Heinzmann GmbH & Co. KG
Engine & Turbine Management

Am Haselbach 1
D-79677 Schönau
Germany

Phone +49 7673 8208 - 0
Fax +49 7673 8208 - 188
E-mail info@heinzmann.de
www.heinzmann.com

V.A.T. No.: DE145551926

HEINZMANN®
Engine & Turbine Management

Safety Systems

Triton OMD
Oil Mist Detection System

CAN Protocol

The appropriate manuals must be thoroughly studied before installation, initial start-up and maintenance.

All instructions pertaining to the system and safety must be followed in full. Non-observance of the instructions may lead to injury to persons and/or material damage.

HEINZMANN shall not be held liable for any damage caused through non-observance of instructions.

Independent tests and inspections are of particular importance for all applications in which a malfunction could result in injury to persons or material damage.

All examples and data, as well as all other information in this manual are there solely for the purpose of instruction and they may not be used for special application without the operator running independent tests and inspections beforehand.

HEINZMANN does not guarantee, neither expressly nor tacitly, that the examples, data or other information in this manual is free from error, complies with industrial standards or fulfils the requirements of any special application.

To avoid any injury to persons and damage to systems, the following monitoring and protective systems must be provided:

– Thermal overload protection

– Overspeed protection independent of the rpm controller

HEINZMANN shall not be held liable for any damage caused through missing or insufficiently rated overspeed protection.

The following must also be provided for ship propulsion:

– Second power supply for the engine control system

If the governor is used for propulsion, a second power supply has to be provided to the engine control system if the governor can't keep its position after power failure.

The following must also be provided for alternator systems:

– Overcurrent protection

– Protection against faulty synchronisation for excessively-large frequency, voltage or phase difference

– Directional contactor

The reasons for overspeeding may be:

– Failure of positioning device, control unit or its auxiliary devices

– Linkage sluggishness and jamming

	**The following must be observed before an installation:** – Always disconnect the electrical mains supply before any interventions to the system. – Only use cable screening and mains supply connections that correspond with the *European Union EMC Directive* – Check the function of all installed protection and monitoring systems
	**Please observe the following for electronically controlled injection (MVC):** – For **common rail** systems each injector line must be equipped with a separate mechanical flow-rate limiter – For **unit pump** (PLD) and **pump-injector unit** (PDE) systems, the fuel enable is first made possible by the solenoid valve's control plunger motion. This means that in the event of the control plunger sticking, the fuel supply to the injection valve is stopped.
	As soon as the positioning device receives power, it can actuate the controller output shaft automatically at any given time. The range of the controller shaft or control linkage must therefore be secured against unauthorised access.
	HEINZMANN expressly rejects any implied guarantee pertaining to any marketability or suitability for a special purpose, including in the event that HEINZMANN was notified of such a special purpose or the manual contains a reference to such a special purpose.
	HEINZMANN shall not be held liable for any indirect and direct damage nor for any incidental and consequential damage that results from application of any of the examples, data or miscellaneous information as given in this manual.
	HEINZMANN shall not provide any guarantee for the design and planning of the overall technical system. This is a matter of the operator its planners and its specialist engineers. They are also responsible for checking whether the performances of our devices match the intended purpose. The operator is also responsible for a correct initial start-up of the overall system.

Table of contents

Page

Revision Index

Revision No.	Date of modification	Name	Description of modification
01	10.10.2019	ReY	First edition
02	18.10.2019	ReY	Section “7.6.2 Error and status report”: Bit description for OMD Error Status Report data items was revised according to the latest software state.
03	20.03.20	ReY	External range for temperature parameters was changed.
04			
05			

1 Safety instructions and related symbols

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

- Personnel

- Product and machine

- The environment

The primary aim of the safety instructions is to prevent personal injury!

The signal words used in this publication are specifically designed to direct your attention to possible damage extent!

⚠ DANGER ***DANGER indicates a hazardous situation which, if not avoided, will result in death or serious injury..***

⚠ WARNING ***WARNING indicates a hazardous situation which, if not avoided, could result in death or serious injury.***

⚠ CAUTION ***CAUTION indicates a hazardous situation which , if not avoided, may result in minor or moderate injury.***

NOTICE *NOTICE indicates a property damage message.*

Safety instructions are not only denoted by a signal word but also by hazard warning triangles. Hazard warning triangles can contain different symbols to illustrate the danger. However, the symbol used is no substitute for the actual text of the safety instructions. The text must therefore always be read in full!

This symbol does not refer to any safety instructions but offers important notes for better understanding the functions that are being discussed. They should by all means be observed and practiced.

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 them.
- Before turning the installation on please verify and make sure that

– only authorized persons 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 superior 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!

1.1 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 hose to clean cabinets or other casings of electric equipment!

1.2 Before putting an installation into service after naintenance and repair works

- Check on all slackened screw connections to have been tightened again!
- 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 System description

2.1 Proper and intended use

CAN protocol, discussed in this document, is used for internode communication in OMD network. The network can consist of one master device (OMD evaluator or superordinate system) and up to 16 OMD sensor devices.

2.2 Functional description

Communication in the OMD network is based on the following key functions:

- HEINZMANN CAN protocol is used for internode communication within the network.
- Each device in the network has its unique ID which is used as node number item of CAN telegram identifiers.
- Automatic ID assignment procedure is used to start communication within the network after system power up.
- After successful ID assignment, sequence for checking of operational parameters set consistency of all OMD sensors on bus is foreseen. If required, master device broadcasts valid parameter set to the OMD sensors.
- Only after successful ID definition and operational parameter set matching, OMD sensors broadcast own operational data.

3 HEINZMANN CAN protocol

HEINZMANN CAN protocol is based on the CAN specification 2.0B with 29 bit identifier.

The identifier contains information about sender and receiver and the command code. Maximum 8 data bytes are therefore available completely for operative data.

3.1 Identifier structure

Bit range	Bit range	Meaning	Value
28…27	p	priority	always 2
26…23	d	type of receiving device	0…15
22…18	m	destination node number	0, 1….31
17	r	reserved	always 0
16…13	s	type of sending device (source)	0…15
12…8	n	source node number	1…31
7…0	c	command	0…255

Tab. 3.1 Identifier structure

Each connection in a HEINZMANN CAN network is therefore a point-to-point connection. A telegram is sent by a unique source to a unique destination. An exception is transmission of a command to all units of the same type using node number 0.

3.2 Node types

Sources and destinations are subdivided in node type (device type) and a node number.

OMD sensor and OMD evaluator are internally treated as AC-devices and have device type item of identifier equal to 5, which is used in telegram identifier.

3.3 Telegram identifier

Basic structure of the 29-bit form of the identifier is as follows:

- mmmmm - are the five bits for the node number of the receiving device;
- nnnnn - is the node number of the sending device;
- cccccccc - is the command code.

Communication direction	p	d	m	r	s	n	c	Identifier
AC to AC	2	5	m	0	5	n	c	10 0110 mmmmm 0 0110 nnnnn cccccccc

Tab. 3.2 Identifier template

3.4 Node numbers

For each device type in the HEINZMANN CAN network each node number in the range between 1 and 31 must be assigned only once.

Node number 0 is not allowed for a single device, since it is used as the number for messages to all nodes of a certain type.

In OMD system the following node numbers are used:

- Evaluator has node number 31:
- Sensors have node numbers from 1 to 16.

3.5 Baud rate

Baud rate is fixed and is equal to 250 kBaud.

3.6 Commands

The possible command codes of the telegrams and the respective data mapping are described in the next section of this document.

3.7 OMD network communication basics

Communication network topology is shown on the picture below:

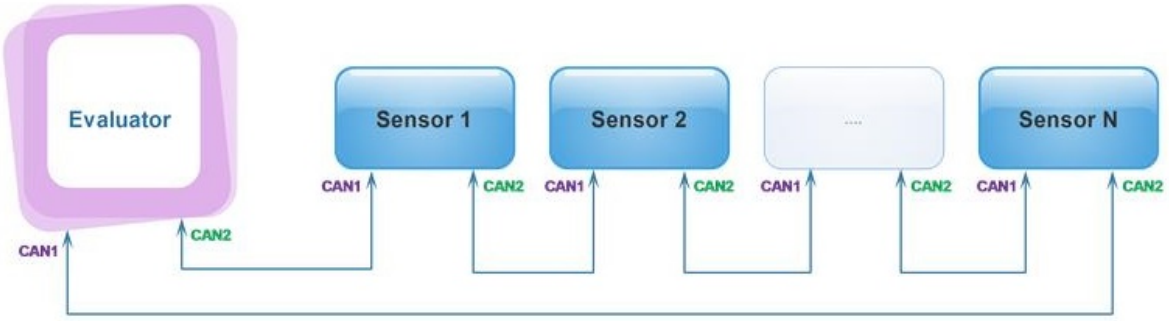

Fig. 3.1 Communication network topology

Communication between master device (evaluator or superordinate system) and HEINZMANN OMD sensors is organized in the following way:

- Only the first sensor - sensor 1 and the last sensor - sensor N (N – actual number of sensors in the network) in the network are connected to master device directly (via CAN1 and CAN2 respectively). Other sensors establish direct communication to the master over partner-devices on both sides via CAN1 and CAN2.
- Communication is point-to-point and is handled as a tunneling connection.
- If destination node number data item of the incoming telegram identifier (mmmmm) is not equal to current receiver node number (ID), then the message is retransmitted to the next node in the network until it reaches the receiver specified with (mmmmm). Otherwise, the message is processed and corresponding response is transmitted to the producer or other predefined actions are performed by the receiver device.

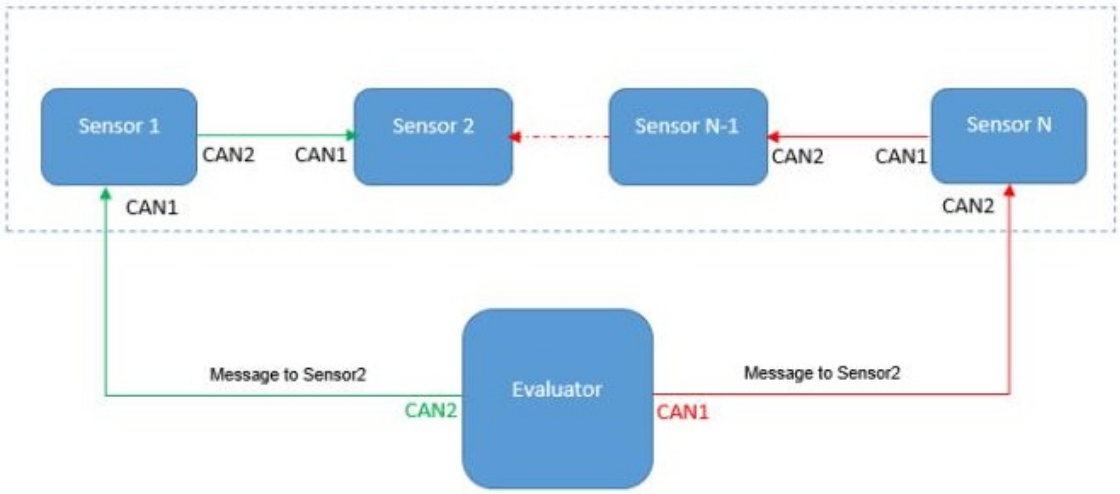

Fig. 3.2 Tunneling connection

4 OMD evaluator commands overview

4.1 Overview of send telegrams

Command	Telegram	Remark	Reference
95	kSensNumDirectCountCommand	Default	5.1 Direct sensor number counter command
97	kNetworkIDDistributRequest	Default	5.2 Network ID distribution command
49	kNetworkTimeSynchCommand	Default	5.3 Network time synchronization command
50	kNetworkTimeUpdateCommand	Default	5.4 Network time update command
51	kSensorConfigDataTel1	Default	5.5 Sensor configuration data telegram
84	kInternFunctionReques*t*	Default	5.6 Sensor internal function execution request telegram

Tab. 4.1 Overview of send telegrams

4.2 Overview of receipt telegrams

Command	Telegram	Remark	Reference
95	kSensNumDirectCountCommand	Default	6.1 Direct sensor number counter command response
96	kSensNumReverseCountCommand	Default	6.2 Revers sensor number counter command response
99	kSensorCRCInvalidReport	Default	6.3 Sensor configuration CRC invalidity report
20	kMesurementValueReport	Default	6.4 Measurement value report
40	kErrorStatusReport	Default	6.5 Error and status report

Tab. 4.2 Overview of receipt telegrams

5 Send telegrams specification

5.1 Direct sensor number counter command

Command: 95
Data bytes: 2
Destination: common designated message from master to all sensors.
Timeout: 1 s
Activation: automatic, if all the following conditions are fulfilled:

- unit number verification sequence is in process;
- timeout for feedback from network units has expired.

	Byte 0	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
	High byte	Low byte	High byte	Low byte	High byte	Low byte	High byte	Low byte
Data item	Sensor Number Counter	Tx. CAN Output Tag	-	-	-		-	-

Tab. 5.1 Command 95

Telegram function details are described in section 7.1 Actual sensor number verification.

5.2 Network ID distribution command

Command: 97
Data bytes: 8
Destination: common designated message from master to all sensors.
Timeout: 0,5 s (for transmit attempt condition check)
Activation: automatic, if all the following conditions are fulfilled:

- ID distribution sequence is in process;
- Timeout for transmit attempt has expired;
- RTC Unix second counter has been changed (incremented).

	Byte 0	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
	High byte	Low byte	High byte	Low byte	High byte	Low byte	High byte	Low byte
Data item	Actual number of Sensors in the network		Masters OMD parameter set CRC		Unix Time High Word		Unix Time Low Word	

Tab. 5.2 Command 97

Telegram function details are described in section 7.2 Automatic sensor ID assignment.

5.3 Network time synchronization command

Command: 49
Data bytes: 4
Destination: common designated message from master to all sensors.
Timeout: 600 s (for transmit attempt condition check)
Activation: automatic, if all the following conditions are fulfilled:

- Network ID distribution sequence is finished
- Timeout for transmit attempt has expired
- RTC Unix second counter has been changed (incremented).

	Byte 0	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
	High byte	Low byte	High byte	Low byte	High byte	Low byte	High byte	Low byte
Data item	Unix Time High Word		Unix Time Low Word		-	-	-	-

Tab. 5.3 Command 49

Telegram function details are described in section 7.3 Network time synchronization.

5.4 Network time update command

Command: 50
Data bytes: 4
Destination: common designated message from master to all sensors.
Timeout: *not periodical telegram*
Activation: automatic, if all the following conditions are fulfilled:

- RTC settings have been changed by the user.

	Byte 0	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
	High byte	Low byte	High byte	Low byte	High byte	Low byte	High byte	Low byte
Data item	Unix Time High Word		Unix Time Low Word		-	-	-	-

Tab. 5.4 Command 50

Telegram function details are described in section 7.3 Network time synchronization.

5.5 Sensor configuration data telegram

Command: 51
Data bytes: 8
Destination: from master individual sensor.
Timeout: *not periodical telegram*
Activation: automatic, if all the following conditions are fulfilled:

- Sensor configuration is invalid;
- Sensor configuration sequence is in process;
- Sensor is online.

Command: 51

	Byte 0	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
	High byte	Low byte	High byte	Low byte	High byte	Low byte	High byte	Low byte
Data item	Configuration Parameter 1 Value		Configuration Parameter 2 Value		Configuration Parameter 3 Value		Configuration Parameter 4 Value	

Tab. 5.5 Command 51

Telegram function details are described in section 7.4 Individual sensor configuration.

5.6 Sensor internal function execution request telegram

Command: 84
Data bytes: 2
Destination: common designated message from master to all sensors.
Timeout: *not periodical telegram*
Activation: It is masters responsibility to activate the transmission of this telegram.
In HEINZMANN OMD evaluator device it activated with DcDesk option "Error – Clear errors" or corresponding GUI option.

	Byte 0	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
	High byte	Low byte	High byte	Low byte	High byte	Low byte	High byte	Low byte
Data item	Internal function execution command		-	-	-	-	-	-

Tab. 5.6 Command 84

Telegram function details are described in section 7.5 Sensor internal function execution request.

6 Receipt telegram specification

6.1 Direct sensor number counter command response

Command: 95
Data bytes: 2
Destination: common designated message to all devices.
Timeout: 1 s

	Byte 0	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
	High byte	Low byte	High byte	Low byte	High byte	Low byte	High byte	Low byte
Data item	Sensor Number Counter	Tx. CAN Output Tag	-	-	-		-	-

Tab. 6.1 Command 95

Telegram function details are described in section 7.1 Actual sensor number verification.

6.2 Reverse sensor number counter command response

Command: 96
Data bytes: 2
Destination: common designated message to all devices.
Timeout: 1 s

	Byte 0	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
	High byte	Low byte	High byte	Low byte	High byte	Low byte	High byte	Low byte
Data item	Sensor Number Counter	Tx. CAN Output Tag	-	-	-	-	-	-

Tab. 6.2 Command 96

Telegram function details are described in section 7.1 Actual sensor number verification.

6.3 Sensor configuration CRC invalidity report

Command: 99
Data bytes: 2
Destination: from individual sensor to master.
Timeout: not periodical telegram

	Byte 0	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
	High byte	Low byte	High byte	Low byte	High byte	Low byte	High byte	Low byte
Data item	Sensor Configuration CRC		-	-	-		-	-

Tab. 6.3 Command 99

6.4 Measurement value report telegram

Command: 20
Data bytes: 8
Destination: from individual sensor to master.
Timeout: 0.2 s

	Byte 0	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
	High byte	Low byte	High byte	Low byte	High byte	Low byte	High byte	Low byte
Data item	OM Concentration [mg/l]		OM Concentration Alarm Percentage		Measured Tempera-ture		Message Counter	

Tab. 6.4 Command 20

Telegram function details are described in section 7.5.1 Measurement value report.

6.5 Error status report telegram

Command: 40
Data bytes: 8
Destination: from individual sensor to master.
Timeout: 0.5 s

	Byte 0	**Byte 1**	**Byte 2**	**Byte 3**	**Byte 4**	**Byte 5**	**Byte 6**	**Byte 7**
	High byte	Low byte	High byte	Low byte	High byte	Low byte	High byte	Low byte
Data item	OMD Error Status (detailed)		Sensor Error State		Sensor Output State		Reserved	Reserved

Tab. 6.5 Command 40

Telegram function details are described in section 7.5.2 Error and status report.

7 General communication functions overview

7.1 Actual sensor number verification

7.1.1 Network connection healthy scenario

Communication in the network starts with verification of preset number of OMD sensors in the network, which is configured with parameter 200 *OMDSensorUnitsNumber*.

Sensor number verification procedure is initiated by the master device (evaluator) by means of common designated service message *kSensNumDirectCountCommand*, sent via CAN1 and CAN2.

Data item within this message:

- Running sensor counter (is equal to 0 in masters tx-message);
- CAN channel Tag (0 - for CAN1 and 1 – for CAN2 in masters tx-message).

On receipt of *kSensNumDirectCountCommand* each sensor increments the running sensor counter data item and retransmitts the telegram (with a modified sensor counter value) to the next unit in the network via opposite CAN-output, i.e. if *kSensNumDirectCountCommand* is received via CAN1 – it will be retransmitted via CAN2, and vice versa.

In the end of the procedure both *kSensNumDirectCountCommand* telegrams, transmitted by the master via CAN1 and CAN2, are received by itself via opposite CAN-ports, e.g. *kSensNumDirectCountCommand*, transmitted from masters CAN1, is received via masters CAN2, and vice versa.

CAN channel Tag data item of the received telegram is used for additional verification of opposite source CAN-port and destination CAN-ports.

Received running sensor counter (direct counter) value represents actual number of sensors in the network and is displayed with parameter 2581 *UnitNumberCounter*.

If direct counter value (parameter 2581 *UnitNumberCounter*) is equal to preset sensor number stored in evaluator (parameter 200 *OMDSensorUnitsNumber*) - verification succeeds, that is displayed with parameter 2580 *UnitNumberApproved* = 1.

In case of unit number verification procedure success, ID distribution sequence can be initiated by evaluator. Otherwise, verification failure scenario should be considered.

7.1.2 Network connection break scenario

In case of connection problem between any two units in the CAN-chain, *kSensNumDirectCountCommand*, sent by the master, will not reach its opposite CAN-input. The problem is detected by the master when *kSensNumDirectCountCommand* is not received within certain timeout, and is displayed with parameter 2582 *CanNetFailure* = 1. Therefore sensor number verification is not possible with *kSensNumDirectCountCommand* in such case. The alternative way to verify unit number is using *kSensNumReversCountCommand*.

On receipt of *kSensNumDirectCountCommand*, in addition to telegram retransmission, sensor also responds to the master with *kSensNumReversCountCommand* telegram, with running sensor counter data item value incremented.

Master can easily calculate the actual number of the sensors using the following logic:

Actual number of sensors in the network is equal to the sum of highest running sensor counter (reverse counter) values, received by master on CAN1 and CAN2 ports.

The calculated number of sensors is displayed with parameter 2581 *UnitNumberCounter*.

If reverse counter value (parameter 2581 *UnitNumberCounter*) is equal to preset sensor number stored in evaluator (parameter 200 *OMDSensorUnitsNumber*) - verification succeeds, that is displayed with parameter 2580 *UnitNumberApproved* = 1.

In case of unit number verification procedure success, ID distribution sequence can be initiated by evaluator. Otherwise, verification failure scenario should be considered.

7.1.3 Sensor number verification failure scenario

If counted sensor number (parameter 2581 *UnitNumberCounter*) value does not correspond to preset sensor number, stored in evaluator (parameter 200 *OMDSensorUnitsNumber*), number of sensor-units in the network is treated as not approved. This is displayed by parameter 2580 *UnitNumberApproved* = 0.

In case when parameter 2581 *UnitNumberCounter* is less than parameter 200 *OMDSensorUnitsNumber*, node assignment procedure can be started, and communication in the network will be handled considering that missing nodes are offline.

In case when parameter 2581 *UnitNumberCounter* is higher than parameter 200 *OMDSensorUnitsNumber*, node assignment procedure **cannot** be started, and further effective communication in the network is not possible. This situation is additionally displayed by parameter 2583 *CanNetConfigInvalid* = 1.

7.2 Automatic sensor ID assignment

Automatic node number assignment procedure in the network is started after unit number verification sequence succeeded.

Master initiates ID assignment in the network with common designated service message *kNetworkIDDistributRequest*, sent on CAN1 and CAN2. The rest of the ID assignment procedure is performed by sensors themselves.

The ID, assigned to each sensor acts as CAN telegram identifier node number item for further communication in the network.

Data items within *kNetworkIDDistributRequest* message:

- Actual number of sensors in the network;
- Masters configuration parameter set CRC value;
- Masters time and date (Unix time).

7.3 Network time synchronization

Master is responsible for time synchronization of each individual sensor in the network. The first synchronization is performed during automatic ID assignment procedure, when master sends its current time stamp within telegram *kNetworkIDDistributRequest*.

After this, master sends common designated message *kNetworkTimeSynchComman* via CAN1 and CAN2 every 600s. Furthermore, master additionally sends *kNetworkTimeUpdateCommand* message if its time and date setting were changed by the user.

The messages are distributed in the network by sensor devices themselves.

Data items within *kNetworkTimeSynchCommand* and *kNetworkTimeUpdateCommand* messages:

- Masters time and date (Unix time).

For the cases when OMD sensor is used with third party superordinate system instead of OMD evaluator, it is important to take into consideration specific feature for correct time synchronization of the whole network. This issue is especially critical for data logging functionality of sensor device.

The transmission of telegrams *kNetworkIDDistributRequest*, *kNetworkTimeSynchCommand*, *kNetworkTimeUpdateCommand* by the master must be triggered by second counter increment event of master clock, e.g. on the edge of new second. Even when all the other condition for transmission are fulfilled, master unit should wait for new second edge before sending one of these messages.

7.4 Individual sensor configuration

After successful automatic ID (node number) assignment procedure each sensor starts communication to master.

Sensor device makes the decision about its configuration validity (or invalidity) on the base of comparison of masters configuration parameter set CRC value (received during automatic node number assignment procedure) to own CRC calculated trough actual configuration parameter set.

In case of configuration parameter set invalidity on individual sensor side, master receives cyclic service message *kSensorCRCInvalidReport* from corresponding sensor. This means that master should broadcast its configuration parameter set to the sensor.

After successful configuration parameter set update with new values, sensor recalculates CRC through updated parameter set. In case new CRC value is equal to masters configuration parameter set CRC value, sensor stops transmission of *kSensorCRCInvalidReport* message and start reporting operational data to the master.

7.4.1 Configuration parameter set

For sensor to be configured with valid OMD parameter set, master sends corresponding parameter values within special configuration telegram *kSensorConfigDataTel1* to the sensor device.

OMD Parameter Set is specified in the table below.

No.	Indicated value	External value range	Unit	Internal value range
117	OMConcAlaramLevel	0.00 – 20.00	mg/l	0 - 65535
118	OMConcPrealaramLevel	0.00 – 20.00	mg/l	0 - 65535
119	TempAlarmLevel	-128.0 – 128.0	°C	0 - 65535
120	TempPrealarmLevel	-128.0 – 128.0	°C	0 - 65535

Tab. 7.1 OMD parameter set

7.4.2 CRC check sum calculation

Both OMD devices, evaluator and sensor, use the same MCU type with embedded cyclic redundancy check (CRC) module. The modules are configured for Standard CCITT CRC16 Check Sum with following specification:

- Width = 16 bits;
- Polynomial = 0x1021;
- Initial value (Seed) = 0xFFFF;
- Input data is 8bits at a time, providing all bytes continuously;
- Input data is not transposed;
- Output CRC is not transposed;
- No XOR is performed on the output CRC.

Values 0x0000 and 0xFFFF are not allowed as a final CRC value, calculated through desired data set. In such cases 0x0001 is returned instead.

7.4.3 Configuration data memory layout

Both OMD devices, evaluator and sensor, use the same MCU type. For the cases when OMD sensor is used with third party superordinate system instead of OMD evaluator, it is important to consider configuration data memory layout of sensor device. This is critical for correct configuration parameter set CRC calculation by superordinate system.

Configuration data memory layout features:

- Little endian data format;
- 16-bit sized parameter data item;
- 2-byte parameter data alignment.

7.5 Sensor internal function execution request

Master can request internal function execution in all sensor-devices in the network by means of telegram *kInternFunctionRequest*.

Data items within this telegram:

- Internal function execution command.

Bit description for internal function execution command data item is provided in the tables below.

Bit Number.	Bit function
0	Clear current errors
1	free
2	free
3	free
4	free
5	free
6	free
7	free
8	free
9	free
10	free
11	free
12	free
13	free
14	free
15	free

Tab. 7.2 Sensor internal function execution command

7.6 Sensor operational data reception

After successful ID assignment and configuration each individual sensor starts cyclical transmission of its operational data to the master.

Operational data are delivered to master by means of the following telegrams:

- kMesurementValueReport;
- kErrorStatusReport;

7.6.1 Measurement value report

Measured data values are delivered to the master by means of telegram *kMesurementValueReport*.

Data items within this telegram:

- Measured OM concentration value [mg/l];
- Measured OM concentration value [concentration% / alarm level];
- Measured temperature value [°C];
- Telegram counter.

As soon as master receives the identical instances of telegrams on CAN1 and CAN2 from each individual sensor, telegram counter data item is used for correct sorting of measurement values on masters side. Note: sensor resets telegram counter value when it reaches 255.

No.	Indicated value	External value range	Unit	Internal value range
2620 – 2635.	OMConcentration_Sensor1 – OMConcentration _Sensor16	0.00 – 20.00	mg/l	0 - 65535
2640 - 2655	OMAllScore _Sensor1 – OMAllScore _Sensor16	0.0 – 100.0	%	0 - 65535
2660 - 2675	Temperature_Sensor1 – Temperature_Sensor16	-128.0 – 128.0	ºC	0 - 65535

Tab. 7.3 Sensor measurement value control parameters

7.6.2 Error and status report

Error state and general status information is delivered to the master by means of telegram *kErrorStatusReport.*

Data items within this telegram:

- OMD Error Status;
- Sensor Error State;
- Sensor Output State.

No.	Indicated value	External value range	Unit	Internal value range
2680 - 2695	OMDError_Sensor1 – OMDError_Sensor16	0 - FFFF	Hex	0 - FFFF
2700 - 2715	ErrorState_Sensor1 – ErrorState _Sensor16	0 - FFFF	Hex	0 - FFFF
2720 - 2735	StatusInfo_Sensor1 – StatusInfo_Sensor16	0 - FFFF	Hex	0 - FFFF

Tab. 7.4 Sensor error and status control parameters

Bit description for OMD error status report data items is provided in the tables below.

Bit Number.	Bit function
0	Broken Wire LERD
1	Broken Wire LRD C
2	Broken Wire LRD D
3	Error Scattered Connected
4	Dirty LERD A
5	Dirty LERD B
6	Primary Alarm
7	Pre-Alarm
8	Maintenance
9	free
10	free
11	free
12	free
13	free
14	Common Error
15	Fatal Error

Tab. 7.5 OMD-error status bits

Bit description for sensor error state data items is provided in the tables below.

Bit Number.	Bit function
0	Error CAN Bus 1
1	Error CAN Bus 2
2	Error CAN Comm 1
3	Error CAN Comm 2
4	Error Data Flash
5	Error EEPROM
6	Error Power Supply
7	Error DRV10983
8	Error Real Time Clock
9	Error Temperature MCP9808
10	Error Internal Temperature CPU
11	Error Configuration
12	Error Internal
13	free
14	free
15	free

Tab. 7.6 Sensor error state bits

Bit description for sensor output state data items is provided in the tables below.

Bit Number.	Bit function
0	OMD Alarm
1	OMD Prealarm
2	Maintenance Level1
3	Maintenance Level2
4	Ready
5	free
6	free
7	free
8	free
9	free
10	free
11	free
12	free
13	free
14	free
15	free

Tab. 7.7 Sensor output state bits

8 Lists

8.1 List of figures

8.2 List of tables

9 Download of manuals

Technical manuals can be downloaded in pdf-format from our homepage:

www.heinzmann.com

If the desired manual should not be available there, send an e-mail to:

info@heinzmann.de

or write to:

HEINZMANN GmbH & Co. KG

Technische Redaktion

Am Haselbach 1

D-79677 Schönau/Germany

Please include the following information:

- your name
- name and address of your company
- e-mail or address where you want the manuals to be sent to (if different from above)
- item code (as on front page, bottom right) and title of the desired manual or alternatively technical data of your HEINZMANN equipment

We would be pleased to receive your comments about the contents and presentation of our publications. Send your comments to the e-mail or address shown above please.