into the correct slot, D or E.



11640_00

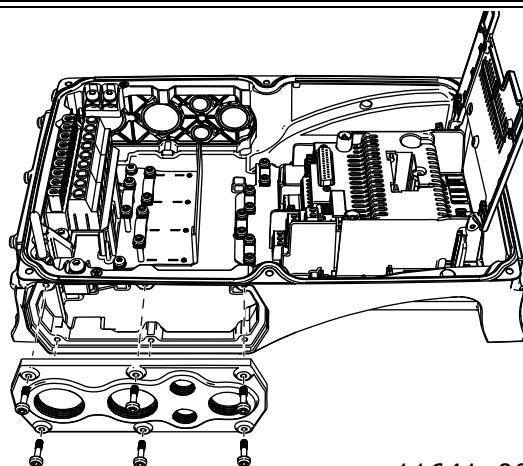
4

Close the option board cover.

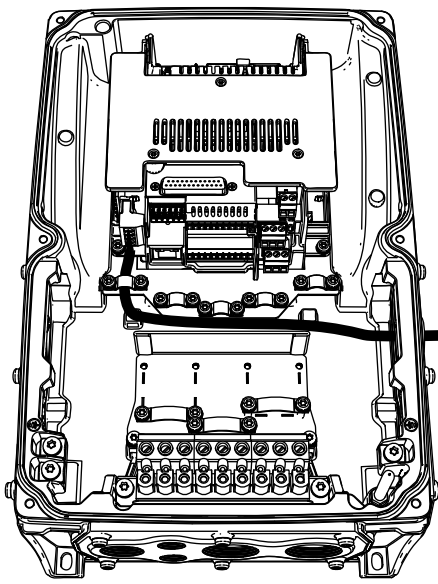
5

Remove the cable entry plate. If you installed the option board in the slot D, use the cable entry plate on the right side. If you installed the option board in the slot E, use the cable entry plate on the left side.

NOTE! The cable entry plate at the bottom of the drive is used only for mains and motor cables.



11641_00

| | |
|---|--|
| 6 | Open the necessary holes in the cable entry plate. Do not open the other holes. See the VACON® 100 X Installation Manual for the dimensions of the holes. |
| 7 | <p>Attach a cable gland on the hole in the cable entry plate. Pull the fieldbus cable through the hole.</p> <p>NOTE! The fieldbus cable must go through the correct cable entry plate to avoid going near the motor cable.</p>  <p>11642_00</p> |
| 8 | Put the cable entry plate back. |
| 9 | Close the cover of the AC drive. |

7. COMMISSIONING

This chapter describes how to commission the OPTE6 board for use. For instructions on how to set-up the AC drive to be controlled over fieldbus, refer to Chapter 11 APPENDIX B - FIEDLBUS PARAMETRIZATION.

7.1 OPTE6 PANEL PARAMETERS

Different number of panel parameters are visible depending on the used AC drive and version.

VACON® NX family option board parameters are found under "Expander boards". Path depends on used slot where OPTE6 is installed (7.4.1 or 7.5.1).

VACON® 100 family option board parameters are found under "I/O and Hardware". Path depends on used slot where OPTE6 is installed (5.3.1 or 5.4.1).

VACON® 20 family option board parameters are under SYS menu.

Table 56. OPTE6 board basic parameters

| # | Parameter | Min | Max | Default | Description |
|----|------------------|-----|-------|---------|---|
| 1 | Node ID | 1 | 127 | 1 | Network-wide unique identifier for each CANopen device |
| 2 | Baud Rate | 3 | 8 | 5 | Data signalling rate. Should be the same on each node in the same bus. See Table 57 |
| 3 | Operate Mode | 1 | 4 | 1 | Selection between OPTE6 operate modes. See Table 58 |
| 4* | Comm. Timeout | 0 | 65535 | 1 | Timeout in seconds for CAN communication faults (PASSIVE, BUS-OFF) |
| 5* | Restore from Set | 0 | 1 | 0 | 0 -> 1 All CANopen parameters are reset to default values based on Operate Mode. |
| 6* | Mode | 0 | 1 | 0 | Set OPTE6 mode. See Table 59 |

* From version V26 (VACON® 100 INDUSTRIAL) and V18 (VACON® 100 FLOW) and V196 (VACON® NXP).

Table 57. OPTE6 Baud Rate settings

| Value | Bit rate |
|-------|-------------------------|
| 1 | 10 kbps (not supported) |
| 2 | 20 kbps (not supported) |
| 3 | 50 kbps |
| 4 | 100 kbps |
| 5 | 125 kbps |
| 6 | 250 kbps |
| 7 | 500 kbps |
| 8 | 1000 kbps |

Table 58. OPTE6 Operate Mode settings

| Value | Name | Description |
|-------|---------------|--|
| 1 | Drive Profile | CiA-402 drive profile, velocity mode active |
| 2 | Bypass | VACON® specific bypass mode active |
| 3 | Puller* | CiA-420 EUROMAP profile, puller mode active |
| 4 | Co-Extruder* | CiA-420 EUROMAP profile, co-extruder mode active |

*Restricted availability, requires use of special application

Baud Rate: used in CANopen communication. Regardless of the selected bitrate, the bit sample timing is set as close as possible to 87.5 % according to the CANopen specification.

Operate Mode: is used to change the modes of operation between standardized profile modes and VACON® specific Bypass mode.

Different operate modes activate different device profile segment objects. Accessing e.g. CiA-402 velocity profile objects in Bypass mode is not possible. See Chapter 8.2 Velocity mode and Chapter 8.3 Bypass mode.

Comm. Timeout: Fieldbus communication timeout is time in seconds after a fault is created from CAN bus communication errors (PASSIVE, BUS-OFF). Setting this value to 0 disables CAN bus errors from triggering a fault in AC drive. This does not affect other CANopen faults, e.g. heartbeat consumer times. This value is the same as object 0x2004 - Communication timeout. See conditions when fieldbus communication timeout is activated in Chapter 9.2 Fieldbus timeout fault (F53).

Table 59. OPTE6 Mode settings

| Value | Description | Details |
|-------|-------------|--|
| 1 | Normal | |
| 2 | Pre V008 | Anyparameter service behaves same as version 7 and earlier in VACON® 100 family devices. See Chapter 8.5 for more details. |

Mode: compatibility mode(s) can be activated using this parameter. This parameter is the same as object 0x2005 - Mode.

7.1.1 OPTE6 ADDITIONAL PANEL PARAMETERS

From VACON® 100 (INDUSTRIAL V027 and FLOW V018) and OPTE6 V009 firmware version forwards, CANopen bus communication parameters are stored and can also be modified directly from panel parameters. The stored parameters are modifiable and stored under the "Parameters" menu and these parameters are restored after power cycle. The active settings are visible under the "Monitor" menu.

These parameters are separated into sub-folders for easier modification. It is recommended to use VACON® Live PC-tool to modify the parameters.

Table 60. OPTE6 board CANopen object parameters under General folder

| Parameter | Default | Reference |
|-------------------------------|----------------------|-----------|
| Sub menu: General | | |
| COB-ID SYNC | 0x80 | 3.7.4 |
| Guard Time | 0 | 3.7.5 |
| Life Time Factor | 0 | 3.7.6 |
| COB-ID EMCY | 0x80 + Node ID | 3.7.7 |
| Sync Counter Ovf. | 0 | 3.7.11 |
| Error Behaviour | 0 | 3.7.12 |
| Sub menu: Heartbeat | | |
| Producer Time | 0 | 3.7.9 |
| Consumer Time 1 | 0 | 3.7.8 |
| Consumer Time 2 | 0 | |
| Consumer Time 3 | 0 | |
| Consumer Time 4 | 0 | |
| Consumer Time 5 | 0 | |
| Consumer Time 6 | 0 | |
| Consumer Time 7 | 0 | |
| Consumer Time 8 | 0 | |
| Sub menu: RPD01 Communication | | |
| COB-ID | 0x200 + Node ID | 3.5.2 |
| Transmission Type | 0xFE | 3.5.3 |
| Inhibit Time | 0 | |
| Event Timer | 0 | |
| Sub menu: RPD02 Communication | | |
| COB-ID | 0x80000300 + Node ID | 3.5.2 |
| Transmission Type | 0xFE | 3.5.3 |
| Inhibit Time | 0 | |
| Event Timer | 0 | |
| Sub menu:RPD03 Communication | | |
| COB-ID | 0x80000400 + Node ID | 3.5.2 |
| Transmission Type | 0xFE | 3.5.3 |
| Inhibit Time | 0 | |
| Event Timer | 0 | |
| Sub menu: RPD01 Mapping | | |

Table 60. OPTE6 board CANopen object parameters under General folder

| Parameter | Default | Reference |
|----------------------------------|----------------------|-----------|
| Number of Entries | 2 | 3.5.4 |
| 1 st Application Obj. | 0x60400010 | |
| 2 nd Application Obj. | 0x60420010 | |
| 3 rd Application Obj. | 0 | |
| 4 th Application Obj. | 0 | |
| Sub menu: RPD02 Mapping | | |
| Number of Entries | 4 | 3.5.4 |
| 1 st Application Obj. | 0x20000410 | |
| 2 nd Application Obj. | 0x20000510 | |
| 3 rd Application Obj. | 0x20000610 | |
| 4 th Application Obj. | 0x20000710 | |
| Sub menu: RPD03 Mapping | | |
| Number of Entries | 4 | 3.5.4 |
| 1 st Application Obj. | 0x20000810 | |
| 2 nd Application Obj. | 0x20000910 | |
| 3 rd Application Obj. | 0x20000A10 | |
| 4 th Application Obj. | 0x20000B10 | |
| Sub menu: TPD01 Communication | | |
| COB-ID | 0x180 + Node ID | 3.5.2 |
| Transmission Type | 0xFE | 3.5.3 |
| Inhibit Time | 100 | |
| Event Timer | 0 | |
| Sync Start Value | 0 | |
| Sub menu: TPD02 Communication | | |
| COB-ID | 0x80000280 + Node ID | 3.5.2 |
| Transmission Type | 0xFE | 3.5.3 |
| Inhibit Time | 1000 | |
| Event Timer | 0 | |
| Sync Start Value | 0 | |
| Sub menu: TPD03 Communication | | |
| COB-ID | 0x80000380 + Node ID | 3.5.2 |
| Transmission Type | 0xFE | 3.5.3 |
| Inhibit Time | 1000 | |
| Event Timer | 0 | |
| Sync Start Value | 0 | |

Table 60. OPTE6 board CANopen object parameters under General folder

| Parameter | Default | Reference |
|----------------------------------|------------|-----------|
| Sub menu: TPD01 Mapping | | |
| Number of Entries | 2 | 3.5.4 |
| 1 st Application Obj. | 0x60410010 | |
| 2 nd Application Obj. | 0x60440010 | |
| 3 rd Application Obj. | 0 | |
| 4 th Application Obj. | 0 | |
| Sub menu: TPD02 Mapping | | |
| Number of Entries | 4 | 3.5.4 |
| 1 st Application Obj. | 0x20010410 | |
| 2 nd Application Obj. | 0x20010510 | |
| 3 rd Application Obj. | 0x20010610 | |
| 4 th Application Obj. | 0x20010710 | |
| Sub menu: TPD03 Mapping | | |
| Number of Entries | 4 | 3.5.4 |
| 1 st Application Obj. | 0x20010810 | |
| 2 nd Application Obj. | 0x20010910 | |
| 3 rd Application Obj. | 0x20010A10 | |
| 4 th Application Obj. | 0x20010B10 | |

Considerations when modifying the CANopen object related parameters:

- All settings written to the CANopen related panel parameters are taken into use immediately if the value is valid
- No error is given if parameter value is invalid, instead the parameter value is restored back to previous value when read back
- Writing of an invalid value (for example 0) to a PDO mapping application object is not allowed. If a mapping must be invalidated, use the number of entries parameter to set the number of used objects
- All settings written via CANopen objects (bus parameters) are not automatically stored and visible in the panel parameters. Only after giving the "Save all parameters" command (object 0x1010,1), the parameters written from bus are stored to parameters
- Parameter Error Behaviour is linked to the Communication error (object 0x1029:01)

7.1.2 PANEL PARAMETER CHANGE REACTION

Changing of a parameter can change CANopen parameterization depending on what parameter is changed and what has been previously parameterized. This chapter clarifies the changes that occur in OPTE6 while parameters are changed from panel parameters or from CANopen objects that are linked to these parameters. Note that there are some differences between OPTE6 versions.

Changing of Node ID:

V008 and earlier: No difference to any stored parameter (listed in Table 60). This means that if the COB-IDs of e.g. PDO communication parameters need to be recalculated using the new Node ID one of the following actions must be performed:

- Change operate mode parameter (or toggle back and forth)
- Restore default parameter set from bus using the 0x1011 object and sending a reset node message

V009 forwards: COB-IDs are recalculated using the new Node ID and validity of COB-ID. Therefore, the following objects are changed:

| Object | Name | New value if enabled | New value if disabled |
|-----------|----------------|----------------------|-----------------------|
| 0x1014 | COB-ID EMCY | 0x80 + Node ID | 0x80000080 + Node ID |
| 0x1400:01 | (RPD01) COB-ID | 0x200 + Node ID | 0x80000200 + Node ID |
| 0x1401:01 | (RPD02) COB-ID | 0x300 + Node ID | 0x80000300 + Node ID |
| 0x1402:01 | (RPD03) COB-ID | 0x400 + Node ID | 0x80000400 + Node ID |
| 0x1403:01 | (RPD04) COB-ID | 0x500 + Node ID | 0x80000500 + Node ID |
| 0x1404:01 | (RPD05) COB-ID | Not changed | |
| 0x1800:01 | (TPD01) COB-ID | 0x180 + Node ID | 0x80000180 + Node ID |
| 0x1801:01 | (TPD02) COB-ID | 0x280 + Node ID | 0x80000280 + Node ID |
| 0x1802:01 | (TPD03) COB-ID | 0x380 + Node ID | 0x80000380 + Node ID |
| 0x1803:01 | (TPD04) COB-ID | 0x480 + Node ID | 0x80000480 + Node ID |
| 0x1804:01 | (TPD05) COB-ID | Not changed | |

The 5th PDOs are not automatically calculated, as the CANopen standard Predefined Connection set supports 4 transmit and receive PDOs. If 4 PDOs per device is not sufficient, you must define custom COB-IDs for PDOs.

Any heartbeat consumer entry using the new Node ID is reset to default value. No other parameters are affected.

If special PDO communication settings are used, these settings are overwritten and must be re-stored. They can be restored from bus, panel or PC-tool. It is recommended to create a user set (with PC-tool) before changing the Node ID setting and restoring it after changing Node ID (and Operate Mode).

Changing of Operate Mode:

V008 and earlier: All CANopen communication parameters are restored to use operate mode specific default values.

V009 forwards: Only PDO communication and mapping parameters are restored to use mode specific default values. CANopen communication objects such as Heartbeat and Node guarding are not modified.

Using "Restore from Set" parameter:

When writing any value (other than zero) to this parameter, all CANopen communication parameters are restored to default parameters and PDO communication and mapping parameters are restored to default parameters of the currently active operate mode.

Node ID, operate mode, baud rate, communication timeout and mode parameters are not affected. After the restore the parameter value is set back to zero automatically.

7.1.3 REPLACING OPTION BOARD AND PARAMETER RESTORE

CANopen parameters must be defined for OPTE6 CANopen board in case the board is replaced in the field. Table 56 defines OPTE6 basic parameters and Table 60 defines CANopen communication parameters.

From the following AC drive models it is possible to take a complete CANopen parameter backup with Keypad or VACON® Live PC tool. The CANopen parameter backup can be restored to another OPTE6 board. This functionality requires that OPTE6 CANopen board contains V009 firmware or newer.

- VACON® 100 INDUSTRIAL (control firmware FW0072V029 or newer)
- VACON® 100 X (control firmware FW0072V029 or newer)
- VACON® 100 FLOW (control firmware FW0159V018 or newer)

From the following AC drive models it is possible to take parameter backup of the OPTE6 board's basic parameters that are defined in Table 56. The backup can be taken and restored from/to the board with Keypad or VACON® Live PC tool. CANopen communication parameters must be defined by writing them via CAN bus.

- VACON® NXP
- VACON® 20
- VACON® 20 X
- VACON® 20 Cold Plate

In all cases CANopen communication parameters (Table 60) can be defined for OPTE6 CANopen board by writing them via CAN bus. CANopen objects used for CANopen parametrization are defined in Chapter 10 APPENDIX A: OBJECT DICTIONARY.

OPTE6 parameters can be restored into default settings by using "Restore from set" parameter with Keypad or VACON® Live PC tool. See Chapter 7.1 OPTE6 panel parameters.

NOTE: OPTE6 board's parameters are stored into the board's flash memory. If OPTE6 board is parametrized, for example, in VACON® 100 FLOW drive and then moved into VACON® NXP drive which does not contain CANopen communication parameters, OPTE6 board uses CANopen communication parameters from its internal memory.

7.2 OPTE6 PANEL MONITOR VALUES

Different number of panel monitoring values are visible depending on the used AC drive and version.

VACON® NX family option board parameters are found under "Expander boards". Path depends on used slot where OPTE6 is installed (7.4.2 or 7.5.2).

VACON® 100 family option board parameters are found under "I/O and Hardware". Path depends on used slot where OPTE6 is installed (5.3.2 or 5.4.2).

VACON® 20 family option board monitoring values are under SYS menu.

Table 61. OPTE6 Monitoring Values

| # | Parameter | Format | Description |
|----|----------------|--------|---|
| 1* | CANopen Status | NNNN.Y | NNNN = A running counter for incoming messages Y = Node CANopen status, see Table 62 |
| 2* | Drive CW | - | Control word sent to AC drive from OPTE6 |
| 3* | Drive SW | - | Status word received from AC drive |
| 4* | Protocol CW | - | Control word received from CANopen protocol |
| 5* | Protocol SW | - | Status word sent to CANopen protocol |

* From version V26 (VACON® 100 INDUSTRIAL) and V18 (VACON® 100 FLOW) and V196 (VACON® NXP).

Table 62. Node CANopen status

| Value | Description |
|-------|---------------------|
| 0 | Initialising |
| 4 | Stopped |
| 5 | Operational |
| 6 | Pre-operational |
| 7 | Reset application |
| 8 | Reset communication |

From VACON® 100 (INDUSTRIAL V027 and FLOW V018) and OPTE6 V009 firmware version forwards, the active CANopen object values are visible as monitoring values. These values are the currently active settings and are equal to values read via CANopen objects. These objects might have different values than the stored parameters, depending if parameters written via bus are stored or not. The values are divided into subfolders same way as the corresponding parameters.

7.3 VACON PC-TOOLS

With VACON PC-tools it is possible to do following operations for OPTE6 CANopen:

- Update firmware into OPTE6 CANopen option board
- Set parameters for OPTE6 CANopen
- Read monitor values of OPTE6 CANopen

7.3.1 PC TOOL SUPPORT

This table describes what PC tools are supported in each AC drive type. The connection type "serial" means a direct serial connection to the AC drive. The connection type "Ethernet" means that Ethernet connection is supported by using for example via VACON® 100 inbuild Ethernet interface or via OPTE9 Dual Port Ethernet option board.

Table 63. The supported PC tools with different AC drives

| | VACON® 100 family | | VACON® NXS/NXP | | VACON® 20 family | |
|---------------|-----------------------------|----------|----------------|----------|------------------|----------|
| Tool | Serial | Ethernet | Serial | Ethernet | Serial | Ethernet |
| VACON® Loader | x | | x | | x | |
| VACON® Live | x | x | | | x | |
| NCIPConfig | Not used with OPTE6 CANopen | | | | | |
| NCDrive | | | x | x | | |
| NCLoad | Not used with OPTE6 CANopen | | | | | |

7.3.2 OPTE6 OPTION BOARD FIRMWARE UPDATE WITH VACON® LOADER

You can update OPTE6 CANopen firmware with VACON® Loader PC-tool. You need to have:

- PC with VACON® Loader installed
- VACON® AC drive in which OPTE6 CANopen option board is installed
- Serial cable:
 - VACON® NXP is connected to PC with RS232 serial cable which is connected from PC to NXP control unit's 9-pin DSUB connector (female). If PC does not contain RS232 serial port, then USB - RS232 converter device is needed between PC and NXP control.
 - VACON® 100 and VACON® 20 are connected to PC with VACON® Serial Cable.

The VACON® Loader can be downloaded from [https://www.danfoss.com/en/service-and-support/->Downloads->Software->select "Drives" as Business unit](https://www.danfoss.com/en/service-and-support/->Downloads->Software->select%20%22Drives%22%20as%20Business%20unit). It is bundled with the VACON® Live software package. After starting the installation program, follow the on-screen instructions.

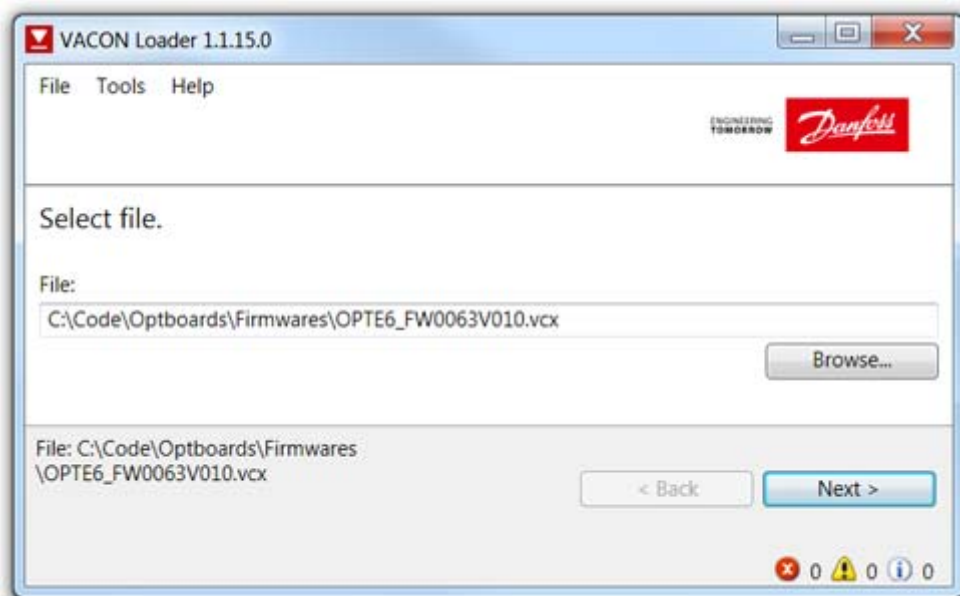
The OPTE6 CANopen firmware can be downloaded from [https://www.danfoss.com/en/service-and-support/->Software->select "Drives" as Business unit -> Fieldbus firmware](https://www.danfoss.com/en/service-and-support/->Software->select%20%22Drives%22%20as%20Business%20unit->Fieldbus%20firmware).

To update the option board firmware, follow the steps below.

NOTE! With VACON® 20, the baud rate 9600 must be used. With VACON® 20 X and VACON® 20 CP, the following baud rates are supported: 9600, 19200, 38400 or 57600. With VACON® 100 and VACON® NXP drives VACON® Loader selects a correct baud rate automatically.

Step 1: Connect your PC to the controller by using the serial cable.

Then select the firmware file which you want to load to the option board and double click it. This will start the VACON® Loader software. You can also start the program from the Windows Start menu. In this case, select the firmware file using the "Browse"-button.

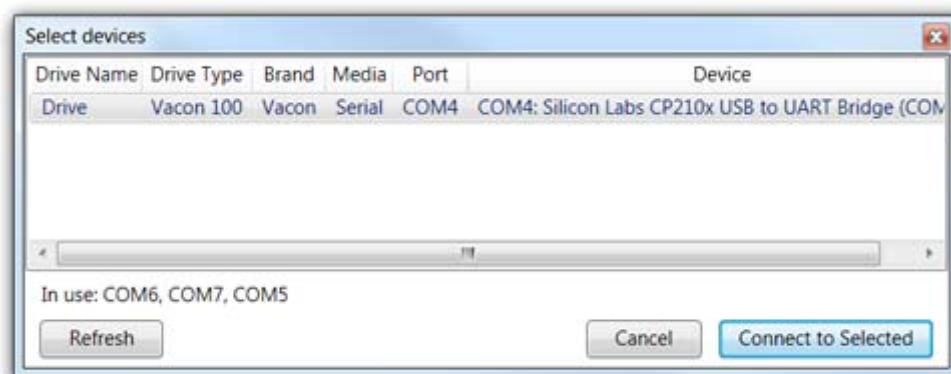


11598A_00

Figure 12. VACON® Loader: File selection

Step 2. Press 'next' and wait for the loader to find the network drives.

Then select a drive from the list and press 'Connect to Selected'.



11599A_00

Figure 13. VACON® Loader: Connecting to drive

Step 3. Select the modules to be updated, press 'next' and wait until the operation is finished.

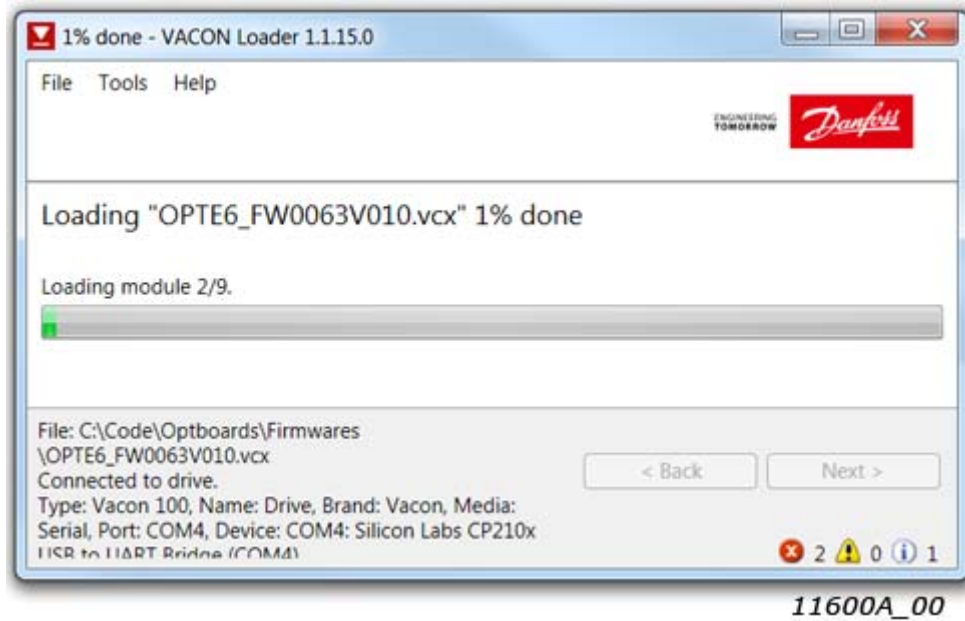


Figure 14. Option board module selection

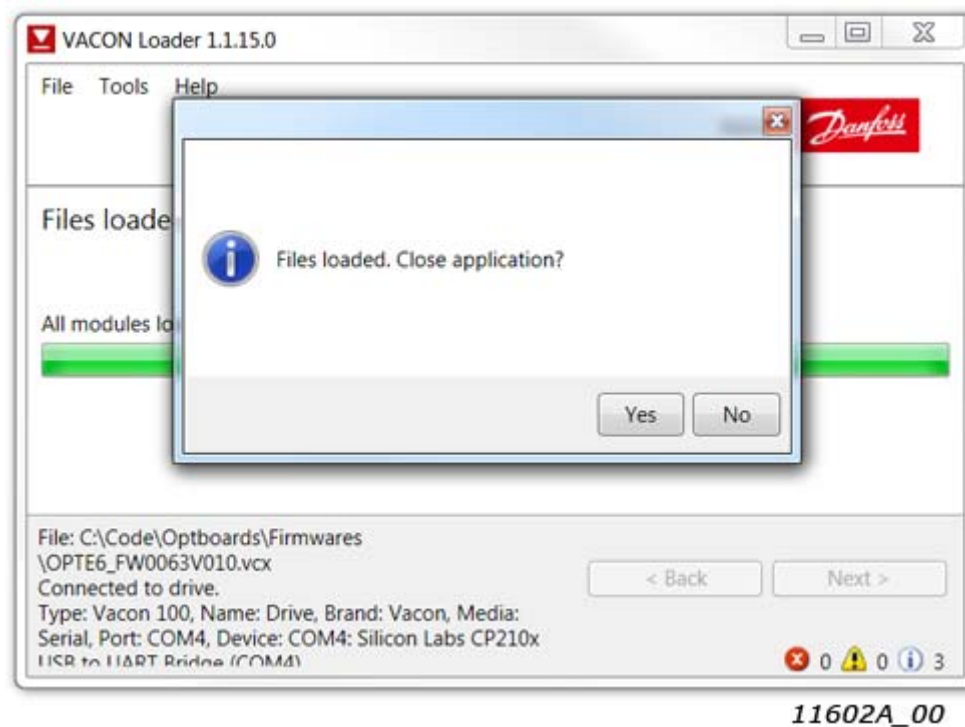


Figure 15. VACON® Loader: Loading is finished

7.3.3 PC TOOLS FOR VACON® NXP: NCDRIVE

You can configure the VACON® NXP AC drive and OPTE6 CANopen parameters with the NCDrive PC-tool. You need to have:

- PC with NCDrive installed
- VACON® NXP drive
- In case of Serial connection:
 - If PC contains RS232 serial port, then connect the serial cable from PC to VACON® NXP control unit's 9-pin DSUB connector (female).
 - If PC does not contain RS232 serial port, then USB - RS232 converter device is needed between PC and NXP control.
- In case of Ethernet connection:
 - Ethernet cable which is connected to option board's Ethernet interface.
 - VACON® NXP requires option board supporting Ethernet communication. For example, OPTE9 Dual Port Ethernet option board.

The NCDrive can be downloaded from [https://www.danfoss.com/en/service-and-support/->Downloads->Software->select "Drives" as Business unit](https://www.danfoss.com/en/service-and-support/->Downloads->Software->select%20%22Drives%22%20as%20Business%20unit). After starting the installation program, follow the on-screen instructions.

Once the program is installed successfully, you can launch it by selecting it in the Windows Start menu. Select Help --> Contents if you want more information about the software features.

7.3.3.1 NCDrive Serial communication settings

Connect your PC to the controller by using the USB/RS485 cable.

Select Tools -> Options... -> Communication tab. Then define settings for your USB - RS232 adapter and press Ok.

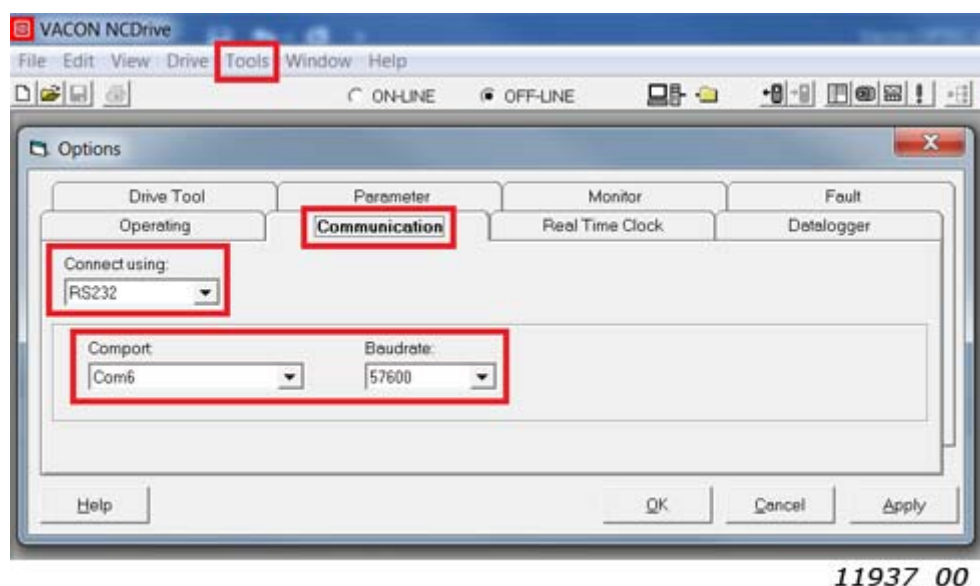


Figure 16. NCDrive: Serial communication settings

7.3.3.2 NCDrive Ethernet communication settings

For NCDrive Ethernet connection you need to have:

- Working Ethernet connection between PC and AC drive
- NCDrive is parametrized to use Ethernet connection

See instructions from Ethernet option board manual. Option board manuals can be downloaded from <https://www.danfoss.com/en/service-and-support/> -> Documentation -> select "Drives" as Business unit -> Select "VACON Option Boards" as Product Series.

7.3.3.3 Connecting to NCDrive

Press the "ON-LINE" button. The NCDrive will connect to the drive and start loading parameter information.

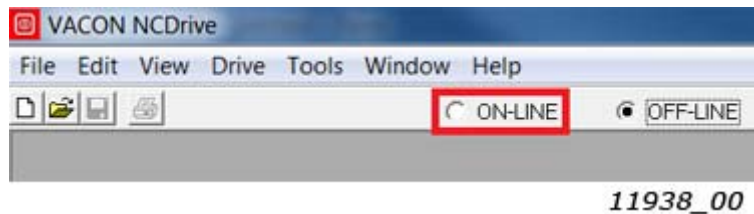


Figure 17. NCDrive: Going online

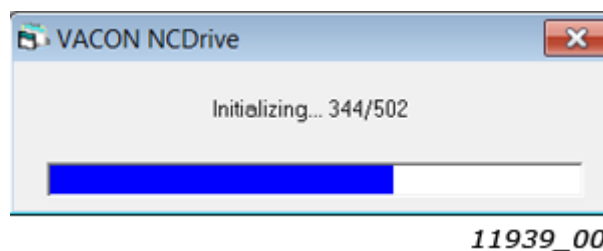


Figure 18. NCDrive: Loading information from the drive

To change the option board settings, navigate to the "M 7 Expander boards" menu and select the slot in which OPTE6 CANopen is connected to. It is possible to change parameters defined in Chapter 7.1 OPTE6 panel parameters.

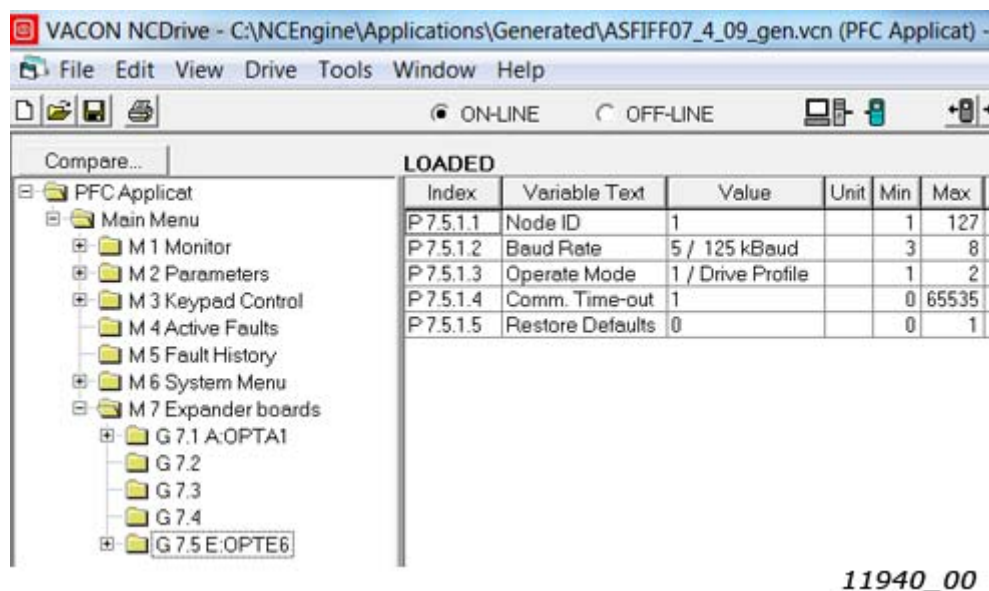


Figure 19. NC Drive: Parameter menu

7.3.4 PC TOOLS FOR VACON® 100 AND VACON® 20: VACON LIVE

You can configure the VACON® 100 AC drives, VACON® 20 AC drives and OPTE6 CANopen parameters with the VACON® Live PC-tool. Also monitor values of these devices can be read with VACON® Live. You need to have:

- PC with VACON® Live installed
- VACON® 100 or VACON® 20 AC drive
- In case of Serial connection:
 - VACON Serial Cable (USB - Serial cable) which is connected from PC to AC drive control unit.
 - In case of VACON® 20 also MCA (Micro Communications Adapter) is required. This adapter is not needed in case of VACON® 20 X / CP.
- In case of Ethernet connection:
 - Ethernet cable which is connected to AC drive's Ethernet interface.
 - In case of VACON® 100 it is possible to use inbuild Ethernet connection or Ethernet option board (for example OPTE9 Dual Port Ethernet).

NOTE! VACON® 20, VACON® 20 X and VACON® 20 Cold Plate do not support VACON® Live connection over Ethernet.

The VACON® Live can be downloaded from [https://www.danfoss.com/en/service-and-support/->Downloads -> Software -> select "Drives" as Business unit](https://www.danfoss.com/en/service-and-support/->Downloads->Software->select%20%22Drives%22%20as%20Business%20unit). After starting the installation program, follow the on-screen instructions.

Once the program is installed successfully, you can launch it by selecting it in the Windows Start menu. Select Help --> Contents if you want more information about the software features.

7.3.4.1 VACON Live Serial communication settings

Step 1: Connect your PC to VACON® AC drive with VACON® Serial Cable.

Step 2: Start VACON® Live. When the program starts, it asks "Select startup mode". Select "Online" startup mode. After this the program scans compatible drives.

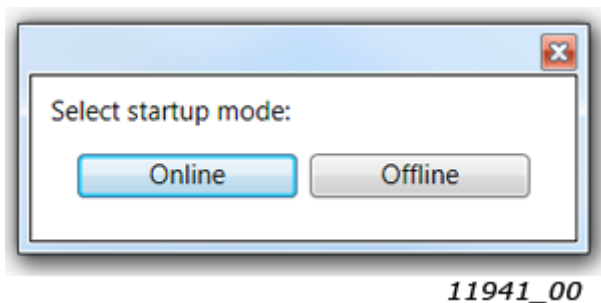
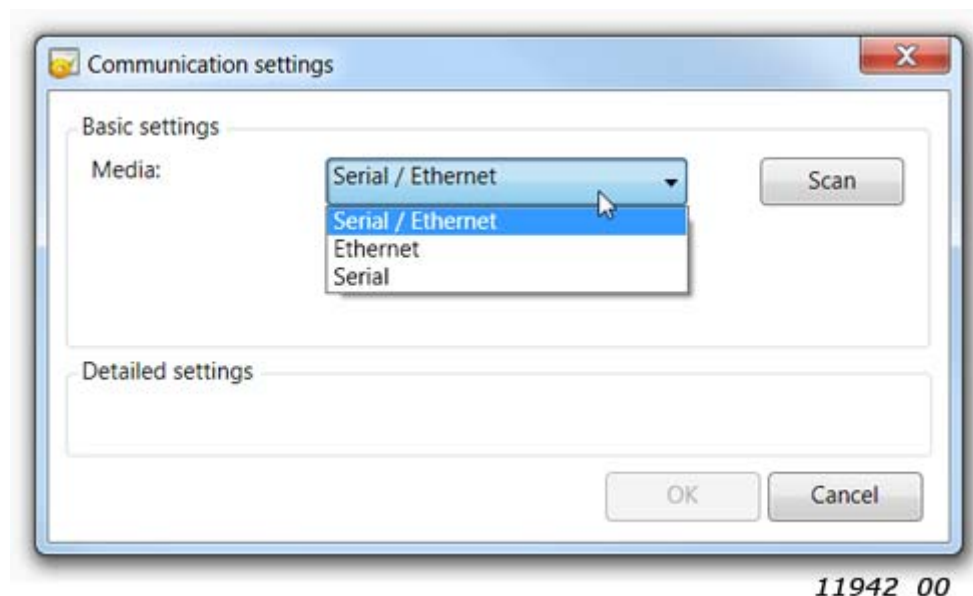


Figure 20. VACON® Live: To online mode

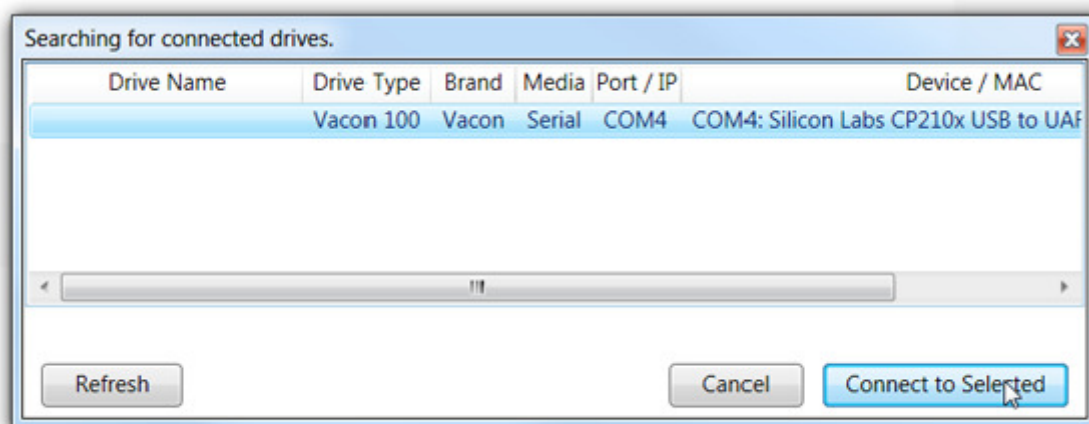
Step 2b: If VACON® Live cannot find your AC drive then ensure that "Serial / Ethernet" or "Serial" is selected. After that press "Scan".



11942_00

Figure 21. VACON® Live: Communication settings

Step 3: After successful scanning VACON Live shows the drive in connected drives window. Select the drive and press "Connected to Selected". After this VACON® Live reads parameter and monitor value tree from the drive.



11943_00

Figure 22. VACON® Live: Communication settings

7.3.4.2 VACON Live Ethernet communication settings

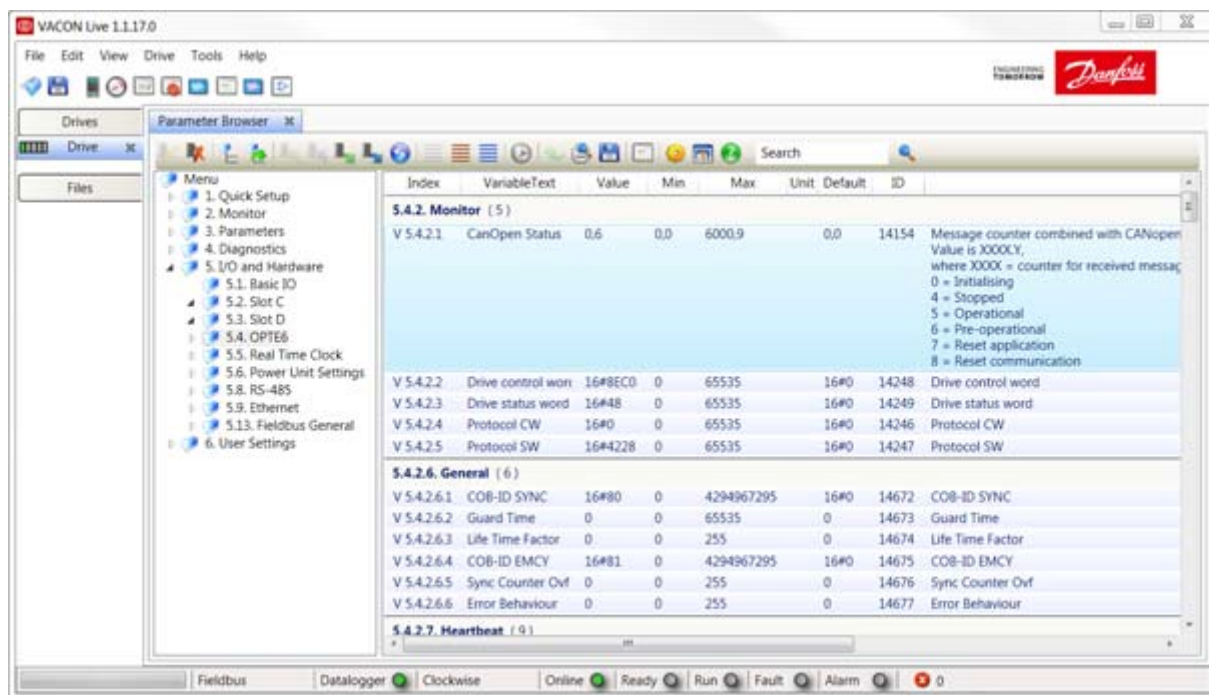
For VACON® Live Ethernet connection you need to have:

- Working Ethernet connection between PC and AC drive
- VACON® Live is parametrized to use Ethernet connection

See instructions from VACON® 100 Modbus, PROFINET I/O, Ethernet/IP, BACnet or OPTE9 Dual Port Ethernet option board manual. Manuals can be downloaded from <https://www.danfoss.com/en/service-and-support/> -> Documentation -> Select "Drives" as Business unit -> Select "VACON Option Boards" as Product Series.

7.3.4.3 *OPTE6 CANopen parameters in VACON® Live*

OPTE6 CANopen parameters and monitor values can be found from "5. I/O and Hardware" menu. With VACON® Live it is possible to modify OPTE6 CANopen parameters and view monitor values.



11944_00

Figure 23. VACON® Live: OPTE6 CANopen parameters.

7.4 QUICK INSTRUCTIONS FOR CONTROLLING THE MOTOR

This chapter gives quick instructions on how to get the motor running using either CiA-402 velocity or Bypass mode using SDO protocol.

1. Set the OPTE6 jumpers, refer to Chapter 5.3
2. Install OPTE6 option board in VACON® AC drive and connect the CAN cable, refer to chapter 6 and 5.1
3. Set the option board parameters to establish CANopen communication. Refer to chapter 7.1.
4. Set the AC drive parameters so that it is controlled over fieldbus. Refer to Chapter 11 APPENDIX B - FIELD BUS PARAMETRIZATION

Drive profile:

Motor control mode should be set to Speed control. If it is not, Statusword has bit 0x4000 set.

If Statusword bit 0x200 is not set, the AC drive is not in fieldbus control mode and cannot be started via CANopen.

5. Set Controlword (0x6040) to 0x00
 - Statusword (0x6041) is 0x270
6. Set Controlword to 0x06
 - Statusword is 0x4231
7. Set Controlword to 0x0F
 - AC drive starts, Statusword is 0x637
8. Set vl target velocity (0x6042) to 500 (rpm)
 - Motor starts to run at 500 rpm (0x6044 vl velocity actual value)
 - If actual velocity does not change, check that AC drive reference is set to fieldbus

Bypass:

Motor control mode must be set to Frequency (if reference is given as percentage of maximum frequency).

5. Set FB Control Word (0x2000:01) to 0x00
 - FB Status Word (0x2001:01) is 0x41
6. Set FB Control Word to 0x01
 - AC drive starts, FB Status Word is 0x23
 - If AC drive does not start, check that AC drive is in fieldbus control mode
7. Set FB Speed Reference (0x2000:03) to 2500 = 25.00% of maximum frequency. Usually default value for maximum frequency is 50.00 Hz, so in this case the reference is 12.50 Hz.
 - FB Actual Speed (0x2001:03) is 2500 = 25.00% output frequency of maximum frequency
 - If actual speed does not change, check that AC drive reference is set to fieldbus

If motor control mode is set to Speed, the reference is given as a percentage of maximum frequency converted to rpm. In this case the FB Actual Speed should not be read as this holds the percentage related to output frequency. Therefore, the actual speed should be from process data out 2 (by default set to speed in rpm).

8. CANOPEN OPTION BOARD INTERFACE

By default, the CANopen option board is configured to operate in Drive Profile mode. The drive profile implementation is 'Velocity Mode' which is defined in CiA 402 specification with PDO set for the AC drive. The board can also be configured into Manufacturer specific mode.

8.1 SUPPORTED DRIVE MODES

Drive modes can be selected by writing the desired mode in Modes of Operation (0x6060) object or by selecting the mode from panel. The active mode of operation can be read from object Modes of Operation Display (0x6061).

The supported drive modes object will return value 0x80000002, which is described in table below.

Table 64. Supported drive modes

| Bit(s) | Description |
|--------|-----------------------------------|
| 31 | Manufacturer specific Bypass mode |
| 30-2 | Not supported |
| 1 | Velocity mode |
| 0 | Not supported |

Table 65. Drive modes

| Value | Mode name | Description |
|-------|------------------|---|
| 2 | Velocity mode | The CiA 402 Drive Profile mode where the control of the drive is done using a control word and speed reference value as specified in the drive profile specification. |
| -1 | Bypass mode | In this mode, the Drive control can be done using raw process data which is exchanged with drive application. The drive profile state machine and the related objects become invalid. |
| -3 | Puller mode | CiA 420 EUROMAP profiles. Restricted availability, requires use of special application. |
| -4 | Co-Extruder mode | |

Table 66. Operating mode related objects

| Index | Description |
|--------|----------------------------|
| 0x6060 | Modes of Operation |
| 0x6061 | Modes of Operation Display |
| 0x6502 | Supported Drive Modes |

8.2 VELOCITY MODE

The velocity mode is one of the specific modes that CIA-402 Drive Profile defines. Common behaviour in all modes are PDS state machine, some control and status bits and certain objects.

8.2.1 PDS STATE MACHINE

The PDS state machine describes the generic start and stop sequence of the drive and the error behavior. The state machine is controlled by the Controlword object and internal events. The following objects are usable in velocity mode:

Table 67. Velocity mode related objects in OD

| Index | Description |
|--------|----------------------------|
| 0x6040 | Controlword |
| 0x6041 | Statusword |
| 0x6042 | Vl Target Velocity |
| 0x6043 | Vl Velocity Demand |
| 0x6044 | Vl Velocity Actual Value |
| 0x6046 | Vl Velocity Min Max Amount |
| 0x6048 | Vl Velocity Acceleration |
| 0x6049 | Vl Velocity Deceleration |

The possible state machine transitions can be seen from Figure 25. The state of the AC drive can be changed by writing the corresponding bits to Controlword data object. The needed bit values for each command can be seen from Figure 24.

The table below explains the actions taken in different state transitions and which event triggers which state transition. If the used drive/application does not support different stop bits in Fixed Control Word, the stop method will always be according to set stop function.

Table 68. State transition events and actions

| Transition | Event(s) | Action(s) |
|------------|--|---|
| 0 | Automatic transition after power-on or reset | Self-initialization is performed |
| 1 | Automatic transition after drive status is 'ready' | None |
| 2, 6 | Shutdown command | None |
| 3 | Switch on command | None |
| 4 | Enable operation command | Drive function is enabled |
| 5 | Disable operation command | Drive function is disabled |
| 7 | Disable voltage or quick stop command | None |
| 8 | Shutdown command | Stop by ramp / stop function |
| 9 | Disable voltage command | Stop by coast / stop function |
| 10, 12 | Disable voltage command | None |
| 11 | Quick stop command | Quick stop / stop function |
| 13 | Fault signal | Go to fault state and stop by stop function |
| 14 | Automatic transition | None |
| 15 | Fault reset command | Reset fault if no fault currently exists on drive |

Current state can be determined by reading the value of Statusword data object and comparing the value of bits to the table below.

Table 69. Statusword bits

| Bits of the statusword | | | | | | | | | PDS state |
|------------------------|---|---|---|---|---|---|---|---|------------------------|
| 15-8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 | |
| x | x | 0 | x | x | 0 | 0 | 0 | 0 | Not ready to switch on |
| x | x | 1 | x | x | 0 | 0 | 0 | 0 | Switch on disabled |
| x | x | 0 | 1 | x | 0 | 0 | 0 | 1 | Ready to switch on |
| x | x | 0 | 1 | x | 0 | 0 | 1 | 1 | Switched on |
| x | x | 0 | 1 | x | 0 | 1 | 1 | 1 | Operation enabled |
| x | x | 0 | 0 | x | 0 | 1 | 1 | 1 | Quick stop active |
| x | x | 0 | x | x | 1 | 1 | 1 | 1 | Fault reaction active |
| x | x | 0 | x | x | 1 | 0 | 0 | 0 | Fault |
| x = Do not care | | | | | | | | | |

8.2.2 CİA-402 OBJECTS

8.2.2.1 0X6040 - Controlword

Controlword is used to control the drive operation according to the PDS state machine. By default, Controlword is mapped into the first two bytes of rxPD01.

Table 70. 0x6040:Controlword

| Bit | | Name | Description |
|-------|-----------------|-------------------------|---|
| 12-15 | na | Not used | Bits 12 through 15 are not in use. |
| 11 | ar | Alarm reset | Rising edge resets alarm |
| 10 | r | Reserved | Bit 10 is not in use |
| 9 | oms | Operation mode specific | Bit 9 is not in use |
| 8 | h | Halt | Bit 8 is not in use |
| 7 | fr | Fault reset | Rising edge resets fault |
| 4-6 | na | Not Used | Bits 4 through 6 are not in use. |
| 3 | eo | Enable operation | Start drive |
| 2 | \overline{qs} | Quick stop | Stops the drive using the drive/application specific stop function used as quick stop |
| 1 | ev | Enable voltage | Enables/disables output voltage |
| 0 | so | Switch on | Enables possibility to start drive together with ev |

8.2.2.2 0X6041 - Statusword

Statusword indicates whether the drive is in remote control and if the target velocity is reached. By default, Statusword is mapped into the first two bytes of txPD01.

Table 71. 0x6041:Statusword

| Bit | | Name | Description |
|-------|-----------------|-------------------------|--|
| 15 | na | Not in use | Bit 15 is not in use |
| 14 | idm | Incorrect drive mode | Indicates that the drive is in incorrect "Control mode" for the used CiA-402 profile |
| 12-13 | oms | Operation mode specific | Bits 12 through 13 are not in use |
| 10 | tr | Target reached | Target velocity reached |
| 9 | rm | Remote | Indicates if the drive is controllable by Fieldbus |
| 8 | na | Not in use | Bit 8 not in use. |
| 7 | w | Warning | The AC drive has an active Alarm. |
| 6 | sod | Switch on disabled | PDS switch on disable |
| 5 | \overline{qs} | Quick stop | PDS quick stop active |
| 4 | ve | Voltage enabled | Voltage is enabled |
| 3 | f | Fault | PDS Fault (indicates fault condition) |
| 2 | oe | Operation enabled | PDS operation enabled (drive is running) |
| 1 | so | Switched on | PDS switched on |
| 0 | rtso | Ready to switch on | PDS ready to switch on |

8.2.2.3 0X6042 - vl Target Velocity

The signed value of motor rpm speed request to drive. A negative value means that the motor is running clockwise. By default, the object is mapped into the last two bytes of rxPDO1.

Range: -32768...32767

| 0x6042:vl Target Velocity | | |
|---------------------------|--|---|
| 15 | | 0 |
| Rpm request to drive | | |

8.2.2.4 0X6043 - vl Velocity Demand

The signed read-only value of the ramp generator output scaled into rpm. A negative value means that the motor is running clockwise. By default, the object is not mapped into any PDO.

Range: -32768...32767

| 0x6043:vl Velocity Demand | | |
|---|--|---|
| 15 | | 0 |
| Drive ramp generator output scaled into rpm | | |

8.2.2.5 0X6044 - vl Velocity Actual Value

The signed value of the motor actual rpm speed. A negative value means that the motor is running clockwise. By default, the object is mapped into the last two bytes of txPDO1.

Range: -32678...32767

| 0x6044:vl Velocity Actual Value | | |
|---------------------------------|--|---|
| 15 | | 0 |
| Motor actual rpm speed | | |

8.2.2.6 0X6046 - vl Velocity Min Max Amount

The minimum and maximum rpm speed of the AC drive's motor. The motor runs on minimum speed defined here when the vl Target Velocity is set to 0.

| 0x6046.01: vl Velocity Min Amount | | |
|-----------------------------------|--|---|
| 31 | | 0 |
| Motor minimum rpm speed | | |

| 0x6046.02: vl Velocity Max Amount | | |
|-----------------------------------|--|---|
| 31 | | 0 |
| Motor maximum rpm speed | | |

Range: 0... 4294967296

8.2.2.7 0X6048 - vI Velocity Acceleration

This object indicates the configured delta speed and delta time of the slope of the acceleration ramp.

| 0x6048.01: Delta speed | | |
|--|--|---|
| 31 | | 0 |
| Maximum change of rpm the motor will accelerate during the time specified in Delta Time. | | |

Range: 0... 4294967296

| 0x6048.02: Delta time | | |
|--|--|---|
| 16 | | 0 |
| Time (in seconds) in which the rpm of the motor will accelerate the amount specified in Delta Speed. | | |

Range: 0... 65536

8.2.2.8 0X6049 - vI Velocity Deceleration

This object indicates the configured delta speed and delta time of the slope of the deceleration ramp.

| 0x6049.01: Delta speed | | |
|--|--|---|
| 31 | | 0 |
| Maximum change of rpm the motor will accelerate during the time specified in Delta Time. | | |

Range: 0... 4294967296

| 0x6049.02: Delta time | | |
|--|--|---|
| 16 | | 0 |
| Time (in seconds) in which the rpm of the motor will accelerate the amount specified in Delta Speed. | | |

Range: 0... 65536

8.2.3 PDO CONFIGURATION

This chapter describes the default PDO mapping configuration when using CiA-402 velocity mode. By default, RPDO/TPDO 2 - 5 are disabled. These settings are restored when:

- changing the operate mode to "Velocity" from either panel or object 0x6060
- using the "Restore from Set" panel parameter when Velocity operate mode is selected, or
- loading a default parameter set with object 0x1011, 1 "Restore all Default parameters"

When the parameter set is restored, the COB-IDs are calculated according to used Node ID.

Table 72. Overview of Receive PDO configuration in CiA-402 vl mode

| Object | COB-ID | Size | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|--------|----------------------|------|-----------------------|----------------------------|-----------------------|-----------------------|-------|-------|-------|-------|
| RPD01 | 0x200 + Node ID | 4 | 0x6040, Controlword | 0x6042, vl Target Velocity | Not used | | | | | |
| RPD02* | 0x80000300 + Node ID | 8 | 0x2000,4 FB PD in 1 | 0x2000,5 FB PD in 2 | 0x2000,6 FB PD in 3 | 0x2000,7 FB PD in 4 | | | | |
| RPD03* | 0x80000400 + Node ID | 8 | 0x2000,8 FB PD in 5 | 0x2000,9 FB PD in 6 | 0x2000,A FB PD in 7 | 0x2000,B FB PD in 8 | | | | |
| RPD04* | 0x80000500 + Node ID | 8 | 0x2000,C FB PD in 9 | 0x2000,D FB PD in 10 | 0x2000,E FB PD in 11 | 0x2000,F FB PD in 12 | | | | |
| RPD05* | 0x80000000 + Node ID | 8 | 0x2000,10 FB PD in 13 | 0x2000,11 FB PD in 14 | 0x2000,12 FB PD in 15 | 0x2000,13 FB PD in 16 | | | | |

* Disabled by default

Table 73. Overview of Transmit PDO configuration in CiA-402 vl mode

| Object | COB-ID | Size | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|--------|----------------------|------|------------------------|----------------------------|------------------------|------------------------|-------|-------|-------|-------|
| TPD01 | 0x180 + Node ID | 4 | 0x6041, Statusword | 0x6044, vl Velocity Actual | Not used | | | | | |
| TPD02* | 0x80000280 + Node ID | 8 | 0x2001,4 FB PD out 1 | 0x2001,5 FB PD out 2 | 0x2001,6 FB PD out 3 | 0x2001,7 FB PD out 4 | | | | |
| TPD03* | 0x80000380 + Node ID | 8 | 0x2001,8 FB PD out 5 | 0x2001,9 FB PD out 6 | 0x2001,A FB PD out 7 | 0x2001,B FB PD out 8 | | | | |
| TPD04* | 0x80000480 + Node ID | 8 | 0x2001,C FB PD out 9 | 0x2001,D FB PD out 10 | 0x2001,E FB PD out 11 | 0x2001,F FB PD out 12 | | | | |
| TPD05* | 0x80000000 + Node ID | 8 | 0x2001,10 FB PD out 13 | 0x2001,11 FB PD out 14 | 0x2001,12 FB PD out 15 | 0x2001,13 FB PD out 16 | | | | |

* Disabled by default

8.3 BYPASS MODE

In bypass mode, some data defined in the profile is invalid, and the drive control is done using raw process data. The raw process data arrays are located in the manufacturer specific objects 0x2000 and 0x2001. There are two arrays: one for incoming data and one for outgoing data.

The drive application defines what data is mapped into the outgoing process data and how the incoming process data is handled. See 12. APPENDIX C - FIELD BUS PROCESS DATA MAPPING AND SCALING for more details.

8.3.1 PDO CONFIGURATION

This chapter describes the default PDO mapping configuration when using Bypass mode. By default, RPDO/TPDO 4 & 5 are disabled. These settings are restored when:

- changing the operate mode to "Bypass" from either panel or object 0x6060
- using the "Restore from Set" panel parameter when Bypass operate mode is selected, or
- loading a default parameter set with object 0x1011,4 "Restore Bypass parameter set" and sending a NMT reset message

When the parameter set is restored, the COB-IDs are calculated according to used Node ID.

Table 74. Overview of Receive PDO configuration in Bypass mode

| Object | COB-ID | Size | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|--------|----------------------|------|--------------------------|--|-------|-----------------------------------|-------|--------------------------|-------|-------|
| RPD01 | 0x200 + Node ID | 6 | 0x2000,1, Controlword | 0x2000,2 FB General Control Word | | 0x2000,3 FB Speed Reference | | Not used | | |
| RPD02 | 0x300 + Node ID | 8 | 0x2000,4 FB PD in 1 | 0x2000,5 FB PD in 2 | | 0x2000,6 FB PD in 3 | | 0x2000,7 FB PD in 4 | | |
| RPD03 | 0x400 + Node ID | 8 | 0x2000,8 FB PD in 5 | 0x2000,9 FB PD in 6 | | 0x2000,A FB PD in 7 | | 0x2000,B FB PD in 8 | | |
| RPD04* | 0x80000500 + Node ID | 8 | 0x2000,C FB PD in 9 | 0x2000,D FB PD in 10 | | 0x2000,E FB PD in 11 | | 0x2000,F FB PD in 12 | | |
| RPD05* | 0x80000000 + Node ID | 8 | 0x2000,10 FB PD in 13 | 0x2000,11 FB PD in 14 | | 0x2000,12 FB PD in 15 | | 0x2000,13 FB PD in 16 | | |

* Disabled by default

Table 75. Overview of Transmit PDO configuration in Bypass mode

| Object | COB-ID | Size | DATA0 | DATA1 | DATA2 | DATA3 | DATA4 | DATA5 | DATA6 | DATA7 |
|--------|----------------------|------|----------------------------|------------------------------------|-----------------------------|---------------------------|-------|-------|-------|-------|
| TPD01 | 0x180 + Node ID | 6 | 0x2001,1 FB Status Word | 0x2001,2 FB General Status Word | 0x2001,3 FB Actual Speed | Not used | | | | |
| TPD02 | 0x280 + Node ID | 8 | 0x2001,4 FB PD out 1 | 0x2001,5 FB PD out 2 | 0x2001,6 FB PD out 3 | 0x2001,7 FB PD out 4 | | | | |
| TPD03 | 0x380 + Node ID | 8 | 0x2001,8 FB PD out 5 | 0x2001,9 FB PD out 6 | 0x2001,A FB PD out 7 | 0x2001,B FB PD out 8 | | | | |
| TPD04* | 0x80000480 + Node ID | 8 | 0x2000,C FB PD out 9 | 0x2000,D FB PD out 10 | 0x2000,E FB PD out 11 | 0x2000,F FB PD out 12 | | | | |
| TPD05* | 0x80000000 + Node ID | 8 | 0x2000,10 FB PD out 13 | 0x2000,11 FB PD out 14 | 0x2000,12 FB PD out 15 | 0x2000,13 FB PD out 16 | | | | |

* Disabled by default

8.4 DEFAULT PROCESS DATA APPLICATION MAPPING

This chapter describes the default mapping of the OPTE6 process data variables to the application data in the drive. It also provides a description of the application data in the drive. Supported control/status word bits might differ depending on used application. See AC drive specific bit definition in 13. APPENDIX D - CONTROL AND STATUS WORD.

8.4.1 FB CONTROL WORD

Table 76. FB Control Word

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|------|------|------|------|-------|------|------|------|-------|-------|------|-----|------|
| - | - | - | ESTP | JOG2 | JOG1 | BREF | BCTRL | ZREF | FRMP | ZRMP | STPM2 | STPM1 | FRST | DIR | STRT |

Table 77. FB Control Word description

| Bit | Name | Description | 0 | 1 |
|-----|-------|-----------------|----------------------------------|--------------------------|
| 12 | ESTP | Emergency Stop | Request as fast stop as possible | Emergency stop |
| 11 | JOG2 | Jogging request | Select jogging with reference 2 | Select ref2 jogging |
| 10 | JOG1 | Jogging request | Select jogging with reference 1 | Select ref1 jogging |
| 9 | BREF | Bus Reference | Force Reference to fieldbus | Selected reference place |
| 8 | BCTRL | Bus Control | Force fieldbus control active | Selected control place |
| 7 | ZREF | Zero Ref | Force reference to zero | Force reference to zero |
| 6 | FRMP | Ramp Freeze | Freeze ramp generator | Freeze ramp generator |
| 5 | QRMP | Quick Ramp Time | Use quick ramp time | Normal ramp time |
| 4 | STPM2 | Stop Mode2 | Stop mode ramping | Stop By Ramp mode |
| 3 | STPM1 | Stop Mode1 | Stop mode coasting | Coasting Stop Mode |
| 2 | FRST | Fault Reset | Request fault reset from drive | Request reset from drive |
| 1 | DIR | Direction | Rotation direction | Clockwise |
| 0 | STRT | Start / Stop | Start / Stop request | Stop |
| | | | | Run |

8.4.2 FB CONTROL WORD EXTENSION (GENERAL CONTROL WORD)

Table 78. FB Control Word Extension

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------|--------|------|----|----|----|---|---|---|---|---|---|---|---|---|---|
| BSTAT2 | BSTAT1 | BFLT | - | - | - | - | - | - | - | - | - | - | - | - | - |

Table 79. FB Control Word extension description

| Bit | Name | Description | 0 | 1 |
|-----|--------|-------------|-------------------------------------|--------------------------|
| 15 | BSTAT2 | Bus status | Informs bus status into application | Status of option board |
| 14 | BSTAT1 | Bus status | Informs bus status into application | |
| 13 | BFLT | Bus fault | Request fieldbus fault | - |
| | | | | Generate field bus fault |

8.4.3 FB SPEED REFERENCE*Table 80. FB Speed Reference*

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| FB Speed Reference | | | | | | | | | | | | | | | |

Table 81. FB Speed reference description

| Name | Description | Min | Max |
|--------------------|---|-----------|--------------------|
| FB Speed Reference | Frequency reference at percentage between minimum and maximum frequency. Control word DIR bit is used to select rotation direction. | 0 (0%) | 10000 (100.00%) |

8.4.4 FB PROCESS DATA INPUT 1...16*Table 82. FB Process Data Input 1...16*

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|--------------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| FB Process data in | | | | | | | | | | | | | | | |

8.4.5 FB PROCESSDATA INPUT MAPPING IN APPLICATION

In addition to sending control commands and speed reference values to the application, 16 process data items can be sent directly to the application to control other features via fieldbus. These can be, for example, activating preset frequencies, activating or deactivating faults, selecting different ramp times, controlling I/O's, and so on. Different applications support different features. For more information, see the application manual. Also, always check the value of the sent data in the application manual to see in which format they are given (for example, % values, bit coded values, and so on).

8.4.6 FB STATUS WORD

Table 83. FB Status Word

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|----|----|----|----|----|----|---|---|------|------|-------|-------|-----|-----|-----|-----|
| - | - | - | - | - | - | - | - | FRDY | ZSPD | ATREF | ALARM | FLT | DIR | RUN | RDY |

Table 84. FB Status word description

| Bit | Name | Description | 0 | 1 |
|-----|-------|--------------|-----------|----------------------|
| 7 | FRDY | Flux ready | - | Flux ready |
| 6 | ZSPD | Zero speed | - | Zero speed condition |
| 5 | ATREF | At reference | - | Reference reached |
| 4 | ALARM | Alarm | - | Drive is in Alarm |
| 3 | FLT | Faulted | - | Drive is Faulted |
| 2 | DIR | Direction | Clockwise | Counter clockwise |
| 1 | RUN | Run | Stopped | Running |
| 0 | RDY | Ready | - | Ready |

8.4.7 FB STATUS WORD EXTENSION (GENERAL STATUS WORD)

Table 85. FB Status word extension

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----|-----|-----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| CP3 | CP2 | CP1 | - | - | - | - | - | - | - | - | - | - | - | - | - |

Table 86. FB Status word extension description

| Bit | Name | Description | 0 | 1 |
|-----|------|---------------|---------------------------------|-------------------------------------|
| 15 | CP3 | Control Place | Drive Control place information | 0,0,1-(1) Fieldbus |
| 14 | CP2 | | | 0,1,0- (2) Panel |
| 13 | CP1 | | | 0,1,1 - (3) Tool 1,0,0 - (4) I/O |

8.4.8 FB ACTUAL SPEED

Table 87. FB Actual Speed

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|-----------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| FB Actual Speed | | | | | | | | | | | | | | | |

Table 88. FB Actual Speed description

| Name | Description | Min | Max |
|-----------------|--|-----------|--------------------|
| FB Actual Speed | Actual output frequency at percentage between minimum and maximum frequency. Status word bit DIR is used to indicate actual direction. | 0 (0%) | 10000 (100.00%) |

8.4.9 FB PROCESSDATA OUTPUT 1...16

Table 89. FB Process data output 1...16

| 15 | 14 | 13 | 12 | 11 | 10 | 9 | 8 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
|---------------------|----|----|----|----|----|---|---|---|---|---|---|---|---|---|---|
| FB Process data Out | | | | | | | | | | | | | | | |

8.4.10 FB PROCESSDATA OUTPUT MAPPING IN APPLICATION

Process data output mappings are selectable and also the default settings vary between drives. Refer to application manual for details. See Chapter 12. APPENDIX C - FIELDBUS PROCESS DATA MAPPING AND SCALING for default values for commonly used applications.

8.5 VACON ANYPARAMETER SERVICE

For access to AC drive parameters and monitoring values, the OPTE6 maps the CANopen manufacturer segment 0x2100 - 0x5FFF into the application IDs of the drive based on the IEC61131 standard. Both read and write access is supported.

The application ID 1 maps to CANopen object 0x2101 and so on, making it possible to read/write any application ID between 1 and 16127 (0x3EFF). To access the application IDs in the drive, make the following calculation:

Index = ID number + 0x2100

NOTE! The response data is in raw format. See application manual for available IDs, decimal count and the unit used for the parameters.

IDs can be read/written as VACON® NX scaled values in all AC drives, or in VACON® 100 family AC drives also as actual raw value. Units such as Hz, rpm, A, V, etc. are given with 4 decimal point precision. See examples for more details.

The used scale in VACON® 100 family depends on the "Mode" parameter setting. This can be written from panel parameter or by using CANopen object 0x2005.

VACON® provides .EDS files which contains the correct data types and IDs for multiple VACON® AC drive applications.

These files can be downloaded from <https://www.danfoss.com/en/service-and-support/> -> Software -> Select "Drives" as Business unit -> Fieldbus configuration files.

8.5.1 ERROR RESPONSES

In case an error occurs during reading or writing a parameter via the anyparameter service, the following SDO abort error codes are returned:

Table 90. SDO abort codes returned by anyparameter service

| Abort code | Description | Error |
|-------------|--|---|
| 0x0602 0000 | Object does not exist in the object dictionary | ID is not found in used application |
| 0x0601 0002 | Attempt to write a read only object | ID is a monitoring value |
| 0x0607 0010 | Data type does not match, length of service parameter does not match | Data length does not match ID data type |
| 0x0609 0030 | Invalid value for parameter (download only). | Data value is invalid |
| 0x0800 0000 | General error | Other error |
| 0x0800 0024 | No data available | Parameter access timeout |

8.5.2 EXAMPLES

In the examples below, the following ID values are used:

- 102 = Maximum frequency (Hz)
- 600 = Motor control mode

Table 91. Example 1: Reading values from different AC drives

| AC drive | Mode | Read command | | Response | | | |
|-------------------|-----------------|--------------|--------|----------|-------------|-----------|--------------|
| | | ID | Index | Length | Hex (LSB) | Dec (MSB) | Actual value |
| VACON® 20 family | Any Pre V008 | 102 | 0x2166 | 2 | 88 13 | 5000 | 50.00 Hz |
| VACON® 100 family | | 600 | 0x2358 | 2 | 01 00 | 1 | 1 = OL Speed |
| VACON® 100 family | Normal | 102 | 0x2166 | 4 | 20 A1 07 00 | 500000 | 50.0000 Hz |
| | | 600 | 0x2358 | 4 | 01 00 00 00 | 1 | 1 = OL Speed |

| AC drive | Mode | Write command | | | | Actual value |
|-------------------|-----------------|---------------|--------|--------|-------------|------------------|
| | | ID | Index | Length | Value (Hex) | |
| VACON® 20 family | Any Pre V008 | 102 | 0x2166 | 2 | 94 11 | 45.00 Hz |
| VACON® 100 family | | 600 | 0x2358 | 2 | 00 00 | 0 = OL Frequency |
| VACON® 100 family | Normal | 102 | 0x2166 | 4 | D0 DD 06 00 | 45.0000 Hz |
| | | 600 | 0x2358 | 4 | 00 00 00 00 | 0 = OL Frequency |

9. FAULT TRACING

When the option board or the AC drive control diagnostics detect an unusual operating condition, the drive opens a notification, for example, on the keypad. The keypad shows the ordinal number of the fault, the fault code and a short fault description.

You can reset the fault with the Reset button on the control keypad, via the I/O terminal or via the used fieldbus protocol. The faults are stored in the Fault history menu, which can be browsed. The fault table presents only the fault conditions related to the fieldbus in use.

NOTE! When you contact a distributor or a factory because of a fault condition, always write down all the texts and codes on the keypad display. Then send detailed problem description together with the Drive Info (Service Info) File to the local distributor. If possible, also send CAN communication log from the fault situation if applicable.

Service Info can be read from the drive with PC-tool.

- In case of VACON® Live connect to the drive and select from VACON® Live menu bar: Drive -> Service information...
- In case of NCDrive connect to the drive and select from NCDrive menu bar: File -> Service Info...

See basic usage of VACON® PC-tools in 7.3 VACON PC-tools.

See local contacts in Danfoss web pages:

<https://www.danfoss.com/en/contact-us/contacts-list/> -> select "Drives" as Business unit.

9.1 TYPICAL FAULT CONDITIONS

Table 92. Typical fieldbus fault conditions

| Fault condition | Possible cause | | Remedy |
|---------------------------|-----------------|--|------------------------|
| No communication | Cabling | Supply or motor cables are located too close to the fieldbus cable | Refer to (part 9) |
| | | Wrong type of fieldbus cable, e.g. insufficient shielding | Refer to 4.3 |
| | | Too long cabling | |
| | | Invalid termination | Refer to 5.3 and 4.3.1 |
| | Grounding | Inadequate or invalid grounding | Refer to 5.3 and 4.3.1 |
| | Connections | Excessive stripping of cables | Refer to |
| | | Conductors in wrong terminals | Refer to 5.1 |
| | | Too loose connections of conductors | |
| Faulty communication | Parametrization | Overlapping Node ID | Refer to 7.1 |
| | | Wrong baud rate | |
| Drive does not start | | PDO mapping is invalid | Refer to 7.1 and 3.5 |
| | | Wrong control place selected | Refer to 11. |
| Drive runs at wrong speed | | Wrong motor control mode selected | |
| Fieldbus fault (F53) | Fieldbus | See chapter 9.2 | |

9.2 FIELDBUS TIMEOUT FAULT (F53)

VACON® fieldbuses create a fieldbus timeout fault (F53) when a fault has occurred in the fieldbus protocol and the AC drive is set to fieldbus control. If the control place is set to e.g. I/O, no fieldbus fault is triggered even if a fault condition is met. The fault response can also be modified in the AC drive application, see chapter 11.2 for more details.

OPTE6 creates a fieldbus fault in the following conditions:

Table 93. OPTE6 Fieldbus fault trigger conditions

| Fault | Description | Remedy |
|--------------------|--|--|
| CAN Passive | <ul style="list-style-type: none"> "Communication timeout" parameter is other than zero Either Transmit Error Counter or Receive Error Counter raises above 127, causing the CAN driver to go into passive state | <ul style="list-style-type: none"> Check cabling and baud rates of all nodes in network Disable communication timeout or increase the time. |
| CAN BUS-OFF | <ul style="list-style-type: none"> "Communication timeout" parameter is other than zero Transmit Error Counter raises above 255, causing the CAN driver to go into bus-off state | |
| Heartbeat Consumer | <ul style="list-style-type: none"> Heartbeat consumer is set At least one valid heartbeat message has been received The heartbeat time has elapsed | <ul style="list-style-type: none"> Check Heartbeat Consumer time Check heartbeat producer time of the producer CANopen node Increase Heartbeat Time Check cabling |
| Node Guard | <ul style="list-style-type: none"> Guard Time and Life Time Factor are set, At least one valid node guard message has been received Node Life Time has elapsed | <ul style="list-style-type: none"> Check Node Life Time (Life Time Factor Guard Time) Check Node Guard producer CANopen node Increase Node Life Time Check cabling |
| PDO Timer | <ul style="list-style-type: none"> RPDO Timer is set At least one valid PDO message has been received RPDO Timer has elapsed | <ul style="list-style-type: none"> Check RPDO Timer time Check PDO producer CANopen node Increase PDO Timer time Check cabling |

If any of the Table 93 fault situations occur (even if no fault is triggered in AC drive), the CANopen NMT state machine state can be affected. This is managed by object 0x1029:01 Communication Error. By default, the error response has value 0 = Change to NMT state Pre-operational.

Any other fault (e.g. overtemperature), does not affect the NMT state machine. The object 0x1029:02 Internal Error is fixed to value 1 = No change to NMT state.

9.3 DETAILED FAULT CODE

In VACON® 100 family a detailed fault code for fieldbuses is "Source3" in the fault history. This feature is available from firmware version V026 (INDUSTRIAL) and V018 (FLOW).

This information can be read with:

- Panel (4.1.x.26): Diagnostics ' Active Faults ' "FB Timeout" ' Details ' Source 3
- Panel (4.3.x.26): Diagnostics ' Fault History ' "FB Timeout" ' Details ' Source 3
- VACON® Live: View ' Fault Diagnostics ' Icon "Load active faults" ' Source3
- VACON® Live: View ' Fault Diagnostics ' Icon "Load fault history" ' Source3

Source3 fault codes are defined as follows:

Table 94. VACON® 100 family fieldbus fault Source3 codes

| Code | Name | Description |
|------|--------------------------------------|---|
| 1* | IO Watchdog | IO connection closed after timeout occurred |
| 2* | IO Connection Closed | IO connection closed by fieldbus master |
| 3* | Explicit Messaging Watchdog | Explicit messaging connection (other than cyclic IO data) closed after timeout occurred |
| 4* | Explicit Messaging Connection Closed | Explicit messaging connection closed by fieldbus master |
| 5* | Cable Disconnected | Fieldbus cable disconnected after device startup |
| 6* | Cable Not Connected | Fieldbus cable was not connected after device startup |
| 7* | Bad Data | Master IO data has changed from GOOD to BAD |
| 8* | Idle State | Connection status changed to IDLE when motor is controlled |
| 9* | Internal System Fault | Internal fieldbus fault, contact technical support |
| 10* | Too Many Bad Messages | Protocol has received too many bad messages in a row and closed the connection |
| 11 | CAN BUS-OFF | CAN driver is in BUS-OFF state |
| 12 | CAN Passive | CAN driver is in passive state |
| 13* | No External Power | No external power detected |
| 14 | Heartbeat Consumer | Heartbeat consumer timeout occurred |
| 15 | Node Guard | Node guarding timeout occurred |
| 16 | PDO Timer | PDO timer event timeout occurred |

* Not relevant to OPTE6 CANopen option board

In case multiple error situations occur at once, the fault that has happened first has its code shown as the Source3 fault code.

10. APPENDIX A: OBJECT DICTIONARY

10.1 COMMUNICATION SEGMENT

| Index | Sub-index | Description | Object Code | Data Type | Access | PDO Mapping | Default value | Unit |
|---|-----------|--------------------------------|-------------|------------|--------|-------------|---------------|------|
| Object 1000h: Device Type | | | | | | | | |
| 1000 | 0 | Device type | Variable | UNSIGNED32 | CONST | No | 0x00010192 | |
| Object 1001h: Error Register | | | | | | | | |
| 1001 | 0 | Error Register | Variable | UNSIGNED8 | RO | No | 0x00 | |
| Object 1003h: Pre-defined Error Field | | | | | | | | |
| 1003 | | Pre-defined Error Field | Array | UNSIGNED32 | | | | |
| | 000 | Number of Errors | | UNSIGNED8 | RW | No | 0x00000000 | |
| | 001 | Standard Error Field | | UNSIGNED32 | RO | No | 0x00000000 | |
| | 002 | Standard Error Field | | UNSIGNED32 | RO | No | 0x00000000 | |
| | 003 | Standard Error Field | | UNSIGNED32 | RO | No | 0x00000000 | |
| | 004 | Standard Error Field | | UNSIGNED32 | RO | No | 0x00000000 | |
| | 005 | Standard Error Field | | UNSIGNED32 | RO | No | 0x00000000 | |
| | 006 | Standard Error Field | | UNSIGNED32 | RO | No | 0x00000000 | |
| | 007 | Standard Error Field | | UNSIGNED32 | RO | No | 0x00000000 | |
| | 008 | Standard Error Field | | UNSIGNED32 | RO | No | 0x00000000 | |
| | 009 | Standard Error Field | | UNSIGNED32 | RO | No | 0x00000000 | |
| Object 1005h: COB-ID SYNC | | | | | | | | |
| 1005 | 0 | COB-ID SYNC | Variable | UNSIGNED32 | RW | No | 0x00000080 | |
| Object 100Ch: Guard Time | | | | | | | | |
| 100C | 0 | Guard Time | Variable | UNSIGNED16 | RW | No | 0x00000000 | ms |
| Object 100Dh: Life Time Factor | | | | | | | | |
| 100D | 0 | Life Time Factor | Variable | UNSIGNED8 | RW | No | 0x00000000 | |
| Object 1010h: Store Parameter Field | | | | | | | | |
| 1010 | | Store Parameter Field | Array | UNSIGNED32 | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x2 | |
| | 001 | Save all Parameters | | UNSIGNED32 | RW | No | - | |
| Object 1011h: Restore Default Parameters | | | | | | | | |
| 1011 | | Restore Default Parameters | Array | UNSIGNED32 | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x4 | |
| | 001 | Restore all Default Parameters | | UNSIGNED32 | RW | No | - | |
| | 004 | Restore Bypass parameter set | | UNSIGNED32 | RW | No | - | |
| Object 1014h: COB-ID EMCY | | | | | | | | |
| 1014 | 0 | COB-ID EMCY | Variable | UNSIGNED32 | RW | No | 0x80 | |
| Object 1016h: Heartbeat Consumer Entries | | | | | | | | |

| Index | Sub-index | Description | Object Code | Data Type | Access | PDO Mapping | Default value | Unit |
|---|-----------|---------------------------------------|-------------|--------------|--------|-------------|---------------|--------|
| 1016 | | Heartbeat Consumer Entries | Array | UNSIGNED32 | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x08 | |
| | 001 | Consumer Heartbeat Time 1 | | UNSIGNED32 | RW | No | 0x00000000 | |
| | 002 | Consumer Heartbeat Time 2 | | UNSIGNED32 | RW | No | 0x00000000 | |
| | 003 | Consumer Heartbeat Time 3 | | UNSIGNED32 | RW | No | 0x00000000 | |
| | 004 | Consumer Heartbeat Time 4 | | UNSIGNED32 | RW | No | 0x00000000 | |
| | 005 | Consumer Heartbeat Time 5 | | UNSIGNED32 | RW | No | 0x00000000 | |
| | 006 | Consumer Heartbeat Time 6 | | UNSIGNED32 | RW | No | 0x00000000 | |
| | 007 | Consumer Heartbeat Time 7 | | UNSIGNED32 | RW | No | 0x00000000 | |
| | 008 | Consumer Heartbeat Time 8 | | UNSIGNED32 | RW | No | 0x00000000 | |
| Object 1017h: Producer Heartbeat Time | | | | | | | | |
| 1017 | 0 | Producer Heartbeat Time | Variable | UNSIGNED16 | RW | No | 0x00000000 | |
| Object 1018h: Identity Object | | | | | | | | |
| 1018 | | Identity Object | Record | IDENTITY | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x4 | |
| | 001 | Vendor Id | | UNSIGNED32 | RO | No | 0x90 | |
| | 002 | Product Code | | UNSIGNED32 | RO | No | - | |
| | 003 | Revision number | | UNSIGNED32 | RO | No | - | |
| | 004 | Serial number | | UNSIGNED32 | RO | No | - | |
| Object 1019h: Synchronous counter | | | | | | | | |
| 1019 | 0 | Synchronous counter | Variable | UNSIGNED8 | RW | No | 0x00000000 | |
| Object 1029h: Error Behaviour | | | | | | | | |
| 1029 | | Error Behaviour | Array | UNSIGNED8 | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x1 | |
| | 001 | Communication Error | | UNSIGNED8 | RW | No | 0x0 | |
| | 002 | Internal Error | | UNSIGNED8 | RW | No | 0x1 | |
| Object 1400h: Receive PDO Communication Parameter 1 | | | | | | | | |
| 1400 | | Receive PDO Communication Parameter 1 | Record | PDO_COMM_PAR | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x05 | |
| | 001 | COB-ID | | UNSIGNED32 | RW | No | 0x200 | |
| | 002 | Transmission Type | | UNSIGNED8 | RW | No | 0xFE | |
| | 003 | Inhibit Time | | UNSIGNED16 | RW | No | 0x0 | 100 µs |
| | 005 | Event Timer | | UNSIGNED16 | RW | No | 0x0 | ms |
| Object 1401h: Receive PDO Communication Parameter 2 | | | | | | | | |

| Index | Sub-index | Description | Object Code | Data Type | Access | PDO Mapping | Default value | Unit |
|--|-----------|---------------------------------------|-------------|--------------|--------|-------------|---------------|--------|
| 1401 | | Receive PDO Communication Parameter 2 | Record | PDO_COMM_PAR | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x05 | |
| | 001 | COB-ID | | UNSIGNED32 | RW | No | 0x80000300 | |
| | 002 | Transmission Type | | UNSIGNED8 | RW | No | 0xFE | |
| | 003 | Inhibit Time | | UNSIGNED16 | RW | No | 0x0 | 100 µs |
| | 005 | Event Timer | | UNSIGNED16 | RW | No | 0x0 | ms |
| Object 1402h: Receive PDO Communication Parameter 3 | | | | | | | | |
| 1402 | | Receive PDO Communication Parameter 3 | Record | PDO_COMM_PAR | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x05 | |
| | 001 | COB-ID | | UNSIGNED32 | RW | No | 0x80000400 | |
| | 002 | Transmission Type | | UNSIGNED8 | RW | No | 0xFE | |
| | 003 | Inhibit Time | | UNSIGNED16 | RW | No | 0x0 | 100 µs |
| | 005 | Event Timer | | UNSIGNED16 | RW | No | 0x0 | ms |
| Object 1403h: Receive PDO Communication Parameter 4 | | | | | | | | |
| 1403 | | Receive PDO Communication Parameter 4 | Record | PDO_COMM_PAR | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x05 | |
| | 001 | COB-ID | | UNSIGNED32 | RW | No | 0x80000500 | |
| | 002 | Transmission Type | | UNSIGNED8 | RW | No | 0xFE | |
| | 003 | Inhibit Time | | UNSIGNED16 | RW | No | 0x0 | 100 µs |
| | 005 | Event Timer | | UNSIGNED16 | RW | No | 0x0 | ms |
| Object 1404h: Receive PDO Communication Parameter 5 | | | | | | | | |
| 1404 | | Receive PDO Communication Parameter 5 | Record | PDO_COMM_PAR | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x05 | |
| | 001 | COB-ID | | UNSIGNED32 | RW | No | 0x80000000 | |
| | 002 | Transmission Type | | UNSIGNED8 | RW | No | 0xFE | |
| | 003 | Inhibit Time | | UNSIGNED16 | RW | No | 0x0 | 100 µs |
| | 005 | Event Timer | | UNSIGNED16 | RW | No | 0x0 | ms |
| Object 1600h: Receive PDO Mapping Parameter 1 | | | | | | | | |
| 1600 | | Receive PDO Mapping Parameter 1 | Record | PDO_MAPPING | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RW | No | 0x02 | |
| | 001 | Mapping Entry 1 | | UNSIGNED32 | RW | No | 0x60400010 | |
| | 002 | Mapping Entry 2 | | UNSIGNED32 | RW | No | 0x60420010 | |
| | 003 | Mapping Entry 3 | | UNSIGNED32 | RW | No | 0x00000000 | |
| | 004 | Mapping Entry 4 | | UNSIGNED32 | RW | No | 0x00000000 | |
| Object 1601h: Receive PDO Mapping Parameter 2 | | | | | | | | |

| Index | Sub-index | Description | Object Code | Data Type | Access | PDO Mapping | Default value | Unit |
|--|-----------|--|-------------|--------------|--------|-------------|---------------|--------|
| 1601 | | Receive PDO Mapping Parameter 2 | Record | PDO_MAPPING | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RW | No | 0x04 | |
| | 001 | Mapping Entry 1 | | UNSIGNED32 | RW | No | 0x20000410 | |
| | 002 | Mapping Entry 2 | | UNSIGNED32 | RW | No | 0x20000510 | |
| | 003 | Mapping Entry 3 | | UNSIGNED32 | RW | No | 0x20000610 | |
| | 004 | Mapping Entry 4 | | UNSIGNED32 | RW | No | 0x20000710 | |
| Object 1602h: Receive PDO Mapping Parameter 3 | | | | | | | | |
| 1602 | | Receive PDO Mapping Parameter 3 | Record | PDO_MAPPING | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RW | No | 0x04 | |
| | 001 | Mapping Entry 1 | | UNSIGNED32 | RW | No | 0x20000810 | |
| | 002 | Mapping Entry 2 | | UNSIGNED32 | RW | No | 0x20000910 | |
| | 003 | Mapping Entry 3 | | UNSIGNED32 | RW | No | 0x20000A10 | |
| | 004 | Mapping Entry 4 | | UNSIGNED32 | RW | No | 0x20000B10 | |
| Object 1603h: Receive PDO Mapping Parameter 4 | | | | | | | | |
| 1603 | | Receive PDO Mapping Parameter 4 | Record | PDO_MAPPING | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RW | No | 0x04 | |
| | 001 | Mapping Entry 1 | | UNSIGNED32 | RW | No | 0x20000C10 | |
| | 002 | Mapping Entry 2 | | UNSIGNED32 | RW | No | 0x20000D10 | |
| | 003 | Mapping Entry 3 | | UNSIGNED32 | RW | No | 0x20000E10 | |
| | 004 | Mapping Entry 4 | | UNSIGNED32 | RW | No | 0x20000F10 | |
| Object 1604h: Receive PDO Mapping Parameter 5 | | | | | | | | |
| 1604 | | Receive PDO Mapping Parameter 5 | Record | PDO_MAPPING | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RW | No | 0x04 | |
| | 001 | Mapping Entry 1 | | UNSIGNED32 | RW | No | 0x20001010 | |
| | 002 | Mapping Entry 2 | | UNSIGNED32 | RW | No | 0x20001110 | |
| | 003 | Mapping Entry 3 | | UNSIGNED32 | RW | No | 0x20001210 | |
| | 004 | Mapping Entry 4 | | UNSIGNED32 | RW | No | 0x20001310 | |
| Object 1800h: Transmit PDO Communication Parameter 1 | | | | | | | | |
| 1800 | | Transmit PDO Communication Parameter 1 | Record | PDO_COMM_PAR | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x06 | |
| | 001 | COB-ID | | UNSIGNED32 | RW | No | 0x180 | |
| | 002 | Transmission Type | | UNSIGNED8 | RW | No | 0xFE | |
| | 003 | Inhibit Time | | UNSIGNED16 | RW | No | 0x64 | 100 µs |
| | 005 | Event Timer | | UNSIGNED16 | RW | No | 0x0 | ms |
| | 006 | Sync start value | | UNSIGNED8 | RW | No | 0x0 | |
| Object 1801h: Transmit PDO Communication Parameter 2 | | | | | | | | |

| Index | Sub-index | Description | Object Code | Data Type | Access | PDO Mapping | Default value | Unit |
|--|-----------|--|-------------|--------------|--------|-------------|---------------|--------|
| 1801 | | Transmit PDO Communication Parameter 2 | Record | PDO_COMM_PAR | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x06 | |
| | 001 | COB-ID | | UNSIGNED32 | RW | No | 0x80000280 | |
| | 002 | Transmission Type | | UNSIGNED8 | RW | No | 0xFE | |
| | 003 | Inhibit Time | | UNSIGNED16 | RW | No | 0x3E8 | 100 µs |
| | 005 | Event Timer | | UNSIGNED16 | RW | No | 0x0 | ms |
| | 006 | Sync start value | | UNSIGNED8 | RW | No | 0x0 | |
| Object 1802h: Transmit PDO Communication Parameter 3 | | | | | | | | |
| 1802 | | Transmit PDO Communication Parameter 3 | Record | PDO_COMM_PAR | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x06 | |
| | 001 | COB-ID | | UNSIGNED32 | RW | No | 0x80000380 | |
| | 002 | Transmission Type | | UNSIGNED8 | RW | No | 0xFE | |
| | 003 | Inhibit Time | | UNSIGNED16 | RW | No | 0x3E8 | 100 µs |
| | 005 | Event Timer | | UNSIGNED16 | RW | No | 0x0 | ms |
| | 006 | Sync start value | | UNSIGNED8 | RW | No | 0x0 | |
| Object 1803h: Transmit PDO Communication Parameter 4 | | | | | | | | |
| 1803 | | Transmit PDO Communication Parameter 4 | Record | PDO_COMM_PAR | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x06 | |
| | 001 | COB-ID | | UNSIGNED32 | RW | No | 0x80000480 | |
| | 002 | Transmission Type | | UNSIGNED8 | RW | No | 0xFE | |
| | 003 | Inhibit Time | | UNSIGNED16 | RW | No | 0x3E8 | 100 µs |
| | 005 | Event Timer | | UNSIGNED16 | RW | No | 0x0 | ms |
| | 006 | Sync start value | | UNSIGNED8 | RW | No | 0x0 | |
| Object 1804h: Transmit PDO Communication Parameter 5 | | | | | | | | |
| 1804 | | Transmit PDO Communication Parameter 5 | Record | PDO_COMM_PAR | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RO | No | 0x06 | |
| | 001 | COB-ID | | UNSIGNED32 | RW | No | 0x80000000 | |
| | 002 | Transmission Type | | UNSIGNED8 | RW | No | 0xFE | |
| | 003 | Inhibit Time | | UNSIGNED16 | RW | No | 0x3E8 | 100 µs |
| | 005 | Event Timer | | UNSIGNED16 | RW | No | 0x0 | ms |
| | 006 | Sync start value | | UNSIGNED8 | RW | No | 0x0 | |
| Object 1A00h: Transmit PDO Mapping Parameter 1 | | | | | | | | |

| Index | Sub-index | Description | Object Code | Data Type | Access | PDO Mapping | Default value | Unit |
|--|-----------|----------------------------------|-------------|-------------|--------|-------------|---------------|------|
| 1A00 | | Transmit PDO Mapping Parameter 1 | Record | PDO_MAPPING | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RW | No | 0x02 | |
| | 001 | Mapping Entry 1 | | UNSIGNED32 | RW | No | 0x60410010 | |
| | 002 | Mapping Entry 2 | | UNSIGNED32 | RW | No | 0x60440010 | |
| | 003 | Mapping Entry 3 | | UNSIGNED32 | RW | No | 0x00000000 | |
| | 004 | Mapping Entry 4 | | UNSIGNED32 | RW | No | 0x00000000 | |
| Object 1A01h: Transmit PDO Mapping Parameter 2 | | | | | | | | |
| 1A01 | | Transmit PDO Mapping Parameter 2 | Record | PDO_MAPPING | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RW | No | 0x04 | |
| | 001 | Mapping Entry 1 | | UNSIGNED32 | RW | No | 0x20010410 | |
| | 002 | Mapping Entry 2 | | UNSIGNED32 | RW | No | 0x20010510 | |
| | 003 | Mapping Entry 3 | | UNSIGNED32 | RW | No | 0x20010610 | |
| | 004 | Mapping Entry 4 | | UNSIGNED32 | RW | No | 0x20010710 | |
| Object 1A02h: Transmit PDO Mapping Parameter 3 | | | | | | | | |
| 1A02 | | Transmit PDO Mapping Parameter 3 | Record | PDO_MAPPING | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RW | No | 0x04 | |
| | 001 | Mapping Entry 1 | | UNSIGNED32 | RW | No | 0x20010810 | |
| | 002 | Mapping Entry 2 | | UNSIGNED32 | RW | No | 0x20010910 | |
| | 003 | Mapping Entry 3 | | UNSIGNED32 | RW | No | 0x20010A10 | |
| | 004 | Mapping Entry 4 | | UNSIGNED32 | RW | No | 0x20010B10 | |
| Object 1A03h: Transmit PDO Mapping Parameter 4 | | | | | | | | |
| 1A03 | | Transmit PDO Mapping Parameter 4 | Record | PDO_MAPPING | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RW | No | 0x04 | |
| | 001 | Mapping Entry 1 | | UNSIGNED32 | RW | No | 0x20010C10 | |
| | 002 | Mapping Entry 2 | | UNSIGNED32 | RW | No | 0x20010D10 | |
| | 003 | Mapping Entry 3 | | UNSIGNED32 | RW | No | 0x20010E10 | |
| | 004 | Mapping Entry 4 | | UNSIGNED32 | RW | No | 0x20010F10 | |
| Object 1A04h: Transmit PDO Mapping Parameter 5 | | | | | | | | |
| 1A04 | | Transmit PDO Mapping Parameter 5 | Record | PDO_MAPPING | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | RW | No | 0x04 | |
| | 001 | Mapping Entry 1 | | UNSIGNED32 | RW | No | 0x20011010 | |
| | 002 | Mapping Entry 2 | | UNSIGNED32 | RW | No | 0x20011110 | |
| | 003 | Mapping Entry 3 | | UNSIGNED32 | RW | No | 0x20011210 | |
| | 004 | Mapping Entry 4 | | UNSIGNED32 | RW | No | 0x20011310 | |

10.2 MANUFACTURER SEGMENT

| Index | Sub-index | Description | Object Code | Data Type | Access | PDO Mapping | Default value | Unit |
|----------------------------------|-----------|---------------------------|-------------|------------|--------|-------------|---------------|------|
| Object 2000h: FB Processdata In | | | | | | | | |
| 2000 | | FB Processdata In | Array | UNSIGNED16 | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | CONST | No | 0xB | |
| | 001 | FB Control Word | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 002 | FB Control Word Extension | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 003 | FB Speed Reference | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 004 | FB Process Data In 1 | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 005 | FB Process Data In 2 | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 006 | FB Process Data In 3 | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 007 | FB Process Data In 4 | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 008 | FB Process Data In 5 | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 009 | FB Process Data In 6 | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 010 | FB Process Data In 7 | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 011 | FB Process Data In 8 | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 012 | FB Process Data In 9 * | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 013 | FB Process Data In 10 * | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 014 | FB Process Data In 11 * | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 015 | FB Process Data In 12 * | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 016 | FB Process Data In 13 * | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 017 | FB Process Data In 14 * | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 018 | FB Process Data In 15 * | | UNSIGNED16 | RWW | Yes | 0x0 | |
| | 019 | FB Process Data In 16 * | | UNSIGNED16 | RWW | Yes | 0x0 | |
| Object 2001h: FB Processdata Out | | | | | | | | |

| Index | Sub-index | Description | Object Code | Data Type | Access | PDO Mapping | Default value | Unit |
|-------------------------------------|-----------|--------------------------|-------------|------------|--------|-------------|---------------|------|
| 2001 | | FB Processdata Out | Array | UNSIGNED16 | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | CONST | No | 0xB | |
| | 001 | FB Status Word | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 002 | FB Status Word Extension | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 003 | FB Actual Speed | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 004 | FB Process Data Out 1 | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 005 | FB Process Data Out 2 | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 006 | FB Process Data Out 3 | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 007 | FB Process Data Out 4 | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 008 | FB Process Data Out 5 | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 009 | FB Process Data Out 6 | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 010 | FB Process Data Out 7 | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 011 | FB Process Data Out 8 | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 012 | FB Process Data Out 9 * | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 013 | FB Process Data Out 10 * | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 014 | FB Process Data Out 11 * | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 015 | FB Process Data Out 12 * | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 016 | FB Process Data Out 13 * | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 017 | FB Process Data Out 14 * | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 018 | FB Process Data Out 15 * | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | 019 | FB Process Data Out 16 * | | UNSIGNED16 | R0 | Yes | 0x0 | |
| Object 2002h: FB Dummy In | | | | | | | | |
| 2002 | | FB Dummy In | Array | UNSIGNED16 | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | CONST | No | 0xB | |
| | 001 | FB Dummy In Offset 0 | | UNSIGNED16 | RW | Yes | 0x0 | |
| | ... | ... | ... | ... | ... | ... | ... | ... |
| | 011 | FB Dummy In Offset 10 | | UNSIGNED16 | RW | Yes | 0x0 | |
| Object 2003h: FB Dummy Out | | | | | | | | |
| 2003 | | | Array | | | | | |
| | 000 | Number of Entries | | UNSIGNED8 | CONST | No | 0xB | |
| | 001 | FB Dummy Out Offset 0 | | UNSIGNED16 | R0 | Yes | 0x0 | |
| | ... | ... | ... | ... | ... | ... | ... | ... |
| | 011 | FB Dummy Out Offset 10 | | UNSIGNED16 | R0 | Yes | 0x0 | |
| Object 2004h: Communication Timeout | | | | | | | | |
| 2004 | 0 | Communication Timeout | Variable | UNSIGNED16 | RW | No | 0x01 | s |
| Object 2005h: Mode | | | | | | | | |
| 2005 | 0 | Mode | Variable | UNSIGNED16 | RW | No | 0x0 | |

* Supported in VACON® NXP when Fast mode or Normal extended mode is enabled. See details in Chapters 14. APPENDIX E - FIELD BUS OPTION BOARD COMMUNICATION and 15. APPENDIX F - PARAMETERS FOR APPLICATION DEVELOPERS.

The objects from 0x2100 to 0x5FFF contain the device-specific parameters and monitor values defined in separate .EDS files. These files can be downloaded from <https://www.danfoss.com/en/service-and-support/> -> Software -> select "Drives" as Business unit -> Fieldbus configuration files.

For more information refer to chapter 8.5.

10.3 DEVICE PROFILE SEGMENT

| Index | Sub-index | Description | Object Code | Data Type | Access | PDO Mapping | Default Value |
|---|-----------|----------------------------|-------------|-------------------|--------|-------------|---------------|
| Object 6040h: Controword | | | | | | | |
| 6040 | 0 | Controlword | Variable | UNSIGNED16 | RWW | Yes | - |
| Object 6041h: Statusword | | | | | | | |
| 6041 | 0 | Statusword | Variable | UNSIGNED16 | RO | Yes | - |
| Object 6042h: vl Target Velocity | | | | | | | |
| 6042 | 0 | vl Target Velocity | Variable | INTEGER16 | RWW | Yes | 0x0000 |
| Object 6043h: vl Velocity Demand | | | | | | | |
| 6043 | 0 | vl Velocity Demand | Variable | INTEGER16 | RO | Yes | - |
| Object 6044h: vl Velocity Actual Value | | | | | | | |
| 6044 | 0 | vl Velocity Actual Value | Variable | INTEGER16 | RO | Yes | - |
| Object 6046h: vl Velocity Min Max Amount | | | | | | | |
| 6046 | | vl Velocity Min Max Amount | Array | UNSIGNED32 | | | |
| | 000 | Number of Entries | | CONST | | No | 0x00000002 |
| | 001 | vl_Velocity_Min_Amount | | UNSIGNED32 | RW | No | 0x2 |
| | 002 | vl_Velocity_Max_Amount | | UNSIGNED32 | RW | No | - |
| Object 6048h: vl Velocity Acceleration | | | | | | | |
| 6048 | | vl Velocity Acceleration | Record | P402_VL_VEL_ACC_T | | | |
| | 000 | NumOfEntries | | CONST | | No | 0x2 |
| | 001 | DeltaSpeed | | UNSIGNED32 | RW | No | 0x0 |
| | 002 | DeltaTime | | UNSIGNED16 | RW | No | 0x0 |
| Object 6049h: vl Velocity Deceleration | | | | | | | |
| 6049 | | vl Velocity Deceleration | Record | P402_VL_VEL_ACC_T | | | |
| | 000 | NumOfEntries | | CONST | | No | 0x2 |
| | 001 | DeltaSpeed | | UNSIGNED32 | RW | No | 0x0 |
| | 002 | DeltaTime | | UNSIGNED16 | RW | No | 0x0 |
| Object 6060h: Modes of Operation | | | | | | | |
| 6060 | 0 | Modes of Operation | Variable | INTEGER8 | RW | No | - |
| Object 6061h: Modes of Operation Display | | | | | | | |
| 6061 | 0 | Modes of Operation Display | Variable | INTEGER8 | RO | No | - |
| Object 6502h: Supported Drive Modes | | | | | | | |
| 6502 | 0 | Supported Drive Modes | Variable | UNSIGNED32 | RO | No | - |

11. APPENDIX B - FIEDLBUS PARAMETRIZATION

The following chapter describes briefly how to parametrize the AC drive in order for the motor to be controllable via fieldbus. These instructions are written for some basic applications. For more information, see the application-specific manual.

In order for the AC drive to accept commands from the fieldbus network, the control place of the AC drive has to be set to fieldbus. The default value of the parameter "Control Place" is usually I/O. Note that if the control unit software is updated, the default settings are restored. In addition, some applications may have the remote speed reference selection set by default to other than fieldbus. In these cases, the speed reference selection must be set to fieldbus, in order for the speed reference to be controlled via fieldbus.

NOTE! The motor control mode should be selected to support the used process and profile.

11.1 FIELDBUS CONTROL AND REFERENCE SELECTION

The following tables list some of the parameters related to fieldbus control in case of standard applications for the VACON® 100 family, VACON® NXP, VACON® 20 and VACON® 20X AC Drives. See the application specific manuals for more detailed information and latest updates.

The parameters can be read and written by using the drive panel, PC Tools or fieldbus protocol. For instructions on fieldbus writing, see the fieldbus specific manual.

Table 95. Fieldbus parameterization for VACON® 100 family (standard application)

| Parameter name | ID | Value | Default | Panel Tree |
|----------------------|------|---|---------|------------|
| Control mode | 600 | 0 = Frequency 1 = Speed 2 = Torque | 0 | P3.1.2.1 |
| Remote control place | 172 | 1 = Fieldbus CTRL | 0 | P3.2.1 |
| Local / remote | 211 | 0 = Remote | 0 | P3.2.2 |
| Fieldbus ref. sel. | 122 | 3 = Fieldbus | 3 | P3.3.1.10 |
| Controlling fieldbus | 2539 | 1 = Automatic 2 = Slot D 3 = Slot E | 1 | P5.13.1 |

Table 96. Fieldbus parametrization for VACON® 20 (standard application)

| Parameter name | ID | Value | Default | Panel Tree |
|--------------------------------------|-----|---|---------|------------|
| Disable showing of Quick menu | - | 0 = Advanced menu 1 = Quick setup parameters | 1 | P 17.2 |
| Motor control mode | 600 | 0 = Frequency 1 = Speed 2 = Torque | 0 | P1.8 |
| Rem. Control place 1 sel. | 172 | 1 = Fieldbus CTRL | 0 | P2.1 |
| Local / remote | 211 | 0 = Remote | 0 | P2.5 |
| Rem. Control place 1 freq. ref. sel. | 122 | 3 = Fieldbus | 7 | P3.3 |

Table 97. Fieldbus parametrization for VACON® 20X (multipurpose application)

| Parameter name | ID | Value | Default | Panel Tree |
|-------------------------|------|----------------------------|---------|------------|
| Motor control mode | 600 | 0 = Frequency 1 = Speed | 0 | P8.1 |
| Control place selection | 125 | 2 = Fieldbus | 0 | P1.11 |
| Frequency ref. sel. | 1819 | 5 = Fieldbus | 0 | P1.12 |

Table 98. Fieldbus parametrization for VACON® NXP (multipurpose application)

| Parameter name | ID | Value | Default | Panel Tree |
|-------------------------|-----|--|---------|------------|
| Motor control mode | 600 | 0 = Frequency 1 = Speed 2 = Torque | 0 | P2.6.1 |
| Control place selection | 125 | 3 = Fieldbus | 1 | P3.1 |
| Frequency ref. sel. | 122 | 9 = Fieldbus | 3 | P2.1.13 |

11.1.1.1 CONTROLLING FIELDBUS PARAMETER

In VACON® 100 family products, if multiple fieldbus options or protocols are active at once, use the "Controlling Fieldbus"-parameter to select the instance from which the process data is sent to the drive application. This parameter is located under I/O and Hardware / Fieldbus General (5.13.1).

By default, the setting is in "Automatic" and the functionality (when receiving process data from multiple sources) is application dependent.

For example, if a fieldbus option board has been installed to slot E and it is used with to control the drive, user can select "Slot E" as value for this parameter. Now only the process data from the option board in slot E is forwarded to the application. All other fieldbuses will still receive process data out. With this setting it is possible to prevent that the monitoring fieldbuses accidentally write process data in.

Table 99.

| Value name | Value | Description |
|-------------|-------|--|
| Automatic | 1 | Process data from all sources is forwardwed to application |
| Slot D | 2 | Only process data from slot D is forward to application. Value is visible only, if option board is installed to slot D. |
| Slot E | 3 | Only process data from slot E is forward to application. Value is visible only, if option board is installed to slot E. |
| RS485 | 4 | Only process data from VACON® 100 family internal RS 485 protocol is forwarded to application |
| PROFINET IO | 5 | Only process data from VACON® 100 family internal PROFI-NET IO protocol is forwarded to application |
| EtherNet/IP | 6 | Only process data from VACON® 100 family internal Ether-Net/IP protocol is forwarded to application |

Table 99.

| Value name | Value | Description |
|----------------|-------|---|
| Modbus TCP/UDP | 7 | Only process data from VACON® 100 family internal Modbus TCP/UDP protocol is forwarded to application |
| Bacnet IP | 8 | Only process data from VACON® 100 family internal Bacnet IP protocol is forwarded to application |

11.2 RESPONSE TO FIELD BUS FAULT

In case of a fieldbus fault (loss of connection etc.), a fieldbus fault is triggered. This fault can be parameterized in application to result in a desired response. Always check the application specific manual for details as responses vary between used applications. For common fault responses used commonly in standard applications, see the table below.

Table 100. Response to fieldbus fault in VACON® AC drives

| ID | AC Drive | Value | Default | Panel Tree |
|-----|-------------------|--|---------|------------|
| 733 | VACON® 100 family | 0 = No action 1 = Alarm 2 = Alarm + preset frequency | 3 | P3.9.1.6 |
| | VACON® 20 | 3 = Fault: Stop function 4 = Fault: Coast | | P13.19 |
| | VACON® 20 X | 0 = No action 1 = Warning 2 = Fault | 2 | P9.15 |
| | VACON® NXP | 0 = No action 1 = Warning 2 = Fault: Stop function 3 = Fault: Coast | 2 | P2.7.22 |

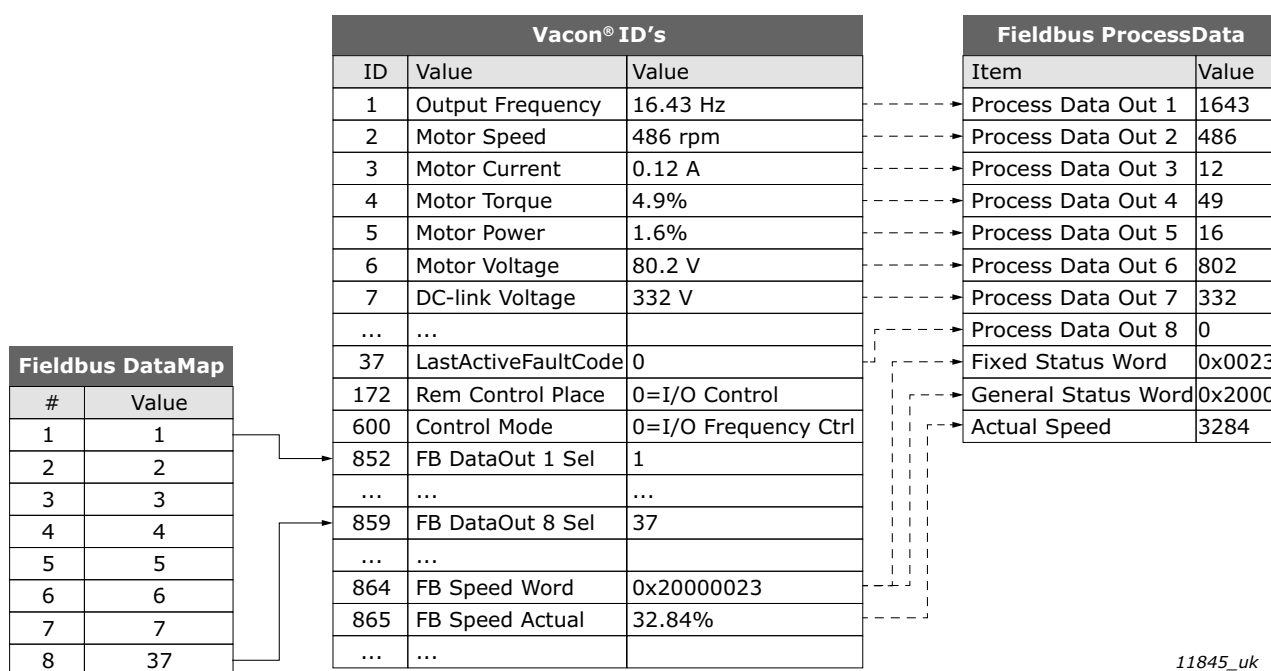
12. APPENDIX C - FIELDBUS PROCESS DATA MAPPING AND SCALING

Fieldbus process data items can be used to write and read variables quickly and periodically to/from VACON® AC drives. Incoming process data can be used for multiple different purposes (e.g. torque reference), and outgoing process data can be used for information about the state of the AC drive.

For fast access to any VACON® AC drive application ID over any fieldbus, generic Process Data Out parameters are defined. The content of the process data items are selected with the FB DataOut Sel parameters. Writing any application ID number to these parameters will then map the data value of that application ID to be sent in the corresponding Process Data Out variable.

Figure 26 illustrates how the fieldbus data mapping (FB DataOut x Sel) affects the data of the corresponding Process Data Out variable. By writing ID value 1 to FB DataOut1 Sel (ID 852), the value of ID 1 (Output Frequency) is sent in Process Data Out 1. The value is always raw value in process data out, so e.g. 16.43 Hz has value 1643. The scaling of the parameters can be checked from application manuals.

The status word and actual speed values cannot be changed, however the values sent by the fieldbus protocol might differ if a profile is used. In Bypass mode, these values are given as is.



11845_uk

Figure 26. Fieldbus Datamapping

Table 101. Fieldbus Process Data Selection Panel Tree for VACON® AC drives

| Parameter name | ID | Panel Tree | | | |
|-------------------------|-----|----------------------|------------------|--------------|-------------------|
| | | VACON® 100 family | VACON® NXP ** | VACON® 20 | VACON® 20 X/CP |
| FB DataOut 1 Selection | 852 | P3.6.1 | P2.13.3 | P10.1 | P11.1 |
| FB DataOut 2 Selection | 853 | P3.6.2 | P2.13.4 | P10.2 | P11.2 |
| ... | ... | ... | ... | ... | ... |
| FB DataOut 8 Selection | 859 | P3.6.8 | P2.13.10 | P10.8 | P11.8 |
| FB DataOut 9 Selection* | * | - | * | - | - |

Table 101. Fieldbus Process Data Selection Panel Tree for VACON® AC drives

| Parameter name | ID | Panel Tree | | | |
|--------------------------|-----|----------------------|------------------|--------------|-------------------|
| | | VACON® 100 family | VACON® NXP ** | VACON® 20 | VACON® 20 X/CP |
| FB DataOut 10 Selection* | * | - | | - | - |
| ... | ... | - | ... | - | - |
| FB DataOut 16 Selection* | * | - | | - | - |

* Supported in VACON® NXP when Fast mode or Normal extended mode is enabled. See details in Chapters APPENDIX E - FIELDBUS OPTION BOARD COMMUNICATION and APPENDIX F - PARAMETERS FOR APPLICATION DEVELOPERS.

** Multipurpose application

Table 102. Default process data mapping for VACON® 100 family and VACON® NXP

| VACON® 100 family | | | | | VACON® NXP | | | | |
|-------------------|-------------------------|----|------|---------|------------|-------------------------|----|------|---------|
| PD | Mapped Application Data | ID | Unit | Scale | PD | Mapped Application Data | ID | Unit | Scale |
| 1 | Output Frequency | 1 | Hz | 0.01 Hz | 1 | Output Frequency | 1 | Hz | 0.01 Hz |
| 2 | Motor Speed | 2 | rpm | 1 rpm | 2 | Motor Speed | 2 | rpm | 1 rpm |
| 3 | Motor Current | 3 | A | Varies* | 3 | Motor Current | 45 | A | 0.1 A |
| 4 | Motor Torque | 4 | % | 0.1 % | 4 | Motor Torque | 4 | % | 0.1 % |
| 5 | Motor Power | 5 | % | 0.1 % | 5 | Motor Power | 5 | % | 0.1 % |
| 6 | Motor Voltage | 6 | V | 0.1 V | 6 | Motor Voltage | 6 | V | 0.1 V |
| 7 | DC Link Voltage | 7 | V | 1 V | 7 | DC Link Voltage | 7 | V | 1 V |
| 8 | Last Active Fault Code | 37 | - | - | 8 | Last Active Fault Code | 37 | - | - |

*Scaling is based on drive nominal power. Scaling can be seen from Table 41.

Table 103. Default process data mapping for VACON® 20 X/CP and VACON® 20

| VACON® 20 X/CP | | | | | VACON® 20 | | | | |
|----------------|-------------------------|----|------|---------|-----------|-------------------------|----|------|---------|
| PD | Mapped Application Data | ID | Unit | Scale | PD | Mapped Application Data | ID | Unit | Scale |
| 1 | Output Frequency | 1 | Hz | 0.01 Hz | 1 | Frequency Reference | 25 | Hz | 0.01 Hz |
| 2 | Motor Speed | 2 | rpm | 1 rpm | 2 | Output Reference | 1 | Hz | 0.01 Hz |
| 3 | Motor Current | 3 | A | Varies* | 3 | Motor Speed | 2 | rpm | 1 rpm |
| 4 | Motor Torque | 4 | % | 0.1 % | 4 | Motor Voltage | 6 | V | 0.1 V |
| 5 | Motor Power | 5 | % | 0.1 % | 5 | Motor Torque | 4 | % | 0.1 % |
| 6 | Motor Voltage | 6 | V | 0.1 V | 6 | Motor Current | 3 | A | Varies* |
| 7 | DC Link Voltage | 7 | V | 1 V | 7 | Motor Power | 5 | % | 0.1 % |
| 8 | Last Active Fault Code | 37 | - | - | 8 | DC Link Voltage | 7 | V | 1 V |

* Scaling is based on drive nominal power. Scaling can be seen from Table 104.

Table 104. Current scaling based on nominal power

| Nominal power | Current scale |
|---------------|---------------|
| < 5 kW | 0.01 A |
| 5 - 100 kW | 0.1 A |
| > 100 kW | 1 A |

13. APPENDIX D - CONTROL AND STATUS WORD

13.1 CONTROL WORD BIT DESCRIPTION

The Control word is composed of 32 bits. FBFixedControlWord consist of the first 16 bits. FBGeneralControlWord consist of the remaining 16 bits. While the functionality of FBFixedControlWord is fixed in the VACON® standard applications, the functionality of FBGeneralControlWord is totally application specific and can vary even in the VACON® standard applications.

The meanings of FBFixedControlWord bits are described below. Unused bits have to be set to zero.

NOTE! This table is valid for VACON® standard applications.

NOTE! There are some control word bit modifications in VACON® NXP AC drive. These modifications are described in Table 106. Unused bits have to be set to zero.

Table 105. FBFixedControlWord bits

| Bit | Function | | Description |
|-----|----------------------------|---|---|
| B0 | Start/Stop | 0 | Stop request from fieldbus. |
| | | 1 | Run request from fieldbus. |
| B1 | Direction | 0 | Requested direction is "FORWARD". |
| | | 1 | Requested direction is "REVERSE". |
| B2 | Fault reset | 0 | No action. |
| | | 1 | No action. Rising edge (0->1) = Active faults, alarms and infos are reset. |
| B3 | Stop mode 1 | 0 | Stop mode is unmodified. |
| | | 1 | Stop mode is overridden to "Coasting". |
| B4 | Stop mode 2 | 0 | Stop mode is unmodified. |
| | | 1 | Stop mode is overridden to "Ramping". |
| B5 | Quick ramp time | 0 | Normal deceleration ramp time. |
| | | 1 | Deceleration ramp time is switched to shorter than normal. |
| B6 | Freeze Setpoint | 0 | Changes in the setpoint value from fieldbus (FB Speed Reference) are taken into use by the application. |
| | | 1 | Changes in the setpoint value from fieldbus (FB Speed Reference) are not taken into use by the application. |
| B7 | Setpoint to Zero | 0 | The setpoint value from fieldbus is taken from FB Speed Reference. |
| | | 1 | The setpoint value from fieldbus is changed to 0. |
| B8 | Request Fieldbus Control | 0 | Control Place is as parameterized in the drive (unchanged). |
| | | 1 | Control Place is overridden to Fieldbus Control. |
| B9 | Request Fieldbus Reference | 0 | Source of the setpoint value is as parameterized in the drive (unchanged). |
| | | 1 | Source of the setpoint value is overridden to Fieldbus. |

Table 105. FBFixedControlWord bits

| Bit | Function | | Description |
|-----|------------|---|---|
| B10 | Jogging 1 | 0 | No action. |
| | | 1 | Jogging request with ref1. |
| B11 | Jogging 2 | 0 | No action. |
| | | 1 | Jogging request with ref2. |
| B12 | Quick stop | 0 | Drive operates as normal. |
| | | 1 | Drive executes quick stop / emergency stop. |
| B13 | Reserved | 0 | - |
| | | 1 | - |
| B14 | Reserved | 0 | - |
| | | 1 | - |
| B15 | Reserved | 0 | - |
| | | 1 | - |

Table 106. FBFixedControlWord modifications in VACON® NXP

| Bit | Function | Value | Description |
|-----|----------------|-------|--------------------|
| 3 | Fieldbus DIN 1 | 0 | Fieldbus DIN 1 off |
| | | 1 | Fieldbus DIN 1 on |
| 4 | Fieldbus DIN 2 | 0 | Fieldbus DIN 2 off |
| | | 1 | Fieldbus DIN 2 on |
| 5 | Fieldbus DIN 3 | 0 | Fieldbus DIN 3 off |
| | | 1 | Fieldbus DIN 3 on |
| 6 | Fieldbus DIN 4 | 0 | Fieldbus DIN 4 off |
| | | 1 | Fieldbus DIN 4 on |
| 7 | Fieldbus DIN 5 | 0 | Fieldbus DIN 5 off |
| | | 1 | Fieldbus DIN 5 on |

13.2 STATUS WORD DESCRIPTIONS

The Status word is composed of 32 bits. FBFixedStatusWord consist of the first 16 bits. FBGeneralStatusWord consist of the remaining 16 bits. While the functionality of FBFixedStatusWord is fixed in the VACON® standard applications, the functionality of FBGeneralStatusWord is totally application specific and can vary even in the VACON® standard applications.

The meanings of FBFixedStatusWord bits are described below. Unused bits have to be set to zero.

In VACON® NX series AC drives the FBFixedStatusWord comes from firmware variable "MCStatus".

Table 107. FBFixedStatusWord bits

| Bit | Function | | Description |
|--------|--------------|---|--|
| B0 | Ready | 0 | Drive is not ready. |
| | | 1 | Drive is ready to run. |
| B1 | Run | 0 | Motor is not running. |
| | | 1 | Motor is running. |
| B2 | Direction | 0 | Motor is running clockwise. |
| | | 1 | Motor is running counterclockwise. |
| B3 | Fault | 0 | No fault active. |
| | | 1 | Drive has an active fault. |
| B4 | Alarm | 0 | No alarm active. |
| | | 1 | Drive has active alarm. |
| B5 | At reference | 0 | Motor is not running at reference speed. |
| | | 1 | Motor is running at reference speed. |
| B6 | Zero speed | 0 | Motor is not at zero speed. |
| | | 1 | Motor is running at zero speed. |
| B7 | Flux ready | 0 | Motor is not magnetized. |
| | | 1 | Motor is magnetized. |
| B8-B12 | Reserved | | |

13.3 CONTROL WORD BIT SUPPORT IN DRIVES

This table describes the control word bit support in different AC drives. Notice that this table is valid only for the VACON® standard applications. Always check the application-specific status from the application manual.

Table 108. Control word

| Bit | Function | VACON® 100 family | VACON® NXS/ NXP | VACON® 20 | VACON® 20 X |
|-----|----------------------------|-------------------------|-----------------------|--------------|----------------|
| B0 | Start/Stop | x | x | x | x |
| B1 | Direction | x | x | x | x |
| B2 | Fault reset | x | x | x | x |
| B3 | Stop mode 1 | x | | | x |
| B4 | Stop mode 2 | x | | | x |
| B5 | Quick ramp time | x | | x | x |
| B6 | Freeze setpoint | x | | | x |
| B7 | Setpoint to zero | x | | | x |
| B8 | Request fieldbus control | x | x | | x |
| B9 | Request fieldbus reference | x | x | | x |
| B10 | Jogging 1 | x | | | |
| B11 | Jogging 2 | x | | | |
| B12 | Quick stop | x | | | x |
| B13 | Reserved | | | | |
| B14 | Reserved | | | | |
| B15 | Reserved | | | | |

13.4 STATUS WORD BIT SUPPORT IN DRIVES

This table describes the status word bit support in different drives. Notice that this table is valid only for the VACON[®] standard applications. Always check the application-specific status from the application manual.

Table 109. Status word

| Bit | Function | VACON [®] 100 family | VACON [®] NXS/ NXP | VACON [®] 20 | VACON [®] 20 X |
|-----|--------------|-------------------------------------|-----------------------------------|--------------------------|----------------------------|
| B0 | Ready | x | x | x | x |
| B1 | Run | x | x | x | x |
| B2 | Direction | x | x | x | x |
| B3 | Fault | x | x | x | x |
| B4 | Alarm | x | x | x | x |
| B5 | At reference | x | x | x | x |
| B6 | Zero speed | x | x | | x |
| B7 | Flux ready | x | x | | |
| B8 | Reserved | | | | |
| B9 | Reserved | | | | |
| B10 | Reserved | | | | |
| B11 | Reserved | | | | |
| B12 | Reserved | | | | |
| B13 | Reserved | | | | |
| B14 | Reserved | | | | |
| B15 | Reserved | | | | |

14. APPENDIX E - FIELDBUS OPTION BOARD COMMUNICATION

The different communication modes can be enabled for fieldbus option board for different features. There are different modes available for different setups:

- Normal mode, for most commonly used setups. 8 process data items.
- Normal extended mode, for setups that requires 16 process data items.
- Fast mode, with low latency process data. 16 process data items.
- Fast safety mode, with safety "black channel" and low latency process data. 16 process data items.
- Fast PROFIBUS mode, for backward compatibility. 8 process data items.

The fast communication modes can be enabled to get minimum communication delay between fieldbus and application.

NOTE! OPTE6 CANopen does not support Fast safety mode and Fast PROFIBUS mode.

14.1 REQUIREMENTS FOR COMMUNICATION MODES

The following table describes the required components for different communication modes:

Table 110. Requirements for different fieldbus communication modes

| | Fast / Normal Extended | Fast safe | Fast PROFIBUS |
|------------------------|---|-------------------------------|---------------------------------|
| Control Board | NXP (serial no. 761 or later) | NXP (serial no. 761 or later) | NXP (serial no. 561 or later) |
| System Software | NXP00003V196.VCN or later | NXP00003V194.VCN or later | NXP00002V171.VCN or later |
| | | | NXP00003V179.VCN or later |
| Applications | Multipurpose V236 or later (Normal Extended Mode) | Any* | System Interface V110 or later |
| | | | Advanced V085 or later |
| | | | Marine V107 or later |
| Fieldbus option slot | OPTE3-E5, V006 or later | OPTE3-E5, V006 or later | OPTC3_10502V014.vcn |
| | OPTE9, V007 or later | - | OPTC3-5_FW0232V001.vcx or later |
| | OPTEA, V001 or later | OPTEA, V001 or later | |
| | OPTEC, V003 or later | - | OPTEC, V001 or later |
| | OPTE6, V010 or later | - | - |
| Advanced safety option | - | OPTBL_FW0227V001 or later | - |

* If safety option is configured to use a safety fieldbus, the fast safe mode is automatically enabled regardless of used application. However, the availability of 16 process items is limited by the application in use. Also the process data application cycle is normally set to 10 ms, instead of 1 ms for fast application.

Refer to application specific manuals for latest information about application support for fieldbus communication modes.

14.2 FIEDLBUS COMMUNICATION MODE FEATURES AND LIMITATIONS

Fast mode:

- 1 ms process data interval
- Available in VACON® NXP slots D and E
 - Possible to run both slots simultaneously
 - Have similar process data latency in both slots
- Service data latency is also reduced
 - Running multiple service data queries at high interval can cause high CPU load in VACON® NXP AC drive.

Fast safe mode:

- 1 ms process data interval
- Includes safety "black channel"
- Activated/deactivated automatically, cannot be set by user or application
- Fast safe mode is available only in slot E
 - Advanced safety option board must be installed into slot D
 - Safety fieldbus must be activated in safety configuration

16 process data items:

- 16 process data items always require support from application
- Available in Fast, Fast safe and Normal extended mode
- If no support is available in the application, the process data out is always '0', while incoming process data items 9-16 are discarded

14.3 NORMAL FIELDBUS COMMUNICATION

The normal fieldbus communication between option board and the AC drive application is visible in Figure 27. In normal communication both process data and service data are transferred in succession at 5 ms interval.

Communication delay for process data can be calculated by summing all delays together:

$$t = t_{IOdatacycle} + t_{updateinterval} + 2 \cdot t_{communicationdelay} + t_{applicationcycle}$$

Example: With fieldbus cycle time of 4 ms and application cycle of 10 ms, the delay is:

$$t = 4ms + 10ms + (2 \cdot 5)ms + 10ms = 34ms$$

NOTE: This value does not include delays of the fieldbus master, jitter in the process data cycle of the communication protocol or resending due to electronic interference.

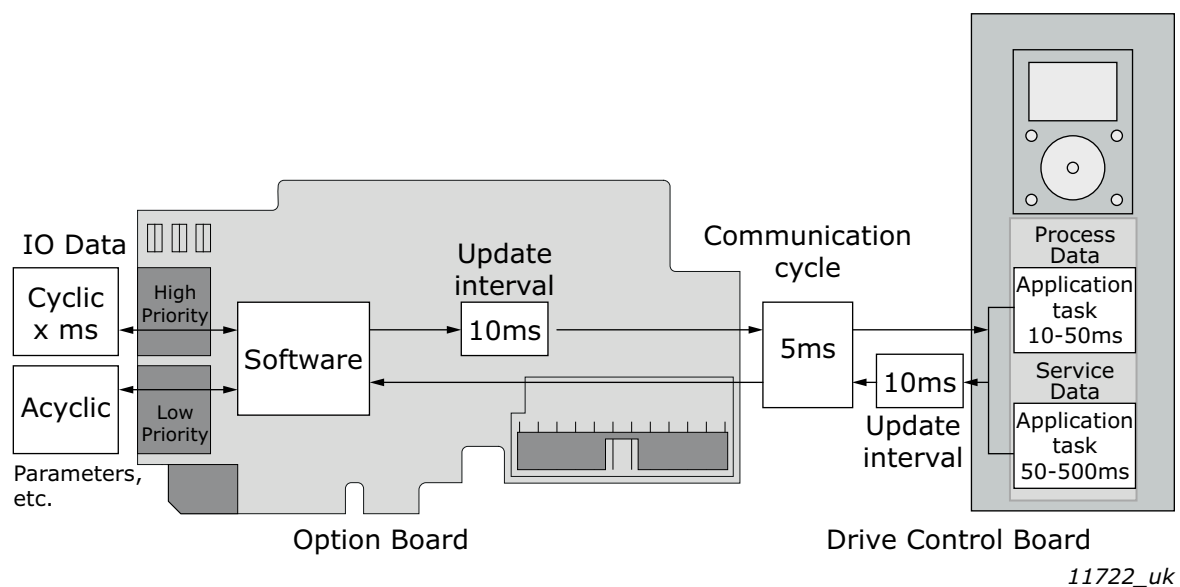


Figure 27. Normal fieldbus communication

14.4 FAST FIELDBUS COMMUNICATION

The fast mode decreases the communication delay between the PLC and the AC drive application significantly by using two communication channels separately for process and service data. The process data interval is set to 1 ms, while other data is sent acyclically. When the fast mode is activated, the application can be synchronized to run with the communication cycle. The Fast communication mode is visible in Figure 28. This mode also includes the ability to transfer 16 process data items.

The communication delay for process data in fast communication mode is (when application task is synchronized with communication):

$$t = t_{IOdata\ cycle} + t_{update\ interval} + t_{application\ cycle}$$

Example: With fieldbus cycle time of 1 ms an application cycle of 1 ms the delay is::

$$t = 1\ ms + 1 + 1\ ms = 3\ ms$$

NOTE: This value does not include delays of the fieldbus master, jitter in the process data cycle of the communication protocol or resending due to electronic interference.

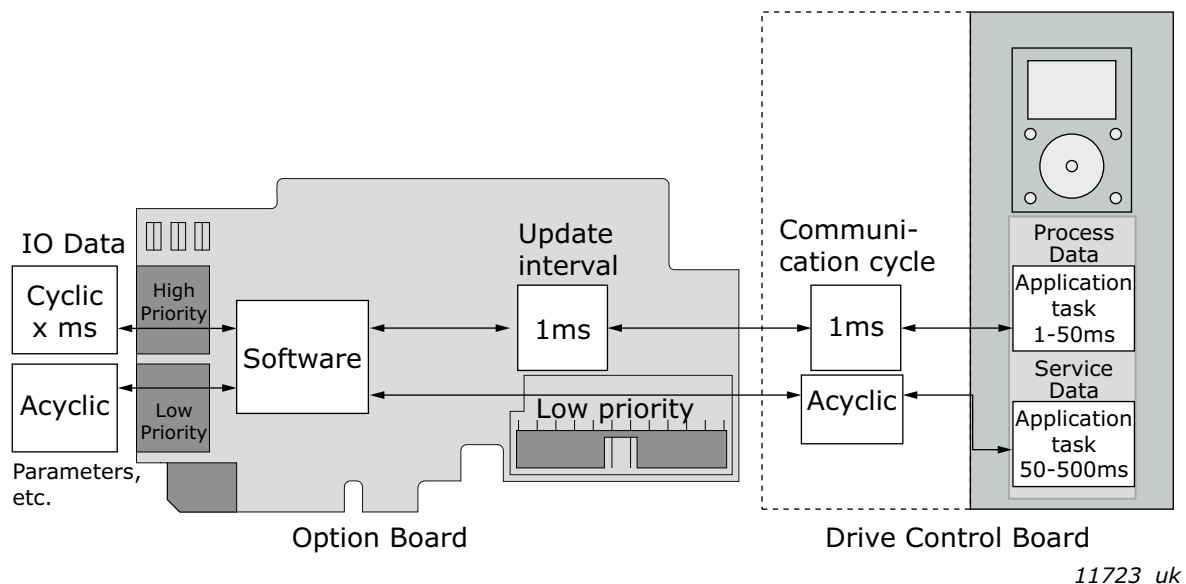


Figure 28. Fast fieldbus communication

14.5 NORMAL EXTENDED MODE

The normal extended mode uses the same communication method as in "Fast mode", but reduces the communication cycle to 10ms.

This can be used in applications where 16 process data items are required but lowest possible communication delay is not needed or the increased CPU load of Fast mode to VACON[®] NXP drives is undesirable.

NOTE! This mode can be automatically enabled in VACON[®] applications supporting 16 process data items.

15. APPENDIX F - PARAMETERS FOR APPLICATION DEVELOPERS

This appendix gives information for the application developers and system integrators on the VACON® NXP system software variables used to activate and control different fieldbus communication modes and features.

Table 111. System software variables for selecting communication modes

| Parameter | Value | Default |
|------------------|---|---------|
| FBModeSlotD_fwu8 | 0 = Normal mode | 0 |
| FBModeSlotE_fwu8 | 1 = Fast safety mode* 2 = Fast mode 3 = Fast PROFIBUS mode ** 4 = Normal extended mode | 0 |

* Fast safety mode is automatically enabled/disabled by system software. Cannot be set by user. This mode is not supported in OPTE6 CANopen.

** Fast PROFIBUS mode is not supported in OPTE6 CANopen.

FBModeSlotX_fwu8 variables are used to select the active fieldbus option board communication mode. If no fieldbus option board is connected to the related slot, the selection of the FBModeSlot parameter is set to 0 = Normal mode.

Table 112. System software variables for monitoring supported communication modes

| Parameter | Value | Default |
|---------------------------|---|---------|
| FBModeSlotDSupModes_fwu16 | 0x00 = Not yet updated. Read again later 0x01 = Fieldbus communication not supported 0x02 = Normal mode supported | 0 |
| FBModeSlotESupModes_fwu16 | 0x04 = Fast safety mode supported 0x08 = Fast mode supported 0x10 = Fast PROFIBUS mode supported 0x20 = Normal extended mode supported | 0 |

FBModeSlotXSupModes_fwu16 variables can be used to determine the different supported modes of the fieldbus option boards. All features are set as bit fields as multiple modes can be supported.

Value '0' is returned while the feature set of the option board is not yet retrieved. Value should be asked again. Any option board not supporting fieldbus communication returns value '1'.

Example 1: OPTE3-E5_FW0083V006 PROFIBUS board returns value: 0x2A, indicating support for Normal, Fast and Normal extended modes.

Example 2: OPTE9_FW0196V006 Dual Port Ethernet board returns value: 0x0A, indicating support for Normal and Fast modes.

Table 113. System software variables for selecting the input process data slot

| Parameter | Value | Default |
|----------------------------|---|---------|
| FBControlSlotSelector_fwu8 | 0 = All slots 4 = Slot D only 5 = Slot E only 6 = Fast PROFIBUS D slot* 7 = Fast PROFIBUS E slot* | 0 |

*For backward compatibility only. Same as FBModeSlotX_fwu8 variable setting '3'. Fast PROFIBUS is not supported in OPTE6 CANopen.

FBControlSlotSelector_fwu8 variable is used to select the controlling fieldbus option board slot. When selected (other than '0'), process data is accepted only from the selected slot and all other process data is discarded. Process data out is still updated normally to all slots.

This selector can be used to support redundant fieldbus connection. In fieldbus redundancy mode two fieldbus option boards are installed to VACON® NXP option board slots D and E. Application selects with FBControlSlotSelector_fwu8 variable which fieldbus option board can deliver process data from fieldbus master to the application.

Default value for FBControlSlotSelector_fwu8 is '0' which means that process data is accepted from both fieldbus option boards.

VACON®

www.danfoss.com

Vacon Ltd
Member of the Danfoss Group
Runsorintie 7
65380 Vaasa
Finland

Document ID:



DPD01091D

Rev. D

Sales code: DOC-OPTE6+DLUK