Heinzmann GmbH & Co. KG
Engine & Turbine Controls
Am Haselbach 1
D-79677 Schönau (Schwarzwald)
Germany
Phone +49 7673 8208-0
Fax +49 7673 8208-188
E-mail info@heinzmann.com
www.heinzmann.com
V.A.T. No.: DE145551926

HEINZMANN®

Digital Electronic Speed Governors

Digital Basis System

CANopen Gateway

for
Digital Governor

Manual DG 04 005-e / 07-04

 **Warning**	Read this entire manual and all other publications appertaining to the work to be performed before installing, operating or servicing your equipment. Practice all plant and safety instructions and precautions.
 **Danger**	Failure to follow instructions may result in personal injury and/or damage to property.
 **Attention! High Voltage**  **Danger**	**Please note before commissioning the installation:** Before starting to install any equipment, the installation must have been switched dead. Be sure to use cable shieldings and power supply connections meeting the requirements of the *European Directive Concerning EMI.* Check the functionability of the existing protection and monitoring systems.
 **Danger**	**To prevent damages to the equipment and personal injuries, it is imperative that the following monitoring and protection systems be installed:** Overspeed protection acting independently of the speed governor Overtemperature protection **Generator installations will in addition require:** Overcurrent protection Protection against faulty synchronization due to excessive frequency, voltage or phase differences Reverse power protection
	Overspeeding can be caused by: Failure of the voltage supply Failure of the control unit or of any accessory device Failure of the actuator Sluggish and blocking linkage

 **Warning**	With electronically controlled injection (MVC) the following provisions must in addition be observed: With **Common Rail** systems, a separate mechanical flow limiter must be provided for every injector pipe. With **Pump-Pipe-Nozzle (PPN)** and **Pump-Nozzle (PNE)** systems, fuel delivery may be enabled only with the control piston of the magnetic valve moving. By this, fuel delivery to the injection nozzle will be inhibited should the control piston be blocked or jamming.
 **Warning**	Any examples, data and any other information contained in this manual are intended exclusively as instruction aids and should not be used in any particular application without independent testing and verification by the person making the application.
 **Danger**	Independent testing and verification are especially important in any application where malfunction might result in personal injury or damage to property.
	HEINZMANN make no warranties, express or implied, that the examples, data, or other information in this volume are free of error, that they are consistent with industry standards, or that they will meet the requirements for any particular application..
	HEINZMANN expressly disclaim the implied warranties of merchantability and of fitness for any particular purpose, even if **HEINZMANN** have been advised of a particular purpose and even if a particular purpose is indicated in the manual.
	HEINZMANN also disclaim all liability for direct, indirect, incidental or consequential damages that result from any use of the examples, data, or other information contained in this manual.
	HEINZMANN make no warranties for the conception and engineering of the technical installation as a whole. This is the responsibility of the user and of his planning staff and specialists. It is also their responsibility to verify whether the performance features of our devices will meet the intended purposes. The user is also responsible for correct commissioning of the whole installation.

Table of Contents

Page

1 Safety Instructions and Related Symbols

This publication offers specific safety instructions, wherever necessary, to indicate inevitable residual risks when operating the engine. These residual risks imply dangers to

- persons
- product and engine
- environment

The symbols used in this publication are in the first place intended to direct your attention to the safety instructions!

Warning

This symbol is to indicate that there may exist dangers to the engine, to the material and to the environment.

Danger

This symbol is to indicate that there may exist dangers to persons. (Danger to life, personal injury).

Attention! High Voltage

This symbol is to indicate that there exist particular dangers due to electrical high tension. (Mortal danger).

Note

This symbol does not refer to any safety instructions but offers important notes for better understanding the functions that are being discussed. They should at any rate be observed and practiced. The respective text is printed in italics.

The primary issue of these safety instructions is to prevent personal injuries.

Whenever some safety instruction is preceded by a warning triangle labelled “Danger” this is to indicate that it is not possible to definitely exclude the presence of danger to persons, engine, material and/or environment.

If, however, some safety instruction is preceded by the warning triangle labelled “Warning” this will indicate that danger of life or personal injury is not involved.

The symbols used in the text do not supersede the safety instructions. So please do not skip the respective texts but read them thoroughly.

In this publication the Table of Contents is preceded by diverse instructions that among other things serve to ensure safety of operation. It is absolutely imperative that these notes be read and understood before commissioning or servicing the installation.

1.1 Basic Safety Measures for Normal Operation

- The installation may be operated only by authorized persons who have been duly trained and who are fully acquainted with the operating instructions so that they are capable of working in accordance with these instructions.
- Before turning the installation on please verify and make sure that
  - only authorized personnel are present within the working range of the engine;
  - nobody will be in danger of suffering injuries by starting the engine.
- Before starting the engine always check the installation for visible damages and make sure it is not put into operation unless it is in perfect condition. On detecting any faults please inform your principal immediately!
- Before starting the engine remove any unnecessary material and/or objects from the working range of the installation/engine.
- Before starting the engine check and make sure that all safety devices are working properly.

1.2 Basic Safety Measures for Servicing and Maintenance

- Before performing any maintenance or repair work make sure the working area of the engine has been closed to unauthorized persons. Put on a sign warning that maintenance or repair work is being done.
- Before performing any maintenance or repair work switch off the master switch of the power supply and secure it by a padlock! The key must be kept by the person performing the maintenance and repair works.
- Before performing any maintenance and repair work make sure that all parts of engine to be touched have cooled down to ambient temperature and are dead!
- Refasten loose connections!
- Replace at once any damaged lines and/or cables!
- Keep the cabinet always closed. Access should be permitted only to authorized persons having a key or tools.

- Never use a water jet to clean cabinets or other casings of electric equipment!

1.3 Before Putting an Installation into Service after Maintenance or Repair Works

- Check on all slackened screw connections to have been retightened.
- Make sure the control linkage has been reattached and all cables have been reconnected.
- Make sure all safety devices of the installation are in perfect order and are working properly.

2 General Description

CANopen Gateway allows to integrate a HEINZMANN Digital Governor as a CANopen Slave device into a CANopen network. It constitutes a CAN interface between the control unit and other devices included in the CANopen network supporting either directions of communication. Via CANopen process data can both be read and written by the control unit.

CANopen Gateway is based on a separate microprocessor having 2 CAN interfaces. The CAN1 interface serves for communication between the Gateway and the control unit on the basis of the HEINZMANN CAN protocol whereas the CAN2 interface is connected with the CANopen network. The Baud rates of the CAN1 and CAN2 interfaces are independent of each other and may be freely parameterized.

By CANopen Gateway it is possible to limit the additional stress of the Digital Governor due to a new CAN protocol since it takes over many of the functions defined in the CANopen protocol such as the CANopen State Machine, the Object Dictionary and SDO communication. The Gateway also guarantees a strict isolation between the HEINZMANN CAN network and the CANopen network. Therefore, any disturbances of either CAN network will have no consequences for the other, and there is but a moderate increase of bus load of the HEINZMANN CAN network since only telegrammes concerning the Digital Governor will be transmitted by the Gateway.

These are the main features of the CANopen Gateway:

- CANopen protocol according to CiA Draft Standard DS301 Version 4.01 „CANopen Application Layer and Communication Profile“.
- Since the CAN Extended Format (29 bits identifiers) is only very rarely used by CANopen, it is only the CAN Standard Format (11 bits identifiers) that is supported by the Gateway for CANopen communication.
- Both directions of communication are supported. Data within the Digital Governor may also be written. (Such write data are CANopen sensors and CANopen switch functions.)
- Since the majority of data exchanged may be characterized as process data, they will be transmitted and/or received via PDO's. For this purpose, the Gateway supports 16 TPDO's and 4 RPDO's. The Object Dictionary, of course, can also be read via SDO.
- Configuration of the CANopen communication in the Gateway (node number, Baud rate, identifier, etc.) is made by means of DcDesk 2000. However, the storing and loading of parameters via CANopen (objects 0x1010 and 0x1011) and the Layer Setting Service are not supported.
- As a monitoring function the heartbeat function is provided.

Figure 1 offers an overview of the concept of the CANopen Gateway.

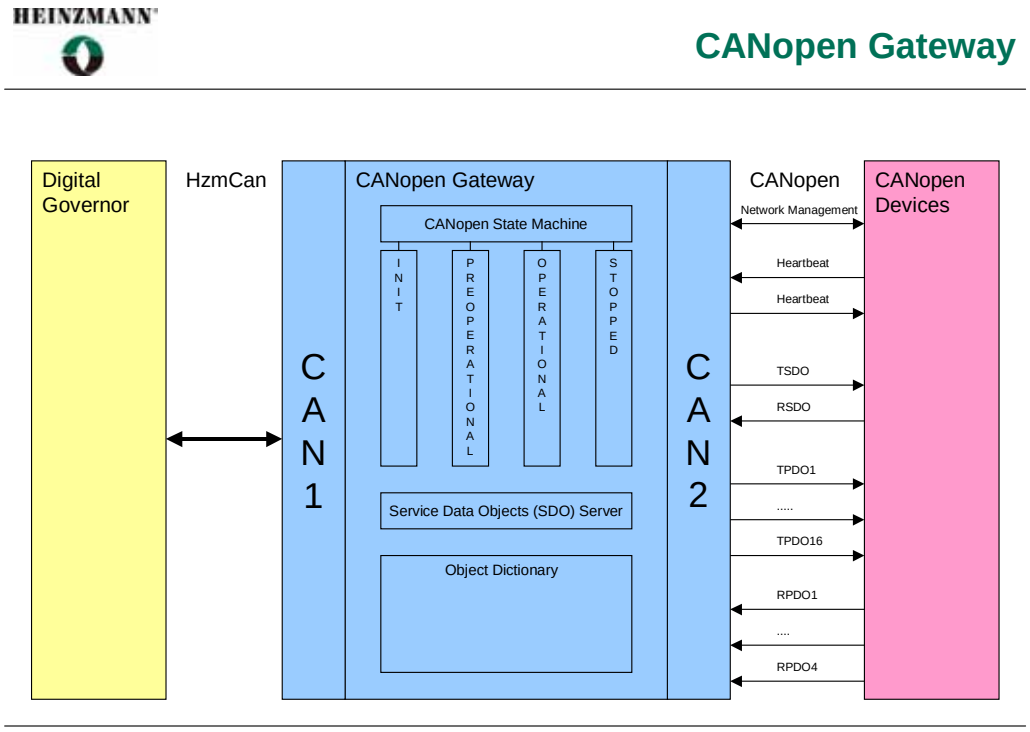


Figure 1: Overview CANopen Gateway

There have new functionalities been implemented both in the Digital Governor and in the Gateway for CANopen Gateway communication. These functionalities are summarized in Table 1.

Functionalities in the Digital Governor	Functionalities in the CANopen Gateway
Selection of the parameter for the TPDO's. Selection of the CANopen sensors. Selection of the CANopen switch functions. Determination of the cycle times for the TPDO's. Determination of the monitoring times for the RPDO's. Periodical sending of TDPO's. Receiving and monitoring RPDO's and assigning them to the CANopen sensors / switch functions. Gateway initialization procedure.	CanOpen State Machine. Initialization and creation of the CANopen Object Dictionary according to the initialization telegrammes of the Digital Governor. CANopen SDO Server. Periodical sending of the Tx heartbeat telegramme. Control of the Rx heartbeat telegramme of a CANopen device. Transfer of the TPDO's to the CANopen partner and of the RPDO's to the Digital Governor in operational state.

Table 1: Distribution of CANopen Functionalities

3 CANopen State Machine

CANopen communication is being controlled by a State Machine. The CANopen State Machine has been realized within the CANopen Gateway. It is, however, also reproduced within the Digital Governor. There are 5 states of the CANopen State Machine:

- Undefined (following a reset or on the occurrence of certain errors; there is no CanOpen communication active)
- Init (Initialization procedure; there is no CANopen communication active)
- Pre-Operational (NMT, SDO, heartbeat active, PDO inactive)
- Operational (NMT, SDO, heartbeat, PDO active)
- Stopped (NMT, heartbeat active, SDO, PDO inactive)

The current state of the CANopen State Machine is represented both in the Digital Governor and in the CANopen Gateway by the parameters 12401 *CanOpStopped* through 12404 *CanOpOperational*

The CANopen Gateway is receiving the CANopen State Machine commands from two sources: from the Digital Governor and from the CANopen Master. The Digital Governor is sending 3 types of commands: Undefined, Init, Pre-Operational. The CANopen Master is capable of sending 4 commands: Init, Pre-Operational, Operational, Stopped; NMT (Network Management) communication, however, will be active only with the states Pre-Operational, Operational and Stopped. The normal procedure after a reset is the following:

- Init command from the Digital Governor
- Initialization procedure (exchange of the initialization telegrammes between Digital Governor and Gateway)
- Command Pre-Operational from the Digital Governor
- Boot-up message from Gateway to CANopen Master
- Command Operational from CANopen Master

Figure 2 gives a complete overview of the CANopen State Machine and its transitions.

Figure 3 shows the Network Management Telegramm from CANopen Master to Gateway.

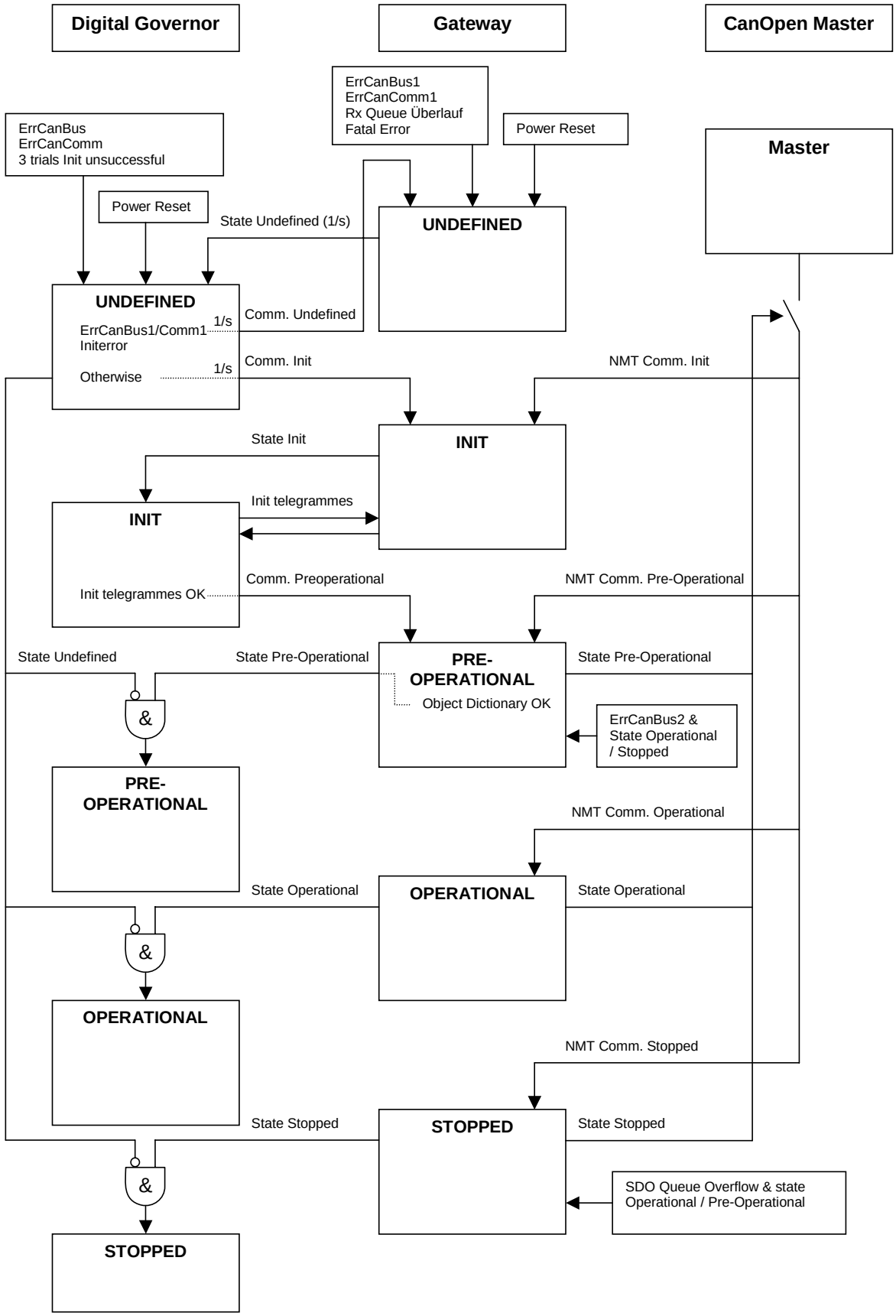


Figure 2: CANopen State Machine

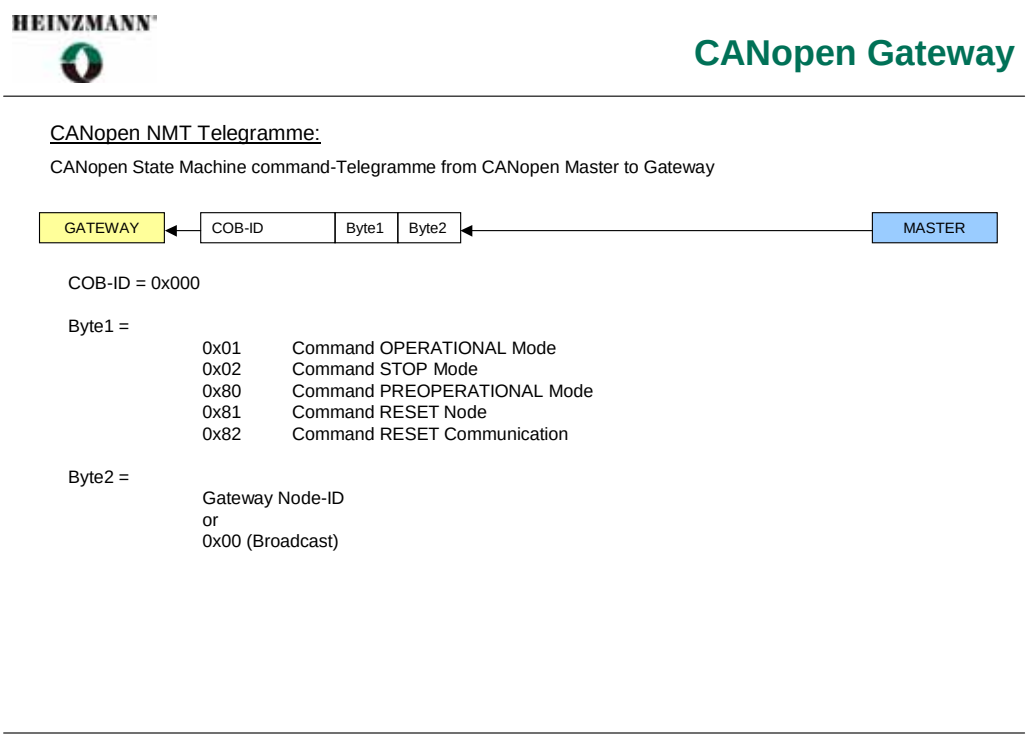


Figure 3: NMT Telegramme

4 Process Data Objects (PDO Communication)

Most of the data communicated via the CANopen Gateway in connection with a Digital Governor are process data and will be exchanged by means of PDO communication. Both directions of communication are supported (TPDO and RPDO), and both binary data and 16 bits analogue data may be transmitted in either direction.

4.1 TPDO

The CANopen Gateway supports a maximum number of 16 TPDO's. 4 of them consist of the TPDO's that by standard are pre-defined via the Gateway-CANopen node number. 12 of them can be additionally defined.

4.1.1 TPDO's for Binary Data

The TPDO's numbered 1–3 are reserved for the communication of binary data. Via TPDO1 and TPDO2 any current errors within the Digital Governor and the Gateway can be transmitted. By TPDO3, the engine state can be made available. Data communicated via TPDO1 – TPDO3 can be retrieved from the Object Dictionary as objects 2000h01-07.


Note

In contrast to the engine state, any current errors within the Digital Governor and the Gateway will be stored in the Object Dictionary of the Gateway even if transmission of these errors via TPDO has not been selected. Thus it will always be possible to read errors via SDO.

Any binary TPDO will not regularly be transmitted but only in Operational state in case the value of any of its data has changed since the last transmission, at most, however, only once per main cycle of the Digital Governor. It is not possible to define some cycle time for the TPDO's 1–3.

4.1.2 TPDO's for Analogue Data

The TPDO's 4–16 have been reserved for the communication of analogue data. ‘Analogue data’ here denotes 8 bits or 16 bits signed or unsigned parameters. Since the TPDO's support only communication of 16 bits data, 8 bits parameters will have to be converted into 16 bits. Single bits (bit parameters) cannot be transmitted by TPDO.

The assignment of each analogue TPDO is made by means of DcDesk 2000. Each TPDO data (maximum 4 data per TPDO) may be assigned some parameter number.

The objects 2001h01-52 of the Object Dictionary contain the unsigned data of the analogue TPDO's.

The objects 2002h01-52 contain the signed data.

In Operational state, the analogue TPDO's are periodically transmitted. The cycle time can individually be parameterized for every TPDO using DcDesk 2000 and may range between 1 and 255 main cycles of the Digital Governor.

Figure 4 offers an overview of TPDO communication.

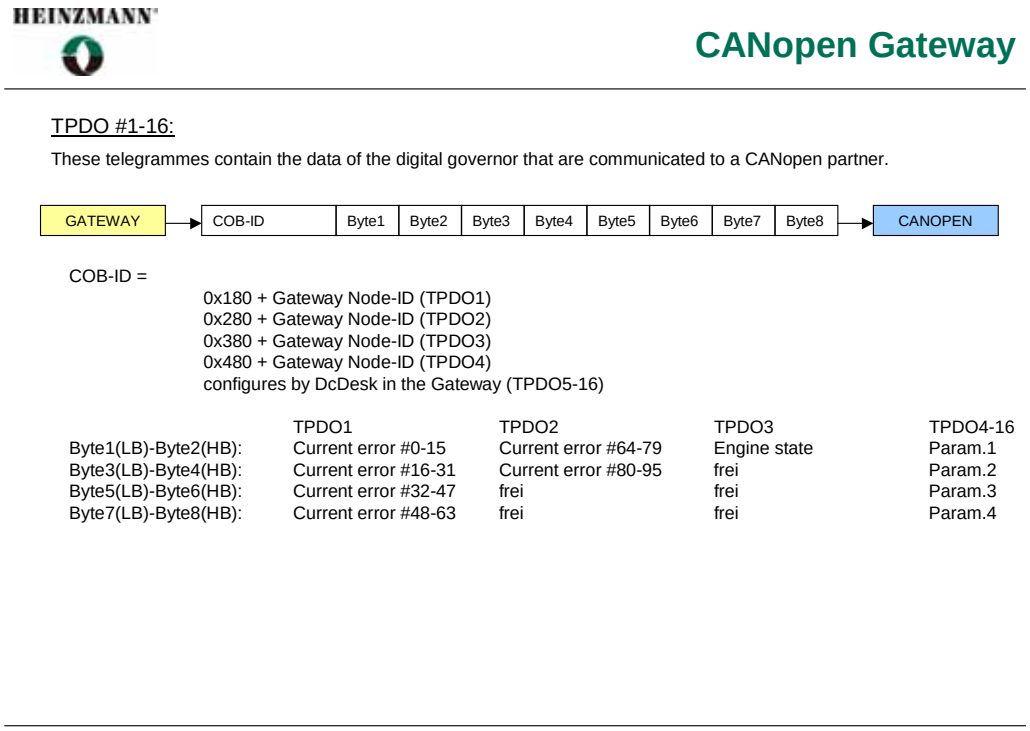


Figure 4: Overview of TPDO Communication

4.2 RPDO

CANopen Gateway supports the 4 RPDO's that have been pre-defined via the Gateway-CANopen node number. Each RPDO can be monitored with respect to time by checking the time difference between 2 telegrammes for not exceeding the monitoring period as defined by means of DcDesk 2000.

4.2.1 RPDO for Binary Data

RPDO1 has been reserved for communication of binary data. Word 1 comprises up to 16 switch functions that can be freely assigned using DcDesk 2000. Word 3 contains up to 12 CANopen sensor error flags depending on how many sensors have been defined in the RPDO's 2–4. Word 2 and Word 4 are not assigned. The binary data are stored in the Object Dictionary by the numbers 2003h01 and 2003h02.

4.2.2 RPDO for Analogue Data

It is by means of RPDO2–4 that 16 bits unsigned analogue data (object 2004h1-12) are communicated to the Digital Governor. The assignment of these analogue data to the respective sensors is achieved via DcDesk 2000. If time monitoring trips the sensors contained in the RPDO will be regarded as faulty.

Figure 5 offers an overview of TPDO communication.

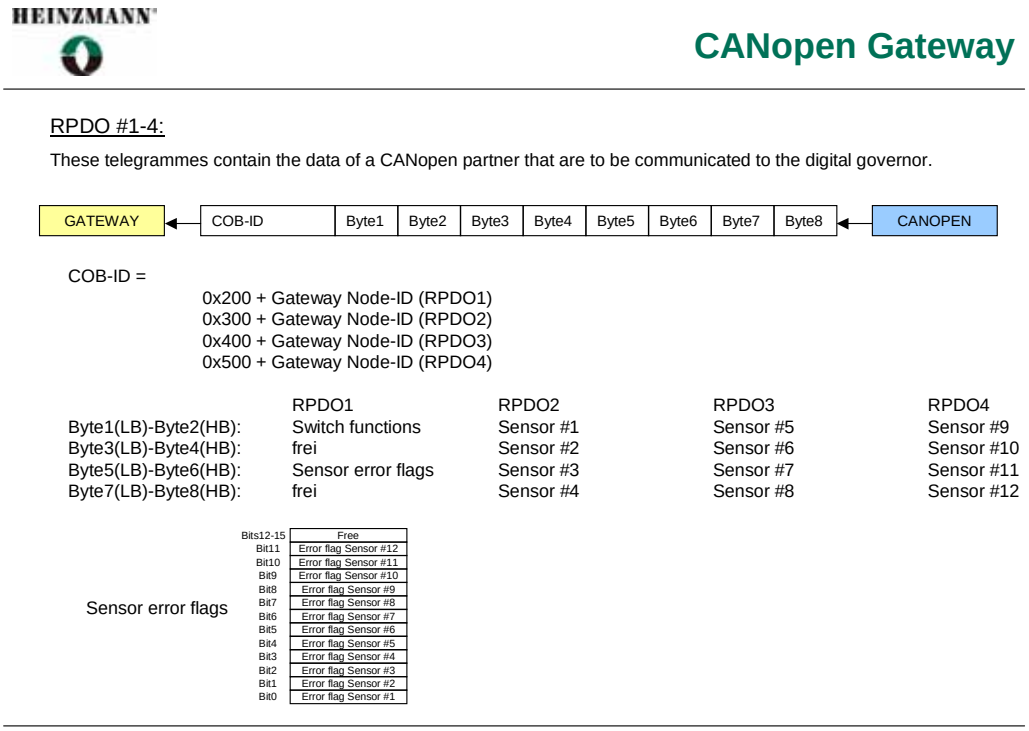


Figure 5: Overview of RDPO Communication

5 Object Dictionary and Service Data Objects (SDO Communication)

The structure and content of the Object Dictionary are easily ascertained from the EDS file of the CANopen Gateway ↑ *10 EDS File*. Since the CANopen standard objects of communication profile (objects 1000h to 1FFFh) do not present any peculiarities they will not have to be dealt with in more detail. The following sections offer some basic facts about manufacturer specific objects. All manufacturer specific objects have a size of 16 bits.

5.1 Manufacturer Specific Objects

5.1.1 Objects 2000h01-07

Each of these objects (maximum number 7) contains 16 digital outputs that can be transmitted by the Digital Governor via TPDO1-3. As digital outputs any errors within the Digital Governor and the Gateway (96 bits) as well as the engine state (8 bits) may be selected via DcDesk 2000. The objects 2000h01-06 will always be present in the Object Dictionary and contain any current errors of the Digital Governor and the Gateway even if they cannot be transmitted via TPDO. They can, however, always be read via SDO.

5.1.2 Objects 2001h01-52

These objects (maximum number 52) consist of the unsigned 16 bits analogue output data that are being transmitted by the Digital Governor via TPDO4-16.

5.1.3 Objects 2002h01-52

These objects (maximum number 52) are the signed 16 bits analogue output data that are being transmitted by the Digital Governor via TPDO4-16.

5.1.4 Objects 2003h01-02

These objects (maximum number 2) contain the digital inputs that can be transmitted to the Digital Governor via RDPO1. Digital input may consist of switch functions (RPDO1 Word1) or of sensor error flags (RPDO1 Word3).

5.1.5 Objects 2004h01-12

These objects (maximum number 12) are the CANopen sensors (16 bits unsigned data) that can be sent to the Digital Governor via RPDO2-4.

5.1.6 Object 2005h00

This object contains the current CANopen error state within the Gateway. It exists always in the Object Dictionary of the CANopen Gateway an can be read in via SDO communication. Its content consists of all causes that may result in a CANopen error indication by 3074 *ErrCanOpen*. The meaning of each individual bit is listed in the below table, see also ↑ *9.5 CANopen Error.*

Bit 15	Free
Bit 14	Free
Bit 13	12424 *CanOpGWErrCanComm2*
Bit 12	12423 *CanOpGWErrCanBus2*
Bit 11	12422 *CanOpGWErrCanComm1*
Bit 10	12421 *CanOpGWErrCanBus1*
Bit 9	12420 *CanOpHeartbeatErr*
Bit 8	12419 *CanOpGWResetErr*
Bit 7	Free
Bit 6	Free
Bit 5	Free
Bit 4	12415 *CanOpDCErrCanComm*
Bit 3	12414 *CanOpDCErrCanBus*
Bit 2	12413 *CanOpTimeCtrlErr*
Bit 1	12412 *CanOpInitErr*
Bit 0	12411 *CanOpDCResetErr*

Table 2: Content of the CANopen Error Object 2005h00

5.2 Service Data Objects (SDO Communication)

Access to the Object Dictionary of the CANopen Gateway is provided by SDO communication. It has been decided that no write access to the Object Dictionary will be allowed within the Gateway since the entire configuration (definition of the CanOpen sensors, switch functions, mappings, cycle times) must be conducted by means of DcDesk 2000. The Gateway represents the SDO server and reacts to client requests (SDO Rx telegrammes). The client requests have been subdivided into different services that can be identified via the Client Command Specifier (CCS, bits 5–7 of byte 1 of the SDO Rx telegramme). Since none of the objects in the Object Dictionary of the Gateway is larger than 4 bytes, segmented and block transfer will not be supported. It is only Expedited

Transfer that will be possible. Table 2 and Figures 6–10 give an overview of the SDO services and of the responses of the Gateway to SDO requests:

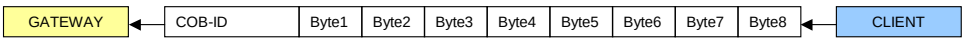
CCS Nr.	SDO Request	Answer of the CANopen Gateway
1	Initiate SDO Download	Abort telegramme, error: 06020000h (Object does not exist) / 06090011h (SubIndex does not exist) / 06070012h/13h (Length too high/low) / 06010002h (Attempt to write a read only obj.)
0	Download SDO Segment	Abort telegramme, error: 05040001h (CSS not valid or unknown)
2	Initiate SDO Upload	Initiate SDO Upload response or Abort telegramme, error: 06020000h (Object does not exist) / 06090011h (SubIndex does not exist)
3	Upload SDO Segment	Abort telegramme, error: 05040001h (css not valid or unknown)
4	Abort SDO Transfer	Any respective request that has not been responded to will be cancelled, no answer
6	Init SDO Block Download End SDO Block Download	Abort telegramme, error: 05040001h (CSS not valid or unknown)
5	Initiate SDO Block Upload Upload SDO Block Segment End SDO Block Upload	Abort telegramme, error: 05040001h (CSS not valid or unknown)

Table 3: SDO Client Command Specifiers and CANopen Gateway Answers

HEINZMANN

CANopen Gateway

SDO Rx: CANopen SDO Protocol – Initiate SDO Download



COB-ID = 0x600 + Gateway Node-ID, 8 Bytes telegramme

Byte1 = 0x2? ➜ Initiate SDO Download protocol

	Bits7-5	Bit4	Bits3-2	Bit1	Bit0
Byte 1 =	css=1	0	n	e	s

Bytes 2-3 = Index
Byte 4 = SubIndex
Bytes 5-8 = Data

s = 0 ➜ Object length not defined
s <> 0 ➜ e = 0 ➜ Object length = Bytes5-8
e <> 0 ➜ Object length = 4 – n

Answer of the CANopen Gateway:

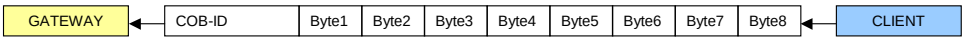
Index does not exist	➜ Abort telegramme, Error 06020000h (Object does not exist)
SubIndex does not exist	➜ Abort telegramme, Error 06090011h (SubIndex does not exist)
Wrong object length	➜ Abort telegramme, Error 06070012h/13h (Length too high/low)
Otherwise	➜ Abort telegramme, Error 06010002h (Attempt to write a read only object)

Figure 6: Initiate SDO Download Telegramme

HEINZMANN

CANopen Gateway

SDO Rx: CANopen SDO Protocol – Initiate SDO Upload



COB-ID = 0x600 + Gateway Node-ID, 8 Bytes telegramme

Byte1 = 0x40 ➜ Initiate SDO Upload protocol

	Bits7-5	Bits4-0
Byte 1 =	css=2	0

Bytes 2-3 = Index
Byte 4 = SubIndex
Bytes 5-8 = Reserved (0)

Answer of the CANopen Gateway:

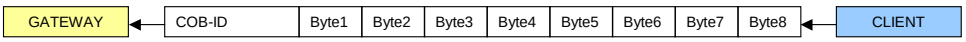
Index does not exist	➜ Abort telegramme, Error 06020000h (Object does not exist)
SubIndex does not exist	➜ Abort telegramme, Error 06090011h (SubIndex does not exist)
Otherwise	➜ Initiate SDO Upload Response telegramme (Send object)

Figure 7: Initiate SDO Upload Telegramme

HEINZMANN

CANopen Gateway

SDO Rx: CANopen SDO Protocol – Abort SDO Transfer



COB-ID = 0x600 + Gateway Node-ID, 8 Bytes telegramme

Byte1 = 0x80 ➜ Abort SDO Transfer Protocol

	Bits7-5	Bits4-0
Byte 1 =	css=4	0

Bytes 2-3 = Index
Byte 4 = SubIndex
Bytes 5-8 = SDO Abort Code

Answer of the CANopen Gateway:

An abort-telegramme does not lead to an answer-telegramme, any respective SDO request not yet responded to will be cancelled.

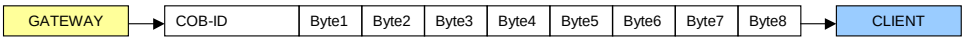
Byte1 <> 0x2?, 0x40, 0x80 ➜ Abort-telegramme, error 05040001h (css not valid or unknown)

Figure 8: Abort SDO Transfer Rx Telegramme

HEINZMANN

CANopen Gateway

SDO Tx: CANopen SDO Protocol – Initiate SDO Upload Response



COB-ID = 0x580 + Gateway Node-ID, 8 Bytes telegramme

Byte1 = 0x4? ➜ Initiate SDO Upload Response

	Bits7-5	Bit4	Bits3-2	Bit1	Bit0
Byte 1 =	scs=2	0	n	e	s

n=4 – Length of object by bytes
e=1 -> Expedited Transfer
s=1 -> Size indicator, n is valid

Bytes 2-3 = Index
Byte 4 = SubIndex
Bytes 5-8 = Data (Object)

Figure 9: Initiate SDO Upload Response Telegramme

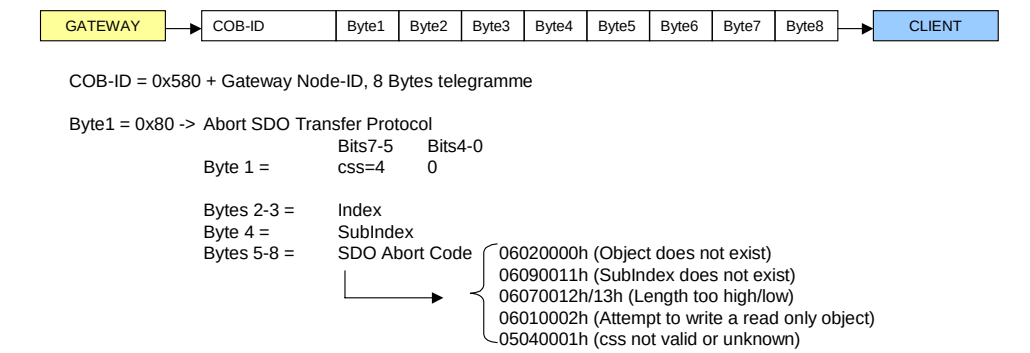


Figure 10: Abort SDO Transfer Tx Telegramme

6 Device Monitoring

For device monitoring, the heartbeat method has been implemented in the Gateway in accordance with the recommendation of the CANopen specification. The Gateway will regularly send its heartbeat telegramme. The cycle time may be parameterized by means of DcDesk 2000. The Gateway will also monitor whether the heartbeat telegramme of some CANopen partner is being regularly received. Configuration of monitoring times and of the CANopen partner node ID is to be made via DcDesk 2000. The figures 11 and 12 offer some more details about the content of the heartbeat telegrammes.

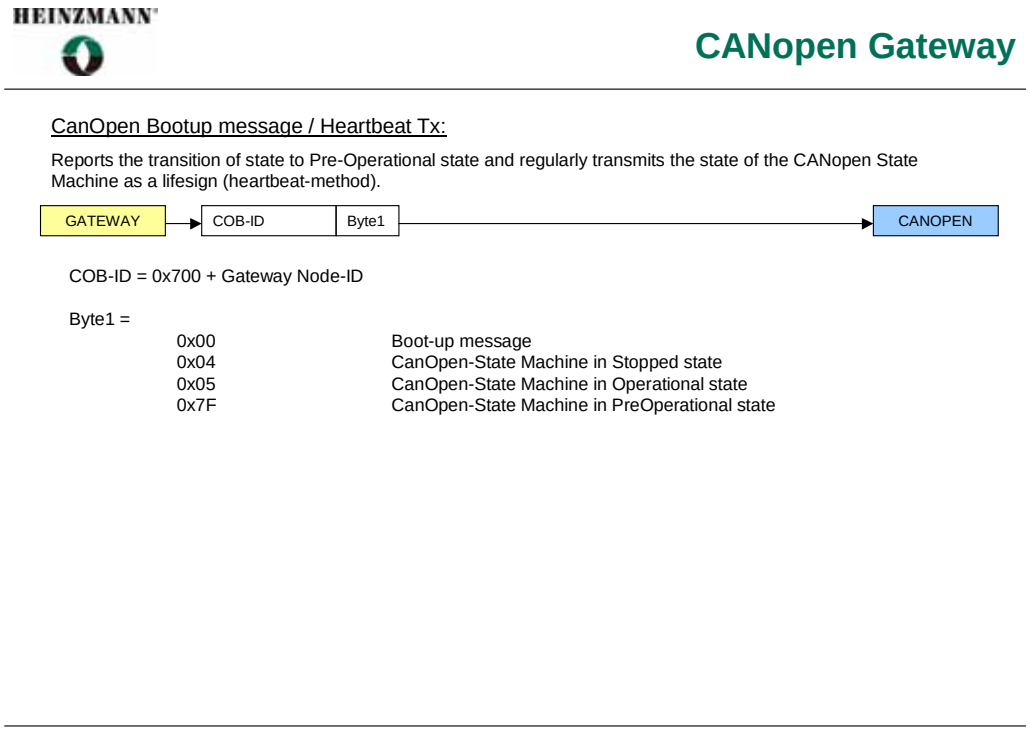


Figure 11: Heartbeat Telegramme from CANopen Gateway

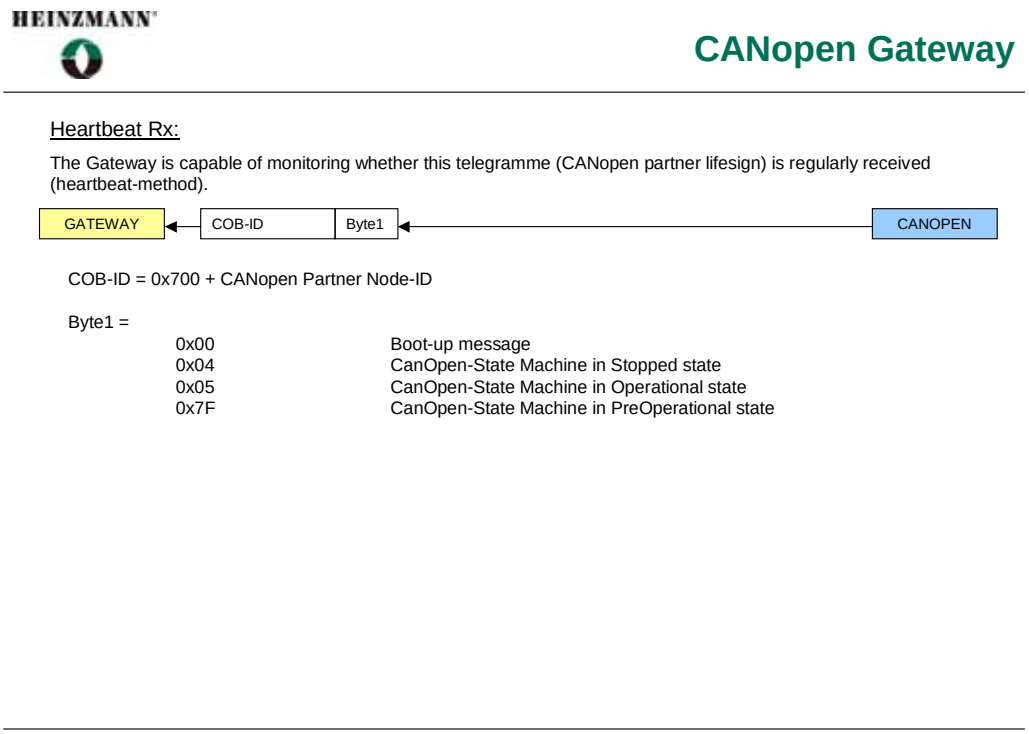


Figure 12: Heartbeat Telegramme from CANopen Partner

7 Configuration

Configuration is conducted both within the Digital Governor and the CANopen Gateway by means of DcDesk 2000. This chapter will describe a complete configuration procedure step by step.

Any new configuration will be taken over into the Digital Governor and into the CANopen Gateway only after saving the data and resetting the control unit. This will apply to all of the parameters described below.

7.1 Configuring the Digital Governor

7.1.1 Configuring the HEINZMANN CAN Bus

The Digital Governor and the CANopen Gateway are interconnected via CAN. For this communication the HEINZMANN CAN protocol is being used.

Any participant in the HEINZMANN CAN bus is specified by its device type and its node number. The device type is fixedly pre-defined, thus, e.g., the Digital Governor's type is DC (Digital Controller) and the CANopen Gateway's type is CM (Customer Module). The node number is freely selectable, but may figure on the HEINZMANN CAN bus only once.

The node number of the Digital Governor is assigned via the parameter 401 *CanMyNodeNumber*. This number must be identical with the parameter 402 *CanDCNodeNumber* in the Gateway.

The node number of the CANopen Gateway is assigned via the parameter 403 *CanCMNodeNumber*. This number must be identical with the parameter 401 *CanMyNodeNumber* in the Gateway.

The CAN Baud rate is automatically adjusted via the parameter 416 *CanBaudrate* whenever the parameter 4416 *CanSegmentOrBaudrate* has not been set.

For manually setting the Baud rate the following parameters are to be used:

410 *CanPrescaler*	Prescaler
411 *CanSyncJumpWidth*	Synchronizing jump width
412 *CanSamplingMode*	Sampling mode
413 *CanPhaseSegment1*	Phase segment 1
414 *CanPhaseSegment2*	Phase segment 2

In this case, the parameter 4416 *CanSegmentOrBaudrate* must have been set, too.

The CAN Baud rate of the HEINZMANN CAN bus must be identical both within the Digital Governor and the CANopen Gateway.

Communication is continuously monitored. However, on powering up the control unit a certain time interval may elapse before any error message is issued. This time delay is to be entered in the parameter 400 *CanStartTimeOutDelay*. All participants in the CAN network should be parameterized to the same time delay. Within this time interval, the entire network must be supplied voltage in order to avoid an error message to be output when turning the system on.

Communication of the Digital Governor with the CANopen Gateway is activated by means of the parameter 4406 *CanCommCMOn*.

Example of Parameterization:

The node number of the Digital Governor is to be 1, the node number of the CANopen Gateway is 1, the Baud rate of the HEINZMANN CAN bus is 250 kBits/s, time delay is to be 1.0 s.

Number	Parameter	Value	Unit
400	*CanStartTimeOutDelay*	*1,0*	*s*
401	*CanMyNodeNumber*	*1*	
403	*CanCMNodeNumber*	*1*	
409	*CanBaudrate*	*250*	
Activation:			
4406	*CanCommCMOn*	*1*	
4409	*CanSegmentOrBaudrate*	*0*	

7.1.2 Content of the TPDO's

7.1.2.1 TPDO 1-3

The TPDO's 1 to 3 are reserved for the communication of binary data (TPDO1-2 contain the current errors within the Digital Governor and the Gateway, TPDO3 the engine state). The content of these telegrammes has been presented in the chapter ↑ *4.1.1 TPDO's for Binary Data* and in Figure 4. For the details about the diverse error bits or engine state bits refer to the description of the respective Digital Governor. Activation of this TPDO is achieved by means of the parameters 14401 *CanOpenSendErrors* and 14402 *CanOpenSendStatus*.

Example of Parameterization:

The current errors within the Digital Governor and the Gateway are to be sent via TPDO1 and 2. TPDO3 is not to be activated (the engine state is not transmitted via TPDO3).

Number	Parameter	Value	Unit
14401	*CanOpenSendErrors*	*1*	
14402	*CanOpenSendStatus*	*0*	

7.1.2.2 TPDO 4-16

The TPDO's 4 to 16 are reserved for the communication of analogue data. The content of these telegrammes has been presented in chapter ↑ *4.1.2 TPDO's for Analogue Data* and in Figure 4. The TPDO 4 to 16 may contain up to 4 16 bits data. To each TPDO data, a parameter can be assigned. This assignment is to be made via the parameters 16400 *CanOpTPDO4Assign(0)* to 16451 *CanOpTPDO16Assign(3)*.


Note

Analogue data are transmitted by their internal value ranges. As to the conversion between internal and external value ranges refer to the description of the respective Digital Governor.


Note

4 parameters can be assigned to the 4 TPDO data. TPDO's do not contain any dummy data. If some assigned parameter is not valid (in case the parameter number is 0 or the parameter does not exist or does not contain analogue data), the corresponding data will not be sent. Length of the TDPO telegramme will be reduced by 2 bytes. For example, if parameters 1-4 are assigned to TPDO-data 1-4 but parameter 2 is not valid, TPDO length will be 6 bytes, word 1 will contain parameter 1, word 2 parameter 3 and word 3 parameter 4.

Example of Parameterization:

The parameters 2000 *Speed*, 2031 *SpeedSetp*, 2300 *ActPos*, 2330 *ActPosSetpoint* are to be transmitted via TPDO4 and the parameters 2905 *OilPressure*, 2908 *ChargeAirTemp*, 2917 *CurrentLoad* via TPDO16.

Number	Parameter	Value	Unit
16400	*CanOpTPDO4Assign(0)*	*2000*	
16401	*CanOpTPDO4Assign(1)*	*2031*	
16402	*CanOpTPDO4Assign(2)*	*2300*	
16403	*CanOpTPDO4Assign(3)*	*2330*	
16404	*CanOpTPDO5Assign(0)*	*0*	

16405	*CanOpTPDO5Assign(1)*	*0*
:	*:*	*:*
16446	*CanOpTPDO15Assign(2)*	*0*
16447	*CanOpTPDO15Assign(3)*	*0*
16448	*CanOpTPDO16Assign(0)*	*2905*
16449	*CanOpTPDO16Assign(1)*	*2908*
16450	*CanOpTPDO16Assign(2)*	*2917*
16451	*CanOpTPDO16Assign(3)*	*0*

7.1.3 TPDO Cycle Times

It is only the analogue TPDO's (4–16) that are regularly trasmitted. Any binary TPDO (1–3) will be sent only if the value of one of its data has changed since the last transmission, at most, however, once per main cycle time of the Digital Governor. Cycle times can be defined only for the TPDO's 4–16. This is done by means of the parameters 16460 *CanOpTPDO4-16Cyc(0)* through 16472 *CanOpTPDO4-16Cyc(12)*.

The accuracy of sending TPDO's will not be by 1 ms but by the Digital Governor's main clock cycle. For this reason, the TPDO cycle times are converted into numbers of main cycles of the Digital Governor. The below formulas present the correspondences between the cycle times of a RPDO as defined via DcDesk 2000 and its effective cycle time, provided the Digital Governor's main cycle time is 15.625 ms:

$$\text{Cycle time}_{DcDesk}[ms] = 0 \Rightarrow \text{TPDO is not active nor is it transmitted.}$$

$$1 \le \text{Cycle time}_{DcDesk}[ms] < 15 \Rightarrow \text{Cycle time}_{TPDO}[ms] = 15{,}625ms$$

$$n \times 15{,}625 \le \text{Cycle time}_{DcDesk}[ms] < (n+1) \times 15{,}625 \Rightarrow \text{Cycle time}_{TPDO}[ms] = n \times 15{,}625$$
$$(n \in \mathrm{N}, 1 \le n \le 255)$$

Example of Parameterization:

TPDO4 is to be sent every 16 ms, TPDO6 every 200 ms and TPDO16 every 100 ms (in reality TPDO4 will be sent every 15.625 ms, TPDO6 every 187.5 ms and TPDO16 every 93.75 ms). All of the other TPDO's are inactive.

Number	Parameter	Value	Unit
16460	*CanOpTPDO4-16Cyc(0)*	*16*	*ms*
16461	*CanOpTPDO4-16Cyc(1)*	*0*	*ms*
16462	*CanOpTPDO4-16Cyc(2)*	*200*	*ms*
16463	*CanOpTPDO4-16Cyc(3)*	*0*	*ms*
16464	*CanOpTPDO4-16Cyc(4)*	*0*	*ms*
16465	*CanOpTPDO4-16Cyc(5)*	*0*	*ms*
16466	*CanOpTPDO4-16Cyc(6)*	*0*	*ms*
16467	*CanOpTPDO4-16Cyc(7)*	*0*	*ms*
16468	*CanOpTPDO4-16Cyc(8)*	*0*	*ms*
16469	*CanOpTPDO4-16Cyc(9)*	*0*	*ms*

16470	*CanOpTPDO4-16Cyc(10)*	*0*	*ms*
16471	*CanOpTPDO4-16Cyc (11)*	*0*	*ms*
16472	*CanOpTPDO4-16Cyc(12)*	*100*	*ms*

7.1.4 Content of the RPDO's

7.1.4.1 Switch Functions

Each individual bit of Word 1 of RPDO1 can be assigned a switch function (logical function). An overview of all switch functions is to be found in the description of the respective Digital Governor.

The selection of a switch function as a CANopen input is to be made via the parameters from 4811 *Type_IdleSpeed* upward. To do so, the value “3” (CANopen) must be entered.

The switch function ‘Engine stop’ (parameters 810 and 4810) should be fixedly wired and must not be transmitted via CANopen.

The assignment of a bit of Word 1 of RPDO1 to some switch function can be made by entering the number of this bit (1–16) in the respective assignment parameter of the function (parameters from 811 *FunctIdleSpeed* onward). Entering 0 signifies that the switch function is not assigned. Any such switch function will always have the value 0.

CANopen switch functions can be high-active, i.e. active if the respective bit in Word 1 of RPDO 1 has been set, or low-active, i.e. active if the respective bit of Word 1 of RPDO1 has not been set. High-active switch functions are to be assigned the bit number positive in sign, and low-active ones the bit number negative in sign.

Example of Parameterization:

Bit number 1 of Word 1 of RPDO1 is to be used to switch between the setpoints 1 and 2. If bit number 5 has not been set load control is to be enabled.

Number	Parameter	Value	Unit
827	*FunctSetpoint2Or1*	*1*	
835	*FunctLoadEnable*	*-5*	
4827	*Type_Setpoint2Or1*	*3*	
4835	*Type_LoadEnable*	*3*	

7.1.4.2 CANopen Sensors

Each word of RPDO 2 to 4 may be assigned to a sensor. Transmission is by the internal value range. An overview of all sensors and details about the conversion between internal and external value ranges are to be found in the description of the respective Digital Governor.

Selection of a sensor as a CANopen sensor is to be made via the parameters from 4900 *ChanType_Setp1ext* onward. To do so, the value “3” (CANopen) must be entered.

49xx *ChanType_yy* = 3 Sensor is transmitted via CANopen.

The assignment of CANopen inputs to sensors is achieved by entering the desired CANopen channel number in the assignment parameters from 900 *AssignIn_Setp1Ext* upward. In this case, the channel numbers are ranging from 1 (RPDO2 Word 1) to 12 (RPDO4 Word 4).

For each CANopen sensor, there is an error flag defined in RPDO1 Word 3. When this bit is set the respective sensor is regarded as faulty.

Examples of Parameterization:

The external setpoint 1 is to be transmitted via RPDO2 Word 1, the external setpoint 2 via RPDO2 Word 2, atmospheric pressure via RPDO3 Word 3 and current load via RPDO4 Word 4.

Number	Parameter	Value	Unit
900	*AssignIn_Setp1Ext*	*1*	
901	*AssignIn_Setp2Ext*	*2*	
906	*AssignIn_AmbPress*	*7*	
917	*AssignIn_CurrentLoad*	*12*	
4900	*ChanType_Setp1Ext*	*3*	
4901	*ChanType_Setp1Ext*	*3*	
4906	*ChanType_AmbPress*	*3*	
4917	*ChanType_CurrentLoad*	*3*	

7.1.5 Monitoring for RPDO's

Every RPDO can be time-monitored: In Operational state the time difference between 2 telegrammes should not exceed the monitoring time as defined by DcDesk 2000. Definition of the monitoring times is to be made via the parameters 16480 *CanOpRPDO1-4Cyc(0)* to 16483 *CanOpRPDO1-4Cyc(3)*. Input is by [ms]. Like with the TPDO cycle times, accuracy is limited to the main cycle time of the Digital Governor. To calculate the effective monitoring times the same formulas can be used as for calculating the TPDO cycle times, see ↑ *7.1.3 TPDO Cycle Times.* Setting the monitoring time parameter to 0 will de-activates the monitoring function. It is

recommended to avoid choosing the values of the monitoring times too small because CAN communication is not in real time and might be delayed due to being processed by the Gateway or due to bus load and possibly necessary arbitration.

Example of Parameterization:

RPDO1 is not to be monitored. RPDO2 is to be received every 100 ms, RPDO3 every 200 ms and RPDO4 every 1000 ms. In reality, however, there will be a check on being received at least once per 6 main cycles (93.75 ms) for RPDO2, at least once per 12 main cycles (187.5 ms) for RPDO3 and at least once per 64 main cycles (1000 ms) for RPDO4.

Number	Parameter	Value	Unit
16480	*CanOpRPDO1-4Cyc(0)*	*0*	*ms*
16481	*CanOpRPDO1-4Cyc (1)*	*100*	*ms*
16482	*CanOpRPDO1-4Cyc (2)*	*200*	*ms*
16483	*CanOpRPDO1-4Cyc (3)*	*1000*	*ms*

7.2 Configuring the CANopen Gateway

7.2.1 Configuring the HEINZMANN CAN Bus

Any participant in the HEINZMANN CAN bus is precisely specified by its device type and its node number. The device type is pre-determined by the type of control unit and cannot be modified whereas the node number is freely selectable but must not figure more than once for any device type.

The node number of the CANopen Gateway in the HEINZMANN CAN network will have to be entered in the parameter 401 *CanMyNodeNumber*. This number must be identical with the parameter 403 *CanCMNodeNumber* in the Digital Governor.

The node number of the Digital Governor within the HEINZMANN CAN network is assigned by means of the parameter 402 *CanDCNodeNumber*. This number must be identical with the parameter 401 *CanMyNodeNumber* in the Digital Governor.

Communication of the CANopen Gateway with the Digital Governor is to be activated by the parameter 4400 *HzmCanCommDCOn*.

The node number of the Digital Governor participating in the HEINZMANN CAN bus and communicating actively with the CANopen Gateway will be indicated by the parameters 2410 *CanDCNodeState31to16* and 2411 *CanDCNodeState15to01*. The values of the parameters 2410 and 2411 are encoded binarily with the bit number corresponding to the node number. Similarly, if a PC is communicating with the CANopen Gateway via DcDesk-2000-CAN its node number will be indicated by the parameters 2424 *CanPCNodeState31to16* and 2425 *CanPCNodeState15to01*.

Example of Parameterization:

The Digital Governor with node number 1 is communicating with the CANopen Gateway, DcDesk 2000 CAN is not being used.

Number	Parameter	Value	Unit
2410	*CanDCNodeState31to16*	*0*	*Hex*
2411	*CanDCNodeState15to01*	*2* *(bit 1 set)*	*Hex*
2424	*CanPCNodeState31to16*	*0*	*Hex*
2425	*CanPCNodeState15to01*	*0*	*Hex*

The Baud rate of the HEINZMANN CAN is adjusted automatically via the parameter 416 *HzmCanBaudrate* if the parameter 4416 *HzmCanSegmOrBaudrate* has not been set. The supported Baud rates are 125, 250, 500 and 1000 kBits/s.

For manually setting the Baud rate the following parameters are to be used:

410 *HzmCanPrescaler*	Prescaler
411 *HzmCanSyncJumpWidth*	Synchronizing jump width
412 *HzmCanSamplingMode*	Sampling mode
413 *HzmCanPhaseSegment1*	Phase segment 1
414 *HzmCanPhaseSegment2*	Phase segment 2
415 *HzmCanPropSegment*	Propagation segment

In this case, the parameter 4416 *HzmCanSegmentOrBaudrate* must have been set, too.

The CAN Baud rate of the HEINZMANN CAN bus must be identical both within the Digital Governor and the CANopen Gateway.

Communication will be continuously monitored. However, on powering up the control unit a certain time interval may elapse before any error message is issued. This time delay is to be entered in the parameter 400 *CanStartTimeOutDelay*. All participants in the CAN network should be parameterized to the same time delay. Within this time interval, the entire network must be supplied voltage in order to avoid an error message to be output on switching the system on.

The following common error messages may be generated:

3070 *ErrCanBus1*	Error of the HEINZMANN CAN bus
3071 *ErrCanComm1*	Error of HEINZMANN CAN communication

On the occurrence of a HEINZMANN CAN bus error, the CAN controller is signalling errors such as BusStatus, ErrorStatus or DataOverrun, or that a transmission could not be fed into the CAN bus. In spite of re-initializing the controller, it is not possible to definitely eliminate the error. In most cases, the causes will be wrong cabling, missing

termination or different Baud rates of the individual participants in the network. The CANopen Gateway will try to obtain an error free condition of the connection by continuously initializing the CAN controller.

In contrast, the HEINZMANN CAN communication error 3071 *ErrCanComm1* is a network error of content, i.e., it does not constitute a physical error, so basically communication will be possible. Information about the communication errors concerning the HEINZMANN CAN bus can be obtained from these parameters:

2401 *CanTxBufferState*	Status of transmission buffer
2402 *CanRxBufferState*	Status of reception buffer
2403 *CanRxTimeout*	Status of reception timeout monitoring
2404 *CanTypeMismatch*	Status of device numbers

The values of the parameters 2401 through 2404 are encoded binarily with the bit numbers corresponding to the device numbers. Any indication by these parameters will also activate the error condition 3071 *ErrCanComm1*.

The transmission and reception buffers are being monitored for overflow for each device type, and are indicated by the parameters 2401 *CanTxBufferState* and 2402 *CanRxBufferState*. The messages must have been completed received within a certain time frame, otherwise the error condition 2403 *CanRxTimeout* will be set. The error 2404 *CanTypeMismatch* will indicate an error of configuration as in this case a second participant with the same device number and the same device type has been connected to the network. In case there is overflow of the transmission or the reception buffer, only this overflow will be signalled, and communication will continue even though one or more messages could not be received and/or transmitted. If too many messages could not be received the error 2403 *CanRxTimeout* will be set. If the messages cannot be transmitted due to overflow of the transmission buffer the opposite station will signal the timeout error.

The error 2403 *CanRxTimeout* will generally be set whenever there is no response from the opposite station. Though in this case the transmission of messages to the opposite station still continues there will be a change-over of content to certain emergency procedures.

Any overflow of the specific CANopen reception buffer will also entail the error 3071 *ErrCanComm1*. In this case, the reception buffer is cleared and CANopen communication re-initiated.

The parameter

2405 *CanOnline*	General condition

serves to indicate whether the control unit is generally ready to communicate via CAN.

Example of Parameterization:

The node number of the Digital Governor is to be 1, the node number of the CANopen Gateway is also 1, the Baud rate of the HEINZMANN CAN bus is to be set to 250 kbits/s, and the time delay is supposed to be 1.0 s.

Number	Parameter	Value	Unit
400	*CanStartTimeOutDelay*	*1,0*	*s*
401	*CanMyNodeNumber*	*1*	
402	*CanDCNodeNumber*	*1*	
416	*HzmCanBaudrate*	*250*	

Activation:

4400	*HzmCanCommDCOn*	*1*
4416	*HzmCanSegmentOrBaudrate*	*0*

7.2.2 Configuring the CANopen Bus

Any CANopen participant is specified by its node number (Node ID). This node number is freely selectable between 1 and 127 but may figure only once.

Assignment of the node number of the CANopen Gateway is to be made by means of the parameter 427 *CanOpenMyNodeID*.

The CANopen Baud rate is automatically adjusted via the parameter 426 *CanOpenBaudrate* if the parameter 4426 *CanOpSegmOrBaudrate* has been set to 0. The supported Baud rates are 125, 250, 500 and 1000 kBits/s.

For manually setting the Baud rate the following parameters are to be used:

420 *CanOpenPrescaler*	Prescaler
421 *CanOpenSyncJumpWidth*	Synchronizing jump width
422 *CanOpenSamplingMode*	Sampling mode
423 *CanOpenPhaseSegment1*	Phase segment 1
424 *CanOpenPhaseSegment2*	Phase segment 2
425 *CanOpenPropSegment*	Propagation segment

In this case, the parameter 4426 *CanOpSegmOrBaudrate* must have been set, too.

The Baud rates of the HEINZMANN CAN bus and of the CANopen network are entirely independent of each other and must not be identical

Communication is continuously monitored. The following common error messages will possibly be generated:

3072 *ErrCanBus2*	Error of the CANopen bus
3073 *ErrCanComm2*	Error of CANopen communication

On the occurrence of a CANopen bus error, the CAN controller will signal errors such as BusStatus, ErrorStatus or DataOverrun, or that a transmission could not be fed into the CANopen bus. In spite of re-initializing the controller, it is not possible to definitely eliminate the error. In most cases, the causes will be wrong cabling, missing termination or different Baud rates of the individual participants in the network. The CANopen Gateway will try to obtain an error free condition of the connection by continuously initializing the CAN controller and by regularly transmitting the boot-up message or the heartbeat telegramme to the CANopen Master.

In contrast, the CANopen communication error 3073 *ErrCanComm2* is a network error of content, i.e., it does not constitute a physical error, so basically communication will be possible but there will occur an overflow in the CANopen reception buffer or in the CANopen SDO transmission buffer ↑ *9.5 CANopen Error* (description of the error 12424 *CanOpGWErrCanComm2*).

The parameter

12405 *CanOpOnline*	General condition

serves to indicate whether the control unit is generally ready to communicate via the CANopen interface.

Example of Parameterization:

The node number (Node ID) of the CANopen Gateway is 1, the Baud rate of the CANopen network is to be set to 125 kBits/s.

Number	Parameter	Value	Unit
427	*CanOpenMyNodeID*	*1*	
426	*CanOpenBaudrate*	*125*	

Activation:

4426	*CanOpSegmentOrBaurdate*	*0*

7.2.3 Configuring Device Monitoring

The Gateway is capable of regularly sending its heartbeat telegramme. The cycle time is adjusted by means of the parameter 429 *CanOpHeartBeatTxTime*. If set to 0 this is to signify that the heartbeat telegramme is not to be transmitted.

Just as with the TPDO cycle times, accuracy of transmission will not be by 1 ms but by the main cycle time of the CANopen Gateway (16 ms). Input via DcDesk 2000 must,

therefore, be regarded as approximate values only. To calculate the cycle times more precisely the following formulas may be used

$$\text{Cycle time}_{DcDesk}[ms] = 0 \Rightarrow \text{Heartbeat is not active nor is it transmitted.}$$

$$1 \le \text{Cycle time}_{DcDesk}[ms] < 15 \Rightarrow \text{Cycle time}_{Heartbeat}[ms] = 16ms$$

$$n \times 16 \le \text{Cycle time}_{DcDesk}[ms] < (n+1) \times 16 \Rightarrow \text{Cycle time}_{Heartbeat}[ms] = n \times 16$$
$$(n \in \mathrm{N}, 1 \le n \le 4095)$$

The Gateway is also capable of monitoring whether the heartbeat telegramme of some CANopen partner is being regularly received. This is achieved by entering the node number of the CANopen partner to be monitored in the parameter 428 *CanOpenPartnerNodeID*. Configuration of the monitoring time is made via the parameter 430 *CanOpHeartBeatRxTime*. Setting it to 0 will de-activate the monitoring function. The formulas given above apply also to calculating the accurate monitoring time.

Example of Parameterization:

The node number (Node ID) of the partner to be monitored is 127, and its heartbeat is to be received every 100 ms. The Gateway will transmit its own heartbeat every 1000 s.

Number	Parameter	Value	Unit
428	*CanOpenPartnerNodeID*	*127*	
429	*CanOpHeartBeatTxTime*	*1000*	*ms*
430	*CanOpHeartBeatRxTime*	*100*	*ms*

7.2.4 Configuring the TPDO Identifiers

The identifiers of 4 RPDO's and 4 TPDO's will already have been determined via the node number of the CANopen Gateway 427 *CanOpenMyNodeID*. Table 3 offers an overview of the identifiers of RPDO1-4 and TPDO1-4.

CANopen Telegramme	Identifier
RPDO1	200h + node number
RPDO2	300h + node number
RPDO3	400h + node number
RPDO4	500h + node number
TPDO1	180h + node number
TPDO2	280h + node number
TPDO3	380h + node number
TPDO4	480h + node number

Table 4: Pre-defined PDO Identifiers

As the CANopen Gateway supports 16 TPDO's the identifiers of TPDO 5 to 16 will have to be defined by DcDesk 2000. This is done by means of the parameters 16404 *CanOpTPDO5-16ID(0)* through 16415 *CanOpTPDO5-16ID(11)*.

Example of Parameterization:

Identifier 282h is to be assigned to TPDO5, identifier 382h to TPDO8, identifier 3h to TDPO12 and identifier 57Fh to TPDO16.

Number	Parameter	Value	Unit
16404	*CanOpTPDO5-16ID(0)*	*282*	*Hex*
16407	*CanOpTPDO5-16ID(3)*	*382*	*Hex*
16411	*CanOpTPDO5-16ID(7)*	*3*	*Hex*
16415	*CanOpTPDO5-16ID(11)*	*57F*	*Hex*

7.3 Errors of Configuration

Besides the structure of the Init telegrammes that are exchanged between the Gateway and the Digital Governor during the initialization phase, the Gateway will also check the consistency of the configuration of CANopen communication. In case of any error, initialization is going to be aborted and the error messages 3074 *ErrCanOpen* and 12412 *CanOpInitErr* will be generated, see ↑ *9.5 CANopen Error*. The present section offers explainations of some frequently occurring configuration errors.

Which TPDO's and RPDO's are active is defined in the Digital Governor via DcDesk 2000:

- TPDO 1 and 2 are active if 14401 *CanOpenSendErrors* is set.
- TPDO 3 is active if 14402 *CanOpenSendStatus* is set.

- TPDO 4-16 are active if at least one of the parameters assigned to the TPDO data is valid (16400 *CanOpTPDO4Assign(0)* through 16451 *CanOpTPDO16Assign(3))* and if the set cycle time is not zero (16460 *CanOpTPDO4-16Cyc(0)* through 16472 *CanOpTPDO4-16Cyc(12)*)
- RPDO 1 is active if any CANopen switch functions have been defined (parameters from 811 *FunctIdleSpeed* upward and from 4811 *Type_IdleSpeed* upward) or if at least one of the RPDO's 2-4 is active (sensor error flags).
- RPDO 2-4 are active if they contain at least one CANopen sensor (parameters from 900 *AssignIn_Setp1Ext* upward and from 4900 *ChanType_Setp1ext* upward).

The CANopen identifiers of the active PDO's are determined within the Gateway either via the Gateway node numbers for the TPDO's 1–4 and RPDO's 1–4 or via the parameters 16404 *CanOpTPDO5-16ID(0)* to 16415 *CanOpTPDO5-16ID(11)* for the TDPO's 5-6.

The Gateway will check whether any two PDO's have identical identifiers. Furthermore, PDO identifiers should not belong to the pre-defined and reserved CANopen identifiers. These reserved identifiers are the following:

- 000h (Identifier of the NMT telegramme)
- 080h (Synchronization telegramme)
- 100h (Time Stamp telegramme)
- 580h + Gateway node number (SDO Tx telegramme)
- 600h + Gateway node number (SDO Rx telegramme)
- 700h + Gateway node number (Heartbeat Tx telegramme)
- 700h + CANopen partner node number (Heartbeat Rx telegramme)

Finally the Gateway will check whether the CANopen node numbers of the Gateway and of the CANopen partner (427 *CanOpenMyNodeID* and 428 *CanOpenPartnerNodeID)* are different.

8 Data Management

The control unit contains several parameters informing about the type of control unit and the software version.

8.1 Serial Number of the Control Unit

Every individual control unit is unambiguously identified by a serial number. Its first four digits will indicate the production year and the month of delivery. The other digits represent the consecutive production number. The serial number can be viewed by the following parameters:

3844 *SerialDate*	Production year and month
3845 *SerialNumber*	Consecutive production number

8.2 Identification of the Control Unit

The application dependent functionality of any control unit is unambiguously defined by the software running one and only one certain hardware type.

3840 *HardwareVersion*	Version number of the device hardware
3841 *AddHardwareVersion*	Version number for hardware modifications
3842 *SoftwareVersion*	Version number of the device software
3843 *BootSoftwareversion*	Version number of the bootloader software

The software version number is made up of an unambiguous two- to four-digit customer number defined by HEINZMANN, a one- to two-digit variant number and a two-digit modification index. DcDesk 2000 and the Hand Programmer will permit the customer to access only control units containing software with the particular customer number. Variants define different applications such as implementation of new CANopen functionalities. Due to software extensions every variant may exist by different stages of modifications where the next higher index will always comprise the next lower one and replace it entirely.

8.3 Identification of the PC Programme and the Hand Programmer

Every HEINZMANN PC Programme and every HEINZMANN Hand Programmer that are required for parameter changes has its own identification number which is handed over to the control unit. The current identification of the PC Programme or the Hand Programmer can be viewed by the parameter 3850 *Identifier*. The identification number of the PC Programme or Hand Programmer that was used for saving the last parameter change is indicated by the parameter 3851 *LastIdentifier*.

9 Error Handling

9.1 General

The CANopen Gateway includes an integrated error monitoring system by which errors may be detected and reported.

The different errors can be viewed by the parameters 3000..3099. A currently set error parameter will read the value “1”, otherwise the value “0”.

In general, the following errors types can be distinguished:

- Errors in configuring the device and adjusting the parameters

  These errors are caused by erroneous input on the part of the user and cannot be intercepted by either the PC or the Hand Held Programmer. They do not occur with controls from series production.

- Errors occuring during operation

  These errors are the most significant ones when using control units produced in series. CANopen communication errors are typical of this category.

- Internal computational errors of the device

  These errors may be due to defective components or other inadmissible operating conditions. Under normal circumstances, they are not likely to occur.

When cancelling an error one should first establish and eliminate its cause before clearing any of the current errors. Some errors are cleared automatically as soon as the failure cause has been eliminated.. Errors can be cleared by means of the PC or by a Hand Programmer. Since the CANopen Gateway and the Digital Governor are connected via CAN clearing an error in one device will cause this error to be also cleared in the other device. Should the system not stop reporting an error, the search for its cause must go on.

Principally, the control starts operating on the assumption that there is no error and will only then begin to check for possible occurrences of errors. This implies that the control can be put into an error free state by a reset of the control unit but will immediately begin to report any errors that are currently active.

9.2 Error Memories

When the CANopen Gateway is powered down it will lose any existing information on actual errors. In order to be able to check on which errors have occurred, a permanent error memory has been incorporated in the control. Any errors that have occurred at least once will be stored there.

For every occurring error an error counter is included in the error memory and displayed by the error memory image of DcDesk 2000.

The values stored in the error memory are treated by the Gateway merely as monitor values and are not any further taken account of. In other words, it is only the errors occurring during operation that the Gateway will respond to.

The error counters within the permanent error memory can be inspected by means of the parameters that have been assigned numbers from 3100 upward so that the numbers of permanently stored errors will differ by 100 from those of the respective actual errors.

The permanent error memory can be cleared by means of the PC or the Hand Programmer only. After clearance, the Gateway will revert to accumulating any occurring errors in the empty error memory.

9.3 Bootloader

The HEINZMANN Digital Governors include what is called a bootloader. This programme section is stored at a specific location of the read-only memory and is programmed once for all at the factory. The bootloader cannot be cleared.

On starting the control programme by powering it up or by a reset, the bootloader programme is always executed first. This programme performs various relevant tests telling whether the actual control programme is or is not operable. Based on these tests the bootloader decides whether further programme execution can be handed on to the control programme or whether execution must remain confined to the bootloader to preclude any risk of personal injury or damage to the engine.


Note

The entire bootloader tests and the subsequent initialization of the main programme will take about 200 ms.

9.3.1 Bootloader Start Tests

The following section describes which tests are performed by the bootloader and which measures may have to be taken. As long as these tests are being conducted, there will be no communication with the device, especially when due to some fatal error the programme is caught in an infinite loop.

- Watchdog Test

  This test is to verify whether the watchdog integrated into the processor is operable. This is to ensure that in case of an undefined programme execution the control will go into a safe state after a pre-defined time. If the outcome of the watchdog test is negative, the bootloader programme will remain in an endless loop.

- Bootloader Programme Test

  By this test, a check-sum is calculated over the memory area containing the bootloader programme and compared with the check-sum pre-programmed at the factory. If the sums do not match, the bootloader programme will remain in an endless loop.

- Bootloader RAM Test

  During this test, various binary patterns are written into the internal processor RAM memory and read out again. If at least one storage location does not contain the expected code, the bootloader programme will enter into an endless loop.

- Control RAM Test

  During this test, various binary patterns are written into the internal processor RAM memory used by the Gateway programme and read out again. If at least one storage location does not contain the expected code, the bootloader programme will got into a state in which the errors can be read out via serial communication. DcDesk 2000 will indicate by means of 3078 *ErrRamTest* that a RAM memory error has occurred. The parameter 3800 *TestStatus* will be set to 1, and the parameter 3801 *TestValue1* will display the faulty address, 3802 *TestValue2* the test value and 3803 *TestValue3* the content of the address as played back.

- Control Programme Test

  By this test, a check-sum is calculated over the memory area containing the Gateway programme and compared with the check-sum pre-programmed at the factory. If the sums do not match, the bootloader will go into a state which is indicated by the error 3087 *ErrMainCheckSum* via serial communication. The parameter 3800 will be set to 0, the parameter 3801 *TestValue1* will contain the expected check sum and 3802 *TestValue2* the calculated check sum. At this point, it will be possible to load a new programme.

- Watchdog Triggering

  The bootloader passes into a state which is indicated as “watchdog error” 3089 *ErrWatchdog* via serial communication. The parameter 3095 *Exception-Number* will be set to 3, 3096 *ExceptionAddrLow* and 3097 *ExceptionAddrHigh* will indicate the programme address at which the watchdog was triggered, and 3098 *ExceptionFlag* the associated programme flags.

9.3.2 Bootloader-Parameterliste

Parameter	Range	Unit	Level	Meaning
3075 ErrClearFlash	0..1		1	Error on clearing programme
3076 ErrEEPROM	0..1		1	No access to EEPROM
3078 ErrRAMTest	0..1		1	Error of RAM memory
3079 ErrProgFlash	0..1		1	Error on storing programme
3087 ErrMainCheckSum	0..1		1	Error of programme check sum
3089 ErrWatchdog	0..1		1	Watchdog triggering
3094 ErrIntern	0..1		1	internal programme error
3095 ExceptionNumber	0..FFFF	Hex	1	internal programme error
3096 ExceptionAddrLow	0..FFFF	Hex	1	internal programme error
3097 ExceptionAddrHigh	0..FFFF	Hex	1	internal programme error
3098 ExceptionFlag	0..FFFF	Hex	1	internal programme error
3800 TestStatus	0..65535		1	Error status of bootloader test
3801 TestValue1	0..FFFF	Hex	1	Error indication of bootloader test
3802 TestValue2	0..FFFF	Hex	1	Error indication of bootloader test
3803 TestValue3	0..FFFF	Hex	1	Error indication of bootloader test
3840 HardwareVersion	0..9999		1	Hardware version
3841 AddHardwareVersion	0..9999		1	Additional hardware version
3842 SoftwareVersion	0..65535		1	Software version
3850 Identifier	0..65535		1	Identification number DcDesk 2000
3870 Timer	0..65535		1	Internal timer

9.3.3 Bootloader Communication

If serial communication with the bootloader can be entered into, this will on the one hand have the effect that errors are being reported; on the other hand, this state may serve as a starting point for downloading a new control programme. By principle, this procedure will always have to be carried out by the bootloader.

The variables 3095 *ExceptionNumber* and 3800 *TestStatus* will indicate the reason why the programme has branched into bootloader communication.

3095 ExceptionNumber

1 AND command (programme code FF)

2 INTO command (overflow after arithmetic operations)

3 Watchdog

with 3096 *ExceptionAddrLow* = error address lowWord

3097 *ExceptionAddrHigh* = error address highWord

3098 *ExceptionFlag* = Flags

3800 TestStatus

1 RAM error in the area outside the bootloader RAM

3801 *TestValue1* = Error address

3802 *TestValue2* = Test value

3803 *TestValue3* = Content of address

2 Main check sum from EEPROM = 0 (programme cleared)

3801 *TestValue1* = Main check sum

3802 *TestValue2* = 3803 *TestValue3* = 0

3 Main length from EEPROM = 0 (programme cleared)

3801 *TestValue1* = Main check sum

3802 *TestValue2* = Main length lowWord

3803 *TestValue3* = Main length highWord

4 Main length from EEPROM > maximum possible length

3801 *TestValue1* = Main check sum

3802 *TestValue2* = Main length lowWord

3803 *TestValue3* = Main length highWord

5 Flash is empty (position NMI address = FFFFFFFF)

3801 *TestValue1* = NMI address lowWord

3802 *TestValue2* = NMI address highWord

3803 *TestValue3* = 0

3800 TestStatus

6 Flash is empty (content of NMI address = FFFF)

3801 *TestValue1* = NMI address lowWord

3802 *TestValue2* = NMI address highWord

3803 *TestValue3* = Content of NMI address

7 Flash is empty (position programme start address = FFFFFFFF)

3801 *TestValue1* = Programme start address lowWord

3802 *TestValue2* = Programme start address highWord

3803 *TestValue3* = 0

8 Flash is empty (content programme start address = FFFF)

3801 *TestValue1* = Programme start address lowWord

3802 *TestValue2* = Programme start address highWord

3803 *TestValue3* = Programme start address content

9 Main check sum wrong

3801 *TestValue1* = Calculated check sum

3802 *TestValue2* = Stored check sum

3803 TestValue3 = 0

10 No access possible to EEPROM (reading out from Page 0)

3801 TestValue1 = 3802 TestValue2 = 3803 TestValue3 = 0

9.4 Gateway Error Parameter List

The below error parameter list contains descriptions of the causes of each single error and of the Gateway's response. It also lists the appropriate actions to be taken to remove the respective error.

The errors are stored in the volatile error memory under the parameter numbers 3000 and higher; at the same they are accumulated in the permanent error memory under the parameter numbers from 3100 onward.

The errors are sorted by ascending numbers with the parameter on the left indicating the actual error as stored in the volatile memory and with the parameter on the right indicating the one stored as a sentinel in the permanent error memory. As explained above, the Gateway will only react to actual errors whereas the permanent error memory serves no other purpose than to accumulate information on the occurrences of errors.

3070 ErrCanBus1 **3170 SErrCanBus1**

Cause: The CAN controller for the HEINZMANN CAN bus reports errors such as BusStatus, ErrorStatus or DataOverrun or that some message could not be fed into the HEINZMANN CAN bus. In spite of re-initializing the controller it is not possible to eliminate the errors permanently. In the majority of cases, this will be caused by wrong cabling, missing terminaton or different Baud rates of the individual participants in the network

Response: - CANopen communication is not possible
- State Machine in state Undefined

Action: - Check CAN connection

- Check CAN termination

- Check CAN Baud rates of the bus participants

3071 ErrCanComm1 **3171 SErrCanComm1**

Cause: There is an overrun in the reception buffer or the transmission buffer or a second participant with identical device number and identical device type is connected to the network or the Digital Governor does not respond.

Response: - Neither HEINZMANN nor CANopen communication is possible
- State Machine in state Undefined

Action: - Check CAN connection

- Check CAN configuration of Gateway and Digital Governor

3072 ErrCanBus2 **3172 SErrCanBus2**

Cause: The CAN controller for the CANopen CAN bus reports errors such as BusStatus, ErrorStatus or DataOverrun or that some message could not be fed into the CANopen bus. In spite of re-initializing the controller it is not possible to eliminate the errors permanently. In the majority of cases, this will be caused by wrong cabling, missing termination or different Baud rates of the individual participants in the network

Response:
- CANopen communication is not possible
- State Machine in state Preoperational
- Peridocal transmission of the boot-up message and/or of the heartbeat telegramme to the CANopen Master

Action:
- Check CANopen connection
- Check CANopen termination
- Check CANopen Baud rates of the bus participants

3073 ErrCanComm2 **3173 SErrCanComm2**

Cause: There is an overrun in the CANopen reception buffer or the SDO-transmission buffer

Response:
- CANopen communication is not possible
- State Machine in state Undefined (overflow reception buffer) or in state Stopped (overflow transmission buffer)

Action:
- Check CAN connection
- Check CANopen bus load and extent of CANopen communication to Gateway

3074 ErrCanOpen **3174 SErrCanOpen**

↑ *9.5 CANopen Error*

3076 ErrParamStore **3176 SErrParamStore**

Cause: Occurrence of an error on programming the Gateway's flash memory.

Response:
- CANopen communication is not possible
- State Machine in state Undefined

Action:
- Restart Gateway by resetting the control unit
- Notify HEINZMANN.

3077 ErrProgramTest **3177 SErrProgramTest**

Cause: Current monitoring of the programme memory reports an error.

Response: - CANopen communication is not possible
- State Machine in state Undefined

Action: - Restart Gateway by resetting the control unit
- Notify HEINZMANN.

3078 ErrRAMTest **3178 SErrRAMTest**

Cause: Current monitoring of the working memory reports an error.

Response: - CANopen communication is not possible
- State Machine in state Undefined

Action: - Write down the values of the parameters 3895 *RAMTestAddr* and 3896 Write down *RAMTestPattern*
- Restart Gateway by resetting the control unit
- Notify HEINZMANN.

3085 ErrVoltage **3185 SErrVoltage**

Cause: The 24V supply voltage is below 7.5 V.

Response: - Error message.
- Error is cleared automatically as soon as the voltage is back within the normal range.

Action: - Check voltage supply.

3090 ErrData **3190 SErrData**

Cause: No data found or check sum over data is wrong.

Response: - CANopen communication is not possible
- State Machine in state Undefined

Action: Check data for correct setting, save parameters in control unit and restart Gateway by a reset of the control unit.

Note: The error will occur only when adjusting and saving parameters.

3093 ErrStack **3193 SErrStack**

Cause: Internal programming or computing error, “stack-overflow”.

Response: - CANopen communication is not possible
- State Machine in state Undefined

Action: - Write down the value of the parameters 3897 *CStackTestFreeBytes* and 3898 *IStackTestFreeBytes* and notify HEINZMANN.
- Restart Gateway by resetting the control unit.

3094 ErrIntern **3194 SErrIntern**
3195 SExceptionNumber
3196 SExceptionAddrLow
3197 SExceptionAddrHigh
3198 SExceptionFlag

Cause: Internal programming or computing error, so-called "EXCEPTION" error.

Response: - CANopen communication is not possible
- State Machine in state Undefined

Action: - Write down the values of the parameters 3195 *ExceptionNumber*, 3196 *ExceptionAddrLow*, 3197 *ExceptionAddrHigh* and 3198 *ExceptionFlag* and notify HEINZMANN.
- Restart Gateway by resetting the control unit.

3099 EEPROMErrorCode

Cause: Internal error of structure of the EEPROM pages

Response: - Error message.

Action: - Write down the value of the parameter and notify HEINZMANN.
- Restart Gateway by resetting the control unit.

Bit	Description of Error	Effect
0	Bootloader page faulty	Programme remains in bootloader
3	Serial number page faulty	The serial number of the control unit cannot be displayed.
4	Error memory pages faulty	The error memory is cleared.
5	Parameter memory pages faulty	Default data are being used
6	Exception memory page faulty	Exception cannot be reported.

9.5 CANopen Error

CANopen communication is monitored by the Digital Governor and the CANopen Gateway. Any error condition is signalled both by the Digital Governor and by the CANopen Gateway through the error 3074 *ErrCanOpen*. The possible causes of this error are indicated by the parameters 12411 *CanOpDCResetErr* through 12422 *CanOpGWErrCanComm2* which will be dealt with in this chapter in more detail. Any CANopen errors resulting in re-initializing CANopen communication between the Digital Governor and the Gateway remain set in the parameters from 12411 *CanOpDCResetErr* on also in the state Preoperational. If the error has been eliminated the parameter can be cleared by clearing all errors or by changing over into Operational state.

12411 CanOpDCResetErr

Cause: The Digital Governor has been reset, but the CANopen Gateway has not.

Response: When in Operational state, PDO communication is aborted. The Digital Governor will re-initialize the Gateway, and the state will change over to Preoperational. The boot-up message is then sent to the CANopen Master.

Action: - Check current supply to Digital Governor
- To restart CANopen communication the Master is to send the NMT command “Operational” to the Gateway.
- The error is cleared by clearing all errors or when in Operational state.

12412 CanOpInitErr

Cause: Configuration of CANopen communication cannot be performed.

Response: After 3 unsuccessful attempts of initialization between the Digital Governor and the Gateway the initialization procedure is aborted. The CANopen State Machine remains in the state Undefined. The Gateway cannot create its Object Dictionary and does not attain the state Preoperational. CANopen communication is not active. Clearing all errors will lead to another 3 attempts at initialization.

Action: - Check and correct the configuration of the Digital Governor and the Gateway, see ↑ *7.3 Errors of Configuration*
- Reset Digital Governor and CANopen Gateway.

12413 CanOpTimeCtrlErr

Cause: The time monitoring of some RPDO has tripped in Operational state, see ↑ *7.1.5 Monitoring for RPDO's*

Response: The respective sensors or switch functions are regarded as faulty. The error will automatically be cleared as soon as RPDO communication is working properly again (according to the monitoring times as defined).

Action: Check RPDO communication in the CANopen partner (cycle time ...).

12414 CanOpDCErrCanBus

Cause: This bit will be set in case the error 3070 *ErrCanBus* has been set in the Digital Governor. The CAN controller of the Digital Governor will report errors such as BusOff, ErrorStatus or DataOverrun, or some message cannot be fed into the CAN bus. In spite of re-initialization, the error cannot be eliminated definitely.

Response: CANopen communication is aborted, and the CANopen State Machine switches into the “Undefined” state. This bit will be reset by clearing all errors in case the CAN bus is healthy again, or in Operational state.

Action: - Check CAN module of Digital Governor
- Check HEINZMANN CAN connection.

12415 CanOpDCErrCanComm

Cause: This bit is set if the error 3071 *ErrCanComm* has been set in the Digital Governor. There has occurred overflow of the reception or transmission buffer, or an opposite station does not respond, or there have been connected 2 devices with identical device type and identical node number to the HEINZMANN CAN bus.

Response: CANopen communication is aborted, and the CANopen State Machine switches into the “Undefined” state. This bit will be reset by clearing all errors in case the CAN bus is healthy again, or when in Operational state.

Action: - Check CAN module of Digital Governor
- Check HEINZMANN CAN connection.

12419 CanOpGWResetErr

Cause: The CANopen Gateway has been reset, but not the Digital Governor.

Response: If in Operational state, PDO communication is aborted. The Digital Governor will re-initialize the Gateway, and the state will change over to Preoperational. The boot-up message is then sent to the CANopen Master.

Action: - Check current supply to CANopen Gateway
- To restart CANopen communication the CANopen Master is to send the NMT command “Operational” to the Gateway.
- The error is cleared by clearing all errors or when in Operational state.

12420 CanOpHeartbeatErr

Cause: The heartbeat monitoring function has tripped in Operational state, see. ↑ *7.2.3 Configuring Device Monitoring*

Response: All CANopen sensors and switch functions in the Digital Governor are regarded as faulty. The error will be automatically cleared as soon as the heartbeat telegrammes are received again in accordance with the monitoring times defined.

Action: Check CANopen partner (condition, cycle time of its heartbeat telegrammes ...).

12421 CanOpGWErrCanBus1

Cause: This bit will be set in case the error 3070 *ErrCanBus1* has been set in the Gateway. The CAN1 controller of the Gateway connected with the Digital Governor will report errors such as BusOff, ErrorStatus or DataOverrun, or that some message cannot be fed into the CAN bus. In spite of re-initialization, the error cannot be eliminated definitely.

Response: CANopen communication is aborted, and the CANopen State Machine switches into the “Undefined” state. This bit will be reset by clearing all errors in case the CAN bus is healthy again, or when in Operational state.

Action: - Check CAN module of Digital Governor
- Check HEINZMANN CAN connection.

12422 CanOpGWErrCanComm1

Cause: This bit is set if the error 3071 *ErrCanComm1* has been set in the Gateway. There has occurred overflow of the HEINZMANN reception or transmission buffer or in the CANopen reception buffer, or the Digital Governor does not respond, or there have been connected 2 devices with identical device type and identical node number to the HEINZMANN CAN bus.

Response: CANopen communication is aborted, and the CANopen State Machine switches into the “Undefined” state. This bit will be reset by clearing all errors in case the CAN bus is healthy again, or when in Operational state.

Action:
- Check CAN module of Digital Governor
- Check HEINZMANN CAN connection
- Reset control units
- Check CANopen bus load and the extent of CANopen communication to the Gateway.

12423 CanOpGWErrCanBus2

Cause: This bit will be set in case the error 3070 *ErrCanBus2* has been set in the Gateway. The CAN2 controller of the Gateway connected with the CANopen network will report errors such as BusOff, ErrorStatus or DataOverrun, or that some message cannot be fed into the CAN bus. In spite of re-initialization, the error cannot be eliminated definitely.

Response: Since communication between Digital Governor and Gateway remains maintained, CANopen communication will be aborted but not re-initialized. The State Machine will switch over into “Preoperational” state. Every half second the CAN2 controller is re-initialized and after that the boot-up message or the heartbeat telegramme respectively will be sent to the CANopen Master. This bit will be reset by clearing all errors in case the CAN bus is healthy again, or when in Operational state.

Action:
- Check CANopen connection.

12424 CanOpGWErrCanComm2

Cause: This bit is set if the error 3073 *ErrCanComm2* has been set in the Gateway. There has occurred overflow in the CANopen reception buffer or in the SDO transmission buffer.

Response: In the event of overflow of the CANopen reception buffer CANopen communication is aborted. The CANopen State Machine will switch into "Undefined" state. The CANopen reception buffer is cleared. The Digital Control will re-initialize the Gateway, and the state will change into "Preoperational". Then the boot-up message is sent to the CANopen Master. In the event of overflow of the CANopen SDO transmission buffer PDO communication is aborted. The State Machine will switch into the state "Stopped" which does not permit of any SDO communication. The CANopen SDO transmission buffer is cleared, and the Gateway will be waiting for an NMT command from the Master.
This bit is reset by clearing all errors in case the CAN bus is healthy again, or when in Operational state.

Action: - Check CANopen bus load and the extent of CANopen communication to the Gateway.

9.6 CANopen Sensor Errors in the Digital Governor

It is by RPDO2–4 that the CANopen sensors are transmitted to the Digital Governor. they are being monitored like fixedly wired sensors. A sensor error will be reported if

- the state of the CANopen State Machine is not Operational (the RPDO's are not active);
- the state of the CANopen State Machine is Operational, the time monitoring of RPDO corresponding to the respective the sensor is active and has tripped (the cycle time as defined for this RPDO in the Digital Governor (parameters 16481 *CanOpRPDO1-4Cyc(1)* through 16483 *CanOpRPDO1-4Cyc(3))* has been exceeded);
- the state of the CANopen State Machine is Operational, the time monitoring of the RPDO corresponding to the respective sensor is not active (the parameters 16481 *CanOpRPDO1-4Cyc(1)* through 16483 *CanOpRPDO1-4Cyc(3)* are 0), and no single RPDO corresponding to the respective sensor has been received in Operational state (this implies that in spite of time monitoring being switched off, at least one telegramme must be received in Operational state);
- the state of the CANopen State Machine is Operational and the sensor error flag in RPDO1 Word 3 is set;

- the state of the CANopen State Machine is Operational, time monitoring of RPDO1 containing the sensor flag is active and has tripped (the monitoring time as defined for RPDO1 (parameters 16480 *CanOpRPDO1-4Cyc(0*)) has been exceeded);
- the state of the CANopen State Machine is Operational and the heartbeat monitoring function is active and has tripped.

As with the fixedly wired sensors, any CANopen sensor error will be reported via the parameters from 3005 *ErrSetpointExtern* upward in the Digital Governor. Further reactions on the occurrence of sensor errors (self-erasing errors, default values) and the required parameterization are described in the manuals of the respective Digital Governor.

9.7 Errors of CANopen Switch Functions in the Digital Governor

The CANopen switch functions sent to the Digital Governor via RPDO's are being monitored. An error will be detected if

- the state of the CANopen State Machine is not Operational (the RPDO's are not active);
- the state of the CANopen State Machine is Operational, time monitoring of RPDO1 is active and has tripped (the monitoring time as defined for RPDO1 (parameters 16480 *CanOpRPDO1-4Cyc(0*)) has been exceeded);
- the state of the CANopen State Machine is Operational, the time monitoring of the RPDO is not active (the parameters 16481 *CanOpRPDO1-4Cyc(1)* through 16483 *CanOpRPDO1-4Cyc(3)* are 0), and no single RPDO1 has been received in Operational state (this implies that in spite of time monitoring being switched off, at least one telegramme must be received in Operational state);
- the state of the CANopen State Machine is Operational and the heartbeat monitoring function is active and has tripped..

As a consequence of any error of switch functions the switch function will be set to 0.

10 EDS File

```
[Comments]
Lines=0

[FileInfo]
FileName=CANopenGateway.eds
FileVersion=1
FileRevision=0
EDSVersion=4
Description=EDS for Heinzmann CANopen Gateway
CreationDate=05-19-2004
CreationTime=00:00AM
CreatedBy=Heinzmann GmbH & Co. KG
ModificationDate=05-19-2004
ModificationTime=00:00AM
ModifiedBy=Heinzmann GmbH & Co. KG

[DeviceInfo]
VendorName=Heinzmann GmbH & Co. KG
VendorNumber=0x00000161

ProductName=CANopen Gateway
ProductNumber=0x00001B59
RevisionNumber=0
OrderCode=CANopen Gateway

BaudRate_10=0
BaudRate_20=0
BaudRate_50=0
BaudRate_125=1
BaudRate_250=1
BaudRate_500=1
BaudRate_800=0
BaudRate_1000=1

SimpleBoot-upMaster=0
SimpleBoot-upSlave=1

Granularity=0

DynamicChannelsSupported=0
GroupMessaging=0
NrOfRXPDO=4
NrOfTXPDO=16
LSS_Supported=0

[DummyUsage]
Dummy0001=0
Dummy0002=0
Dummy0003=0
Dummy0004=0
Dummy0005=0
Dummy0006=1
Dummy0007=0

[MandatoryObjects]
SupportedObjects=3
1=0x1000
2=0x1001
3=0x1018

[1000]
ParameterName=Device Type
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=0x00000000
PDOMapping=0

[1001]
ParameterName=Error Register
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0x00
PDOMapping=0

[1018]
ParameterName=Identity Object
ObjectType=0x8
SubNumber=5

[1018sub0]
ParameterName=Number of entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0x04
PDOMapping=0
```

[1018sub1]
ParameterName=Vendor ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=0x00000161
PDOMapping=0

[1018sub2]
ParameterName=Product Code
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=0x0x00001B59
PDOMapping=0

[1018sub3]
ParameterName=Revision Number
ObjectType=0x7
DataType=0x0007
AccessType=ro
PDOMapping=0

[1018sub4]
ParameterName=Serial Number
ObjectType=0x7
DataType=0x0007
AccessType=ro
PDOMapping=0

[OptionalObjects]

SupportedObjects=43

;Consumer Heartbeat Time
1=0x1016
;Producer Heartbeat Time
2=0x1017

;SDO Parameters
3=0x1200

;RPDO1 Com
4=0x1400
;RPDO2 Com
5=0x1401
;RPDO3 Com
6=0x1402
;RPDO4 Com
7=0x1403

;RPDO1 Map
8=0x1600
;RPDO2 Map
9=0x1601
;RPDO3 Map
10=0x1602
;RPDO4 Map
11=0x1603

;TPDO1 Com
12=0x1800
;TPDO2 Com
13=0x1801
;TPDO3 Com
14=0x1802
;TPDO4 Com
15=0x1803
;TPDO5 Com
16=0x1804
;TPDO6 Com
17=0x1805
;TPDO7 Com
18=0x1806
;TPDO8 Com
19=0x1807
;TPDO9 Com
20=0x1808
;TPDO10 Com
21=0x1809
;TPDO11 Com
22=0x180A
;TPDO12 Com
23=0x180B
;TPDO13 Com
24=0x180C
;TPDO14 Com
25=0x180D
;TPDO15 Com
26=0x180E
;TPDO16 Com

27=0x180F

;TPDO1 Map
28=0x1A00
;TPDO2 Map
29=0x1A01
;TPDO3 Map
30=0x1A02
;TPDO4 Map
31=0x1A03
;TPDO5 Map
32=0x1A04
;TPDO6 Map
33=0x1A05
;TPDO7 Map
34=0x1A06
;TPDO8 Map
35=0x1A07
;TPDO9 Map
36=0x1A08
;TPDO10 Map
37=0x1A09
;TPDO11 Map
38=0x1A0A
;TPDO12 Map
39=0x1A0B
;TPDO13 Map
40=0x1A0C
;TPDO14 Map
41=0x1A0D
;TPDO15 Map
42=0x1A0E
;TPDO16 Map
43=0x1A0F

[1016]

ParameterName=Consumer Heartbeat Time
ObjectType=0x8
SubNumber=2

[1016sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0x01
PDOMapping=0

[1016sub1]
ParameterName=Consumer Heartbeat Time
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=0x00000000
PDOMapping=0

[1017]
ParameterName=Producer Heartbeat Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

;********
;SDO PARAMETERS
;********
[1200]

ParameterName=Server SDO Parameter
ObjectType=0x8
SubNumber=3

[1200sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0x2
PDOMapping=0

[1200sub1]
ParameterName=COB-ID Client -> Server
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x600
PDOMapping=0

[1200sub2]

ParameterName=COB-ID Server -> Client
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x580
PDOMapping=0

;*****************

;RPDO1 COMMUNICATION PARAMETERS

;*****************

[1400]

ParameterName=RPDO1 Communication Parameter
ObjectType=0x9
SubNumber=6

[1400sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[1400sub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000200
PDOMapping=0

[1400sub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[1400sub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[1400sub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[1400sub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************

;RPDO2 COMMUNICATION PARAMETERS

;*****************

[1401]

ParameterName=RPDO2 Communication Parameter
ObjectType=0x9
SubNumber=6

[1401sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[1401sub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000300
PDOMapping=0

[1401sub2]
ParameterName=Transmission Type
ObjectType=0x7

DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[1401sub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[1401sub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[1401sub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************

;RPDO3 COMMUNICATION PARAMETERS

;*****************

[1402]

ParameterName=RPDO3 Communication Parameter
ObjectType=0x9
SubNumber=6

[1402sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[1402sub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000400
PDOMapping=0

[1402sub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[1402sub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[1402sub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[1402sub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************

;RPDO4 COMMUNICATION PARAMETERS

;*****************

[1403]

ParameterName=RPDO4 Communication Parameter
ObjectType=0x9
SubNumber=6

[1403sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[1403sub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000500
PDOMapping=0

[1403sub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[1403sub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[1403sub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[1403sub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************
;RPDO1 MAPPING PARAMETERS
;*****************
[1600]
ParameterName=RPDO1 Mapping Parameter
ObjectType=0x8
SubNumber=1

[1600sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0
PDOMapping=0
LowLimit=0
HighLimit=3

;*****************
;RPDO2 MAPPING PARAMETERS
;*****************
[1601]
ParameterName=RPDO2 Mapping Parameter
ObjectType=0x8
SubNumber=1

[1601sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0
PDOMapping=0
LowLimit=0
HighLimit=4

;*****************
;RPDO3 MAPPING PARAMETERS
;*****************
[1602]

ParameterName=RPDO3 Mapping Parameter
ObjectType=0x8
SubNumber=1

[1602sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0
PDOMapping=0
LowLimit=0
HighLimit=4

;*****************
;RPDO4 MAPPING PARAMETERS
;*****************
[1603]
ParameterName=RPDO4 Mapping Parameter
ObjectType=0x8
SubNumber=1

[1603sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0
PDOMapping=0
LowLimit=0
HighLimit=4

;*****************
;TPDO1 PARAMETERS
;*****************
[1800]
ParameterName=TPDO1 Communication Parameter
ObjectType=0x9
SubNumber=6

[1800sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[1800sub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000180
PDOMapping=0

[1800sub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[1800sub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[1800sub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[1800sub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************
;TPDO2 PARAMETERS
;*****************

[1801]

ParameterName=TPDO2 Communication Parameter

ObjectType=0x9

SubNumber=6

[1801sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=5

PDOMapping=0

[1801sub1]

ParameterName=COB-ID

ObjectType=0x7

DataType=0x0007

AccessType=ro

DefaultValue=$NODEID+0x80000280

PDOMapping=0

[1801sub2]

ParameterName=Transmission Type

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0xFE

PDOMapping=0

[1801sub3]

ParameterName=Inhibit Time

ObjectType=0x7

DataType=0x0006

AccessType=ro

DefaultValue=0x0000

PDOMapping=0

[1801sub4]

ParameterName=Compatibility Entry

ObjectType=0x7

DataType=0x0005

AccessType=ro

PDOMapping=0

[1801sub5]

ParameterName=Event Timer

ObjectType=0x7

DataType=0x0006

AccessType=ro

DefaultValue=0

PDOMapping=0

;*****************

;TPDO3 PARAMETERS

;*****************

[1802]

ParameterName=TPDO3 Communication Parameter

ObjectType=0x9

SubNumber=6

[1802sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=5

PDOMapping=0

[1802sub1]

ParameterName=COB-ID

ObjectType=0x7

DataType=0x0007

AccessType=ro

DefaultValue=$NODEID+0x80000380

PDOMapping=0

[1802sub2]

ParameterName=Transmission Type

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0xFE

PDOMapping=0

[1802sub3]

ParameterName=Inhibit Time

ObjectType=0x7

DataType=0x0006

AccessType=ro

DefaultValue=0x0000

PDOMapping=0

[1802sub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[1802sub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************
;TPDO4 PARAMETERS
;*****************
[1803]
ParameterName=TPDO4 Communication Parameter
ObjectType=0x9
SubNumber=6

[1803sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[1803sub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000480
PDOMapping=0

[1803sub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[1803sub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[1803sub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[1803sub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************
;TPDO5 PARAMETERS
;*****************
[1804]
ParameterName=TPDO5 Communication Parameter
ObjectType=0x9
SubNumber=6

[1804sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[1804sub1]
ParameterName=COB-ID
ObjectType=0x7

DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000000
PDOMapping=0

[1804sub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[1804sub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[1804sub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[1804sub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************

;TPDO6 PARAMETERS

;*****************

[1805]

ParameterName=TPDO6 Communication Parameter
ObjectType=0x9
SubNumber=6

[1805sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[1805sub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000000
PDOMapping=0

[1805sub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[1805sub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[1805sub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[1805sub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************

;TPDO7 PARAMETERS

;*****************

[1806]

ParameterName=TPDO7 Communication Parameter

ObjectType=0x9

SubNumber=6

[1806sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=5

PDOMapping=0

[1806sub1]

ParameterName=COB-ID

ObjectType=0x7

DataType=0x0007

AccessType=ro

DefaultValue=$NODEID+0x80000000

PDOMapping=0

[1806sub2]

ParameterName=Transmission Type

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0xFE

PDOMapping=0

[1806sub3]

ParameterName=Inhibit Time

ObjectType=0x7

DataType=0x0006

AccessType=ro

DefaultValue=0x0000

PDOMapping=0

[1806sub4]

ParameterName=Compatibility Entry

ObjectType=0x7

DataType=0x0005

AccessType=ro

PDOMapping=0

[1806sub5]

ParameterName=Event Timer

ObjectType=0x7

DataType=0x0006

AccessType=ro

DefaultValue=0

PDOMapping=0

;*****************

;TPDO8 PARAMETERS

;*****************

[1807]

ParameterName=TPDO8 Communication Parameter

ObjectType=0x9

SubNumber=6

[1807sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=5

PDOMapping=0

[1807sub1]

ParameterName=COB-ID

ObjectType=0x7

DataType=0x0007

AccessType=ro

DefaultValue=$NODEID+0x80000000

PDOMapping=0

[1807sub2]

ParameterName=Transmission Type

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0xFE

PDOMapping=0

[1807sub3]

ParameterName=Inhibit Time

ObjectType=0x7

DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[1807sub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[1807sub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************
;TPDO9 PARAMETERS
;*****************
[1808]
ParameterName=TPDO9 Communication Parameter
ObjectType=0x9
SubNumber=6

[1808sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[1808sub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000000
PDOMapping=0

[1808sub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[1808sub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[1808sub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[1808sub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************
;TPDO10 PARAMETERS
;*****************
[1809]
ParameterName=TPDO10 Communication Parameter
ObjectType=0x9
SubNumber=6

[1809sub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[1809sub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000000
PDOMapping=0

[1809sub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[1809sub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[1809sub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[1809sub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************
;TPDO11 PARAMETERS
;*****************
[180A]
ParameterName=TPDO11 Communication Parameter
ObjectType=0x9
SubNumber=6

[180Asub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[180Asub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000000
PDOMapping=0

[180Asub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[180Asub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[180Asub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[180Asub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006

AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************
;TPDO12 PARAMETERS
;*****************
[180B]
ParameterName=TPDO12 Communication Parameter
ObjectType=0x9
SubNumber=6

[180Bsub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[180Bsub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000000
PDOMapping=0

[180Bsub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[180Bsub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[180Bsub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[180Bsub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************
;TPDO13 PARAMETERS
;*****************
[180C]
ParameterName=TPDO13 Communication Parameter
ObjectType=0x9
SubNumber=6

[180Csub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[180Csub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000000
PDOMapping=0

[180Csub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[180Csub3]

ParameterName=Inhibit Time

ObjectType=0x7

DataType=0x0006

AccessType=ro

DefaultValue=0x0000

PDOMapping=0

[180Csub4]

ParameterName=Compatibility Entry

ObjectType=0x7

DataType=0x0005

AccessType=ro

PDOMapping=0

[180Csub5]

ParameterName=Event Timer

ObjectType=0x7

DataType=0x0006

AccessType=ro

DefaultValue=0

PDOMapping=0

;*****************

;TPDO14 PARAMETERS

;*****************

[180D]

ParameterName=TPDO14 Communication Parameter

ObjectType=0x9

SubNumber=6

[180Dsub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=5

PDOMapping=0

[180Dsub1]

ParameterName=COB-ID

ObjectType=0x7

DataType=0x0007

AccessType=ro

DefaultValue=$NODEID+0x80000000

PDOMapping=0

[180Dsub2]

ParameterName=Transmission Type

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0xFE

PDOMapping=0

[180Dsub3]

ParameterName=Inhibit Time

ObjectType=0x7

DataType=0x0006

AccessType=ro

DefaultValue=0x0000

PDOMapping=0

[180Dsub4]

ParameterName=Compatibility Entry

ObjectType=0x7

DataType=0x0005

AccessType=ro

PDOMapping=0

[180Dsub5]

ParameterName=Event Timer

ObjectType=0x7

DataType=0x0006

AccessType=ro

DefaultValue=0

PDOMapping=0

;*****************

;TPDO15 PARAMETERS

;*****************

[180E]

ParameterName=TPDO15 Communication Parameter

ObjectType=0x9

SubNumber=6

[180Esub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[180Esub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000000
PDOMapping=0

[180Esub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[180Esub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[180Esub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[180Esub5]
ParameterName=Event Timer
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0
PDOMapping=0

;*****************

;TPDO16 PARAMETERS

;*****************

[180F]
ParameterName=TPDO16 Communication Parameter
ObjectType=0x9
SubNumber=6

[180Fsub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=5
PDOMapping=0

[180Fsub1]
ParameterName=COB-ID
ObjectType=0x7
DataType=0x0007
AccessType=ro
DefaultValue=$NODEID+0x80000000
PDOMapping=0

[180Fsub2]
ParameterName=Transmission Type
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0xFE
PDOMapping=0

[180Fsub3]
ParameterName=Inhibit Time
ObjectType=0x7
DataType=0x0006
AccessType=ro
DefaultValue=0x0000
PDOMapping=0

[180Fsub4]
ParameterName=Compatibility Entry
ObjectType=0x7
DataType=0x0005
AccessType=ro
PDOMapping=0

[180Fsub5]

ParameterName=Event Timer

ObjectType=0x7

DataType=0x0006

AccessType=ro

DefaultValue=0

PDOMapping=0

;*****************

;TPDO1 MAPPING PARAMETERS

;*****************

[1A00]

ParameterName=TPDO1 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A00sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=4

;*****************

;TPDO2 MAPPING PARAMETERS

;*****************

[1A01]

ParameterName=TPDO2 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A01sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=2

;*****************

;TPDO3 MAPPING PARAMETERS

;*****************

[1A02]

ParameterName=TPDO3 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A02sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=1

;*****************

;TPDO4 MAPPING PARAMETERS

;*****************

[1A03]

ParameterName=TPDO4 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A03sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=4

;*****************

;TPDO5 MAPPING PARAMETERS

;*****************

[1A04]

ParameterName=TPDO5 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A04sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=4

;*****************

;TPDO6 MAPPING PARAMETERS

;*****************

[1A05]

ParameterName=TPDO6 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A05sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=4

;*****************

;TPDO7 MAPPING PARAMETERS

;*****************

[1A06]

ParameterName=TPDO7 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A06sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=4

;*****************

;TPDO8 MAPPING PARAMETERS

;*****************

[1A07]

ParameterName=TPDO8 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A07sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=4

;*****************

;TPDO9 MAPPING PARAMETERS

;*****************

[1A08]

ParameterName=TPDO9 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A08sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=4

;*****************

;TPDO10 MAPPING PARAMETERS

;*****************

[1A09]

ParameterName=TPDO10 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A09sub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=4

;*****************

;TPDO11 MAPPING PARAMETERS

;*****************

[1A0A]

ParameterName=TPDO11 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A0Asub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=4

;*****************

;TPDO12 MAPPING PARAMETERS

;*****************

[1A0B]

ParameterName=TPDO12 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A0Bsub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=4

;*****************

;TPDO13 MAPPING PARAMETERS

;*****************

[1A0C]

ParameterName=TPDO13 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A0Csub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=4

;*****************

;TPDO14 MAPPING PARAMETERS

;*****************

[1A0D]

ParameterName=TPDO14 Mapping Parameter

ObjectType=0x8

SubNumber=1

[1A0Dsub0]

ParameterName=Number of Entries

ObjectType=0x7

DataType=0x0005

AccessType=ro

DefaultValue=0

PDOMapping=0

LowLimit=0

HighLimit=4

;*****************

;TPDO15 MAPPING PARAMETERS

;*****************

[1A0E]

ParameterName=TPDO15 Mapping Parameter

ObjectType=0x8
SubNumber=1

[1A0Esub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0
PDOMapping=0
LowLimit=0
HighLimit=4

;*****************
;TPDO16 MAPPING PARAMETERS
;*****************
[1A0F]
ParameterName=TPDO16 Mapping Parameter
ObjectType=0x8
SubNumber=1

[1A0Fsub0]
ParameterName=Number of Entries
ObjectType=0x7
DataType=0x0005
AccessType=ro
DefaultValue=0
PDOMapping=0
LowLimit=0
HighLimit=4

[ManufacturerObjects]
SupportedObjects=6
1=0x2000
2=0x2001
3=0x2002
4=0x2003
5=0x2004
6=0x2005

;*****************
;DIGITAL OUTPUTS
;*****************
[2000]
SubNumber=7
ParameterName=Digital Outputs
ObjectType=0x8

[2000sub0]
ParameterName=Number of entries
ObjectType=0x07
DataType=0x0005
AccessType=ro
PDOMapping=0
Defaultvalue=6
LowLimit=6
HighLimit=7

[2000sub1]
ParameterName=Errors 0-15
ObjectType=0x07
DataType=0x0006
AccessType=ro
PDOMapping=1

[2000sub2]
ParameterName=Errors 16-31
ObjectType=0x07
DataType=0x0006
AccessType=ro
PDOMapping=1

[2000sub3]
ParameterName=Errors 32-47
ObjectType=0x07
DataType=0x0006
AccessType=ro
PDOMapping=1

[2000sub4]
ParameterName=Errors 48-63
ObjectType=0x07
DataType=0x0006
AccessType=ro
PDOMapping=1

[2000sub5]
ParameterName=Errors 64-79
ObjectType=0x07
DataType=0x0006

AccessType=ro
PDOMapping=1

[2000sub6]
ParameterName=Errors 80-95
ObjectType=0x07
DataType=0x0006
AccessType=ro
PDOMapping=1

;*****************
;UNSIGNED ANALOG OUTPUTS
;*****************
[2001]
SubNumber=1
ParameterName=Unsigned Analog Outputs
ObjectType=0x8

[2001sub0]
ParameterName=Number of entries
ObjectType=0x07
DataType=0x0005
AccessType=ro
PDOMapping=0
Defaultvalue=0
LowLimit=0
HighLimit=52

;*****************
;SIGNED ANALOG OUTPUTS
;*****************
[2002]
SubNumber=1
ParameterName=Signed Analog Outputs
ObjectType=0x8

[2002sub0]
ParameterName=Number of entries
ObjectType=0x07
DataType=0x0005
AccessType=ro
PDOMapping=0
Defaultvalue=0
LowLimit=0
HighLimit=52

;*****************
;DIGITAL INPUTS
;*****************
[2003]
SubNumber=1
ParameterName=Digital Inputs
ObjectType=0x8

[2003sub0]
ParameterName=Number of entries
ObjectType=0x07
DataType=0x0005
AccessType=ro
PDOMapping=0
Defaultvalue=0
LowLimit=0
HighLimit=2

;*****************
;ANALOG INPUTS
;*****************
[2004]
SubNumber=1
ParameterName=Analog Inputs
ObjectType=0x8

[2004sub0]
ParameterName=Number of entries
ObjectType=0x07
DataType=0x0005
AccessType=ro
PDOMapping=0
Defaultvalue=0
LowLimit=0
HighLimit=12

;*****************

;CANOPEN ERRORS

;*****************

[2005]

ParameterName=CanOpen Errors

ObjectType=0x7

DataType=0x0006

AccessType=ro

DefaultValue=0x0000

PDOMapping=0

11 Parameter Description

11.1 Synoptical Table

In the below table, all parameters of the CANopen Gateway have been listed with their numbers and identifiers in adjacent columns such as to illustrate the functional interrelation between the individual parameters. The parameters added in the Digital Governor for CANopen communication are described in the respective manual of the Digital Governor.

Parameters		Measurements		Functions		Curves	
400	CanStartTimeOutDelay			4400	HzmCanCommDCOn		
401	CanMyNodeNumber	2401	CanTxBufferState				
402	CanDCNodeNumber	2402	CanRxBufferState				
		2403	CanRxTimeout				
		2404	CanTypeMismatch				
		2405	CanOnline				
		2406	CanState				
410	HzmCanPrescaler	2410	CanDCNodeState31to16				
411	HzmCanSyncJumpWidth	2411	CanDCNodeState15to01				
412	HzmCanSamplingMode						
413	HzmCanPhaseSegment1						
414	HzmCanPhaseSegment2						
415	HzmCanPropSegment						
416	HzmCanBaudrate			4416	HzmCanSegmOrBaudrate		
420	CanOpenPrescaler						
421	CanOpenSyncJumpWidth						
422	CanOpenSamplingMode						
423	CanOpenPhaseSegment1						
424	CanOpenPhaseSegment2	2424	CanPCNodeState31to16				
425	CanOpenPropSegment	2425	CanPCNodeState15to01				
426	CanOpenBaudrate			4426	CanOpSegmOrBaudrate		
427	CanOpenMyNodeID						
428	CanOpenPartnerNodeID						
429	CanOpHeartBeatTxTime						
430	CanOpHeartBeatRxTime						
		3070	ErrCanBus1				
		3071	ErrCanComm1				
		3072	ErrCanBus2				
		3073	ErrCanComm2				
		3074	ErrCanOpen				
		3076	ErrParamStore				
		3077	ErrProgramTest				
		3078	ErrRAMTest				
		3085	ErrVoltage				
		3090	ErrData				
		3093	ErrStack				
		3094	ErrIntern				
		3099	EEPROMErrorCode				
		3170	SErrCanBus1				
		3171	SErrCanComm1				
		3172	SErrCanBus2				
		3173	SErrCanComm2				
		3174	SErrCanOpen				
		3176	SErrParamStore				
		3177	SErrProgramTest				
		3178	SErrRAMTest				

Parameters		Measurements		Functions		Curves	
		3185	SErrVoltage				
		3190	SErrData				
		3193	SErrStack				
		3194	SErrIntern				
		3195	SExceptionNumber				
		3196	SExceptionAddrLow				
		3197	SExceptionAddrHigh				
		3198	SExceptionFlag				
		3600	PowerSupply				
1800	Level						
		3840	HardwareVersion				
		3841	AddHardwareVersion				
		3842	SoftwareVersion				
		3843	BootSoftwareVersion				
		3844	SerialDate				
		3845	SerialNumber				
		3850	Identifier				
		3851	LastIdentifier				
		3865	CalculationTime				
		3870	Timer				
		3895	RAMTestAddr				
		3896	RAMTestPattern				
		3897	CStackTestFreeBytes				
		3898	IStackTestFreeBytes				
		12400	CanOpState				
		12401	CanOpStopped				
		12402	CanOpInit				
		12403	CanOpPreOperational				
		12404	CanOpOperational			16404	CanOpTPDO5-16ID
		12405	CanOpOnline				
		12410	CanOpErr				
		12411	CanOpDCResetErr				
		12412	CanOpInitErr				
		12413	CanOpTimeCtrlErr				
		12414	CanOpDCErrCanBus				
		12415	CanOpDCErrCanComm				
		12419	CanOpGWResetErr				
		12420	CanOpHeartbeatErr				
		12421	CanOpGWErrCanBus1				
		12422	CanOpGWErrCanComm				
		12423	CanOpGWErrCanBus2				
		12424	CanOpGWErrCanComm				

11.2 List 1: Parameters

400 CanStartTimeOutDelay

Level:	6	Delay time of monitoring of CAN connection after reset
Range:	0..100 s	
Page(s):	27, 29	

401 CanMyNodeNumber

Level:	6	Own node number in HEINZMANN CAN network
Range:	1..31	
Page(s):	26, 29	

402	**CanDCNodeNumber**		
	Level:	6	Node number of Digital Governor HEINZMANN CAN network
	Range:	1..31	
	Page(s):	26, 29	
410	**HzmCanPrescaler**		
	Level:	6	Prescaler of CAN Baud rate of HEINZMANN CAN network
	Range:	0..63	
	Page(s):	27	
411	**HzmCanSyncJumpWidth**		
	Level:	6	Synchronizing Jump Width of CAN Baud rate of HEINZMANN CAN network
	Range:	0..3	
	Page(s):	27	
412	**HzmCanSamplingMode**		
	Level:	6	Sampling mode for CAN Baud rate of HEINZMANN CAN network
	Range:	0..1	
	Page(s):	27	
413	**HzmCanPhaseSegment1**		
	Level:	6	Phase segment 1 for CAN Baud rate of HEINZMANN CAN network
	Range:	0..7	
	Page(s):	27	
414	**HzmCanPhaseSegment2**		
	Level:	6	Phase segment 2 for CAN Baud rate of HEINZMANN CAN network
	Range:	0..7	
	Page(s):	27	
415	**HzmCanPropSegment**		
	Level:	6	Propagation segment for CAN Baud rate of HEINZMANN CAN network
	Range:	0..7	
	Page(s):	27	
416	**HzmCanBaudrate**		
	Level:	6	CAN Baud rate of HEINZMANN CAN network
	Range:	125..1000	
	Page(s):	27, 29	
420	**CanOpenPrescaler**		
	Level:	6	Prescaler of CAN Baud rate of CANopen network
	Range:	0..63	
	Page(s):	29	
421	**CanOpenSyncJumpWidth**		
	Level:	6	Synchronizing Jump Width for CAN-Baudrate of CANopen network
	Range:	0..3	
	Page(s):	29	
422	**CanOpenSamplingMode**		
	Level:	6	Sampling mode for CAN Baud rate of CANopen network
	Range:	0..1	
	Page(s):	29	
423	**CanOpenPhaseSegment1**		
	Level:	6	Phase segment 1 for CAN Baud rate of CANopen network
	Range:	0..7	
	Page(s):	29	
424	**CanOpenPhaseSegment2**		
	Level:	6	Phase segment 2 for CAN Baud rate of CANopen network
	Range:	0..7	
	Page(s):	29	

425	**CanOpenPropSegment**		
	Level:	6	Propagation segment for CAN-Baud rate of CANopen network
	Range:	0..7	
	Page(s):	29	
426	**CanOpenBaudrate**		
	Level:	6	CAN Baud rate of CANopen network
	Range:	125..1000	
	Page(s):	29, 30	
427	**CanOpenMyNodeID**		
	Level:	6	Own node number in CANopen network
	Range:	1..127	
	Page(s):	29, 30	
428	**CanOpenPartnerNodeID**		
	Level:	6	Node number of CANopen device to be monitored via the heartbeat procedure
	Range:	0..127	
	Page(s):	31	
429	**CanOpHeartBeatTxTime**		
	Level:	6	Cycle time of own heartbeat telegramme
	Range:	0..65535 ms	
	Page(s):	30, 31	
430	**CanOpHeartBeatRxTime**		
	Level:	6	Monitoring time of the heartbeat telegramme of the CANopen partner
	Range:	0..65535 ms	
	Page(s):	31	
1800	**Level**		
	Level:	1	User level
	Range:	1..7	
	Page(s):		

11.3 List 2: Measurements

2401	**CanTxBufferState**		
	Level:	1	State of HEINZMANN CAN transmission buffer
	Range:	0000..FFFF Hex	
	Page(s):	28	
2402	**CanRxBufferState**		
	Level:	1	State of HEINZMANN CAN reception buffer
	Range:	0000..FFFF Hex	
	Page(s):	28	
2403	**CanRxTimeout**		
	Level:	1	State of HEINZMANN CAN reception timeout monitoring
	Range:	0000..FFFF Hex	
	Page(s):	28	
2404	**CanTypeMismatch**		
	Level:	1	State of HEINZMANN CAN device number
	Range:	0..1	
	Page(s):	28	
2405	**CanOnline**		
	Level:	1	CANopen Gateway is ready to communicate via CAN over the HEINZMANN CAN bus
	Range:	0..1	
	Page(s):	28	

2410	**CanDCNodeState31to16**		
	Level:	6	Node number of Digital Governor on HEINZMANN CAN bus
	Range:	0000..FFFF Hex	
	Page(s):	26, 27	
2411	**CanDCNodeState15to01**		
	Level:	6	Node number of Digital Governor on HEINZMANN CAN bus
	Range:	0000..FFFF Hex	
	Page(s):	26, 27	
2424	**CanPCNodeState31to16**		
	Level:	6	Node number of PC on HEINZMANN CAN bus
	Range:	0000..FFFF Hex	
	Page(s):	26, 27	
2425	**CanPCNodeState15to01**		
	Level:	6	Node number of PC on HEINZMANN CAN bus
	Range:	0000..FFFF Hex	
	Page(s):	26, 27	
3070	**ErrCanBus1**		
	Level:	1	Error indication of HEINZMANN CAN bus
	Range:	0..1	
	Page(s):	27, 41	
3071	**ErrCanComm1**		
	Level:	1	Error indication of HEINZMANN CAN communication via CAN bus
	Range:	0..1	
	Page(s):	27, 28, 41	
3072	**ErrCanBus2**		
	Level:	1	Error indication of CANopen bus
	Range:	0..1	
	Page(s):	30, 42	
3073	**ErrCanComm2**		
	Level:	1	Error indication of CAN communication via CAN2 bus interface
	Range:	0..1	
	Page(s):	30, 42	
3074	**ErrCanOpen**		
	Level:	1	Error indication of CANopen communication ↑ *9.5 CANopen Error*
	Range:	0..1	
	Page(s):	42, 46	
3076	**ErrParamStore**		
	Level:	1	Error indication on programming the permanent memory
	Range:	0..1	
	Page(s):	43	
3077	**ErrProgramTest**		
	Level:	1	Error indication on checking the programme memory
	Range:	0..1	
	Page(s):	43	
3078	**ErrRAMTest**		
	Level:	1	Error reported during RAM Test
	Range:	0..1	
	Page(s):	43	
3085	**ErrVoltage**		
	Level:	1	Error indication of 24 V voltage supply
	Range:	0..1	
	Page(s):	43	

3090	**ErrData**		
	Level:	1	Error indication of data block
	Range:	0..1	
	Page(s):	44	
3093	**ErrStack**		
	Level:	1	Error indication of „Stack-Overflow“
	Range:	0..1	
	Page(s):	45	
3094	**ErrIntern**		
	Level:	1	Error indication for internal software fault
	Range:	0..1	
	Page(s):	45	
3099	**EEPROMErrorCode**		
	Level:	6	Error indication on checking EEPROM structure
	Range:	0000..FFFF Hex	
	Page(s):	45	
3170	**SErrCanBus1**		
	Level:	1	Sentinel for the occurrence of 3070 *ErrCanBus1*
	Range:	0..255	
	Page(s):	27, 41	
3171	**SErrCanComm1**		
	Level:	1	Sentinel for the occurrence of 3071 *ErrCanComm1*
	Range:	0..255	
	Page(s):	27, 28, 41	
3172	**SErrCanBus2**		
	Level:	1	Sentinel for the occurrence of 3072 *ErrCanBus2*
	Range:	0..255	
	Page(s):	30, 42	
3173	**SErrCanComm2**		
	Level:	1	Sentinel for the occurrence of 3073 *ErrCanComm2*
	Range:	0..255	
	Page(s):	30, 42	
3174	**SErrCanOpen**		
	Level:	1	Sentinel for the occurrence of 3074 *ErrCanOpen*
	Range:	0..255	
	Page(s):	42, 46	
3176	**SErrParamStore**		
	Level:	1	Sentinel for the occurrence of 3076 *ErrParamStore*
	Range:	0..255	
	Page(s):	43	
3177	**SErrProgramTest**		
	Level:	1	Sentinel for the occurrence of 3077 *ErrProgramTest*
	Range:	0..255	
	Page(s):	43	
3178	**SErrRAMTest**		
	Level:	1	Sentinel for the occurrence of 3078 *ErrRAMTest*
	Range:	0..255	
	Page(s):	43	
3185	**SErrVoltage**		
	Level:	1	Sentinel for the occurrence of 3085 *ErrVoltage*
	Range:	0..255	
	Page(s):	43	

3190	**SErrData**		
	Level:	1	Sentinel for the occurrence of 3090 *ErrData*
	Range:	0..255	
	Page(s):	44	
3193	**SErrStack**		
	Level:	1	Sentinel for the occurrence of 3093 *ErrStack*
	Range:	0..255	
	Page(s):	45	
3194	**SErrIntern**		
	Level:	1	Sentinel for the occurrence of 3094 *ErrIntern*
	Range:	0..255	
	Page(s):	45	
3195	**SExceptionNumber**		
	Level:	1	Error code for Exception errors
	Range:	0..65535	
	Page(s):	45	
3196	**SExceptionAddrLow**		
	Level:	1	Error code for Exception errors
	Range:	0000..FFFF Hex	
	Page(s):	45	
3197	**SExceptionAddrHigh**		
	Level:	1	Error code for Exception errors
	Range:	0000..FFFF Hex	
	Page(s):	45	
3198	**SExceptionFlag**		
	Level:	1	Error condition in case of Exception errors
	Range:	0000..FFFF Hex	
	Page(s):	45	
3600	**PowerSupply**		
	Level:	1	Current value of supply voltage
	Range:	0..55 V	
	Page(s):		
3840	**HardwareVersion**		
	Level:	1	Version number of control hardware
	Range:	0..9999	
	Page(s):	34	
3841	**AddHardwareVersion**		
	Level:	1	Additional version number of control hardware
	Range:	0..9999	
	Page(s):	34	
3842	**SoftwareVersion**		
	Level:	1	Version number of software
	Range:	0..65535	
	Page(s):	34	
3843	**BootSoftwareVersion**		
	Level:	1	Version number of boot software
	Range:	0..65535	
	Page(s):	34	
3844	**SerialDate**		
	Level:	1	Serial date of control hardware
	Range:	0..9912	
	Page(s):	34	

3845	**SerialNumber**		
	Level:	1	Serial number of control hardware
	Range:	0..65535	
	Page(s):	34	
3850	**Identifier**		
	Level:	1	Identification number of PC-programme /
	Range:	0..65535	Hand Programmer
	Page(s):	34	
3851	**LastIdentifier**		
	Level:	1	Identification number of the PC-programme / Hand
	Range:	0..65535	Programmer used for the parameter change stored last
	Page(s):	34	
3865	**CalculationTime**		
	Level:	1	Calculation time required by main processor
	Range:	0..16,384 ms	
	Page(s):		
3870	**Timer**		
	Level:	1	Internal millisecond timer
	Range:	0..65,535 s	
	Page(s):		
3895	**RAMTestAddr**		
	Level:	6	Value of currently tested memory address
	Range:	0000..FFFF Hex	
	Page(s):	43	
3896	**RAMTestPattern**		
	Level:	6	Value of test pattern for RAM test
	Range:	0000..FFFF Hex	
	Page(s):	43	
3897	**CStackTestFreeBytes**		
	Level:	6	Indication of free bytes in C-stack
	Range:	0000..0200 Hex	
	Page(s):		
3898	**IStackTestFreeBytes**		
	Level:	6	Indication of free bytes in interrupt stack
	Range:	0000..0200 Hex	
	Page(s):		
12401	**CanOpStopped**		
	Level:	6	CANopen State Machine in Stopped state
	Range:	0..1	
	Page(s):	6	
12402	**CanOpInit**		
	Level:	6	CANopen State Machine in Init state
	Range:	0..1	
	Page(s):	6	
12403	**CanOpPreOperational**		
	Level:	6	CANopen State Machine in Preoperational state
	Range:	0..1	
	Page(s):	6	
12404	**CanOpOperational**		
	Level:	6	CANopen State Machine in Operational state
	Range:	0..1	
	Page(s):	6	

12405	**CanOpOnline**		
	Level:	6	CANopen Gateway is ready to communicate by the CANopen bus via CAN
	Range:	0..1	
	Page(s):	30	
12411	**CanOpDCResetErr**		
	Level:	6	The Digital Governor has been reset but the CANopen Gateway has not
	Range:	0..1	
	Page(s):	46	
12412	**CanOpInitErr**		
	Level:	6	Configuration of CANopen communication cannot be performed
	Range:	0..1	
	Page(s):	46	
12413	**CanOpTimeCtrlErr**		
	Level:	6	The time monitoring of some RPDO has tripped in Operational state
	Range:	0..1	
	Page(s):	47	
12414	**CanOpDCErrCanBus**		
	Level:	6	CANopen communication error caused by 3070 *ErrCanBus* in the Digital Governor
	Range:	0..1	
	Page(s):	47	
12415	**CanOpDCErrCanComm**		
	Level:	6	CANopen communication error caused by 3071 *ErrCanComm* in the Digital Governor
	Range:	0..1	
	Page(s):	47	
12419	**CanOpGWResetErr**		
	Level:	6	CANopen Gateway has been reset but the Digital Governor has not
	Range:	0..1	
	Page(s):	48	
12420	**CanOpHeartbeatErr**		
	Level:	6	The heartbeat monitoring function has tripped in Operationale state
	Range:	0..1	
	Page(s):	48	
12421	**CanOpGWErrCanBus1**		
	Level:	6	CANopen communication error caused by 3070 *ErrCanBus1* in the CANopen Gateway
	Range:	0..1	
	Page(s):	48	
12422	**CanOpGWErrCanComm1**		
	Level:	6	CANopen communication error caused by 3071 *ErrCanComm1* in the CANopen Gateway
	Range:	0..1	
	Page(s):	49	
12423	**CanOpGWErrCanBus2**		
	Level:	6	CANopen communication error caused by 3072 *ErrCanBus2* in the CANopen Gateway
	Range:	0..1	
	Page(s):	49	
12424	**CanOpGWErrCanComm2**		
	Level:	6	CANopen communication error caused by 3073 *ErrCanComm2* in the CANopen Gateway
	Range:	0..1	
	Page(s):	50	

11.4 List 3: Functions

4400	**HzmCanCommDCOn**		
	Level:	6	Activation of CAN communication with the Digital Governor
	Range:	0..1	
	Page(s):	26, 29	
4416	**HzmCanSegmOrBaudrate**		
	Level:	6	Activation of manual adjustment of the Baud rate of the HEINZMANN CAN bus
	Range:	0..1	
	Page(s):	27, 29	
4426	**CanOpSegmOrBaudrate**		
	Level:	6	Activation of manual adjustment of the Baud rate of the CANopen bus
	Range:	0..1	
	Page(s):	29, 30	

11.5 List 4: Curves and Maps

16404	**CanOpTPDO5-16ID(0)**		
	Level:	6	CANopen identifiers of TPDO 5 to 16
to	Range:	0000..07FF Hex	
16415	Page(s):	32	

12 Index of Figures

13 Index of Tables

14 Order Specifications for Manuals

There is no charge for technical manuals ordered in reasonable quantities.

Order the necessary manuals on our speed governors from your nearest

HEINZMANN location.

(Please click on “HEINZMANN location” to see the list of our subsidiaries and agents in the world).

Please include the following information:

- your name,
- the name and address of your company (you can simply include your business card),
- The address where you want the manuals sent (if different from above),
- the number(s) and title(s) of the desired manual(s),
- or technical data of your HEINZMANN equipment,
- the quantity you want.

You can directly use the following fax-form for ordering one or several manuals.

In the meantime, most of our publications are available in PDF format. On request, they can be forwarded by e-Mail.

We solicit comments about the content and the presentation of our publications. Please, send your comments to:

HEINZMANN GmbH & Co. KG

Service Department

Am Haselbach 1

D-79677 Schönau

Germany

Fax Reply

Order for HEINZMANN technical manuals

Fax-Hotline +49 7673 8208 194

☐ **Please send me the following manuals:**

Quantity	No. of the manual	Title

☐ **Please send me your new sales documentation about**

( ) the HEINZMANN Analogue Governors Application:.....

( ) the HEINZMANN Digital Governors Application:.....

( ) the HEINZMANN Gas Engine Equipment Application:.....

Company .....

Contact Person.....

Department .....

Address..... Country/Code/Town.....

Phone..... Fax .....

E-Mail.....

Activity.....

Date .....