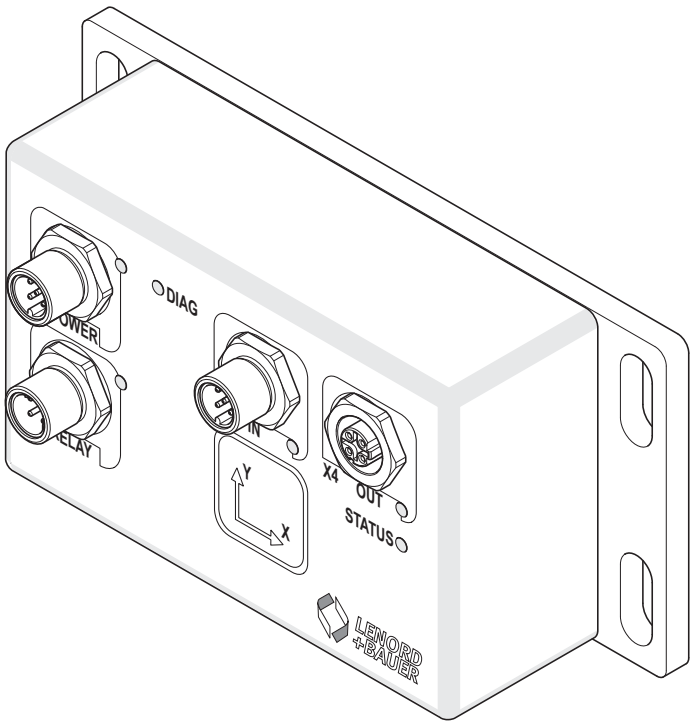
LENORD +BAUER
... automates motion.

Reference

Fieldbus interface



D-02R-3010CO (1.0)

Device manufacturer and publisher:

Lenord, Bauer & Co. GmbH
Dohlenstraße 32
46145 Oberhausen ● Germany
Phone: +49 208 9963–0 ● Fax: +49 208 676292
Internet: www.lenord.de ● E-Mail: info@lenord.de

Doc. no. D-02R-3010CO (1.0)

Table of contents

1 General

1.1 About these instructions

The following description covers the CANopen integration of the Tower Oscillation Monitor **GEL 3010 CO** x x x x x x x

It is aimed at persons who are already familiar with the principle of operation of an oscillation monitor and who have a basic understanding of the CANopen fieldbus interface. For further information refer to the related standards published by the organisation *CAN in Automation* (CiA) (www.canopen.org).

You will find information on the function and usage as well as the technical data for the Tower Oscillation Monitor in the additional information and the technical information (available in the download area at www.lenord.de).

Numerical data:

Unless explicitly stated, decimal values are given as integers without any additional information (e.g. 1408). Binary values are marked with a “b” (e.g. 1101b) and hexadecimal values with an “h” (e.g. 680h) after the integers.

Abbreviations and glossary:

The term **CO x …** is part of the type code (order code) for the product and is not further stated in the following.

The term **monitor** is often used as a synonym for Tower Oscillation Monitor.

1.2 Description

The Tower Oscillation Monitor is designed as a CANopen slave in accordance with the communication profile CiA 301 and supports a large number of the objects defined in the device profile CiA 401. You can obtain a complete specification of the individual profiles from *CAN in Automation e.V.*:

- Protocol layer: *CiA 301 V4.2.0 – CANopen application layer and communication profile, February 2011* (EN 50325-4)
- Device profile: *CiA 401 V3.0.0 – CANopen device profile for generic I/O modules, June 2008*

The communication and device properties supported as well as the functions of the monitor are defined in the related EDS file.

The monitor is integrated into an existing bus line using two M12 connections.

1.3 Bootloader

The GEL 3010 has a so-called bootloader that in normal operation loads and starts the device-specific application after switching on.

However, the device can also be specifically set to the bootloader state, e.g. to update the firmware. This action is undertaken via the bus communication or by means of a specific switch on procedure:

Switch on power supply briefly 5 times (1 to 6 seconds) and then switch off again. The next time the device is switched on, the device remains in the bootloader state. After switching off and on once more, the device works normally again, provided the firmware is working correctly.

- This state can also occur unintentionally if the power supply is not free of contact bounce. For this reason:
  - ▶ Do not disconnect or make connections if electrically live. Ensure the power supply is free of interference due to contact bounce.

1.4 Firmware update

If the device is in the bootloader state, new application software can be installed. However:

- The firmware is only allowed to be updated in consultation with Lenord+Bauer.

  The warranty will be rendered void if an update is not undertaken by a Lenord +Bauer service engineer.

Starting bootloader

Via CANopen run NMT command (COB ID 00h) with 2 data bytes:
Byte 1 = node number of the device
Byte 2 = 70h

Updating firmware

The firmware used can be read or a new version written using the SDO command 1F50h, subindex 1. This procedure is described in more detail in *CiA Draft Standard Proposal 302, Part 3, April 2010: Configuration and program download*.

Quitting bootloader

Via CANopen run Reset command 81h (NMT Reset Node).

2 Connection and display elements

2.1 Connector M12

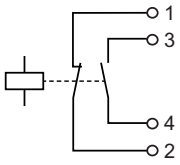
Power supply (X2)

(Male)

1 $+U_B$
2 —
3 GND
4 —

Relay (X1)

(Male)



Bus (X3/X4)

X3 IN

(Male)

X4 OUT

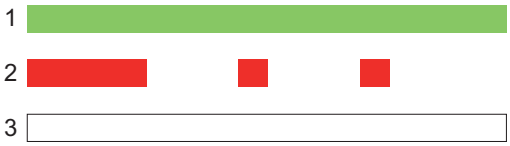
(Female)

1 Screen
2 $+U_B$
3 GND
4 CAN_H
5 CAN_L

2.2 LED displays (1)

Device

- X1 RELAY



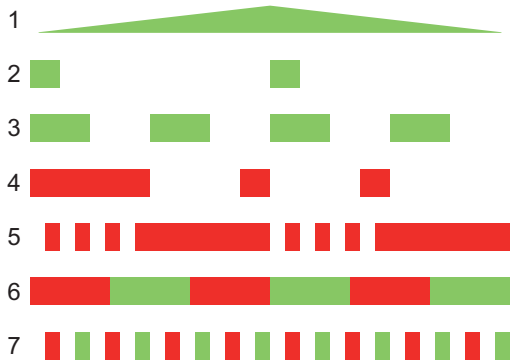
1 Ready
2 Error
3 Triggered (safety circuit opened)

- X2 POWER

1 Internal device power supply in order (orange)
2 24 V supply voltage in order, but no internal supply

(1) Representation on printing in black and white: ■ ≙ green, ■ ≙ red; the area shown covers a duration of approx. 3 seconds

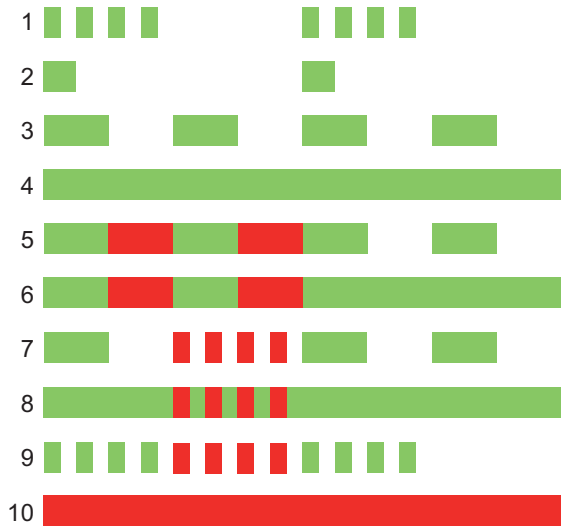
- DIAG



1 Ready (safety circuit closed)
2 Bus error (safety circuit opened)
3 Relay inhibit time elapsed
4 Relay error
5 Sensor error
6 Relay has triggered, switching back on not allowed yet (safety circuit opened)
7 Bootloader mode active, application not started

Bus

- X3 IN and X4 OUT



1 *Init*
2 *Stopped*
3 *Pre-operational*
4 *Operational*
5 *Pre-operational*, Bus warning
6 *Operational*, Bus warning
7 *Pre-operational*, Bus passive
8 *Operational*, Bus passive
9 *Init*, Bus passive
10 *Bus off*

- STATUS

1 Bus ok
2 Invalid baud rate
3 Invalid node address

3 Object list

The object list contains all the CANopen properties supported by the monitor. The data are in the device's non-volatile flash memory and are copied to the memory (RAM) on power-on or reset. If data in the object list are changed, the change is only made in the RAM. If the data are to be saved permanently, they must be transferred to the flash memory via the object 1010h. The original data will then be overwritten.

SDO services are used to access the object list.

The object list is divided into three areas:

- Communication parameters as per CiA 301
- Device parameters as per CiA 401
- Manufacturer-specific parameters

The entries in the object list are addressed using a 16-bit index. Each index entry can be further sub-divided using a subindex.

Information on the object list given below:

- Acc. (access type): ro = read only, rw = read and write
- (Data) type: U*xx* = Unsigned*xx* (xx = 8/16/32 → 1/2/4 bytes without sign), S*xx* = Signed*xx* (xx = 16/32 → 2/4 bytes with sign), STR = ASCII character string
- Sub = Subindex (type: U8)

3.1 Communication parameters in accordance with CiA 301 (1xxxh)

Index	Last name	Acc.	Type	Significance
1000h	*Device type*	ro	U32	870191h Profile 401 (191h), digital inputs/outputs + analogue inputs + manufacturer-specific PDOs (87h)
1001h	*Error register*	ro	U8	Bit 0: 1 = General error (monitor alarm message) Bit 1–6: *Not used* Bit 7: 1 = Manufacturer-specific error

<table>
<tr><th>Index</th><th>Last name</th><th>Acc.</th><th>Type</th><th>Significance</th></tr>
<tr><td>1003h</td><td>Pre-defined error field</td><td>ro</td><td>U32</td><td>
<table>
<tr><th>Sub</th><th>Contents</th></tr>
<tr><td>00h</td><td>Number ≤ 20 (type: rw)</td></tr>
<tr><td>01h</td><td>Last error</td></tr>
<tr><td>02h</td><td>Penultimate error</td></tr>
<tr><td>⋮</td><td></td></tr>
<tr><td>14h</td><td>First of the last 20 errors</td></tr>
</table>
Clear error memory: 00h → Subindex 0
Possible errors:
7300h = Sensor error
7301h = Relay error
8400h = Acceleration over alarm threshold
</td></tr>
<tr><td>1008h</td><td>Manufacturer device name</td><td>ro</td><td>STR</td><td>Product name in ASCII code</td></tr>
<tr><td>1009h</td><td>Hardware version</td><td>ro</td><td>STR</td><td>E.g. “1.01”</td></tr>
<tr><td>100Ah</td><td>Software version</td><td>ro</td><td>STR</td><td>E.g. “3.05”</td></tr>
<tr><td>100Ch</td><td>Guard time – Node monitoring time</td><td>rw</td><td>U16</td><td rowspan="2">The node guarding function is obsolete; CiA recommends the usage of the heartbeat function (Consumer / Producer), → Object 1016h / 1017h.</td></tr>
<tr><td>100Dh</td><td>Life time factor</td><td>rw</td><td>U16</td></tr>
<tr><td>1010h</td><td>Store parameters (power failure-proof)</td><td>rw</td><td>U32</td><td>
Transfer the parameter values from RAM to the flash memory

Write
Write code word “save” in reverse notation (65766173h) to the related subindex
Read
The value 1 is always output

<table>
<tr><th>Sub</th><th>Contents</th></tr>
<tr><td>00h</td><td>Number of save options = 4 (type: ro)</td></tr>
<tr><td>01h</td><td>All parameters</td></tr>
<tr><td>02h</td><td>Only communication parameters (CiA 301)</td></tr>
<tr><td>03h</td><td>Only device parameters (CiA 401)</td></tr>
<tr><td>04h</td><td>Only manufacturer-specific parameters</td></tr>
</table>
</td></tr>
</table>

<table>
<tr><th>Index</th><th>Last name</th><th>Acc.</th><th>Type</th><th>Significance</th></tr>
<tr><td>1011h</td><td>Restore default parameters</td><td>rw</td><td>U32</td><td>Device parameters are reset to their factory settings, not to the values saved using object 1010h
Write
Write code word “load” in reverse notation (64616F6Ch) to the related subindex
Read
The value 1 is always output
<table><tr><th>Sub</th><th>Contents</th></tr><tr><td>00h</td><td>Number of reset options = 4 (type: ro)</td></tr><tr><td>01h</td><td>All parameters</td></tr><tr><td>02h</td><td>Only communication parameters (CiA 301)</td></tr><tr><td>03h</td><td>Only device parameters (CiA 401)</td></tr><tr><td>04h</td><td>Only manufacturer-specific parameters</td></tr></table></td></tr>
<tr><td>1016h</td><td>Consumer Heartbeat time – Repetition time of the master in ms</td><td>rw</td><td>U32</td><td>The heartbeat function (1016h/1017h) should be used instead of node guarding (100Ch/100Dh) according to CiA recommendations.<table><tr><th>Sub</th><th>Contents</th></tr><tr><td>00h</td><td>Number of values = 127 (type: ro)</td></tr><tr><td>01h
⋮
7Fh</td><td>Master node number (bits 23–16) and repetition time (bits 15–0)</td></tr></table>Recommended time setting: Heartbeat Consumer = 3× Heartbeat Producer (1017h)</td></tr>
<tr><td>1017h</td><td>Producer Heartbeat time – Repetition time of the device in ms</td><td>rw</td><td>U16</td><td>Value ≠ 0 deactivated node guarding</td></tr>
</table>

Index	Last name	Acc.	Type	Significance
1018h	*Identity object* – Object identification	ro	U32	Sub: 00h – Contents: Number of IDs = 4 Sub: 01h – Contents: Manufacturer's ID: 1C5h Sub: 02h – Contents: Product code: 3010C0h Sub: 03h – Contents: Revision no.: e.g. 00000002h Sub: 04h – Contents: Serial no.: xxxxxxxxh
1400h	*1st receive PDO parameter* – RxPDO1 configuration	rw	U32	Sub: 00h – Contents: Number of IDs = 2 (type: ro) Sub: 01h – Contents: COB ID used by PDO (default: 200h + node ID) Sub: 02h – Contents: Transmission type for the PDO (default: 01h, cyclic)
1600h	*1st receive PDO mapping* – RxPDO1 mapping	rw	U32	Sub: 00h – Contents: Number of entries = 1 Sub: 01h – Contents: 1st application object: control word (enable operation, 20100008h)
1800h	*1st transmit PDO parameter* – TxPDO1 configuration	rw	U32	Sub: 00h – Contents: Number of IDs = 5 (type: ro) Sub: 01h – Contents: COB ID used by PDO (default: 180h + node ID) Sub: 02h – Contents: Transmission type for the PDO (default: 01h, cyclic) Sub: 03h – Contents: Minimum waiting time for the PDO (in ms) Sub: 04h – Contents: Not used Sub: 05h – Contents: Event timer for the PDO (in ms), after the time has elapsed the PDO is sent automatically

Index	Last name	Acc.	Type	Significance	
				Sub	Contents
1A00h	*1st transmit PDO mapping* – TxPDO1 mapping	rw	U32	00h	Number of entries = 5
				01h	1st application object: acceleration X axis (64010110h)
				02h	2nd application object: acceleration Y axis (64010210h)
				03h	3rd application object: acceleration Z axis (64010310h) (1)
				04h	4th application object: relay status (60000108h)
				05h	5th application object: heartbeat counter (64000108h)

RxPDO structure

Control word 1 byte

Bit 7–1: Reserved	Bit 0: enable operation (1)

TxPDO structure

2 bytes	2 bytes	2 bytes	1 byte	1 byte
X acceleration	Y acceleration	Z acceleration	Status	Lz counter

Lz = Heartbeat

(1) This entry is provided for a future expansion – it always provides the value 0.

3.2 Device parameters in accordance with CiA 401 (6xxxh)

Index	Last name	Acc.	Type	Significance
6000h	Relay status	ro	U8	Sub 00h: Number of entries = 1 Sub 01h: Switch state of the relay Bit 0: Off/on (0/1) Bit 1: Bus status (1 = Bus uncertain) Bit 2: Sensor error (1) Bit 3: Relay error (1) Bit 4: Inhibit time active (1)
6400h	Heartbeat counter	ro	S8	Sub 00h: Number of entries = 1 Sub 01h: Heartbeat counter (0…255)
6401h	Actual acceleration	ro	S16	Sub 00h: Number of entries = 3 Sub 01h: In X direction Sub 02h: In Y direction Sub 03h: In Z direction (1) Values in 1/100 m/s^2

3.3 Manufacturer-specific objects (2xxxh)

Index	Last name	Acc.	Type	Significance
2010h	Enable operation	rw	U8	1 = Enable operation (*Operation release*)
2016h	Alarm limits	ro	S16	Sub 00h: Number of entries = 6 Sub 01h: Acceleration in -X direction Sub 02h: Acceleration in +X direction Sub 03h: Acceleration in -Y direction Sub 04h: Acceleration in +Y direction Sub 05h: Acceleration in -Z direction (1) Sub 06h: Acceleration in +Z direction (1)
2017h	Relay inhibit time	ro	U16	30 s (1Eh)

(1) This entry is provided for a future expansion – it always provides the value 0.

<table>
<tr><th>Index</th><th>Last name</th><th>Acc.</th><th>Type</th><th>Significance</th></tr>
<tr><td>2200h</td><td>Result of self-test</td><td>ro</td><td>U16</td><td>
<table>
<tr><th>Sub</th><th>Contents</th></tr>
<tr><td>00h</td><td>Number of entries = 3</td></tr>
<tr><td>01h</td><td>Acceleration in X direction</td></tr>
<tr><td>02h</td><td>Acceleration in Y direction</td></tr>
<tr><td>03h</td><td>Acceleration in Z direction (1)</td></tr>
</table>
</td></tr>
<tr><td>2201h</td><td>Self-test</td><td>rw</td><td>U8</td><td>Write: 1 = Start
Read: Number of self-tests undertaken since switch on</td></tr>
</table>

(1) This entry is provided for a future expansion – it always supplies the value 0.

4 SDO communication

The service data objects (SDO) form the communication channel for the transmission of device parameters. As these parameters are transmitted acyclically (e.g. only once on booting the network), the SDOs have a lower priority (high COB identifier).

Structure of the SDO message

COB ID	DLC	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7	Byte 8
SDO identifier	Data length	Com-mand	Index L	Index H	Subin-dex	Data 0	Data 1	Data 2	Data 3

The SDO identifier is defined as follows:

Client (control system) → Server (absolute rotary encoder): **600**h + node ID

Server (absolute rotary encoder) → Client (control system): **580**h + node ID

The data length (DLC) is always 8: 1 command byte + 2 index bytes (object) + 1 sub-index byte + 4 data bytes

The command defines whether data are to be written (download) or read (upload) and the number of user data bytes:

Com-mand	Description	User data	Function
22h	SDO(rx), Download Request	Undefined	Send parameters to the monitor
23h		4 bytes	
2Bh		2 bytes	
2Fh		1 byte	
60h	SDO(tx), Download Response	—	Confirm parameter acceptance to the client
40h	SDO(rx), Upload Request	—	Request parameters from the moni-tor
42h	SDO(tx), Upload Response	Undefined	Send parameters to the client
43h		4 bytes	
4Bh		2 bytes	
4Fh		1 byte	
80h	SDO(tx), Abort Domain Transfer (abort due to error)	4 bytes	monitor signals error code to the client

In the case of an error, an error message with the command 80h (SDO Abort Message) replaces the normal confirmation (response). Index and subindex belong to the object stated previously. The error code (Abort code) is given in bytes 5 to 8:

Abort codes	Error
05040001h	Command byte is not supported
06010000h	Incorrect access to an object
06010001h	Read access to a write-only object
06010002h	Write access to a read-only object
06020000h	Object is not supported
06090011h	Subindex is not supported
06090030h	Parameter value outside the limits
06090031h	Parameter value too high
06090032h	Parameter value too low
08000000h	General error
08000020h	Incorrect memory signature (“save”)
08000021h	Not possible to save parameters