GEFRAN

KHC

User Manual

Code 85206 **Edition** 12-2015

SUMMARY

1. INTRODUCTION

The GEFRAN KHC transducer is a Digital Pressure Sensor with CANopen interface. It implements the standard CANopen communications protocol defined by CiA (CAN in Automation).

The CANopen standards supported by the device are listed in the following table.

CiA standard	Description	Version
DS 301	CANopen application layer and communication profile	4.2.0
DS 305	Layer setting services (LSS) e Protocolli	3.0.1
DS 404	Device profile for measuring devices and closed-loop controllers	1.0.1
DR 303-2	Representation of SI units and prefixes	1.5.0

Table 1 - Supported CANopen standards

This document describes the CANopen implementation on the GEFRAN KHC CANopen device. It is addressed to CANopen network system integrators and to CANopen device designers who already know the content of the above-mentioned standards defined by CiA.

The details of aspects defined by CANopen do not pertain to the purpose of this text. For further information on the CANopen protocol see www.can-cia.de

2. GET STARTED PROCEDURE

2.1 NODE PARAMETERS SETTING

Before connecting the GEFRAN KHC sensor to a fully configured and working CAN bus, some basic configuration actions have to be performed. The configuration involves the Node-ID and the Baud rate of the CANopen device.

The configuration is mandatory if at least one of these conditions is true:

1) The Node-ID of the GEFRAN KHC sensor is identical to the Node-ID of another CANopen device connected to the CAN bus.
2) The GEFRAN KHC sensor operates with a baud rate different from the CAN bus baud rate.

If the condition at point 2 in not verified, the configuration can also be performed on that CAN bus, but all the other CANopen devices on the CAN bus should be taken in power-off state during the configuration process in order to avoid errors or conflicts.

If the baud rate configuration has to be performed, the GEFRAN KHC sensor must be connected to a CAN bus that works at the same baud rate of the sensor. The baud rate of the actual CAN bus (with all devices connected to it) can also be temporary set equal to the sensor baud rate until configuration is done. The configuration is made using LSS (Layer Setting Services).

Switching to LSS configuration mode

The first operation is to switch the sensor into LSS configuration mode. If the sensor is the only device on the CAN bus (with the LSS master), the LSS Switch State Global command can be used.

Source	COB-ID	DLC	Data	Destination
Controller	7E5h	08h	04h; 01h; 00h; 00h; 00h; 00h; 00h; 00h	Sensor

Figure 1 - LSS Switch State Global command

If there are other devices on the CAN bus (except the LSS master), the LSS Switch State Selective command must be used. Refer to the LSS Services section for details.

Setting the Node-ID

If the Node-ID of the sensor has to be changed, the LSS Configure Node-ID command must be used.

Source	COB-ID	DLC	Data	Destination
Controller	7E5h	08h	11h; **7Eh***; 00h; 00h; 00h; 00h; 00h; 00h	Sensor
Sensor	7E4h	08h	11h; **00h****; 00h; 00h; 00h; 00h; 00h; 00h	Controller

Figure 2 - LSS Configure Node-ID command

* the Node-ID value to be configured, within 1..127 (126 in this example).
** if value is 1, it means Node-ID out of range, i.e. the command was not accepted.

Setting the baud rate

If the baud rate of the sensor has to be changed, the LSS Configure Bit Timing Parameters command must be used.

Source	COB-ID	DLC	Data	Destination
Controller	7E5h	08h	13h; 00h; **02h***; 00h; 00h; 00h; 00h; 00h	Sensor
Sensor	7E4h	08h	13h; **00h****; 00h; 00h; 00h; 00h; 00h; 00h	Controller

Figure 3 - LSS Configure Bit Timing Parameters command

* the table-index of the corresponding bit rate (500kbit/s in this example).
Refer to the Table index in the LSS Configure Bit Timing Parameters section for details.
** If the value is 1 means that the bit timing is not supported; the command was not accepted.

Storing configuration settings

To save the previously configured Node-ID and Baud rate permanently (to non-volatile memory of the device) the LSS Store Configuration command must be used.

Source	COB-ID	DLC	Data	Destination
Controller	7E5h	08h	17h; 00h; 00h; 00h; 00h; 00h; 00h; 00h	Sensor
Sensor	7E4h	08h	17h; **00h***; 00h; 00h; 00h; 00h; 00h; 00h	Controller

Figure 4 - LSS Store Configuration command

* value other than 0, means store operation failed.

Verifying configuration setting

To check if the configuration settings of the device have been correctly executed and stored, proceed as follows:

1. power off the device
2. set the baud rate of the CAN bus to the correct value
3. power on the device

If the boot-up message is received, it means that the device baud rate setting is correct. The Node-ID of the device is contained inside the COB-ID of the message (boot-up COB-ID = 700h + Node-ID).

The format of the boot-up message is specified in the following figure.

Source	COB-ID	DLC	Data	Destination
Sensor	700h + Node-ID	01h	00h	Controller

Figure 5 – Boot-up message format

2.2 OPERATING PARAMETERS SETTING

After configuring the node parameters, the sensor can be integrated in the CANopen network. When powering on, the sensor transmits the boot-up message, and it goes into the Pre-operational state.

Before requesting process data, configuration of operating parameters of the sensor can be performed. Configuration of operating parameters is made through SDO Services (Service Data Objects). Through SDO services, it is possible for example to change the transmission type of the PDO (Process Data Object) selecting the synchronous (through SYNC messages) or asynchronous (through event-timer) mode, change the transmission time (event timer) of the asynchronous PDO, change the PDO mapping, etc.

It is possible to save changed parameters in non-volatile memory accessing the Store Parameters object through SDO, or restore default parameters with the Restore Default Parameters object.

It is possible to access all the objects specified in the Object Dictionary of the device (see Object Dictionary section).

SDO Services are available in Pre-operational and Operational states only (see NMT Services section).

2.3 REQUESTING PROCESS DATA

The GEFRAN KHC CANopen pressure sensor provides one Transmit PDO (TPDO1), with two mapped objects by default:

- 1^{st} application mapped object: pressure data (object 9130h or 6130h or 2090h)
- 2^{nd} application mapped object: status (object 6150h)

A third object, temperature data, can be mapped (see PDO mapping).

TPDO1 data format

Pressure and status data are mapped in TPDO1 as shown in the following figure.

COB-ID	DLC	D0	D1	D2	D3	D4
180h + Node-ID	5	Pressure data				Status

Figure 6 - TPDO1 mapped data

The physical unit of the pressure data can be set through the object 6131h (AI physical unit PV). If the pressure data mapped in TPDO1 is of integer type (i.e. mapped objects are 2090h or 9130h), the value has to be rescaled considering the value of the object 6132h (AI decimal digits).

If the pressure data mapped in TPDO1 is of float type, the value has not to be rescaled.

Byte ordering of pressure data inside TPDO1 follows the LSB..MSB ordering scheme.

TPDO1 data transmission

The transmission of the Process Data Object is made when the sensor is in Operational state. To start data transmission, the master sends the NMT "Start" command, as shown in the following figure.

Source	COB-ID	DLC	Data	Destination
Controller	000h	02h	01h; **00h***	Sensor

Figure 7 - NMT "Start" command

* 00h: all nodes, nnh: only the node with Node-ID equal to nnh

To stop data transmission the master sends the "Enter NMT Pre-operational state" command, as shown in the following figure.

Source	COB-ID	DLC	Data	Destination
Controller	000h	02h	80h; **00h***	Sensor

Figure 8 - NMT "Enter NMT pre-operational" command

* 00h: all nodes, nnh: only the node with Node-ID equal to nnh

The GEFRAN KHC CANopen device also supports the auto-operational mode. If the auto-operational mode is activated, by setting to 1 the object 2330h, the transition to NMT operational state is automatic when the device is powered on and the initialization is completed.

The transmission of the Process Data Object automatically starts after the boot-up message has been transmitted.

When the auto-operational mode is active, the device still accepts all NMT commands. After a NMT reset command, the device goes in NMT pre-operational state.

The auto-operational mode is typically used when no NMT master is available.

2.4 DEFAULT PARAMETERS SETTINGS

The GEFRAN KHC parameters default settings are listed in the following table

Parameter Name/Description	Object (Index,Subindex)	Default Value
Transmission speed	2320,0	250 kbps*
Node-ID	2321,0	1*
Number of mapped objects	1A00,0	2
PDO mapping, 1st object	1A00,1	9130h (AI input PV, integer32)*
PDO mapping, 2nd object	1A00,2	6150h (AI status)
PDO mapping, 3rd object	1A00,3	2091h (Temperatura)
COB-ID SYNC	1005,0	80h
COB-ID EMCY	1014,0	80h + Node-ID
COB-ID SDO rx	1200,1	600h + Node-ID
COB-ID SDO tx	1200,2	580h + Node-ID
COB-ID TPDO	1800,1	180h + Node-ID
AI physical Unit PV	6131,1	004E0000h (bar)
AI decimal digits	6132,1	2

Table 2 – Parameters default values

Parameters values marked with (*) can be selected during the ordering phase of the GEFRAN KHC sensor. The allowed range of selectable values is listed in the following table.

Parameter Name/Description	Selectable values
Transmission speed	20, 50, 100, 125, 250, 500, 800, 1000 kbps
Node-ID	1..127
PDO mapping, 1st object	9130h (AI input PV, integer32) 6130h (AI input PV, float)

Table 3 - Selectable parameters during ordering phase

The values of the above listed parameters can also be modified through the SDO services.

3. LSS SERVICES

LSS services and protocols are used to inquire or to change the settings of three parameters of the CANopen device:

- Node-ID of the CANopen device
- Bit timing parameters of the physical layer (bit rate)
- LSS address compliant to the identity object (1018h)

3.1 LSS SWITCH STATE SERVICES

LSS switch state global

By means of this service, the LSS master device switches all LSS slave devices in the network into LSS waiting state or LSS configuration state.

The LSS master sends this message to switch the LSS slave(s) into configuration state:

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	04h	01h	00h	00h	00h	00h	00h	00h

Figure 9 - LSS switch state global - configuration state - message

The LSS master sends this message to switch back the LSS slave(s) to waiting state:

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	04h	00h	00h	00h	00h	00h	00h	00h

Figure 10 - LSS switch state global - waiting state - message

LSS switch state selective

By means of this service, the LSS master device switches the LSS slave device, whose LSS address equals the LSS address specified by the messages, into LSS configuration state.

The transmitted LSS address shall be equal to the identity object (object 1018h) of the related LSS slave.

The LSS address for the GEFRAN KHC CANopen device is specified in the following table.

	Address Field	Value
LSS Address	Vendor-ID	00000093h
	Product code	4343484Bh*
	Revision Number	Actual KHC r.n.**
	Serial Number	Actual KHC s.n. (printed on the label)***

Figure 11 - KHC LSS Address

* If read as string data type, it equals to the signature “KHCC” (KHC with CANopen Output)

** Actual Revision number can vary. The user can inquire the Revision number with LSS Inquire Revision Number command (see LSS Inquire Services) , or through an SDO read command of the object (1018, 3).

*** Actual Serial number is device specific. It is printed on the label attached to the GEFRAN KHC transducer case, or it can be inquired with the LSS Inquire Serial Number command (see LSS Inquire Services), or through an SDO read command of the object (1018, 4). The value printed on the label must be intended as expressed in hexadecimal format..

The LSS master sends this message sequence to switch the GEFRAN KHC CANopen device into configuration state (the slave sends the response message):

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	40h	93h	00h	00h	00h	00h	00h	00h
7E5h	Rx	8	41h	52h	4Bh	35h	53h	00h	00h	00h
7E5h	Rx	8	42h	01h*	00h*	01h*	00h*	00h	00h	00h
7E5h	Rx	8	43h	34h**	12h**	01h**	15h**	00h	00h	00h
7E4h	Tx	8	44h	00h	00h	00h	00h	00h	00h	00h

Figure 12 - LSS switch state selective message sequence

* The Revision number used for this example is 00010001h
** The Serial number used for this example is: 15011234h

The Serial Number is assigned by GEFRAN to the KHC sensor in accordance with the following scheme.
SERIAL NUMBER : YY WW NNNN, where:
YY: year of production
WW: week of production
NNNN: progressive number inside the week, starting from 1

3.2 LSS CONFIGURATION SERVICES

LSS configure node-ID

By means of this service, the LSS master device configures the pending node-ID of the LSS slave device. The LSS slave device confirms the success or the failure of the service execution.

The allowed node-ID values are in the range 1..127 (01h..7Fh). The LSS master sends this message to configure the value of the node-ID (the slave sends the response message):

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	11h	Node ID	00h	00h	00h	00h	00h	00h
7E4h	Tx	8	11h	Error code	00h	00h	00h	00h	00h	00h

Figure 13 - LSS configure node-ID message

where Error code: 00h (Protocol successfully completed) or 01h (Node-ID out of range)

The pending node-ID becomes active only after the master sends a NMT reset communication command. The node-ID is not automatically saved to the non-volatile memory of the slave device. In order to save the persistent node-ID, refer to the LSS store configuration service.

When the pending node-ID becomes active, or when the node-ID is stored in non volatile memory, the following COB-IDs are automatically updated according to their default values:

- COB-ID EMCY (1014h)
- COB-ID SDO rx (1200h, sub 1)
- COB-ID SDO tx (1200h, sub 2)
- COB-ID TPDO (1800h, sub 1)

At the power on, the active node-ID equals the persistent node-ID.

LSS configure bit timing parameters

By means of this service, the LSS master device configures the pending bit rate of the LSS slave device. The LSS slave device confirms the success or the failure of the service execution.

The allowed bit rate values with the associated table index, are specified in the following table.

Table index	Bit rate (kbit/s)
0	1000
1	800
2	500
3	250
4	125
5	100
6	50
7	20

Table 4 - Table index for bit timing table

Note: it is not possible to set the bit rate to 20kbps when the Auto-operational mode is active and the Event-timer of the TPDO1 is set between 1 and 9.

The LSS master sends this message to configure the bit rate (the slave sends the response message):

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	13h	00h	Table index	00h	00h	00h	00h	00h
7E4h	Tx	8	13h	Error code	00h	00h	00h	00h	00h	00h

Figure 14 - LSS configure bit timing message

where Error code: 00h (Protocol successfully completed) or 01h (Bit timing not supported).

The pending bit rate becomes active only after the master sends the LSS activate bit timing parameter service, or with the next power-on after the execution of the LSS store configuration service.

The bit rate is not automatically saved to the non-volatile memory of the slave device. In order to save the persistent bit rate, refer to the LSS store configuration service.

At the power on, the active bit rate equals the persistent bit rate.

LSS activate bit timing parameters

By means of this service, the LSS master activates simultaneously the bit rate at the LSS communication interface of all CANopen devices in the network.

Therefore the reception of this command triggers at the LSS slave the copying process of the currently pending bit rate to the active bit rate.

The LSS master sends this message to activate the bit timing parameters:

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	15h	Switch delay		00h	00h	00h	00h	00h

Figure 15 - LSS activate bit timing parameters message

where Switch delay is the time, in ms, multiplied by 2 when the new bit timing settings becomes active (Intel format byte ordering)

The Switch delay parameter specifies the length of two delay periods of equal length, which are necessary to avoid operating the network with different bit rates.

After “Switch delay” has elapsed the first time after service indication, the slave device stops communicating on the bus.

After “Switch delay” has elapsed one more time, the slave device resume the communication on the bus using the new active bit rate.

LSS store configuration

By means of this service, the LSS master device requests the LSS slave device to store the configured local layer settings (node-ID and bit rate) to non-volatile memory. On execution of this command the pending node-ID and bit rate are copied to the persistent node-ID and bit rate.

The LSS master sends this message to store the LSS configuration (the slave sends the response message):

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	17h	00h	00h	00h	00h	00h	00h	00h
7E4h	Tx	8	17h	Error code	00h	00h	00h	00h	00h	00h

Figure 16 - LSS store configuration message

where Error code: 00h (Protocol successfully completed) or 02h (Storage media access error).

3.3 LSS INQUIRY SERVICES

LSS inquire node-ID

By means of this service, the LSS master device inquires the active node-ID of the LSS slave device that is in LSS configuration state. The LSS slave device responds indicating his active node-ID.

The LSS master sends this message to inquire the node-ID (the slave sends the response message):

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	5Eh	00h	00h	00h	00h	00h	00h	00h
7E4h	Tx	8	5Eh	Node ID	00h	00h	00h	00h	00h	00h

Figure 17 - LSS inquire node-ID message

where Node-ID is the LSS slave’s active node-ID.

LSS inquire LSS address

By means of this service, the LSS master device inquires the LSS address of the LSS slave device. The LSS slave device responds indicating his LSS address.

The LSS master sends this message to inquire the Vendor-ID (the slave sends the response message):

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	5Ah	00h	00h	00h	00h	00h	00h	00h
7E4h	Tx	8	5Ah	Vendor ID				00h	00h	00h

Figure 18 - LSS inquire identity Vendor-ID message

where Vendor-ID is the LSS slave’s identity Vendor-ID (Intel format byte ordering).

The LSS master sends this message to inquire the Product-code (the slave sends the response message):

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	5Bh	00h	00h	00h	00h	00h	00h	00h
7E4h	Tx	8	5Bh	Product code				00h	00h	00h

Figure 19 - LSS inquire identity Product-code message

where Product-code is the LSS slave's identity Product-code (Intel format byte ordering).

The LSS master sends this message to inquire the Revision number (the slave sends the response message):

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	5Ch	00h	00h	00h	00h	00h	00h	00h
7E4h	Tx	8	5Ch	Revision number				00h	00h	00h

Figure 20 - LSS inquire identity Revision number message

where Revision number is the LSS slave's identity Revision number (Intel format byte ordering).

The LSS master sends this message to inquire the Serial number (the slave sends the response message):

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	5Dh	00h	00h	00h	00h	00h	00h	00h
7E4h	Tx	8	5Dh	Serial number				00h	00h	00h

Figure 21 - LSS inquire identity Serial number message

where Serial number is the LSS slave's identity Serial number (Intel format byte ordering).

3.4 LSS IDENTIFICATION SERVICES

LSS identify remote slave

By means of this service, the LSS master device requests all LSS slave devices to identify themselves by means of the 'LSS identify slave' service, whose LSS address meets the LSS_Address_sel.

The LSS_Address_sel consists of a single vendor-ID and a single product code and a span of revision and serial numbers determined by a low and high number.

The protocol defined in the following figure implements the LSS identify remote slave service. All LSS slave devices with matching vendor-ID and product-code and whose major revision-number and serial-numbers are located within the given ranges, identify themselves with the LSS identify slave service.

The boundaries are included in the interval.

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	46h	Vendor-ID				00h	00h	00h
7E5h	Rx	8	47h	Product-code				00h	00h	00h
7E5h	Rx	8	48h	Revision number low				00h	00h	00h
7E5h	Rx	8	49h	Revision number high				00h	00h	00h
7E5h	Rx	8	4Ah	Serial number low				00h	00h	00h
7E5h	Rx	8	4Bh	Serial number high				00h	00h	00h

Figure 22 - LSS identify remote slave message sequence

Where:

Vendor-ID is the LSS slave’s identity Vendor-ID (Intel format byte ordering).

Product-code is the LSS slave’s identity Product-code (Intel format byte ordering).

Revision number low and Revision number high identity the Revision number span (Intel format byte ordering).

Serial number low and Serial number high identity the Serial number span (Intel format byte ordering).

LSS identify slave

By means of this service, an LSS slave device indicates that it is a slave device with an LSS address within the LSS_Address_sel given by an LSS identify remote slave service executed prior to this service.

The protocol is defined in the following figure.

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E4h	Tx	8	4Fh	00h	00h	00h	00h	00h	00h	00h

Figure 23 - LSS identify slave message

LSS identify non-configured remote slave

By means of this service, the LSS master device requests all LSS slave devices to identify themselves by means of the 'LSS identify non-configured slave' service, who got stuck in NMT Initialization state, whose pending node-ID is invalid (FFh) and who have no active node-ID.

The protocol is defined in the following figure.

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E5h	Rx	8	4Ch	00h	00h	00h	00h	00h	00h	00h

Figure 24 - LSS identify non-configured slave message

LSS identify non-configured slave

By means of this service, an LSS slave device indicates that it is an LSS slave device that got stuck in NMT Initialization state, owns an invalid (FFh) pending node-ID and no active node-ID.

This service is executed in case a LSS identify non-configured remote slave service was initiated by the LSS master device.

The protocol is defined in the following figure.

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
7E4h	Tx	8	50h	00h	00h	00h	00h	00h	00h	00h

Figure 25 - LSS identify non-configured slave message

4. SDO SERVICES

SDO services provide direct access to the object entries of a CANopen device's object dictionary. The device initiating the SDO transfer is called the SDO client.

The CANopen device hosting the accessed object dictionary is called the SDO server.

SDO download

The SDO client uses this service for transferring data to the object dictionary of the SDO server. SDO download service is therefore used to configure (write) communication, device and manufacturer parameters of the GEFRAN KHC CANopen device.

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
600h + node-ID	Rx	8	Cs	Index		Sub index	Data			
580h + node-ID	Tx	8	60h	Index		Sub index	00h	00h	00h	00h

Figure 26 - SDO download message

where:

Cs is the command specifier of the SDO download request, whose value depends on the number of bytes of Data field:

Cs=23h 4 transmitted data bytes
Cs=27h 3 transmitted data bytes
Cs=2Bh 2 transmitted data bytes
Cs=2Fh 1 transmitted data bytes

Data is the data to be copied in the object dictionary value (Intel format byte ordering)
Index is the object dictionary parameter index (Intel format byte ordering)
Sub index is the object dictionary parameter sub index

SDO upload

The SDO client uses this service for transferring the data from the server (owner of the object dictionary) to the client. SDO upload service is therefore used to check (read) communication, device and manufacturer parameters of the GEFRAN KHC CANopen device.

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
600h + node-ID	Rx	8	40h	Index		Sub index	00h	00h	00h	00h
580h + node-ID	Tx	8	42h	Index		Sub index	Data			

Figure 27 - SDO upload message

where:

Index is the object dictionary parameter index (Intel format byte ordering)
Sub index is the object dictionary parameter sub index
Data is the data value read from object dictionary (Intel format byte ordering)

SDO abort transfer

The SDO abort transfer service aborts the SDO download or the SDO upload service of an SDO referenced by its number.

As result of an SDO abort transfer event, the SDO server sends this message to the SDO client:

COB-ID	Rx/Tx	DLC	Data							
			D0	**D1**	**D2**	**D3**	**D4**	**D5**	**D6**	**D7**
580h + node-ID	Tx	8	80h	Index		Sub index	Abort code			

Figure 28 - SDO abort response message

where:

Index is the object dictionary parameter index (Intel format byte ordering)

Sub index is the object dictionary parameter sub index

Abort code explain the reason of the SDO abort event.

The following table contains the abort codes provided by the protocol SDO abort transfer of the GEFRAN KHC CANopen device.

Abort code	Description
05040001h	Client/server command specifier not valid or unknown
05040005h	Out of memory
06010001h	Attempt to read a write only object
06010002h	Attempt to write a read only object
06020000h	Object does not exist in the object dictionary
06040041h	Object cannot be mapped to the PDO
06040042h	The number and length of the objects to be mapped would exceed PDO length.
06070010h	Data type does not match, length of service parameter does not match
06090011h	Sub-index does not exist
06090030h	Invalid value for parameter (download only)
08000021h	Data cannot be transferred or stored to the application because of local control.
08000022h	Data cannot be transferred or stored to the application because of the present device state.

Figure 29 - SDO abort codes

4.1 OBJECT DICTIONARY

The object dictionary of the GEFRAN KHC CANopen device is specified in the following tables.

Communication Profile Area

Index	Sub index	Name	Type	Access	Default value	Comment
1000h	0	Device type	Unsigned32	RO	80020194h	Analogue input with device-specific PDO mapping and ds404 device profile
1001h	0	Error register	Unsigned8	RO	-	0: no error 1: generic error
1005h	0	COB-ID SYNC	Unsigned32	RW	00000080h	Configured COB-ID of the synchro-nization object (SYNC)
1008h	0	Manufacturer device name	Visible string	RO	KHC	Name of the device
1009h	0	Manufacturer HW version	Visible string	RO	-	Hardware version description
100Ah	0	Manufacturer SW version	Visible string	RO	-	Software version description
100Ch	0	Guard time	Unsigned16	RW	0	Multiplied with object 100Dh gives the lifetime value used by the node guarding protocol
100Dh	0	Life time factor	Unsigned8	RW	0	Multiplied with object 100Ch gives the lifetime value used by the node guarding protocol
1010h	0	Store parameters	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned32	RW	00000001h	Writing the signature “save” (73h, 61h, 76h, 65h) stores all parame-ters in flash memory
1011h	0	Restore default parameters	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned32	RW	00000001h	Writing the signature “load” (6Ch, 6Fh, 61h, 64h) restores all parame-ters in flash to their default values
1014h	0	COB-ID EMCY	Unsigned32	RW	00000080h + Node-ID	Configured COB-ID for the EMCY write service
1015h	0	Inhibit time EMCY	Unsigned16	RW	0000h	Configured inhibit time for the EMCY service
1017h	0	Producer heartbeat time	Unsigned16	RW	0	Configured cycle time of the heart-beat (ms)
1018h	0	Identity object	Unsigned8	RO	4	Highest sub-index supported
	1		Unsigned32	RO	00000093h	Vendor-ID
	2		Unsigned32	RO	4343484Bh	Product code
	3		Unsigned32	RO	-	Revision number
	4		Unsigned32	RO	-	Serial number
1200h	0	SDO1 server parameter	Unsigned8	RO	2	Highest sub-index supported
	1		Unsigned32	RO	00000600h + Node-ID	COB-ID client --> server (rx)
	2		Unsigned32	RO	00000580h + Node-ID	COB-ID server --> client (tx)
1800h	0	TPDO1 communication parameter	Unsigned8	RO	5	Highest sub-index supported
	1		Unsigned32	RW	00000180h + Node-ID	COB-ID del TPDO1
	2		Unsigned8	RW	FFh	Transmission type
	5		Unsigned16	RW	1	Event-timer

Index	Sub index	Name	Type	Access	Default value	Comment
1A00h	0	TPDO1 mapping parameter	Unsigned8	RW	2	Number of mapped application objects in TPDO1
	1		Unsigned32	RW	91300120h	1st application object (pressure)
	2		Unsigned32	RW	61500108h	2nd application object (status)
	3		Unsigned32	RW	20910010h	3rd application object (temperature)

Manufacturer Profile Area

Index	Sub index	Name	Type	Access	Default value	Comment
2010h	0	Minimum nominal pressure	Unsigned16	RO	-	Minimum nominal pressure value
2011h	0	Maximum nominal pressure	0	RO	-	Maximum nominal pressure value
2020h	0	Minimum value storage	Real32	RO	-	Minimum measured pressure value (volatile)
2021h	0	Maximum value storage	Real32	RO	-	Maximum measured pressure value (volatile)
2090h	0	Process value as integer	Integer32	RO	-	AI input PV as 32 bit integer data format. Identical to 9130h
2091h	0	Temperature	Integer16	RO	-	Actual working temperature of the electronic given in 0.5°C
2100h	0	User device name	Unsigned32	RW	FFFFFFFFh	User defined name for the device
2201h	0	Last calibration date year	Unsigned8	RW	-	Year of the last calibration (last two digits)
2202h	0	Last calibration date month	Unsigned8	RW	-	Month of the last calibration
2203h	0	Last calibration date day	Unsigned8	RW	-	Day of the last calibration
2207h	0	Date of production year	Unsigned8	RO	-	Year of production (last two digits)
2208h	0	Date of production month	Unsigned8	RO	-	Month of production
2209h	0	Date of production day	Unsigned8	RO	-	Day of production
2320h	0	Persistent Node-ID	Unsigned8	RW	01h	Node-ID stored in non-volatile memory of the device
2321h	0	Persistent bit timing table index	Unsigned8	RW	3	Bit rate stored in non-volatile memory
2322h	0	Node-ID and baud rate SDO write disable	Unsigned8	RW	0	Disables Node-ID and baud rate change by SDO. 00h: write enabled 01h: write disabled
2330h	0	Auto-operational mode	Unsigned8	RW	0	00h: Disabled 01h: After boot-up the device enters the NMT Operational state automatically
2340h	0	EMCY pressure exceeded reset hysteresis	Real32	RW	5	Sensitivity referred to the exceeded pressure for the EMCY error reset condition

Device Profile Area

Index	Sub index	Name	Type	Access	Default value	Comment
6110h	0	AI sensor type	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned16	RO	90	AI sensor type 1
6114h	0	AI ADC sample rate	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned32	RW	1000	AI ADC sample rate 1
6121h	0	AI input scaling 1 PV (float)	Unsigned8	RO	1	Highest sub-index supported
	1		Real32	RW	-	AI input scaling 1 PV 1 (float)
6123h	0	AI input scaling 2 PV (float)	Unsigned8	RO	1	Highest sub-index supported
	1		Real32	RW	-	AI input scaling 2 PV 1 (float)
6124h	0	AI input offset (float)	Unsigned8	RO	1	Highest sub-index supported
	1		Real32	RW	-	AI input offset 1 (float)
6125h	0	AI autozero	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned32	WO	-	AI autozero 1
6130h	0	AI input PV (float)	Unsigned8	RO	1	Highest sub-index supported
	1		Real32	RO	-	AI input PV 1 (float)
6131h	0	AI physical unit PV	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned32	RW	004E0000h	AI physical unit PV 1
6132h	0	AI decimal digits PV	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned8	RW	2	AI decimal digits PV 1
6148h	0	AI span start (float)	Unsigned8	RO	1	Highest sub-index supported
	1		Real32	RW	-	AI span start 1 (float)
6149h	0	AI span end (float)	Unsigned8	RO	1	Highest sub-index supported
	1		Real32	RW	-	AI span end 1 (float)
6150h	0	AI status	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned8	RO	-	AI status 1
61A0h	0	AI filter type	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned8	RW	0	AI filter type 1
61A1h	0	AI filter constant	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned8	RW	1	AI filter constant 1
7100h	0	AI input FV (integer16)	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned16	RO	-	AI input FV 1(integer16)
7120h	0	AI input scaling 1 FV (integer16)	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned16	RO	-	AI input scaling 1 FV 1 (integer16)
7122h	0	AI input scaling 2 FV (integer16)	Unsigned8	RO	1	Highest sub-index supported
	1		Unsigned16	RO	-	AI input scaling 2 FV 1 (integer16)
9121h	0	AI input scaling 1 PV (integer32)	Unsigned8	RO	1	Highest sub-index supported
	1		Integer32	RW	-	AI input scaling 1 PV 1 (integer32)
9123h	0	AI input scaling 2 PV (integer32)	Unsigned8	RO	1	Highest sub-index supported
	1		Integer32	RW	-	AI input scaling 2 PV 1 (integer32)
9124h	0	AI input offset (integer32)	Unsigned8	RO	1	Highest sub-index supported
	1		Integer32	RW	-	AI input offset 1 (integer32)
9130h	0	AI input PV (integer32)	Unsigned8	RO	1	Highest sub-index supported
	1		Integer32	RO	-	AI input PV 1 (integer32)
9148h	0	AI span start (integer32)	Unsigned8	RO	1	Highest sub-index supported Sottoindice massimo supportato
	1		Integer32	RO	-	AI span start 1 (integer32)
9149h	0	AI span end (integer32)	Unsigned8	RO	1	Highest sub-index supported
	1		Integer32	RO	-	AI span end 1 (integer32)

4.2 SDO OBJECTS

1000h – Device type

This object describes the type of the device and its functionality. It is composed of a 16-bit field that describes the device profile or the application profile that is used and a second 16-bit field, which gives additional information about optional functionality of the device.

The structure of the device parameter is represented in the following figure.

31 16	**15 0**
Additional Information	Device Profile Number

Figure 30 - Structure of the Device type parameter

Additional information = 8002h
Device Profile Number = 0194h

Object description

Index	Name
1000h	Device type

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Device type	RO	Unsigned32	80020194h	80020194h

1001h – Error register

This object provides error information. The CANopen device maps internal errors into this object. It is a part of an emergency object.

For the GEFRAN KHC CANopen device the Generic error indication is given when one or more errors are generated.

The error register contains one of the error codes described in the following table.

Error code	Description
0	No error
1	Generic error

Table 5 - Error codes in the Error register

Object description

Index	Name
1001h	Error register

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Error register	RO	Unsigned8	0,1	-

1005h – COB-ID SYNC

This object indicates the configured COB-ID of the synchronization object (SYNC). It also defines whether the CANopen device generates the SYNC.

The structure of this object is specified in the following figure.

31	30	29	28 — 11	10 — 0
x	gen.	frame	Reserved (0 0000h)	11-bit CAN-ID

Figure 31 - Structure of SYNC COB-ID

The value definition is given in the following table.

Field name	Value	Description
x	0	Do not care
gen	0	Device does not generate SYNC message
frame	0	11-bit CAN-ID valid (CAN base frame)
11 bit CAN-ID	80h	11-bit CAN-ID of the CAN base frame

Table 6 - COB-ID SYNC message field

Object description

Index	Name
1005h	COB-ID SYNC

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	COB-ID SYNC	RW	Unsigned32	Unsigned32 (*)	00000080h

*) The 11-bit CAN-ID of the COB-ID must be compliant to the restricted CAN-ID definitions (see Restricted CAN-ID section). A restricted CAN-ID cannot be used.

1008h – Manufacturer device name

This object provides the name of the device as given by the manufacturer.

Object description

Index	Name
1008h	Manufacturer device name

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Manufacturer device name	RO	Visible_string	KHC	KHC

1009h – Manufacturer hardware version

This object provides the manufacturer hardware version description

Object description

Index	Name
1009h	Manufacturer hardware version

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Manufacturer hardware version	RO	Visible_string	Visible_string	-

100Ah – Manufacturer software version

This object provides the manufacturer software version description..

Object description

Index	Name
100Ah	Manufacturer software version

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Manufacturer software version	RO	Visible_string	Visible_string	-

100Ch – Guard time

This object indicates the configured guard time. The guard time multiplied with the lifetime factor gives the lifetime for the life guarding protocol.

The value of 0 disables the lifeguarding, all other values up to 65535 are valid.

Object description

Index	Name
100Ch	Guard time

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Guard time	RW	Unsigned16	0..65535	0

100Dh – Life time factor

The lifetime factor multiplied with the guard time gives the lifetime for the life guarding protocol. The value of 0 disables the life guarding, all other values, up to 255, are valid.

Object description

Index	Name
100Dh	Life time factor

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Life time factor	RW	Unsigned8	0..255	0

1010h – Store parameters

This object controls the saving of parameters in non-volatile memory.

In order to avoid storage of parameters by mistake, storage is only executed when the signature "save" is written to the sub-index 1, so that all parameters are saved in non-volatile memory.

The storage write access structure is specified in the following figure

31			**0**
e (65h)	v (76h)	a (61h)	s (73h)
MSB			LSB

Figure 32 - Storage write access structure

By read access the sub-index 1 of this object, the device provides information about its saving capabilities. Giving the value of 1, it means that the device saves parameters on command.

Object description

Index	Name
1010h	Store parameters

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	Save all parameters	RW	Unsigned32	Read access: 00000001h Write access: 65766173h (ASCII: "save")	Read access: 00000001h Write access: 65766173h (ASCII: "save")

1011h – Restore default parameters

This object controls the restore of parameters in non-volatile memory to their default values, according to the communication and device profile.

In order to avoid restoring of parameters by mistake, restoring is only executed when the signature "load" is written to the sub-index 1, so that all parameters are restored in non-volatile memory.

The restore default parameters write access structure is specified in the following figure.

31			**0**
d (64h)	a (61h)	o (6Fh)	l (6Ch)
MSB			LSB

Figure 33 - Restore write access structure

By read access the sub-index 1 of this object, the device provides information about its restoring capabilities. Giving the value of 1, it means that the device can restore parameters on command.

The default values are set valid after the device is power cycled.

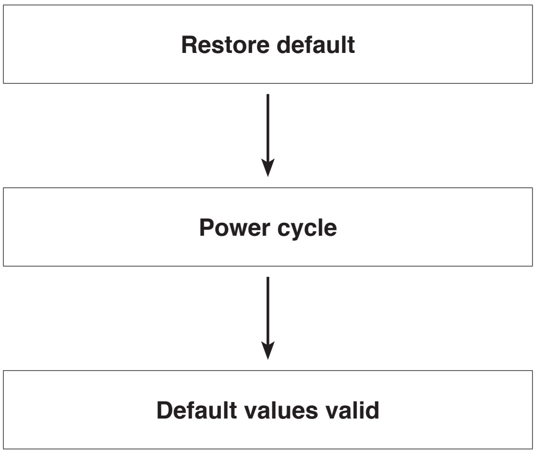

Figure 34 - Restore procedure

For the GEFRAN KHC CANopen device, the Restore default parameters command does not apply to these objects:

- COB-ID EMCY (1014h)
- COB-ID of TPDO1 (1800h, sub-index 1)
- COB-IDs of 1st SDO (1200h, sub-index 1 and 2)

The value of the above listed objects is modified only after a change of the Node-ID value.

Object description

Index	Name
1011h	Restore default parameters

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	Restore all default parameters	RW	Unsigned32	Read access: 00000001h Write access: 64616F6Ch (ASCII: “load”)	Read access: 00000001h Write access: 64616F6Ch (ASCII: “load”)

1014h – COB-ID EMCY

This object indicates the configured COB-ID of the EMCY write service.
The structure of this object is specified in the following figure.

31	30	29	28 11	10 0
valid	res.	frame	Reserved (0 0000h)	11-bit CAN-ID

Figure 35 - Structure of EMCY COB-ID

The value definition is given in the following table.

Field name	Value	Description
valid	0	EMCY exists / is valid
reserved	0	Reserved (always 0)
frame	0	11-bit CAN-ID valid (CAN base frame)
11 bit CAN-ID	80h + Node-ID (default) or user defined	11-bit CAN-ID of the CAN base frame

Table 7 - COB-ID EMCY message fields

Object description

Index	Name
1014h	COB-ID EMCY

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	COB-ID EMCY	RO	Unsigned32	00000080h + Node-ID	00000080h + Node-ID

1015h – Inhibit time EMCY

This object indicates the configured inhibit time of the EMCY write service.
The inhibit time EMCY defines the minimum time that elapses between two consecutive invocations of the EMCY service.
The value is given in multiples of 100us. The accepted values must be multiples of 10, i.e. 1ms. The value 0 disables the inhibit time.

Object description

Index	Name
1015h	Inhibit time EMCY

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Inhibit time EMCY	RW	Unsigned16	0..65535 as multiples of 10	0

NOTE
When using low baudrate values, setting the Inhibit time EMCY to the proper value can avoid possible bus overloads due to the high frequency rate of transmission of the EMCY messages under certain circumstances.

1017h – Producer heartbeat time

The producer heartbeat time indicates the configured cycle time of the heartbeat, given in 1 ms. The value 0 disables the producer heartbeat.

Object description

Index	Name
1017h	Producer heartbeat time

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Producer heartbeat time	RW	Unsigned16	0..65535	0000h

1018h – Producer heartbeat time

This object provides general identification information of the device.

- Sub-index 1: contains the unique value that is allocated uniquely to each vendor of a CANopen device. For GEFRAN s.p.a. manufacturer it is 00000093h.
- Sub-index 2: contain the unique value that identifies a specific type of CANopen device. For the GEFRAN KHC CANopen device it is 2043484Bh.
- Sub-index 3: contains the major revision number and the minor revision number of the revision of the device. Its value is device specific.
- Sub-index 4: contains the serial number that identifies uniquely the device. Its value is device specific.

Object description

Index	Name
1018h	Identity object

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	4	4
1	Vendor-ID	RO	Unsigned32	00000093h	00000093h
2	Product code	RO	Unsigned32	4343484Bh	4343484Bh
3	Revision number	RO	Unsigned32	-	-
4	Serial number	RO	Unsigned32	-	-

The user can also get the identity object values using the LSS inquire identity services (see LSS protocol description section).
The Product code, when read as string data type, equals to the “KHCC” signature (KHC with CANopen Output).
The Revision number can vary depending on HW/FW updates. The Serial Number is unique for each device. The Serial number is also printed on the label attached to the case of the device..

1200h – SDO1 server parameter

This object describes the first SDO used on the device.

The values at sub-index 1 and sub-index 2 specify the COB-IDs for the first SDO. The object structure is specified in the following figure.

31	30	29	28 … 11	10 … 0
valid	dyn	frame	Reserved (0 0000h)	11-bit CAN-ID

Figure 36 - Structure of SDO1 COB-ID

The value definition is given in the following table.

Field name	Value	Description
valid	0	SDO exists / is valid
dyn	0	Value is assigned statically

Field name	Value	Description
reserved	0	11-bit CAN-ID valid (CAN base frame)
11 bit CAN-ID	00000600h + Node-ID (default rx) or 00000580h + Node-ID (default tx)	11-bit CAN-ID of the CAN base frame

Table 8 - SDO1 COB-ID fields

Object description

Index	Name
1200h	SDO1 server parameter

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	2	2
1	COB-ID client --> server (rx)	RO	Unsigned32	Unsigned32 (*)	00000600h + Node-ID
2	COB-ID server --> client (tx)	RO	Unsigned32	Unsigned32 (*)	00000580h + Node-ID

(*) The 11-bit CAN-ID of the COB-ID must be compliant to the restricted CAN-ID definitions (see Restricted CAN-ID section). A restricted CAN-ID cannot be used.

1800h – TPDO1 communication parameter

This object contains the communication parameters for the PDOs the CANopen device is able to transmit.

Sub-index 1 contains the COB-ID of the TPDO1.
The object structure is specified in the following figure.

31	30	29	28 11	10 0
valid	RTR	frame	Reserved (0 0000h)	11-bit CAN-ID

Figure 37 - Structure of TPDO1 COB-ID

The value definition is given in the following table.

Field name	Value	Description
valid	0	PDO exists / is valid
	1	PDO does not exists / is not valid
RTR	0	RTR is processed on this PDO
frame	0	11-bit CAN-ID valid (CAN base frame)
11 bit CAN-ID	00000180h + Node-ID (default) or user defined	11-bit CAN-ID of the CAN base frame

Table 9 - TPDO1 COB-ID fields

The user can change the default TPDO1 COB-ID value in the range of the allowed values, ensuring that no conflicts with other COB-IDs are generated.

The value is also automatically changed in accordance with the “default scheme” when changing the Node-ID value.

Sub-index 2 defines the transmission type of the TPDO.

Three types of PDO transmission are defined:

1. Synchronous: means that the PDO is transmitted after the SYNC
2. RTR-only: means that the PDO is not transmitted normally it shall be requested via RTR
3. Event-driven: means that the PDO may be transmitted at any time based on the occurrence of a CANopen device internal event

Transmission type settings are explained in the following table.

Value	Description
0	Synchronous (acyclic)
1	Synchronous (cyclic every 1 SYNC)
2	Synchronous (cyclic every 2 SYNC)
3	Synchronous (cyclic every 3 SYNC)
... ...	... ...
240	Synchronous (cyclic every 240 SYNC)
241	RESERVED
... ...	RESERVED
251	RESERVED
252	RTR-only
253	RTR-only
254	Event-driven (asynchronous)
255	Event-driven (asynchronous)

Table 10 - TPDO1 transmission type description

Sub-index 5 contains the event-timer. The time is the maximum interval for PDO transmission if the transmission type is set to FEh and FFh.

Its value is given in multiples of 1 ms. The value of 0 disables the event-timer (no PDO is transmitted).

Object description

Index	Name
1800h	TPDO1 communication parameter

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	5	5
1	COB-ID used by TPDO1	RW	Unsigned32	Unsigned32 (*)	00000180h + Node-ID
2	Transmission type	RW	Unsigned8	0..240 and 252..255	254
5	Event-timer	RW	Unsigned16	0..65535	1

(*) The 11-bit CAN-ID of the COB-ID must be compliant to the restricted CAN-ID definitions (see Restricted CAN-ID section). A restricted CAN-ID cannot be used.

Note:

It is not possible to set the Event-timer between 1 and 9 when the Auto-operational mode is active and the bit rate is 20kbps

1A00h – TPDO1 mapping parameter

This object contains the mapping for the PDOs the device is able to transmit.

Sub-index 1 and sub-index 2 contain the information of the mapped application objects. The object describes the content of the PDO by their index, sub-index and length, as specified in the following figure.

31 16	15 8	7 0
Index	Sub-index	Length

Figure 38 - Structure of TPDO1 mapping

The value definition is given in the following table.

Field name	Description
Index	The content of the PDO described by the index
Sub-index	The content of the PDO described by the sub-index
Length	The length of the application object in bit

Table 11 - TPDO1 mapping fields

Object description

Index	Name
1A00h	TPDO1 mapping parameter

Entry description

Sub index	Name	Access	Data Type	Value Range	Default
0	Number of mapped application objects in TPDO1	RO	Unsigned8	0..3	2
1	1st application object (pressure)	RW	Unsigned32	20900020h, 20910010h, 61300120h, 61500108h, 91300120h	91300120h
2	2nd application object (status)	RW	Unsigned32	20900020h, 20910010h, 61300120h, 61500108h, 91300120h	61500108h
3	3rd application object (temperature)	RW	Unsigned32	20900020h, 20910010h, 61300120h, 61500108h, 91300120h	20910010h

2010h – Minimum nominal pressure

This object indicates the minimum nominal pressure. The value is given in 1 bar.

Object description

Index	Name
2010h	Minimum nominal pressure

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Minimum nominal pressure	RO	Unsigned16	-	-

2011h – Maximum nominal pressure

This object indicates the maximum nominal pressure. The value is given in 1 bar.

Object description

Index	Name
2011h	Maximum nominal pressure

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Maximum nominal pressure	RO	Unsigned16	-	-

2020h – Minimum value storage

This object indicates the minimum value of the AI input PV (object 6130h) registered since the power-on or reset of the device.

The storage is volatile.

Object description

Index	Name
2020h	Minimum value storage

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Minimum value storage	RW	Real32	Real32	-

A write access clears the registered value.

2021h – Maximum value storage

This object indicates the maximum value of the AI input PV (object 6130h) registered since the power-on or reset of the device.

The storage is volatile.

Object description

Index	Name
2021h	Maximum value storage

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Maximum value storage	RW	Real32	Real32	-

A write access clears the registered value.

2090h – Process value as integer

This object gives the value of the measured pressure as integer data type format. This object is the same as the object 9130h.

Object description

Index	Name
2090h	Process value as integer

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Process value as integer	RO	Integer32	Integer32	-

2091h – Temperature

This object gives the value of the actual working temperature of the electronic of the device. The value is given in 0.5°C unit.

Object description

Index	Name
2091h	Temperature

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Temperature	RO	Integer16	Integer16	-

6000h – Operating parameters

This object indicates the configuration of the operating parameters of the encoder.

Object description

Index	Name
6000h	Operating parameters

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Operating parameters	RO	Unsigned16	0000h	0000h

This object is not supported by the GEFRAN KHC CANopen device.

2100h – User device name

This object contains the value of the device name specified by the user.

Object description

Index	Name
2100h	User device name

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	User device name	RW	Unsigned32	Unsigned32	FFFFFFFFh

2201h – Last calibration date year

This object contains the year of the last calibration date.

Object description

Index	Name
2201h	Last calibration date year

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Last calibration date year	RW	Unsigned8	0..99	-

2202h – Last calibration date month

This object contains the month of the last calibration date.

Object description

Index	Name
2202h	Last calibration date month

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Last calibration date month	RW	Unsigned8	1..12	-

2203h – Last calibration date day

This object contains the day of the last calibration date.

Object description

Index	Name
2203h	Last calibration date day

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Last calibration date day	RW	Unsigned8	1..31	-

2207h – Date of production year

This object contains the year of the production date of the device

Object description

Index	Name
2207h	Date of production year

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Date of production year	RO	Unsigned8	0..99	-

2208h – Date of production month

This object contains the month of the production date of the device

Object description

Index	Name
2207h	Date of production month

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Date of production month	RO	Unsigned8	1..12	-

2209h – Date of production day

This object contains the day of the production date of the device

Object description

Index	Name
2209h	Date of production day

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Date of production day	RO	Unsigned8	1..31	-

2320h – Persistent Node-ID

This object contains the value of the actual persistent Node-ID stored in non-volatile memory. A write access stores the new Node-ID value in non-volatile memory. The “save parameters” command is not required for this object.

The following COB-IDs are also automatically updated according to their default values:

- COB-ID EMCY (1014h)
- COB-ID SDO rx (1200h, sub 1)
- COB-ID SDO tx (1200h, sub 2)
- COB-ID TPDO (1800h, sub 1)

The updated Node-ID and COBs values becomes active only after a power cycle of the device.

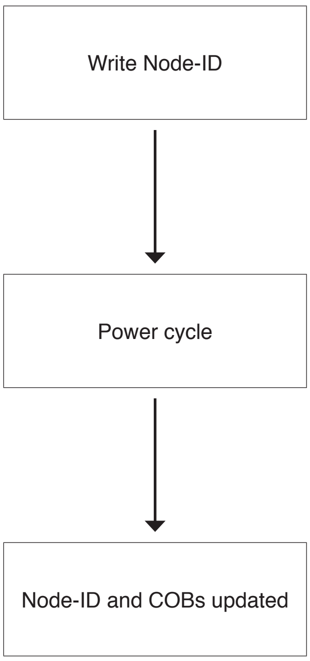

Figure 39 - Setting Node-ID by SDO write

Normally, the change of the Node-ID is made through LSS services (see LSS configure Node-ID).

For security reasons, it is possible to disable the possibility to change the Node-ID through SDO write, configuring the object 2322h.

Object description

Index	Name
2320h	Persistent Node-ID

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Persistent Node-ID	RW	Unsigned8	01h..7Fh	01h

2321h – Persistent bit timing table index

This object contains the value of the actual bit timing table index, which determines the baud rate settings, stored in non-volatile memory.

A write access stores the new bit timing table index value, i.e. the new baud rate settings, in non-volatile memory.

The “save parameters” command is not required for this object.

The allowed bit timing table indexes are specified in the following table.

Table index	Bit rate (kbit/s)
0	1000
1	800
2	500
3	250
4	125
5	100
6	50
7	20

Table 12 - Bit timing table indexes

The new baud rate settings become active only after a power cycle of the device.

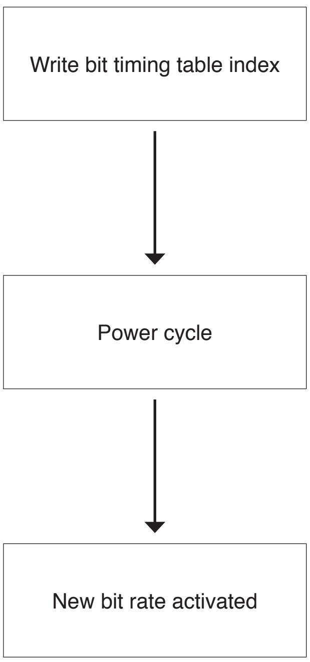

Figure 40 – Setting bit rate by SDO write

Normally, the change of the baud rate settings is made through LSS services (see LSS configure bit timing parameters). For security reasons, it is possible to disable the possibility to change the baud rate through SDO write, configuring the object 2322h.

Object description

Index	Name
2321h	Persistent bit timing table index

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Persistent bit timing table index	RW	Unsigned8	0..7	3

Note:
It is not possible to set baud rate to 20kbps when the Auto-operational mode is active and the Event-timer of the TPDO1 is set between 1 and 9.

2322h – Node-ID and baud rate SDO write disable

This object gives the possibility to disable, for security reasons, the functionality of changing the Node-ID and Baud rate value through SDO write operation.

If the value is set to 1, an SDO write access to objects 2320h (Persistent Node-ID) and 2321h (Persistent bit timing table index) is denied, resulting in SDO abort operation.

Anyway, the change of the Node-ID and baud rate settings is always possible through LSS services.

Object description

Index	Name
2322h	Node-ID and baud rate SDO write disable

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Node-ID and baud rate SDO write disable	RW	Unsigned8	0, 1	0

2330h – Auto-operational mode

This object gives the possibility to force the device to automatically enter the NMT Operational state after power-on. If the value is set to 1, the device automatically enters the Operational mode after power-on.

Object description

Index	Name
2330h	Auto-operational mode

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Auto-operational mode	RW	Unsigned8	0, 1	0

Note:
It is not possible to set the Auto-operational mode to 1 when the Event-timer of the TPDO1 is set between 1 and 9, and the baudrate is set to 20kbps.

2340h – EMCY pressure exceeded reset hysteresis

This object sets the sensitivity of the pressure exceeded reset condition of the Emergency message. The hysteresis is given as percentage of the full scale.

When an EMCY message with “Min. allowed pressure exceeded” has been send, the EMCY message with “Error Reset” error code is not send until the AI Process Value goes above the “AI Span Start + Hysteresis” value.

When an EMCY message with “Max. allowed pressure exceeded” has been send, the EMCY message with “Error Reset” error code is not send until the AI Process Value goes below the “AI Span End – Hysteresis” value.

With higher value of hysteresis, the emergency message can be send less frequently. With lower values it can be send more frequently. If the value is set equal to 0 the hysteresis is disabled.

The ideal value of hysteresis depends on the specific application, so it can be set by the user.

See also: - EMCY Services
- AI span start (6148h or 9148h
- AI span end (6149h or 9149h).

Object description

Index	Name
2340h	EMCY pressure exceeded reset hysteresis

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	EMCY pressure exceeded reset hysteresis	RW	Real32	0..10	5

6110h – AI Sensor Type

This object indicates the configured type of sensor, which is connected to the analog input. The value read indicates a pressure sensor.

Object description

Index	Name
6110h	AI sensor type

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
0	AI sensor type 1	RO	Unsigned16	90	90

6114h – AI ADC sample rate

This object indicates the configured conversion rate used by the A/D converter. The value is given in multiples of micro-seconds. Only values multiple of 1000 microseconds are valid.

Object description

Index	Name
6110h	AI ADC sample rate

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
0	AI ADC sample rate	RW	Unsigned32	1000..255000 as multiples of 1000	1000

6121h – AI input scaling 1 PV (float)

This object indicates the configured PV of the first calibration point for the analog input channel. It is scaled in physical unit of PV (see object 6131h). The data type is floating point number.

For more details about this object usage see also the section "DS 404 specific functionalities".

Object description

Index	Name
6121h	AI input scaling 1 PV (float)

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI input scaling 1 PV (float)	RW	Real32	Real32	-

6123h – AI input scaling 2 PV (float)

This object indicates the configured PV of the second calibration point for the analog input channel. It is scaled in physical unit of PV (see object 6131h). The data type is floating point number.

For more details about this object usage see also the section "DS 404 specific functionalities"..

Object description

Index	Name
6123h	AI input scaling 2 PV (float)

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI input scaling 2 PV (float)	RW	Real32	Real32	-

6124h – AI input offset (float)

This object indicates the configured additional offset value for the analog input channel. It is scaled in physical unit of PV (see object 6131h). The data type is floating point number.

For more details about this object usage see also the section “DS 404 specific functionalities”.

Object description

Index	Name
6124h	AI input offset (float)

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI input offset 1 (float)	RW	Real32	Real32	-

6125h – AI autozero

Writing a signature value of “zero“ to this object causes a modification of the AI input offset (objects 6124h and 9124h) in such a way that the actual AI input PV becomes zero.

The autozero write access structure is specified in the following figure.

31 **0**

o (6Fh)	r (72h)	e (65h)	z (7Ah)

MSB LSB

Figure 41 – AI autozero write access structure

For more details about this object usage see also the section “DS 404 specific functionalities”.

Object description

Index	Name
6125h	AI autozero

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI autozero 1	WO	Unsigned32	Write access: 6F72657Ah (ASCII: “zero”)	-

6130h – AI input PV (float)

This object provides the result of the input scaling block and gives the measured quantity scaled in the used physical unit of the process value (PV) set by AI physical unit PV (see object 6131h).

The data type is floating point number.

Object description

Index	Name
6130h	AI input PV (float)

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI input PV 1 (float)	RW	Real32	Real32	-

6131h – AI physical unit PV

This object indicates the configured SI units and prefixes for the process value within the analog input FB.

The physical units supported by the GEFRAN KHC CANopen device are listed in the following table.

Value	Physical unit
004E0000h	bar
00AB0000h	psi
00220000h	pascal
00A10000h	at
00A20000h	mmH20
00A30000h	mHg
00A40000h	atm

Table 13 - Physical units supported for the process value

Note:
After changing the AI physical unit PV the value of the AI decimal digits PV (object 6132h) is automatically set to the default value.

Object description

Index	Name
6131h	AI physical unit PV

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI physical unit PV 1	RW	Unsigned32	(see table)	004E0000h

6132h – AI decimal digits PV

This object indicates the configured number of decimal digits following the decimal point for interpretation of data types INTEGER8, INTEGER16 and INTEGER32.

The objects whose value is affected by the AI decimal digits PV are the following:

2090h: Process value as integer
9121h: AI input scaling 1 PV (integer32)
9123h: AI input scaling 2 PV (integer32)
9124h: AI input offset (integer32)
9130h: AI input PV (integer32)
9148h: AI span start (integer32)
9149h: AI span end (integer32)

Example: A FV of 1.23 (REAL32) is coded in INTEGER32 format as:

- 1 if number of decimal digits is set to 0
-12 if number of decimal digits is set to 1
-123 if number of decimal digits is set to 2
-1230 if number of decimal digits is set to 3

In order to avoid overflow conditions, the maximum value of decimal digits that can be set depends on the actual physical unit set for the PV (see object 6131).

The allowed range of decimal digits values for a specific physical unit, and the default value, is listed in the following table.

Physical unit	Decimal digits range	Decimal digits default value
bar	0..5	2
psi	0..3	1
pascal	0	0
at	0..4	2
mmH20	0	0
mHg	0..6	2
atm	0..6	2

Table 14 - Decimal digits range and default values

Note:
When changing the physical unit, the value of the AI decimal digits PV (object 6131h) is automatically set to the default value.

Object description

Index	Name
6132h	AI decimal digits PV

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI decimal digits PV 1	RW	Unsigned8	(see table)	2

6148h – AI span start (float)

This object indicates the configured lower limit of the expected Process Value. When the PV is lower than this limit, it is marked as negative overloaded (see AI status, object 6150h). It is scaled in physical unit of PV (see object 6131h).

The data type is floating point number.

Object description

Index	Name
6148h	AI span start (float)

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI span start 1 (float)	RW	Real32	Real32	-

Note:
The value is set at the 5%FS below the minimum nominal pressure value by default.
The user can define a specific value. The value is refused if it is below the 10%FS of the minimum nominal pressure value.
The value can not be higher than the AI span end value (see object 6149h).
This object influences the EMCY service.

6149h – AI span end (float)

This object indicates the configured upper limit of the expected Process Value. When the PV exceeds this limit, it is marked as positive overloaded (see AI status, object 6150h).

It is scaled in physical unit of PV (see object 6131h). The data type is floating point number..

Object description

Index	Name
6149h	AI span end (float)

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI span end 1 (float)	RW	Real32	Real32	-

Note:
The value is set to the maximum nominal pressure value by default.
The user can define a specific value. The value is refused if it is above the 10%FS of the maximum nominal pressure value.
The value can not be lower than the AI span start value (see object 6148h).
This object influences the EMCY service.

6150h – AI status

This object provides the status of the analog input channel as defined in the following figure:

7 ... 3	2	1	0
Reserved	Negative overload	Positive overload	Not valid

MSB ... LSB

Examples:

Value	Description
00h	Measure is valid, normal working condition
01h	Measure is not valid
02h	AI span end exceeded, measure still valid
03h	Pressure over the 10%FS of the maximum nominal pressure value, measure is not valid
04h	Measured value below AI span start, measure still valid
05h	Pressure below the minimum nominal pressure value for more than 10%FS, measure is not valid

Object description

Index	Name
6150h	AI status

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI status	RO	Unsigned8	-	-

61A0h – AI filter type

This object indicates the type of filter to be used for calculation.
The filter types used by GEFRAN KHC CANopen device are specified in the following table.

Value	Description
0	No filter (measure unfiltered)
1	Moving average
2	Repeating average
100	Average of the last n measures

Table 15 - Filter types

If the selected filter type is not “0”, a proper filter constant value has to be specified for the correct filter operation (see object 61A1h).

Object description

Index	Name
61A0h	AI filter type

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI filter type 1	RO	Unsigned8	(see table)	0

61A1h – AI filter constant

This object indicates the configured constant value used for the filter calculation (see object 61A0h).

Object description

Index	Name
61A1h	AI filter constant

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI filter constant 1	RW	Unsigned8	1..64	1

The value of the filter constant should be set depending on the type of filter used (see object 61A0). The same value of the filter constant gives different results with different filter types.

Note:
The calculation result is also influenced by the value of the AI ADC sample rate value (see object 6114h), so the choice of the AI filter constant should be done depending also on the value of that parameter.

7100h – AI input FV

This object provides the converted value of the analog input module, which is not yet scaled to the physical unit of the pressure, called field value (FV).

Object description

Index	Name
7100h	AI input FV

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI input FV	RO	Unsigned16	Unsigned16	-

7120h – AI input scaling 1 FV

This object indicates the configured FV of the first calibration point for the analog input channel.

Object description

Index	Name
7120h	AI input scaling 1 FV

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI input scaling 1 FV	RO	Unsigned16	Unsigned16	-

7122h – AI input scaling 2 FV

This object indicates the configured FV of the second calibration point for the analog input channel.

Object description

Index	Name
7122h	AI input scaling 2 FV

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI input scaling 2 FV	RO	Unsigned16	Unsigned16	-

9121h – AI input scaling 1 PV (integer32)

This object indicates the configured PV of the first calibration point for the analog input channel. It is scaled in physical unit of PV, considering the actual number of decimal digits (see objects 6131h and 6132h).

The data type is 32 bit signed integer.

For more details about this object usage see also the section “DS 404 specific functionalities”.

Object description

Index	Name
9121h	AI input scaling 1 PV (integer32)

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI input scaling 1 PV 1 (integer32)	RW	Integer32	Integer32	-

9123h – AI input scaling 2 PV (integer32)

This object indicates the configured PV of the second calibration point for the analog input channel. It is scaled in physical unit of PV, considering the actual number of decimal digits (see objects 6131h and 6132h). The data type is 32 bit signed integer.

For more details about this object usage see also the section “DS 404 specific functionalities”.

Object description

Index	Name
9123h	AI input scaling 2 PV (integer32)

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI input scaling 2 PV 1 (integer32)	RW	Integer32	Integer32	-

9124h – AI input offset (integer32)

This object indicates the configured additional offset value for the analog input channel. It is scaled in physical unit of PV, considering the actual number of decimal digits (see objects 6131h and 6132h). The data type is 32 bit signed integer.

For more details about this object usage see also the section “DS 404 specific functionalities”.

Object description

Index	Name
9124h	AI input offset (integer32)

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI input offset 1 (integer32)	RW	Integer32	Integer32	-

9130h – AI input PV (integer32)

This object provides the result of the input scaling block and gives the measured quantity scaled in the used physical unit of the process value set by AI physical unit PV (object 6131h), considering the actual number of decimal digits (object 6132h).

The data type is 32 bit signed integer..

Object description

Index	Name
9130h	AI input PV (integer32)

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI input PV 1 (integer32)	RW	Integer32	Integer32	-

9148h – AI span start (integer32)

This object indicates the configured lower limit of the expected Process Value. When the PV is lower than this limit, it is marked as negative overloaded (see AI status, object 6150h).

It is scaled in physical unit of PV, considering the actual number of decimal digits (see objects 6131h and 6132h).

The data type is 32 bit signed integer.

Object description

Index	Name
9148h	AI span start (integer32)

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI span start 1 (integer32)	RW	Integer32	Integer32	-

Note:
The value is set at the 5%FS below the minimum nominal pressure value by default.
The user can define a specific value. The value is refused if it is below the 10%FS of the minimum nominal pressure value.
The value can not be higher than the AI span end value (see object 9149h).
This object influences the EMCY service.

9149h – AI span end (integer32)

This object indicates the configured upper limit of the expected Process Value. When the PV exceeds this limit, it is marked as positive overloaded (see AI status, object 6150h).

It is scaled in physical unit of PV, considering the actual number of decimal digits (see objects 6131h and 6132h).

The data type is 32 bit signed integer.

Object description

Index	Name
9149h	AI span end (integer32)

Entry description

SUB Index	Name	Access	Data Type	Value Range	Default
0	Highest sub-index supported	RO	Unsigned8	1	1
1	AI span end 1 (integer32)	RW	Integer32	Integer32	-

Note:
The value is set equal to the maximum nominal pressure value by default.
The user can define a specific value. The value is refused if it is above the 10%FS of the maximum nominal pressure value.
The value can not be higher than the AI span end value (see object 9149h).
This object influences the EMCY service.

5. PDO SERVICES

The real-time data transfer is performed by means of "Process Data Objects (PDO)". Data type and mapping of application objects into a PDO is determined by a corresponding default PDO mapping structure within the object dictionary. For the PDO1 see object 1A00h.

Communication parameters of the PDO, as COB-ID, transmission mode and transmission frequency, are also specified in the object dictionary. For the PDO1 see object 1800h.

Since the GEFRAN KHC CANopen device is a PDO producer, its PDO is also called Transmit PDO (TPDO).

5.1 PDO MESSAGE FORMAT

The format of the Transmit PDO message is explained in the following figure.

COB-ID	Rx/Tx	DLC	Data				
			D0	D1	D2	D3	D4
180h + Node-ID	Tx	5	Pressure LSB	Pressure	Pressure	Pressure MSB	Status

Figure 42 - Transmit PDO1 (TPDO1) message format

5.2 PDO DATA TYPES

Two types of data are mapped in PDO1 by default: Pressure and Status.
Pressure data can be an INTEGER32 data type or REAL32 data type.
Status data is an UNSIGNED8 data type.
A third type of data can be mapped in PDO1: Temperature (INTEGER16 data type).

Assuming that the data is expressed as a bit sequence of length 32 for INTEGER32 data type and REAL32 data type (b0..b31), and as a bit sequence of length 8 for UNSIGNED8 data type (b0..b7), the transfer syntax is explained in the following figure.

Octet number	1	2	3	4
INTEGER32 REAL32	$b_7..b_0$	$b_{15}..b_8$	$b_{23}..b_{16}$	$b_{31}..b_{24}$
UNSIGNED8	$b_7..b_0$	$b_{15}..b_8$	-	-

Figure 43 - Transfer syntax for different data type

Floating point numbers

Data of the basic data types REAL32 have values in the real numbers.
The data type REAL32 is represented as a bit sequence with the length 32.
The IEEE 32 bit implementation of floating point number is represented in the following table.

Bit	b31	b30..b23	b22..b0
Function	S (sign)	E (exponent)	F (mantissa)

The bit sequence b = b0..b31 assigns the following value (finite non-zero number):
REAL32(b) = $(-1)^S \times 2^{E-127} \times (1+F)$
where
$S = b_{31}$, is the sign
$E = b_{30} \times 2^7 + \ldots + b_{23} \times 2^0$, $0 < E < 255$, is the un-biased exponent
$F = 2^{-23} \times (b_{22} \times 2^{22} + \ldots + b_1 \times 2^1 + b_0 \times 2^0)$ is the fractional part of the number (mantissa)

Notes:
E = 0 is used to represent + 0.
E = 255 is used to represent infinities or NaN (not a number).

Example:

Hex: 40C8 0000HEX

Binary: 0100 0000 1100 1000 0000 0000 0000 0000BIN

Calculation of sign, exponent and mantissa:

S = 0

E = 1000 0001BIN = $1 \times 2^7 + 1 \times 2^0$ = 129DEC

F = $1 \times 2^{-1} + 1 \times 2^{-4}$ = 0,5 + 0,0625 = 0,5625DEC

Calculation of the floating point number:

40C8 0000HEX = $(-1)^0 \times 2^{129-127} \times (1+0,5625)$ = 6,25

5.3 PDO MAPPING

The GEFRAN KHC CANopen supports a variable PDO mapping. When the device is in NMT state Pre-operational, the following procedure is used for re-mapping:

1. Destroy TPDO1 by setting bit valid of COB-ID used by TPDO1 in TPDO1 communication parameter object (1800h, sub-index 1) to 1b
2. Disable mapping by setting Number of mapped application objects in TPDO1 mapping parameter object (1A00, sub-index 0) to 0
3. Modify mapping by changing the value of
   a. 1st application object in TPDO1 mapping parameter object (1A00, sub-index 1)
   b. 2nd application object in TPDO1 mapping parameter object (1A00, sub-index 2)
   c. 3rd application object in TPDO1 mapping parameter object (1A00, sub-index 3)
   to one of the values listed in the following table.

Mappable object	Value	Number of bytes
2090h: Process value as integer	20900020h	4
6130h: AI input PV (float)	61300120h	4
9130h: AI input PV (integer 32)	91300120h	4
6150h: Ai status	61500108h	1
2091h: Temperature	20910010h	2

Table 16 - Mappable objects in TPDO1

4. Enable mapping by setting Number of mapped application objects in TPDO1 mapping parameter object (1A00, sub-index 0) to the desired value (1..3)
5. Create TPDO1 by setting bit valid to 0b of COB-ID used by TPDO1 in TPDO1 communication parameter object (1800h, sub-index 1) to 0b

Note:
the total number of bytes mapped in TPDO1 cannot exceed the value of 8.
Otherwise, a mapping error is given when enabling mapping (step 4).

5.4 PDO TRANSMISSION TYPES

The PDO transmission type for the KHC CANopen device can be changed.
There are three types of transmission mode:

1. Synchronous transmission
2. Asynchronous transmission with RTR frames
3. Asynchronous transmission with event-timer

Synchronous Transmission

The transmission of the PDO is performed after the CANopen device receive the n-th SYNC object, when the transmission type is set to n, with n in the range of 1..240.

The SYNC message format is described in the SYNC services description.

Asynchronous Transmission with RTR frames

The transmission of the PDO is performed after the CANopen device receive the PDO remote frame.
The format of the PDO remote frame is explained in the following figure.

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
PDO COB-ID + RTR bit	Rx	0	-	-	-	-	-	-	-	-

Figure 44 - RTR message format

Asynchronous Transmission

The transmission of the PDO is performed cyclically after the event-timer has elapsed. The transmission period, expressed in multiples of 1 ms, can be changed through the object 1800h sub-index 5 (PDO event timer) or through the object 6200h (cyclic timer).

6. NMT SERVICES

Through NMT services the NMT master controls the state of the NMT slave devices.

The state attribute is one of these:

- √ Initialization
- √ Pre-operational
- √ Operational
- √ Stopped

6.1 NMT DEVICE STATES

Initialization state

In the NMT state initialization the CANopen device is initialized. The CANopen device parameters are set to their power-on values (last stored parameters in non-volatile memory).

The NMT state initialization own the sub-states Reset application and Reset communication, which are processed automatically one after the other :

1) Reset application: the CANopen device resets all application-related CANopen device parameters and initializes the CANopen node-ID.
2) Reset communication: the CANopen device reset all communication-related CANopen device parameters and set the CANopen node-ID.

Pre-operational state

In the pre-operational state the behaviour of the CANopen device at its communication interface can be configured.

This can take place by SDO or LSS services. PDO communication is not allowed.

Operational state

In the operational state all communication objects are active. Object Dictionary Access via SDO is possible and the node can handle PDO communication.

Stopped state

In the stopped state the device stops the communication. In this state no communication object is supported, except of Error control services and the reception of NMT commands.

6.2 NMT NODE CONTROL

After power-on, the CANopen device initializes. The initialization state terminates with the transmission of the boot-up message, after which the device enters autonomously the pre-operational state.

In order to change the NMT state of a CANopen device, the NTM master sends the message shown in the following figure.

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
0	Tx	2	CS	Node-ID	-	-	-	-	-	-

Figure 45 - NMT message format

The bit fields and their values are explained in the following table.

Bit field	Value range	Description
CS	1	Start. Enter NMT Operational state
	2	Stop. Enter NMT Stopped state
	128	Enter NMT Pre-operational state
	129	Enter NMT Reset application state
	130	Enter NMT Reset communication state
Node-ID	0	All devices must perform the commanded transition
	1 to 127	Only the device that claims the indicated Node-ID must execute the commanded transition

All possible NMT states and state transitions are shown in the following figure.

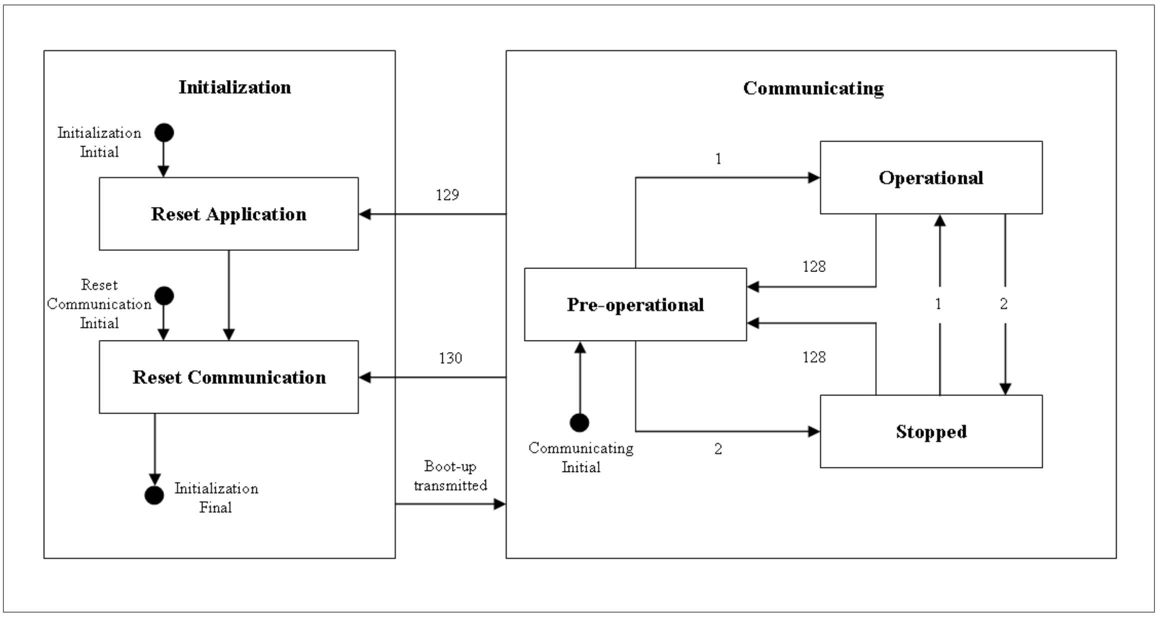

Figure 46 - NMT states and state transitions

6.3 NMT STATES AND COMMUNICATION OBJECTS

Specific services can only be executed if the devices involved in the communication are in the appropriate communication states.

The relationship between communication states and communication objects is shown in the following table.

Object	**Reset application**	**Reset communication**	**Pre-operational**	**Operational**	**Stopped**
PDO				X	
SDO			X	X	
Boot up		X			
SYNC			X	X	
EMCY			X	X	
NMT error control (Heartbeat and Node guarding)			X	X	X
NMT node control			X	X	

Table 17 - NMT states and communication objects

6.4 Restricted CAN-IDs

Restricted CAN-ID can't be used as a CAN-ID by any configurable communication object, neither for SYNC, EMCY, PDO, and SDO. They are listed in the following table.

CAN-ID	used by COB
0 (000h)	NMT
1 (001h) – 127 (07Fh)	reserved
257 (101h) – 384 (180h)	reserved
1409 (581h) – 1535 (5FFh)	default SDO (tx)
1537 (601h) – 1663 (67Fh)	default SDO (rx)
1760 (6E0h) – 1791 (6FFh)	reserved
1793 (701h) – 1919 (77Fh)	NMT error control
1920 (780h) – 2047 (7FFh)	reserved

Table 18 - Restricted CAN-IDs

7. BOOT-UP SERVICES

Through this service, the NMT slave indicates that a local state transition occurred from the state Initialization to the state Pre-operational.

The protocol uses the same identifier as the error control protocol. The format of the boot-up message is explained in the following figure.

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
700h + Node-ID	Tx	1	00h	-	-	-	-	-	-	-

Figure 47 – Boot-up message format

8. SYNC SERVICES

The SYNC object can be broadcasted periodically by the SYNC producer. This SYNC object provides the basic network synchronization mechanism.

If the CANopen devices operates synchronously (see object 1800, sub-index 2), it uses the SYNC object to synchronize its own timing, as the PDO transmission, with that of the synchronization object producer.

The format of the SYNC object is explained in the following figure.

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
80h	Rx	0	-	-	-	-	-	-	-	-

Figure 48 - SYNC message format

The COB-ID of the SYNC message can be changed by object 1005h (SYNC COB-ID).

9. EMCY SERVICES

Emergency objects are triggered by the occurrence of the CANopen device internal error situation. An emergency object is transmitted only once per 'error event'. No further emergency objects are transmitted as long as no new errors occur on the CANopen device. If one or more error conditions change, the CANopen device transmits the emergency object with the updated error code. The error register value inside the EMCY object is also updated.

For the GEFRAN KHC CANopen device the “Generic error” condition is defined.

The possible EMCY error codes are shown in the following table.

Error code	Description
0000h	Error reset or no error
1000h	Generic error

Table 19 - EMCY error codes for the KHC CANopen device

About the content of the error register see the description of the object 1001h (Error register).

The format of the EMCY message is explained in the following figure.

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
80h + Node-ID	Tx	8	EMCY error code LSB	EMCY error code MSB	Error register (1001h)	Manufacturer specific error field				

Figure 49 - EMCY message format

The COB-ID of the EMCY message can be changed by object 1014h (EMCY COB-ID).

The Manufacturer specific error field, inside the EMCY message, is defined as follows.

D3	D4	D5	D6	D7
xxxxxxx1: flash error xxxxx1xx: max. allowed pressure exceeded xxxx1xxx: min. allowed pressure exceeded	00h	00h	00h	00h

Figure 50 - Manufacturer specific error field

The objects involved with the EMCY service are the following:
- 1015h: Inhibit time EMCY
- 2340h: EMCY pressure exceeded reset hysteresis
- 6148h: AI span start (float)
- 6149h: AI span end (float)
- 9148h: AI span start (integer32)
- 9149h: AI span end (integer32)

10. ERROR CONTROL SERVICES

The error control services are used to detect failures within a CAN-based network. Error control services are achieved principally through periodically transmission of messages by a CANopen device.

Two possible mechanism exist to perform the error control: Node guarding and Heartbeat.

The GEFRAN KHC CANopen device makes use of both the mechanism..

10.1 Node guarding protocol

The slave uses the guard time (object 100Ch) and life time factor (object 100D) from its object dictionary to calculate the node lifetime, as follows:

node lifetime = guard time x life time factor

If node lifetime is 0, the slave does not handle the guarding mechanism of the NMT master. The guarding is achieved by transmitting guarding requests (node guarding protocol) by the NMT master.

If a NMT slave has not responded within a defined span of time (node life time) or if the NMT slave's communication status has changed, the NMT master informs its NMT master application about that event.

If the NMT slave is not guarded within its lifetime, the NMT slave informs its local application about that event. Guarding starts for the NMT slave when the first RTR for its guarding CAN-ID is received. This may be during the boot-up phase or later.

For the KHC CANopen device the node guarding is disabled by default. It can be programmed through objects 100Ch and 100Dh.

10.2 Heartbeat protocol

The heartbeat mechanism is established by cyclically transmitting the heartbeat message. If the heartbeat cycle fails for the heartbeat producer the local application on the heartbeat consumer, aware of this heartbeat message, will be informed about that event.

The format of the heartbeat message is explained in the following figure.

COB-ID	Rx/Tx	DLC	Data							
			D0	D1	D2	D3	D4	D5	D6	D7
700h + Node-ID	Tx	1	NMT state	-	-	-	-	-	-	-

Figure 51 - Heartbeat message format

The first byte of the heartbeat message data field contains the actual CANopen Network Management State of the

CANopen device, as shown in the following table.

Bit field	Value	Description
NMT state	0	Reserved (see boot-up protocol)
	4	Stopped
	5	Operational
	127	Pre-operational

Table 20 - NMT state field in heartbeat message

For the KHC CANopen device the heartbeat is disabled by default. It can be programmed through object 1017h.

11. DS404 PROFILE SPECIFIC FUNCTIONALITIES

In this section specific functionalities defined in ds404 profile are explained.

11.1 Calibration

The analog input Functional Block (FB) converts Field Values (FVs) into Process Values (PVs).
The unscaled readings from the A/D converter are the Field Values.
The FVs are converted to the physical dimension SI units of the measured quantity.
These converted values are called Process Values. Pressure values in bar, psi, Pa, etc. are Process Values.
The conversion from FVs to PVs is described as a linear transformation. This is defined by two pairs of FVs and corresponding PVs called calibration point 1 and calibration point 2.

Calibration point 1: (Input scaling 1 FV, Input scaling 1 PV)
Calibration point 2: (Input scaling 2 FV, Input scaling 2 PV)

This is illustrated in the following figure.:

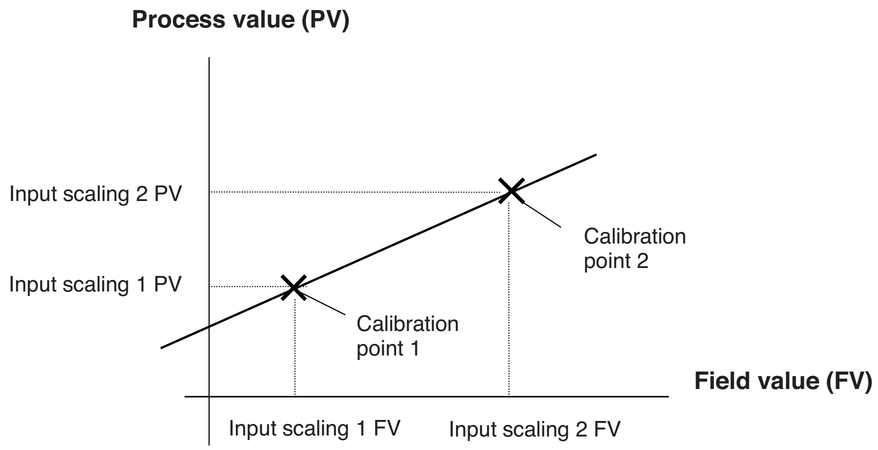

Figure 52 – Calibration

The calibration of point 1 is carried out via the objects 6121h (data is float) or 9121h (data is integer32).
The calibration of point 2 is carried out via the objects 6123h (data is float) or 9123h (data is integer32).
The objects 7120h and 7122h are read only.

The GEFRAN KHC transducer is yet calibrated by the manufacturer.
The user can perform his own calibration, if needed. The user-defined calibration can be discarded through a Restore default parameters action (see object 1011h).

The device can be calibrated following the instructions described below.

11.2 Pre-calibration recommendations

Before calibrating the device, it is suggested to set to zero the value of the AI input offset (object 6124h or 9124h), so that the user can verify that the pressure value after calibration equals the value set for P1 (object 6121h or 9121h) or P2 (object 6123h or 9123h).
Otherwise the user must remember that the output pressure value is affected by the AI input offset value.

Calibration of point 1

1. The user apply the required pressure value (reference value) of the calibration point 1
2. The user waits until the pressure value is stable at the reference value
3. The user writes the value that the device is supposed to indicate under the pressure currently applied to the object 6121h (float data), or to the object 9121h (32 bit signed integer data)

Calibration of point 2

1. The user apply the required pressure value (reference value) of the calibration point 2
2. The user waits until the pressure value is stable at the reference value
3. The user writes the value that the device is supposed to indicate under the pressure currently applied to the object 6123h (float data), or to the object 9123h (32 bit signed integer data)

Notes:
The value written to the objects 6121h, 6123h, 9121h and 9123h, is expressed with the physical unit currently used (see object 6131h)
The value written in objects 9121h and 9123h has to take in consideration the number of decimal digits currently used (see object 6132h)
The calibration is refused if the calculated characteristic differs too much from the manufacturer calculated characteristic, in particular if the value of the new calculated k coefficient (slope of the characteristic) is 5%FS above the value of the k coefficient set by manufacturer.

Example 1
KHC sensor with nominal pressure range 0..250 bar
The user has set the pressure physical unit in psi unit with a number of decimal digits of 2.
AI physical unit PV (6131h): psi
AI decimal digits (6132h): 2
The user is used to operate with integer values, so has mapped the object 9130h (AI input PV (integer32) in TPDO1.
The nominal pressure range of 0..250 bar corresponds to 0..3625 psi.

Calibration of point 1

The user apply a reference pressure of 0 psi.
When the pressure is stable, a pressure of 0,65 psi is measured by the reference pressure sensor.
Since the number of decimal digits (6132h) is 2, the value that the user writes to object 9121h is 0,65 x 102 = 65 = 00000041h.
The user must send the SDO write command shown in the following image.

COB-ID	DLC	CS	index		sub-index	value			
		D0	D1	D2	D3	D4	D5	D6	D7
600h + Node-ID	8	22h	21h	91h	01h	41h	00h	00h	00h

Figure 53 – Calibration of point 1 (example 1)

Calibration of point 2

The user apply a reference pressure of 3625 psi.
When the pressure is stable, a pressure of 3624.12 psi is measured by the reference pressure sensor.
Since the number of decimal digits (6132h) is 2, the value that the user writes to object 9123h is 3624.12 x 102 = 362412 = 000587ACh.
The user must send the SDO write command shown in the following image.

COB-ID	DLC	CS	index		sub-index	value			
		D0	D1	D2	D3	D4	D5	D6	D7
600h + Node-ID	8	22h	23h	91h	01h	ACh	87h	05h	00h

Figure 54 – Calibration of point 2 (example 1)

Example 2
KHC sensor with nominal pressure range 0..250 bar
The user has set the pressure physical unit in bar.
AI physical unit PV (6131h): bar
The user is used to operate with floating point numbers, so has mapped the object 6130h (AI input PV (float)) or the object 2090h in TPDO1.

Calibration of point 1

The user apply a reference pressure of 0 bar.
When the pressure is stable, a pressure of 0.3 bar is measured by the reference pressure sensor.
The value 0.3 in floating point data format corresponds to 3ECCCCCDh.
The user must send the SDO write command shown in the following image.

COB-ID	DLC	CS	index		sub-index	value			
		D0	D1	D2	D3	D4	D5	D6	D7
600h + Node-ID	8	22h	21h	61h	01h	CDh	CCh	CCh	3Eh

Figure 55 – Calibration of point 1 (example 2)

Calibration of point 2

The user apply a reference pressure of 250 bar.
When the pressure is stable, a pressure of 250.2 bar is measured by the reference pressure sensor.
The value 250.2 in floating point data format corresponds to 437A3333h.
The user must send the SDO write command shown in the following image.

COB-ID	DLC	CS	index		sub-index	value			
		D0	D1	D2	D3	D4	D5	D6	D7
600h + Node-ID	8	22h	23h	91h	01h	33h	33h	7Ah	43h

Figure 56 – Calibration of point 2 (example 2)

11.3 Offset adjustment

Using the offset adjustment, the calibration characteristic is shifted by an additional input offset value.
The calibration characteristic is shifted down by positive values of the offset.
The calibration characteristic is shifted up by negative values of the offset.
This is illustrated in the following figure.

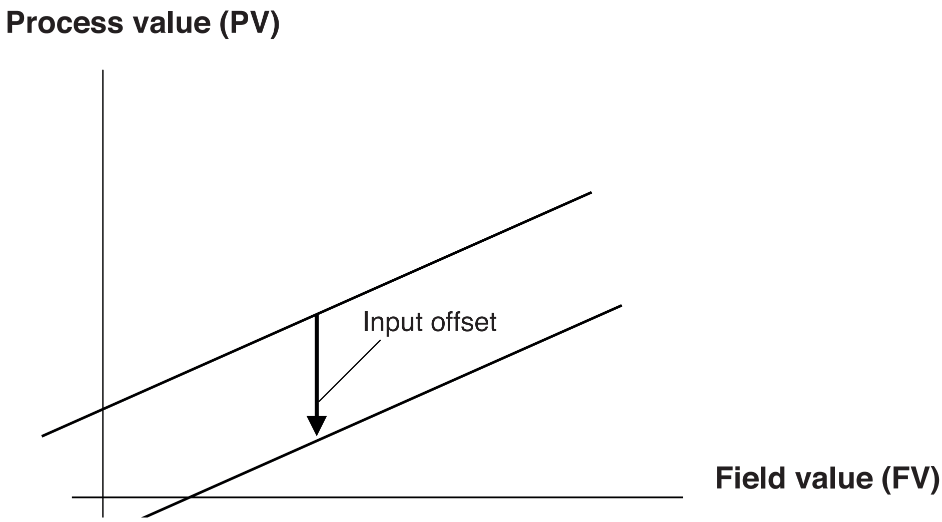

Figure 57 – Offset adjustment

The user can use the offset adjustment functionality to get the required exact reading value from the device at a specific pressure level.

Notes:

- The value written to the objects 6124h or 9124h is expressed with the physical unit currently used (see object 6131h)
- The value written to the object 9124h (32 bit signed integer) has to take in consideration the value of the number of decimal digits currently used (see object 6132h)
- The maximum offset value that can be accepted is within a range of ±10%FS.
  If the value set is outside this range, an SDO abort will be given, and the value discarded.

Example
The user has set the pressure physical unit in bar unit with a number of decimal digits of 1.
The user has set the pressure to 100.0 bar, but the device indicates 100.2 bar.
The user wants to get a 100 bar reading from the device.
The offset value to get a 100.0 bar output value from the device is 0.2 bar.

When using floating point numbers, the user has to write the value 0.2 to the object 6124h.
The value 0.2 in floating point data format corresponds to 3E4CCCCDh.
The user must send the following SDO write command:

COB-ID	DLC	CS	index		sub-index	value			
		D0	D1	D2	D3	D4	D5	D6	D7
600h + Node-ID	8	22h	24h	61h	01h	CDh	CCh	4Ch	3Eh

Figure 58 – Offset adjustment request (6124h)

The response message is the following:

COB-ID	DLC	CS	index		sub-index	value			
		D0	D1	D2	D3	D4	D5	D6	D7
580h + Node-ID	8	60h	24h	61h	01h	00h	00h	00h	00h

Figure 59 – Offset adjustment response (6124h)

When using 32 bit integer numbers, considering the number of decimal digits is 1, the user has to write the value 0.2 x 10 = 2 = 00000002h to the object 9124h.
The user must send the following SDO write command:

COB-ID	DLC	CS	index		sub-index	value			
		D0	D1	D2	D3	D4	D5	D6	D7
600h + Node-ID	8	22h	24h	91h	01h	02h	00h	00h	00h

Figure 60 – Offset adjustment request (9124h)

The response message is the following:

COB-ID	DLC	CS	index		sub-index	value			
		D0	D1	D2	D3	D4	D5	D6	D7
580h + Node-ID	8	60h	24h	91h	01h	00h	00h	00h	00h

Figure 61 – Offset adjustment response (9124h)

11.4 Auto-zero

The auto-zero command sets the zero offset value so that the instantaneous measured PV becomes zero.
The autozero command is executed by writing the signature value of “zero“ to the object 6125h.
The offset value (Ai Input offset) is automatically calculated. It can be read through the objects 6124h (float) or 9124h (integer32).
Similarly to the offset adjustment, after the autozero is performed, the whole calibration characteristic is shifted by the calculated offset.

The autozero procedure is described below:

1. The user applies a pressure of zero (e.g. 0 bar)
2. The user launches the autozero command through an SDO write command (see below)
3. The user waits for the correct SDO write response from the device

In order to launch the autozero command, the user must send the following SDO write command:

COB-ID	DLC	CS	index		sub-index	value			
		D0	D1	D2	D3	D4	D5	D6	D7
600h + Node-ID	8	22h	25h	61h	01h	7Ah	65h	72h	6Fh

Figure 62 – Autozero command request

The response message is the following:

COB-ID	DLC	CS	index		sub-index	value			
		D0	D1	D2	D3	D4	D5	D6	D7
580h + Node-ID	8	60h	25h	61h	01h	00h	00h	00h	00h

Figure 63 – Autozero command response

Note:

- The autozero command must be executed when the pressure is near 0 bar (or equivalent pressure value).
  The device automatically detects this condition. The offset value, calculated by the autozero function, that can be accepted must be within a range of ±10%FS.
  Otherwise, an SDO abort will be given, and the autozero procedure will not be completed..

GEFRAN spa
via Sebina, 74 - 25050 PROVAGLIO D'ISEO (BS) - ITALIA
tel. 0309888.1 - fax. 0309839063 Internet: http://www.gefran.com