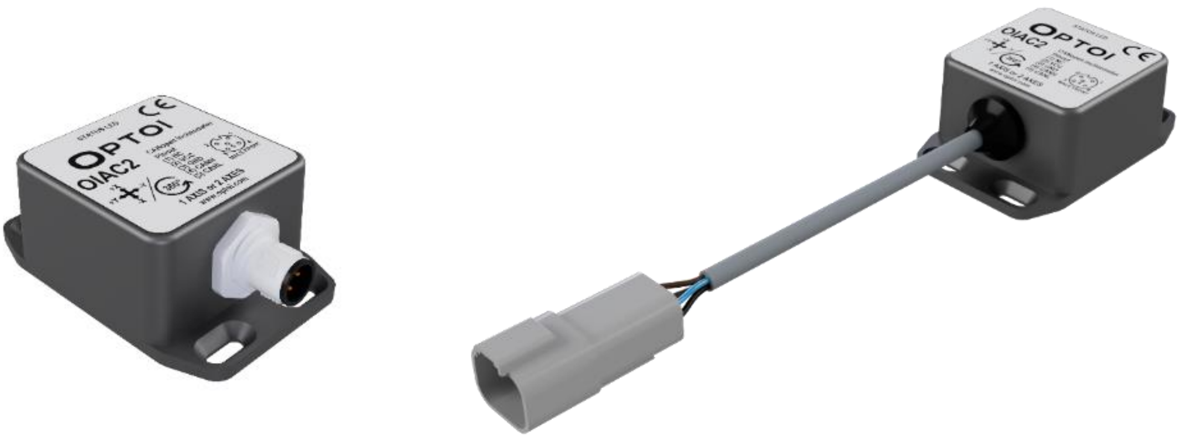
OIAC2

CANopen inclination sensors

User Reference Manual



Date	Document Revision	Notes
01-08-2024	0	First release

INDEX

A general knowledge of CAN OPEN protocol is reccomended to fully understand this manual. All the numbers contained are written in decimal format. Hexadecimal values are marked with a final "h" to distinguish them from decimal.

1 TECHNICAL INFORMATION

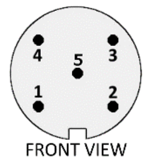
Please refer to the OIAC2 datasheet for all the technical details and specifications.

1.1 Electrical connections

OIAC2 inclinometer has different connection options. The complete list of such options is detailed in the updated technical datasheet on the website.

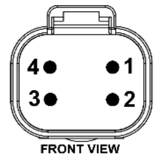
M12-5 POLES MALE PLUG CONNECTOR VERSION

No	Colour	Name	Function
1		SHIELD	Optional CAN shield
2		CAN_V+	Positive power supply
3		CAN_GND	Ground / 0V / V-
4		CAN_H	CAN H bus line (dominant high)
5		CAN_L	CAN L bus line (dominant low)



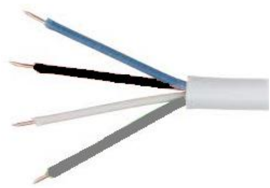
AMPHENOL AT04-4P CONNECTOR ON 0.1M CABLE

No	Colour	Name	Function
1		CAN_GND	Ground / 0V / V-
2		CAN_H	CAN H bus line (dominant high)
3		CAN_L	CAN L bus line (dominant low)
4		CAN_V+	Positive power supply



4 POLES CABLE VERSION

No	Colour	Name	Function
	BLUE	CAN_GND	Ground / 0V / V-
	BLACK	CAN_H	CAN H bus line (dominant high)
	GREY	CAN_L	CAN L bus line (dominant low)
	WHITE	CAN_V+	Positive power supply



1.2 CANBUS connections and termination resistors

The OIAC2 inclinometer doesn't have internal bus line terminator resistor. The user must ensure two 120Ω termination resistors are installed between the CANH and CANL lines. Typically, one is located near the network master at the start of the bus. The other should be at the end of the CANBUS, on the furthest node of the CAN network.

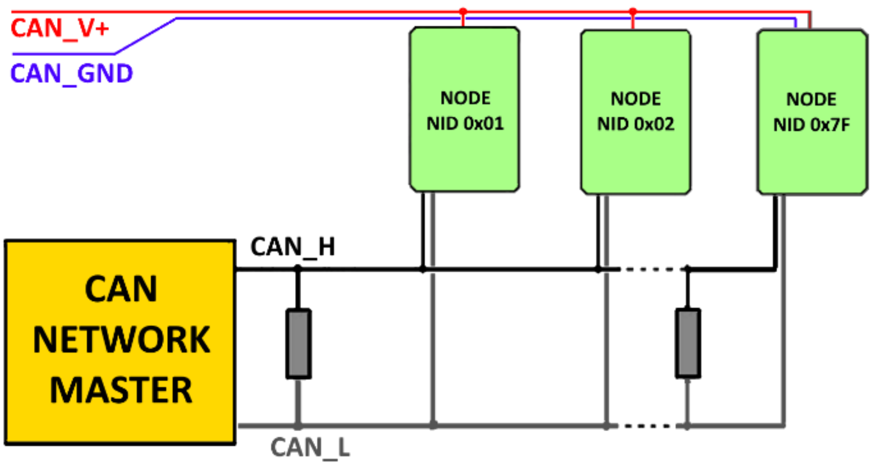
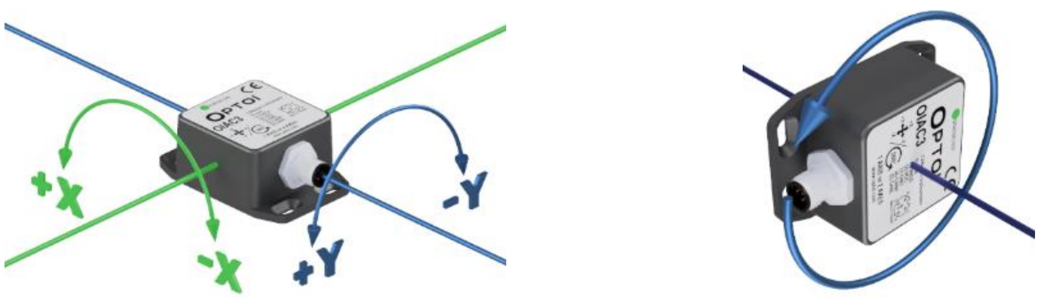


Figure 1 – CAN BUS network typical wiring scheme

2 OIAC2 AXES DEFINITION

The OIAC2 inclinometer is available in two-axes and single-axis versions.



2 AXES

1 AXIS

Table 1: OIAC2 inclinometers axis orientation for two-axes and single-axis variants

3 CANOPEN GENERAL INFORMATION

CANopen (EN 50325-4) is a high-level CAN-based communication protocol, developed for embedded networking applications (e.g.: in-vehicle networks). CANopen comprises and standardizes many aspects of a communication protocol like network programming framework, device descriptions, interface definitions and application profiles, enabling different devices and applications from different manufacturers to communicate.
OIAC2 inclinometer meets CiA 301 v.4.2.0 requirements for a general-purpose CANopen network node. Device profile DSP-410 specifies OIAC2 device as single or double-axes inclination sensors with 16-bit data resolution.

4 SUPPORTED COMMUNICATION OBJECTS

OIAC2 only supports CAN frames with 11-bit node identifiers. The first 7-bit (less significant) of the 11-bit communication object identifier (COB-ID) specifies the node identifier value (NID) while the remaining 4-bit (the most significant) specifies the function code. During the initialization phase, all the different COB-ID identifiers values are computed starting from the dictionary object values.

Communication object	Direction[1]	COB - ID		Object description
NMT	RX	00h		NMT services
SYNC	RX	80h		Sync object
EMCY	TX	80h	+ NID	Emergency object
TPDO1	TX	180h	+ NID	Measured angles values
SDO (Client→Server)	RX	600h	+ NID	Access to a node object dictionary
SDO (Server→Client)	TX	580h	+ NID	Node reply to SDO request
Boot Up \ Heartbeat	TX	700h	+ NID	Boot Up and Heartbeat messages

Table 2 – supported communication object and communication object identifiers

5 BOOT-UP FRAME

The OIAC2's initialization process starts as soon as the device is powered on. This process takes around 100 milliseconds to finish. As soon as initialization is complete, the device sends a boot-up frame (see Table 3) to advise it has entered the pre-operational state. Then the OIAC2 starts measuring the angle, letting the user to transmit and receive messages over the CANopen network.

COB-ID	Byte 0
700h + NID	00h

Table 3 –Boot-Up message frame

[1] *Direction is considered from the point of view of OIAC2. CANopen SDO standard communication protocol defines the device on which the object dictionary table resides as master device.*

6 NMT FINITE STATE MACHINE

A CANopen device's behavior depends on the state of a *finite state machine* (see Figure 2). Each state defines the node's behavior, the valid communication objects and the possible actions.

To change the active state of an NMT finite state machine, you need to use an NMT command frame (refer to Table 4), that contains information about the new state. The first data byte of the NMT command indicates whether the command applies to all nodes in the network or just one specific node. If the recipient node address in the second data byte (byte 1) is 00h, the NMT command frame is sent to all nodes in the network, otherwise the second data byte specifies the address of the target node for the command.

When a device is in the *pre-operational* state, it can start sending heartbeat messages, if this service is configured, but PDO communication is not available. To configure or read data from a device in the *pre-operational* state, you must use the SDO protocol (see section 0).

Only nodes in the *operational* state support all available communication objects, including PDO, SDO, and heartbeat messages. Note that PDO communication objects are only available in the *operational* state. Devices in the *stopped* state only process NMT commands. A *stopped* device indicates its current NMT state by supporting the error control protocol (heartbeat).

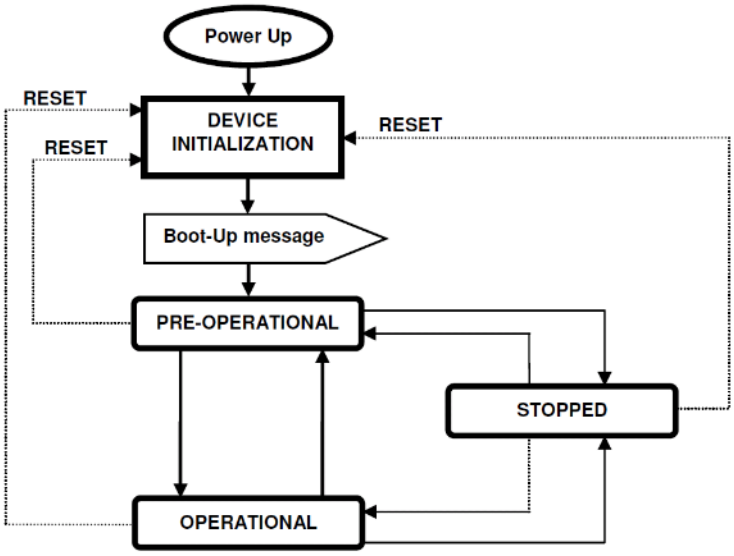


Figure 2 – NMT state machine of CANopen nodes

The various NMT states are entered only after the reception of an NMT command frame. This type contains a command code field, that specifies the NMT state to enter. Below are the reported NMT frame structure (Table 4) and the valid command code values (Table 5).

COB ID (11bit)	Byte0	Byte1	
000h	COMMAND CODE (see Table 5)	00h	broadcast NMT command to all the nodes of the network
		NID	send NMT command to a specific node

Table 4 – NMT command frame organization

COMMAND CODE	EFFECT ON NMT STATE MACHINE
01h	Go to OPERATIONAL
02h	Go to STOPPED
80h	Go to PRE – OPERATIONAL
81h	Go to RESET NODE
82h	Go to RESET COMMUNICATIONS

Table 5 – NMT command code values list

7 PDO - PROCESS DATA OBJECT

Messages that consist solely of process data (e.g.:measured tilt angles, temperature,…) are referred to as *Process Data Objects* (PDO). These message frames are ideal for the cyclical exchange of process data. PDO message frames eliminate the additional fields found in *Service Data Objects* (SDO) messages, such as index, subindex, and data length, allowing the CAN message to contain only the necessary data. PDO messages can only be transmitted when the NMT state machine is in the *operational* state (refer to section 6). The OIAC2 supports one *Transmit Process Data Object* (TPDO) number 1 frame with a COB-ID of 0x180 + NID. The length and data format of the TPDO1 frame differ between dual-axes and single-axis inclinometers.

7.1 Dual-axes inclinometers TPDO1 frame

COB ID	Byte0	Byte1	Byte2	Byte3	Byte4
0x180 + NID	Longitudinal inclination (LSB first)		Lateral inclination (LSB first)	Internal temperature	

Table 6 – default TPDO1 frame structure of a dual-axis OIAC2 inclinometer (axis swap OFF[2])

Measured inclination values can be found at index 6010h (see par.13.2) for the X-axis (longitudinal) and at index 6020h (see par.13.6) for the Y-axis (lateral). The angle values are represented as 16-bit two's complement (signed) fixed-point numbers. The finest resolution (see par.13.1) is limited to one-hundredth of a degree (0.01 degrees). The internal temperature is at index 5000h (see par.12.9). The temperature's value is in Celsius degrees represented by a signed 8-bit value in two's complement.

7.2 Single-axis inclinometers TPDO1 frame

COB ID	Byte0	Byte1	Byte2
0x180 + NID	Inclination (LSB first)	Internal temperature	

Table 7 – TPDO1 frame structure for single-axis mode

Measured inclination value can be found at index 6010h (see par.13.2). These values are represented as a 16-bit signed fixed-point number in two's complement (signed) format. The finest resolution (see par.13.1) is limited to one-hundredth of a degree (0.01 degrees). The internal

[2] *Axis swap options permits to swap the mapped axis in the TPDO1 frame. When turned ON, Byte 0 and Byte 1 contains the measured lateral inclination value while Byte 2 and Byte 3 contains the measured longitudinal inclination value (see par. 12.6).*

temperature is at index 5000h (see par.12.9). The temperature's value is in Celsius degrees represented by a signed 8-bit value in two's complement.

7.3 How to read the transmitted angles

Let's consider a dual-axes OIAC2 configured with a resolution object value of 10 (see paragraph 13.1). This means a base angle resolution of 0.001 degrees (defined in the standard device profile) multiplied by 10, yielding an angle resolution of 0.01 degrees.
The two 16-bit signed integers transmitted with the TPDO1 frame are as follows:

1. X-axis measured angle is equal to F3B1h, which corresponds to -3151 in decimal ;
2. Y-axis measured angle is equal to 0493h, which corresponds to +1171 in decimal.

When applying the computed angle resolution to these values, the measured angles are as follows:

3. X-axis (longitudinal): -3151 * 0.01 = -31.51 degrees
4. Y-axis (lateral): +1171 * 0.01 = +11.71 degrees

7.4 TPDO transmission types

The scheduling of *Transmit Process Data Objects* (TPDOs) is based on the transmission type sub-index (02h) of the TPDO communication parameters objects at index 1800h (see par.11.15). The table below lists all the possible TPDO transmission types for a generic CANopen device and indicates which of these are available for the OIAC2 inclinometer.

Values	Description	Supported
00h	Acyclic synchronous: triggered if SYNC is received and one of the mapped process values has changed after the last transmission	NO
01h…F0h	Cyclic synchronous: triggered when the number of received SYNC matches the values of the transmission type object[3]	**YES**
F1h…FBh	Reserved	-
FCh	Synchronous RTR only: not recommended	NO
FDh	Asynchronous RTR only: not recommended	NO
FEh	Asynchronous: triggered by one or more manufacturer defined internal event. e.g.: the TPDO1 event timer expire (see par.11.15)	**YES**
FFh	Asynchronous: triggered by one or more internal events defined in the device profile and the application profile	NO

Table 8 - PDO transmission types supported by OIAC2 inclinometers

7.5 Supported TPDO transmission types

TPDO frames are only transmitted when a CANopen node is in NMT operational state.

7.5.1 Cyclic synchronous transmission

COB ID
80h

Table 9 - SYNC frame

[3] *e.g.: with transmission type sub-index equal to 01h the TPDO is cyclic every 1 received SYNC; with 02h the TPDO is cyclic every 2 received SYNC; …; with F0h the TPDO is cyclic every 240 received SYNC*

The transmission of the PDO can be synchronized with the reception of the SYNC message. The SYNC message plays a crucial role in coordinating the transmission of TPDO frames across the CANopen network. This synchronization protocol is typically managed by a SYNC producer, which is usually the network master, although other nodes within the network can also serve this function. The SYNC frame corresponds to a single CAN frame, that does not contain any additional data (see Table 9). The value of the SYNC object COB-ID is determined by the object located at index 1005h (see.par.11.5), in the object dictionary. Additionally, it is possible to define multiple SYNC objects COB-ID values for different nodes within a CANopen network to synchronize communication effectively.

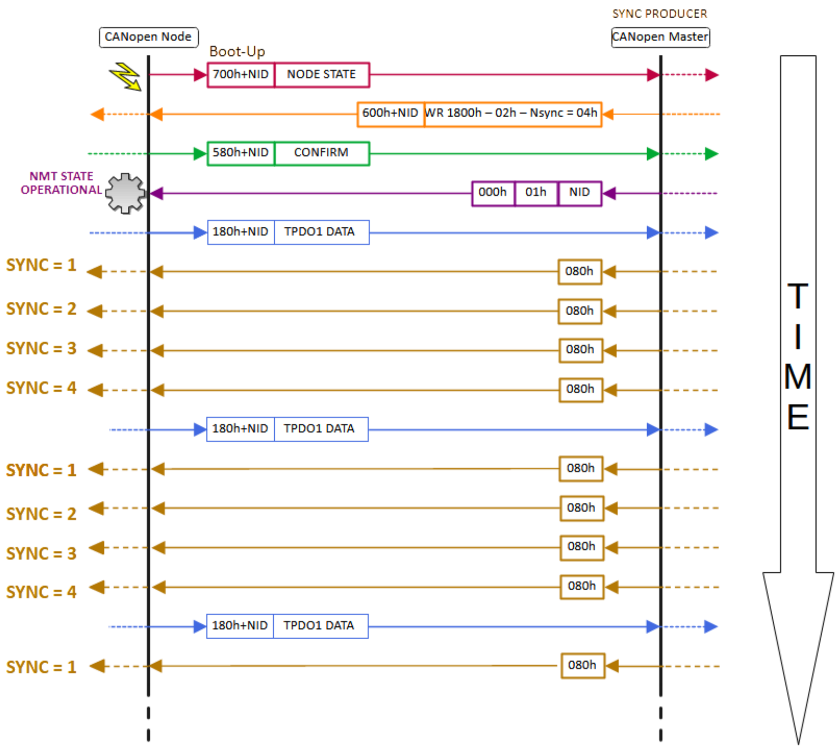


Figure 3: behavior of a SYNC based TPDO1 transmission

7.5.2 Manufacturer event driven asynchronous transmission

When the event-driven asynchronous mode is selected, the TPDO1 is transmitted after a specific internal trigger event occurs. This may happens, for example, when the temperature exceeds a certain limit, or when the event timer elapses, or when the measured angle changes.
To enable the manufacturer-specified event-driven transmission type, set sub-index (02h) of the TPDO communication parameters objects at index 1800h (see par.11.15) equal to 254 (FEh). For the OIAC2 the unique asyncronous trigger for the TPDO1 transmission is the event timer.

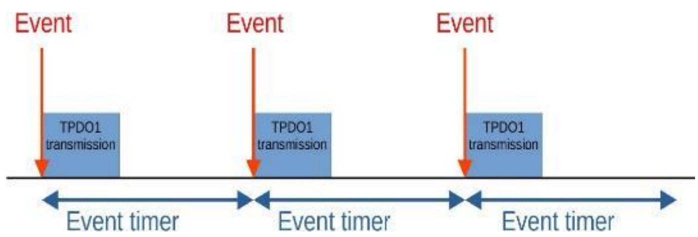


Figure 4 - event timer mechanism for cyclical TPDO1 transmission

The event timer period is defined by the value in the sub-index (05h) of the TPDO communication parameters objects at index 1800h. If the event timer period is different from zero, the TPDO1 frame will be transmitted cyclically each time the event timer elapses the period value. Conversely, if the event timer value is set to zero, periodic transmission is disabled, and no data will be sent.

7.5.3 RTR: remote transmission request

The RTR (Remote Transmission Requests) frame is a request that the NMT master sends directly to a specific CANbus node. The OIAC2 implements RTRs for TPDO1 transmissions and for Lifeguarding/Nodeguarding (see paragraph 10.2).
Upon receiving an RTR, if the OIAC2 is in operational mode, it responds with the object that corresponds to the requested RTR COB-ID. For a TPDO1 request, the COB-ID sent by the master is as follows:

COB ID
180h + Node ID

Table 10 – RTR frame structure for a TPDO1 transmission

**** CAUTION ****

RTRs are the least recommended type of transmission in a CAN system.

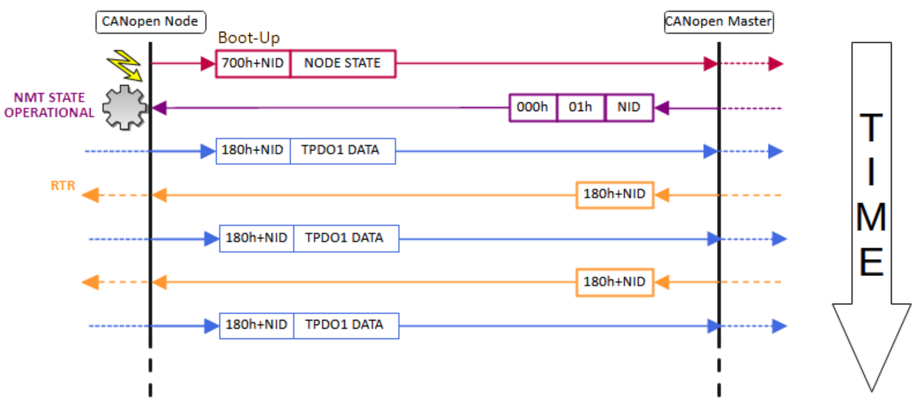


Figure 5: RTR frame reception based TPDO1 transmission mechanism

8 SDO SERVICE DATA OBJECTS

SDO (Service Data Objects) allow reading and writing access to all the registers in the object dictionary table of a CANopen device. Each SDO requires two CAN frames with different identifiers to complete the process. The SDO begins with a request frame (COB-ID = 600h + NID) and concludes with a response frame (COB-ID = 580h + NID), confirming the service.
This process initiates peer-to-peer communication between a server and a client device. The device that owns the object dictionary functions as the SDO server, while the accessing device acts as the SDO client. Under this framework, reading from the object dictionary is referred to a SDO upload, that means server to client communication. Conversely, writing to an object dictionary entry is known as an SDO download, where data is sent from the client back to the server.

The COB-ID for both the request frame (client to server) and the response frame (server to client) can be found at index 1200h in the object dictionary.

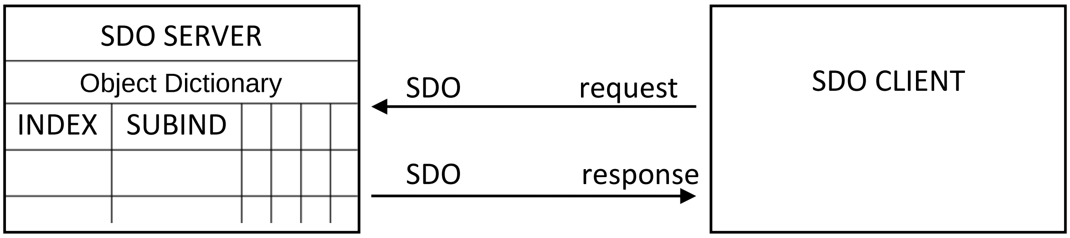


Figure 6 – SDO general client-server behavior scheme

8.1 SDO READ of an OD objet value

A read operation for a devices object dictionary entry starts with a SDO upload request frame. The frame first data byte (byte 0) indicates the data type of the entry to be read from the dictionary.

COB ID	Byte0	Byte1	Byte2	Byte3
600h + NID	RD data type[4] (Table 12)	Index Low (LSB)	Index High (MSB)	Sub-Index

Table 11 – organization of an SDO upload request frame.

OD entry type	RD data type
Read any length	40h
1 byte	4Fh
2 bytes	4Bh
4 bytes	43h
STRING	41h

Table 12 – possible RD data type values for SDO upload request frame.

A successful reading operation concludes with the client receiving a SDO response frame that the server device transmits in reply to the upload request. The value returned in the response can be an 8-bit, 16-bit, or 32-bit number or 1 to 4 ASCII characters for string-type entries. This data fits within the frame's value field. If the returned value is less than 32 bits, the unused bytes can be omitted, resulting in a shorter upload response frame. Additionally, 16-bit and 32-bit values are returned starting from the least significant byte (LSB) to the most significant byte (MSB).

COB ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
580h + NID	Data type (Table 12)	Index Low (LSB)	Index High (MSB)	Sub-Index	Entry Value (LSB to MSB)			

Table 13 – organization of an SDO upload response data frame.

[4] *If the entry data type isn't known it's* possible to use the "Read any length" (40h) data type in the SDO upload frames. Devices will return the value data type in the upload request frame.

** EXAMPLE ** Let's consider an OIAC2 inclinometer with a node identifier (NID) of 09h and a TPDO1 event timer set to 1.25 seconds. The TPDO1 event timer value can be found in the sub-index 05h of the TPDO1 parameters object at index 1800h (see par.11.15). The COB-ID for the SDO upload request frame is calculated as 600h plus the NID, resulting in 609h (as outlined in paragraph 0). The frame that needs to be sent to the OIAC2 is:

COB ID	Byte 0	Byte 1	Byte 2	Byte 3
609h	40h	00h	18h	05h

The OIAC2 answers with the following SDO response frame, which contains the event timer value:

COB ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
589h	4Bh	00h	18h	05h	E2h	04h

The SDO response frame has a COB-ID of 580h + NID = 509h. The data type (byte 0) is 4Bh, indicating that the read value is 2 bytes long (see Table 12). The requested index and sub-index values are included in the SDO response frame. The TPDO1 event timer value is found in byte 4 and byte 5. Following the LSB first convention, the value is equal to 4E2h, which translates to 1250 milliseconds.

8.2 SDO WRITE to an OD objet

A write operation to a device object dictionary entry starts with an SDO download request frame.

COB ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
600h + NID	WR data type[5] (Table 15)	Index Low (LSB)	Index High (MSB)	Sub-Index	Write Value (LSB to MSB)			

Table 14 – organization of an SDO download request frame.

The first data byte (byte 0) indicates the data type to be written. The value can be an 8-bit, 16-bit, or 32-bit number, fitting into the value field (bytes 4 to 7), and must match the object dictionary entry type (see Table 20). If the value is less than 32 bits, transmitting unused bytes is unnecessary, making the frame shorter. For 16-bit and 32-bit values, transmit from the least significant byte (LSB) to the most significant byte (MSB).

OD entry type	WR data type
Write any length	22h
1 byte	2Fh
2 bytes	2Bh
4 bytes	23h

Table 15 – possible WR data type values for SDO download request frame

An error-free write operation is confirmed by a SDO response frame, where byte 0 equals 60h, indicating successful writing of the new value. The index and sub-index values are repeated in the response, and bytes 4 to 7 are all set to zero.

[5] If the entry data type isn't known it's *possible to use the "Write any length" (22h) data type in the SDO download frames. Devices will check if the value in the download request frame is compatible with the indexed entry.*

COB ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
580h + NID	60h	Index Low (LSB)	Index High (MSB)	Sub-Index	00h	00h	00h	00h

Table 16 – organization of an SDO download response data frame.

** EXAMPLE ** It is requested to set an OIAC2 TPDO1 event timer to 550 milliseconds. The TPDO1 event timer value is contained at the sub-index 05h of the TPDO1 parameters object at index 1800h (see par.11.15). The correct frame to be sent is:

COB ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5
615h	2Bh	00h	18h	05h	26h	02h

If the writing procedure is successful, the new event timer value will be written correctly and the OIAC2 will answer with the following SDO response frame:

COB ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5		
595h	60h	00h	18h	05h	00h	00h	00h	00h

8.3 SDO Errors codes

When an error occurs while servicing an SDO request frame, CANopen device answers with an SDO abort frame. The first byte of the SDO abort frame (byte 0) is set to 80h to indicate that an error condition has arisen during the processing of the request frame. The index value and the sub-index value are included in the SDO abort frame to specify which entry has encountered the error. Additionally, the value field in the SDO abort frame contains a 4-byte error code, that identifies the reason for the error. This 4-byte code, known as the SDO abort code (refer to Table 18), is written in the data field of the SDO abort frame.

COB ID	Byte0	Byte1	Byte2	Byte3	Byte4	Byte5	Byte6	Byte7
580h + NID	80h	Index Low (LSB)	Index High (MSB)	Sub-Index	SDO abort code (Table 18) (LSB to MSB)			

Table 17 – organization of an SDO download request frame.

SDO abort code	Bytes 4 to 7	Error
0504 0001h	01h; 00h; 04h; 05h	SDO command not valid or unknown
0601 0002h	02h; 00h; 01h; 06h	Attempt to write a read-only object
0602 0000h	00h; 00h; 02h; 06h	Object does not exist in the object dictionary
0604 0043h	43h; 00h; 04h; 06h	General incompatibility reason
0607 0010h	10h; 00h; 07h; 06h	Data type does not match, length of service does not match
0609 0011h	11h; 00h; 09h; 06h	Sub-index does not exist
0609 0030h	30h; 00h; 09h; 06h	Parameter invalid value (download only)

SDO abort code	Bytes 4 to 7	Error
0609 0031h	31h; 00h; 09h; 06h	Value of written parameter too high
0609 0032h	32h; 00h; 09h; 06h	Value of written parameter too low
0800 0000h	00h; 00h; 00h; 08h	General Error
0800 0020h	20h; 00h; 00h; 08h	Data cannot be transferred or stored to/in the application
0800 0024h	24h; 00h; 00h; 08h	No data available

Table 18 – SDO abort codes list for SDO abort frames

9 OBJECT DICTIONARY

In a CANopen device's object dictionary all control registers and data entries are organized and listed. These entries link the application layer (such as tilt measurement) with the communication objects used to interact with the device (e.g.: SDO, TPDO, SYNC, EMCY,…). Accessing the object dictionary for reading or writing is accomplished using the SDO protocol (see par.8).

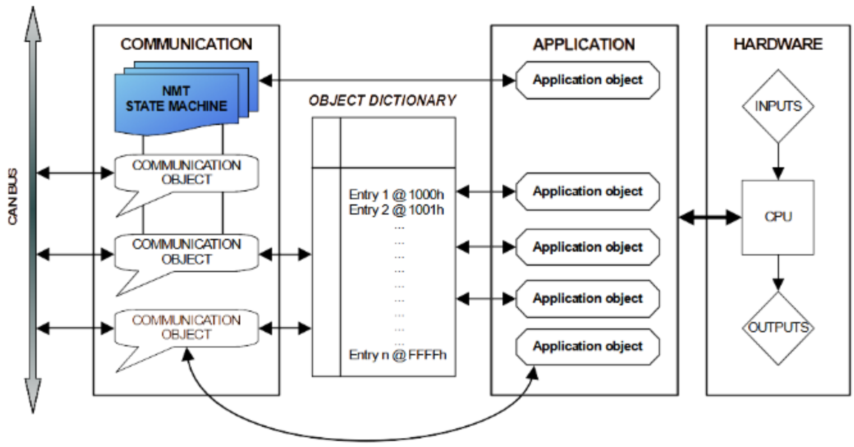


Figure 7 – CANopen general device model: object dictionary table as the interface between communication objects and application objects.

The object dictionary table of all the CANopen devices is organized in multiple sections, based on the function of the contained entry. The objects within the dictionary have a 16bit index to be addressed to read or write.

Index	Object
0000h	Not used
0001h – 001Fh	Static data types
0020h – 003Fh	Complex data types
0040h – 005Fh	Manuf. specific complex data types
0060h – 007Fh	Dev. Prof. specific static data types

0080h – 009Fh	Dev. Prof. specific complex data types
00A0h – 0FFFh	Reserved for further use
1000h – 1FFFh	Communication objects
2000h – 5FFFh	Manufacturer specific objects
6000h – 9FFFh	Standard device profile objects
A000h – BFFFh	Standard interface profile objects
C000h – FFFFh	Reserved for further use

Table 19 – complete list of the different sections for a generic CANopen object dictionary (some sections may not be implemented in the OIAC2)

9.1 OIAC2 object dictionary

Index	Sub Index	Parameter description	Data Type	Access	Default value	Range	Store
1000h	00h	Device type	U32	RO	2019Ah	2019Ah 1019Ah	
1001h	00h	Error register	U8	RO	0		
1002h	00h	Manufacturer status register	U32	RO	0		
1003h	Pre-defined error field						
	00h	Number of errors	U8	RW	0		
	01h	New error code [i]	U32	RO			
	02h	Error code [i-1]	U32	RO			
	03h	Error code [i-2]	U32	RO			
	04h	Error code [i-3]	U32	RO			
	05h	Oldest Error code [i-4]	U32	RO			
1005h	00h	SYNC message COB-ID [1]	U32	RW	80h	1...2047	YES
100Ah	00h	Manufacturer firmware version	U32	const.			
100Ch	00h	Guard time [multiple of 1ms]	U16	RW	0	0...65535	YES
100Dh	00h	Life time factor	U8	RW	0	0...255	YES
1010h	Store parameters						
	00h	Largest supported sub-index	U8	RO	1		
	01h	Save ALL parameters (“save” = 73617665h)	U32	RW	1		
1011h	Restore default parameters						
	00h	Largest supported sub-index	U8	RO	1		
	01h	Reload ALL default parameters ("load" = 6C6F6164h)	U32	RW	1		
1014h	00h	EMCY COB-ID	U32	RO	80h+NID		
1017h	00h	Producer Heartbeat time [multiple of 1ms, 0 = disabled]	U16	RW	0	0...65535	YES

Index	Sub Index	Parameter description	Data Type	Access	Default value	Range	Store
1018h	Identity Object						
	00h	Largest supported sub-index	U8	RO	4		
	01h	Vendor ID	U32	RO	3C4h		
	02h	Product code	U32	RO	414332h	“AC2”	
	03h	Revision number	U32	RO	30h	“0”	
	04h	Serial number	U32	RO	{dev.dep.}		
1200h	Server SDO1 parameters						
	00h	Largest supported sub-index	U8	RO	2		
	01h	COB-ID Client > Server	U32	RO	600h+NID		
	02h	COB-ID Server > Client	U32	RO	580h+NID		
1800h	TPDO1 communication parameters						
	00h	Largest supported sub-index	U8	RO	5		
	01h	COB ID	U32	RO	180h+NID		
	02h	Transmission type [1] [synchronous, manufacturer specific]	U8	RW	FEh	0…240 or 254	YES
	04h	Reserved	U8	RW	0		
	05h	Event timer for cyclical transmission [multiple of 1ms, 0 = disabled]	U16	RW	0	0...65535	YES
1A00h	TPDO1 mapping parameters (fixed mapping)						
	00h	Largest supported sub-index	U8	RO	2		
	01h	Inclination value X-axis parameters	U32	RO	60100010h		
	02h	Inclination value Y-axis parameters	U32	RO	60200010h		
	03h	Device internal temperature [°C]	U32	RO	50000008h		
2000h	00h	Node ID [1]	U8	RW	09h	1...127	YES
2001h	00h	Bit Rate [1]	U8	RW	1	[0] 125kbps [1] 250kbps [2] 500kbps [3] 1000kbps	YES
2002h	00h	Restore default NID value [2] ("load" = 6C6F6164h)	U32	RW	1		
3000h	00h	Digital Filter	U16	RW	4	0...9	YES
3002h	00h	Single axis data format [3] [0] for 0-360°; [1]:for ±180°	U8	RW	0	1 or 0	YES
3003h	00h	Axis Swap in TPDO1 [1] [4] [1] = ON; [0] = OFF	U8	RW	0	1 or 0	YES
4000h	Measure Range						
	00h	Largest supported sub-index	U8	RO	2		
	01h	X range [deg]	U16	RW	75 - 2ax 360 - 1ax	(see.par.12.7)	YES
	02h	Yrange [deg] [4]	U16	RW	75	1…90	YES
4001h	00h	Sensor operational mode [1] 2 Axes or 1 Axis	U8	RO	2	1 or 2	
5000h	00h	Device Internal temperature [°C]	S8	RO			

Index	Sub Index	Parameter description	Data Type	Access	Default value	Range	Store
5002h	00h	Boot delay [1] [multiple of 10ms; 0 = disabled]	U16	RW	10	0…3000	YES
5003h	00h	CANBUS error control register	U8	RW	0	0…7	YES
5004h	00h	Auto Operational [1] [1] = ON; [0] = OFF	U8	RW	0	1 or 0	YES
5005h	00h	Accelerometer X axis value unfiltered, 1kS/sec	U16	RO		0…4096	
5006h	00h	Accelerometer Y axis value unfiltered, 1kS/sec	U16	RO		0…4096	
5544h	Factory reserved						
5555h	Factory reserved						
6000h	00h	Resolution [multiple of 0.001°]	U16	RW	10	10,100,1000	YES
6010h	00h	Measured X axis value	S16	RO			
6011h	00h	Inversion of X axis range [1] = ON; [0] = OFF	U8	RW	0	1 or 0	YES
6012h	00h	X axis preset value	S16	RW	0	-Rng…+Rng	
6013h	00h	X axis offset value	S16	RW	0	-Rng…+Rng	YES
6020h	00h	Measured Y axis value [4]	S16	RO			
6021h	00h	Inversion of Y axis range [4] [1] = ON; [0] = OFF	U8	RW	0	1 or 0	YES
6022h	00h	Y axis preset value [4]	S16	RW	0	-Rng…+Rng	
6023h	00h	Y axis offset value [4]	S16	RW	0	-Rng…+Rng	YES

Table 20 – OIAC2 inclinometers object dictionary

[1] To make this parameter effective, the configuration must be saved using the *save all* command. After that, the device must be restarted. During the device initialization phase, the configuration will be updated according to the new value of the parameter.

[2] To make this command effective the device must be restarted. During the device initialization phase, the configuration will be updated based on the new value of the parameter.

[3] This register is applicable only if the inclinometer is a single-axis device (indicated by the device type register at index 1000h, sub-index 00h, which equals 1019Ah).

[4] This register is applicable only if the inclinometer is a dual-axis device (indicated by the device type register at index 1000h, sub-index 00h, which equals 2019Ah).

9.2 Save the user configuration

To save the device configuration into permanent memory, the word *save*, represented in ASCII codes, must be written to the object at index 1010h (see par.11.9), sub-index 01h of the object dictionary. The string consists of 4 bytes, which equals 73617665h. Only after the *save all* command has been received and confirmed, the user configuration will be stored in permanent memory. From that point on, the user configuration will be retained even after a reset command or power loss.

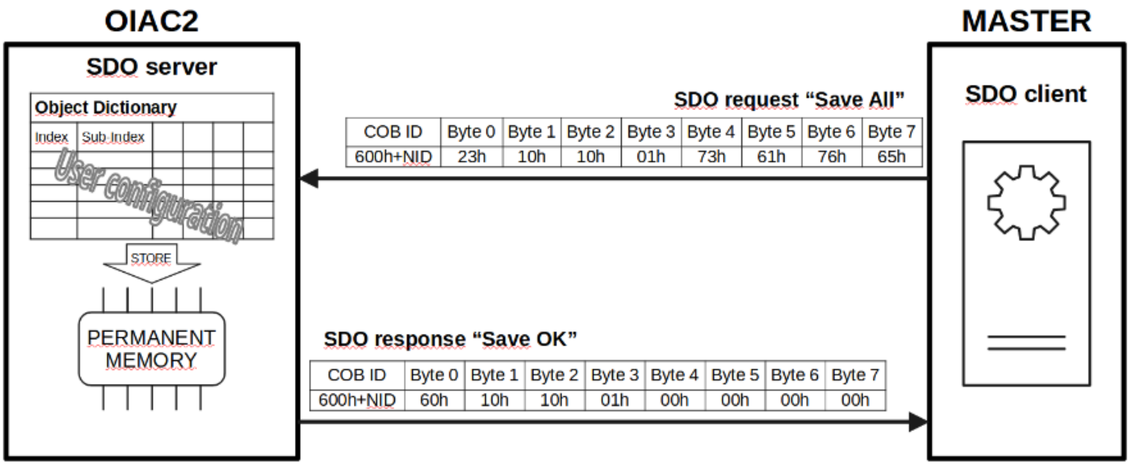


Figure 8 – SDO request and SDO response frame for saving the user configuration to an OIAC2 permanent memory.

For some modifications of the dictionary parameters a reset is required after giving the *save all* command, to make the new entry values fully effective. This is the case of the NID and the bit rate registers that cannot be changed during a SDO service execution. If these values change immediately, communication with the node will be lost.

9.3 Restore the default configuration

To restore the factory default configuration of the OIAC2 inclinometer, you will need to know the node identifier and the baud rate, which are available after receiving the boot-up frame. To perform the *restore* the word *load*, represented in ASCII codes, must be written into the object index 1011h (see par.11.10), sub-index 01h of the dictionary. The string value to be sent is 4 bytes long and corresponds to the hexadecimal value 6C6F6164h. After sending this command, you must wait the the OIAC2 confirmation *load all* command, before having effect. Then the factory default configuration will be restored in permanent memory.

** IMPORTANT ** The *load* command, that restores the factory default configuration does not affect the NID and the bit rate settings. To change these values, always use an SDO *download request* frame. The new values will only take effect after they have been stored in permanent memory, followed by a device reboot.

** IMPORTANT ** If the OIAC2 has custom configuration (such as resolution, event-timer, filter settings, etc.), these parameters will be reverted to the standard default configuration as outlined in the *default values* column of the object dictionary table (refer to section 9.1, Table 20).

10 NODE MONITORING

Heartbeat and other guarding mechanisms are methods used to monitor nodes in a CANopen network. These monitoring techniques are particularly useful for nodes that operate in an asynchronous, event-driven mode, as they require the nodes to regularly send their status to the master device. However, only one of these methods can be active at a time. If an attempt is made to activate both simultaneously, the guarding protocol will be deactivated and the heartbeat mechanism will take priority (heartbeat wins).

10.1 Heartbeat

The heartbeat protocol is a mechanism for monitoring nodes, which is managed by the nodes themselves. When a CANopen device has the heartbeat feature activated, it will cyclically send its NMT state (see par.6) transmitting a heartbeat message frame (refer to Table 21). If the master device does not receive the heartbeat message from a specific node before the heartbeat consumer time elapses, it will assume that the node has failed. This protocol is the least demanding in terms of bus load among the available node monitoring mechanisms. To enable the cyclic transmission of the heartbeat frame, you need to write a non-zero value to the object in the producer heartbeat interval time at index 1017h (see par.11.12). Writing zero will disable the heartbeat frame periodic transmission. The producer heartbeat interval time is in milliseconds and represents the transmission period for the heartbeat frame.

COB ID	Byte 0	Corresponding NMT state
700h + NID	00h	Boot up
	04h	Stop condition
	05h	Run condition
	7Fh	Pre-operational condition

Table 21 – Heartbeat frame structure and meaning of Byte 0

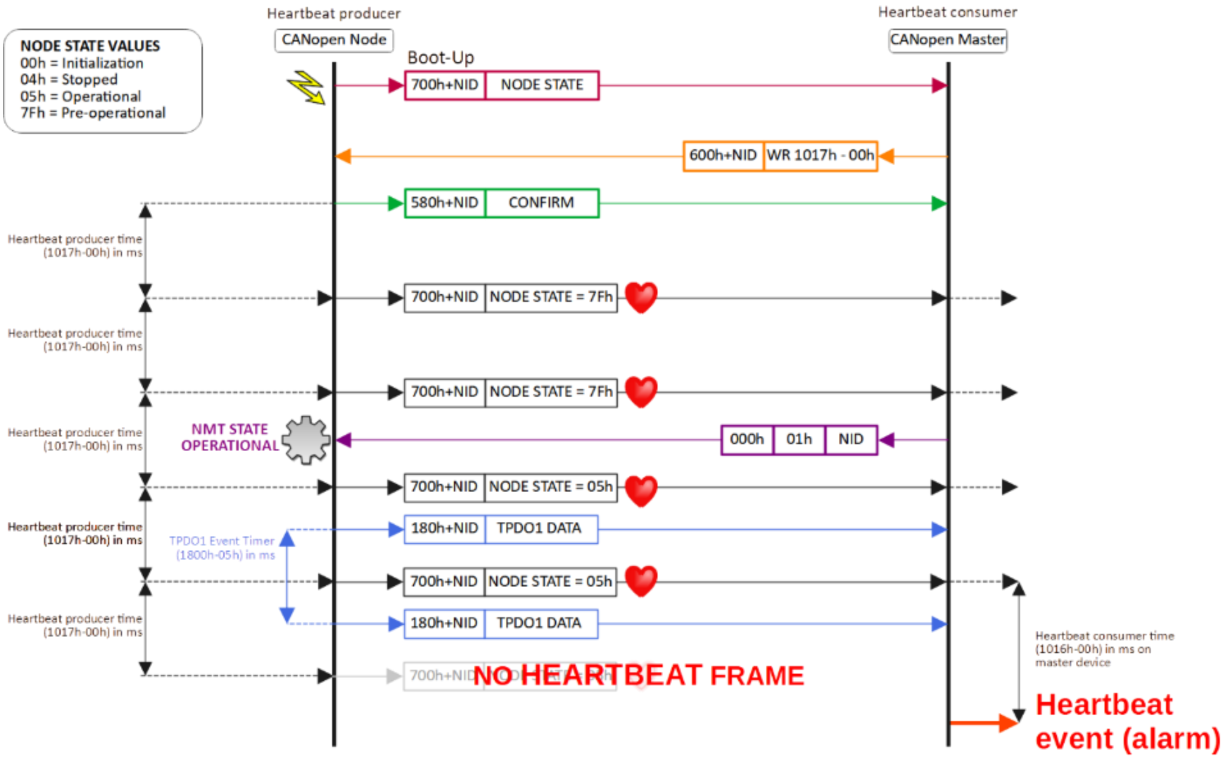


Figure 9 - Heartbeat protocol configuration ad utilization

10.2 Nodeguarding and Lifeguarding

The guarding mechanism utilizes a CANopen master device to monitor one or more nodes connected to the network. The master periodically sends *Remote Transmission Request* (RTR) frames to the guarding identifier of the node being monitored. The node answers sending its status along with a toggle bit. This toggle bit should change with every received remote request. If the toggle bit does not match the expected value from the master or if there is no response, the master interprets this as a potential malfunction of the slave node. Additionally, this guarding mechanism can be employed by slave nodes to detect failures in the master. When the master regularly transmits remote request frames with precise timing, the slave nodes can identify if the master has failed. To enable slave nodes to monitor potential master failures, two parameters in the object dictionary must be configured: the guard time and the lifetime factor. The guard time defines the interval for the master's remote frame transmissions, while the lifetime factor indicates the number of guard time intervals after which a slave node assumes that the master has failed. The total duration is referred to as the node lifetime and is calculated as follows:

NODE LIFETIME = Guard Time × Lifetime Factor

When a master failure is detected, the node immediately sends an *Emergency* (EMCY) frame and then returns to a *pre-operational* state.

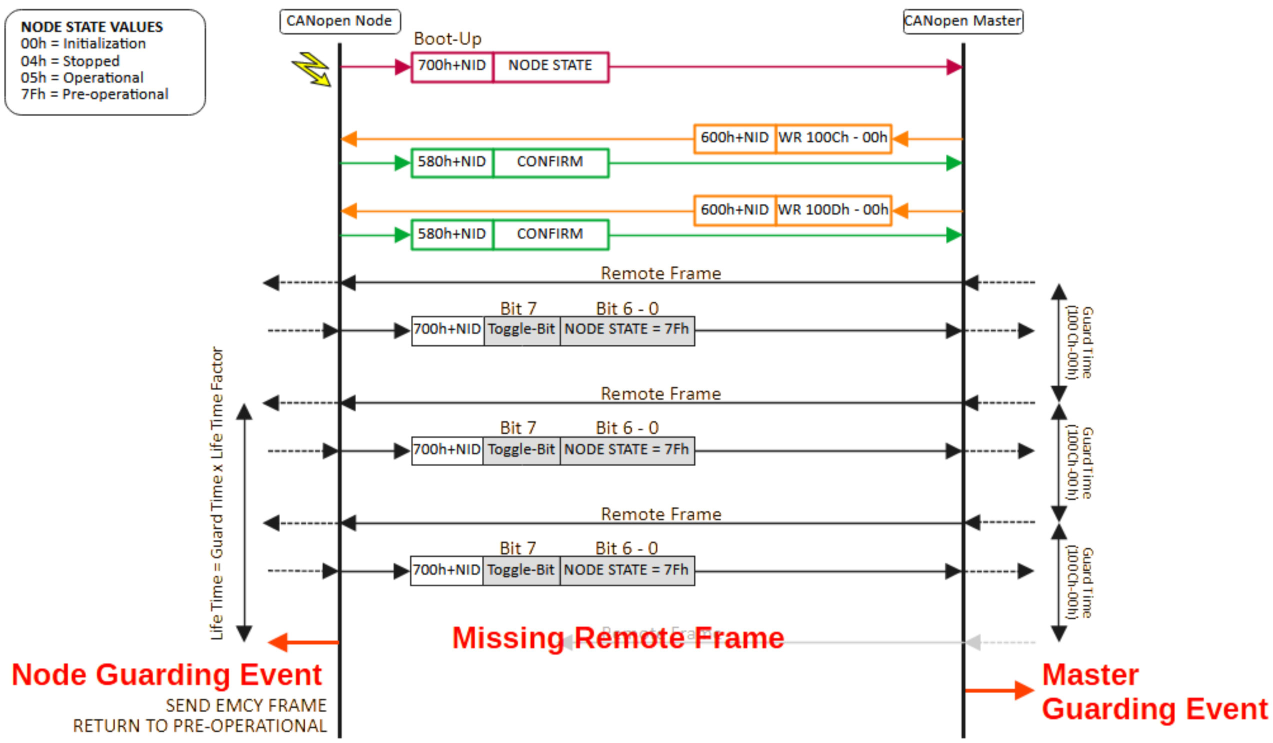


Figure 10 - Nodeguarding protocol configuration ad utilization

11 COMMUNICATION OBJECTS SPECIFICATIONS

11.1 1000h - Device type

Sub Index	00h
Data Type	Unsigned 32bit
Access	Read Only
Default value	1019Ah 2019Ah

According to the CiA device profile for inclinometers DSP-410 v.1.3.0, the device type entry (index 1000h, sub-index 00h) differs between dual-axes and single-axis OIAC2. For dual-axes the device type entry is equal to 2019Ah (two-axis with resolution max. 16-bit). For the single-axis the device type entry is equal to 1019Ah (one-axis with resolution max. 16-bit).

11.2 1001h - Error register

Sub Index	00h
Data Type	Unsigned 8bit
Access	Read Only
Default value	0

This object includes several flag bits, that activate to indicate potential errors related to working conditions, communications, or internal status. The register can only be accessed in a read-only mode and is part of the emergency objects (refer to section 14, EMCY). The default value for this object is 00h, which signifies that no errors have been detected.

ERROR REGISTER							
Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
Manufacturer Specific	Not Used (always 0)	Profile Specific Error	Communication Error	Temperature Error	Not Used (always 0)		At least one active error

Table 22 - Structure of the error register and the meaning of individual error bits.

11.3 1002h - Manufacturer status register

Sub Index	00h
Data Type	Unsigned 32bit
Access	Read Only
Default value	0

The manufacturer status register displays the current state of all detectable errors. Each flag bit within this register corresponds to a specific error, that can be checked on the communication interface or the device's operating conditions.

The manufacturer status register is read-only and the two least significant bytes (bits 0 to 15) are included in emergency objects (EMCY, see section 14). The default value for this object is 00h, indicating that no errors have been detected.

MANUFACTURER ERROR REGISTER																
Bit 31…16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Not Used	COMMUNICATION ERROR FIELD								DEVICE ERROR FIELD							

Table 23 - Structure of the manufacturer error register and the meaning of individual bits.

COMMUNICATION ERROR FIELD							
Bit 15	14	13	12	11	10	9	8
Guarding Error	Not Used				Receive queue overrun	CAN BUS-OFF state reached	CAN WARNING limit reached

Table 24 - Structure of the communication error field and the meaning of individual bits.

DEVICE ERROR FIELD							
Bit 7	6	5	4	3	2	1	0
Hardware	Not used		Memory	Temperature	Y - Axis	X - Axis	Not used

Table 25 – Structure of the device error field and the meaning of individual bits.

11.4 1003h - Pre-defined error field

Sub Index	00h
Data Type	Unsigned 8bit
Access	Read and Write
Default value	0
Range	00h – 05h
Write Effect	immediate

This object contains the error history for the last five errors that have occurred and been signaled via the emergency object (see par.14). Sub-index 00h shows the number of actual errors recorded in the predefined error field, ranging from the most recent error at sub-index 01h to the oldest error at sub-index 05h. Writing the value 00h to sub-index 00h will clear the error history and empty the array.

Sub Index	01h - 05h
Data Type	Unsigned 32bit
Access	Read only
Default value	0

If all entries in the array become full and a new error condition arises, the oldest error (previously at sub-index 05h) will be discarded. The entries from sub-indexes 01h to 04h will be shifted to the next higher sub-index (from 02h to 05h), and the new error will be recorded at sub-index 01h. If the error history is empty (indicated by a value of 00h in sub-index 00h) an SDO *abort* frame will be transmitted in response to an SDO *upload request* (read access) for sub-index 01h, with an *abort* code of 0800 0024h.

PRE-DEFINED ERROR FIELD STRUCTURE		
Bit 31…24	Bit 23…16	Bit 15…0
Additional Information Field		EMCY Error Code (see Table 38)
Communication Error Field (see Table 24)	Device Error Field (see Table 25)	

Table 26 - Structure of a single pre-defined error field entry and the meaning of individual bits.

11.5 1005h - SYNC message COB-ID

Sub Index	00h
Data Type	Unsigned 32bit
Access	Read and Write
Default value	80h
Range	001h – 7FFh
Write Effect	after save all + restart
NVM saved	after save all

This object specifies the valid COB-ID value for SYNC messages used within the SYNC protocol. The COB-ID must be a non-zero 11-bit CAN identifier. To accept SYNC messages with a new COB-ID value, the CANopen node must be restarted after saving the new configuration to permanent memory (see par.9.2). Note that the generation of SYNC messages is not supported, as bit 30 is always set to zero.

11.6 100Ah - Manufacturer firmware version

Sub Index	00h
Data Type	Unsigned 32bit
Access	Read Only

This object reports the software version in form of string, consisting of four characters.

	Byte 4	Byte 5	Byte 6	Byte 7
ASCII	33h	31h	33h	30h
Character	3	1	3	0

Table 27: Hot to understanding the organization and meaning of the firmware version register.

Considering the four bytes as an array of hexadecimal ASCII codes, the manufacturer's firmware version is displayed as “03.1.1” (the periods are fixed).

11.7 100Ch - Guard time [multiple of 1ms]

Sub Index	00h
Data Type	Unsigned 16bit
Access	Read and Write
Default value	0000h
Range	0000h – FFFFh
Write Effect	immediate
NVM saved	after save all

This object defines the guard time value, that is used along with the lifetime factor object (index 100Dh) for the lifeguarding/nodeguarding protocol (refer to par. 10.2). The guard time value is measured in milliseconds, with acceptable values ranging from 0 ms (guarding OFF) to 65,535 ms (maximum value).

IMPORTANT: The heartbeat protocol takes precedence over the lifeguarding/nodeguarding protocols. If both monitoring protocols are activated simultaneously, the lifeguarding/nodeguarding mechanism will be silently deactivated, and no EMCY or error frames will be transmitted.

11.8 100Dh - Life time factor

Sub Index	00h
Data Type	Unsigned 8bit
Access	Read and Write
Default value	00h
Range	00h – FFh
Write Effect	immediate
NVM saved	after save all

This object contains the lifetime factor value, which is used in conjunction with the guard time object (index 100Ch) for the lifeguarding/nodeguarding protocol (see par. 10.2). The lifetime is a dimensionless value and serves as the guard time multiplier to compute the node lifetime value. The acceptable range of values is from 0 (guarding OFF) to 255 (maximum value).

** IMPORTANT ** The heartbeat protocol takes precedence over the lifeguarding/nodeguarding protocols. If both monitoring protocols are activated simultaneously, the lifeguarding/nodeguarding mechanism will be silently deactivated, and no EMCY or error frames will be transmitted.

11.9 1010h - Store parameters

Sub Index	01h
Data Type	Unsigned 32bit
Access	Read and Write
Default value	01h
Range	“save”; 73h 61h 76h 65h
Write Effect	immediate

This object initiates the process of storing device configuration parameters in its permanent memory. A read access returns information about the device's saving capabilities. Each non-zero sub-index indicates whether it is possible to store a parameter group: a value of 1 means yes, and a value of 0 means no. Currently, only the *save all* parameters group (sub-index 01h) is available. To prevent accidental saving of configuration parameters, the permanent memory writing process begins only when a specific signature is written to sub-index 01h. This signature is "save," represented in hexadecimal as 65h 76h 61h 73h (refer to par. 9.2). Once the device receives the *save* signature, it stores the parameters in the permanent memory and confirms the success of the saving process with an SDO download *response* frame.

11.10 1011h - Restore default parameters

Sub Index	01h
Data Type	Unsigned 32bit
Access	Read and Write
Default value	01h
Range	“load” 6Ch 6Fh 61h 64h
Write Effect	immediate

This object allows to restore the device's default configuration, as specified in the *default values* column of the object dictionary table (refer to par.9.1, Table 20). When accessed for reading, it provides information about the device's restoring capabilities. Each non-zero sub-index indicates whether it is possible to restore the corresponding parameter group: a value of 1 means it can be restored, while a value of 0 means it cannot. Currently, only the *restore all* (sub-index 01h) parameter group is available. To prevent accidental restore of default configuration parameters, the permanent memory writing process is initiated only when a specific signature is written to sub-index 01h. This signature is the word *load*, represented as 64h 61h 6Fh 65h (refer to par.9.3). Upon receiving the *load* signature the device restores all parameters in its permanent memory to their default values and confirms the success of the process with an SDO download response frame. It is strongly recommended to reset the sensor by performing a power cycle after the default values have been restored. If an incorrect signature is written to index 1011h, sub-index 01h, the device will not restore any parameters and will send back a SDO abort frame containing the abort code: 0800 0020h.

** IMPORTANT ** The restore default factory configuration command does not affect the OIAC2 node identifier or the baud rate settings. Changes to the node identifier and baud rate must always be made using a SDO download request frame. To make the new values effective, they must first be stored in permanent memory, after which the device will reboot.

** IMPORTANT ** If the device has a custom configuration (e.g., resolution, event timer, filter, etc.), its parameters will be reverted to the standard default configuration, as indicated in the *default values* column of the object dictionary table (refer to par.9.1, Table 20).

11.11 1014h - Emergency object COB-ID

Sub Index	00h
Data Type	Unsigned 32bit
Access	Read only
Default value	80h + NID

This object defines the COB-ID for the device's emergency service communication object (see par. 14) in the CANopen network. The COB-ID for the EMCY object is read-only, and its value is fixed at 80h + NID.

11.12 1017h - Producer heartbeat interval time [ms]

Sub Index	00h
Data Type	Unsigned 16bit
Access	Read and Write
Default value	0000h
Range	0000h – FFFFh
Write Effect	Immediate
NVM saved	after save all

This object specifies the cycle time for transmitting heartbeat frames from the producer device (see par. 10.1). A value of zero disables the cyclical heartbeat frame transmission. The producer heartbeat interval must be a multiple of 1 ms.

11.13 1018h - Identity object

This object provides general information about the device. The sub-index 01h includes the vendor ID of the manufacturer, which is a unique identifier assigned to each manufacturer by CiA. Sub-index 02h contains the manufacturer-specific product code, which identifies the product family of the device. Sub-index 03h holds the manufacturer-specific revision number, indicating the major revision of the device. Finally, sub-index 04h displays the device's serial number.

Sub Index	01h – Vendor ID
Data Type	Unsigned 32bit
Access	Read Only
Default value	03C4h

Sub Index	02h – Product Code
Data Type	Unsigned 32bit
Access	Read Only
Default value	"AC2" - 41h 43h 32h

Sub Index	03h – Revision Number
Data Type	Unsigned 32bit
Access	Read Only
Default value	00h

Sub Index	04h – Serial Number
Data Type	Unsigned 32bit
Access	Read Only
Default value	{device serial number}

11.14 1200h - Server SDO1 parameters

This object includes the COB-ID parameters for the SDO objects where the device acts as the server. When the device NID value (located at index 2000h, sub-index 00h) is modified (refer to par. 12.1), the values of the two COB-ID fields will change after the next reset or power cycle of the device.

Sub Index	01h – Client → Server
Data Type	Unsigned 32bit
Access	Read Only
Default value	600h + NID

Sub Index	02h – Server → Client
Data Type	Unsigned 32bit
Access	Read Only
Default value	580h + NID

11.15 1800h - TPDO1 communication parameters

This object contains all the communication parameters for the TPDO1 message. Sub Index:

1. COB-ID used by TPDO1. Recalculated during the device initialization phase;
2. transmission type of the TPDO1 object (see par. 7.4);
3. not supported;
4. no functional purpose; it exists solely for compatibility reasons;
5. event timer value, expressed in milliseconds [ms] (see par.7.5.2).

Sub Index	01h – COB-ID
Data Type	Unsigned 32bit
Access	Read Only
Default value	180h + NID

Sub Index	02h – Transmission type
Data Type	Unsigned 8bit
Access	Read and Write
Default value	FEh
Range	01h - F0h; FEh
Write Effect	after save all + restart
NVM saved	after save all

Sub Index	05h – Event Timer
Data Type	Unsigned 16bit
Access	Read and Write
Default value	00h
Range	0000h – FFFFh
Write Effect	immediate
NVM saved	after save all

11.16 1A00h - TPDO1 mapping parameters

This section describes the mapping parameters for TPDO1. Sub-index 00h indicates the total number of entries in the mapping record. Each entry corresponds to an application variable, that is transmitted within the TPDO1 frame. Sub-indexes 01h through the total number of mapped application variables in the TPDO1 frame contain the object index from the object dictionary, the associated sub-index, and the data length (refer toTable 28). The data length field within a mapping entry specifies the number of bits in the variable's data type.

Sub Index	01h – TPDO1 entry 1
Data Type	Unsigned 32bit
Access	Read Only
Default value	60100010h

Sub Index	02h – TPDO1 entry 2
Data Type	Unsigned 32bit
Access	Read Only
Default value	60200010h

Sub Index	03h – TPDO1 entry 3
Data Type	Unsigned 32bit
Access	Read Only
Default value	50000008h

Bit 31…16	Bit 15…8	Bit 7…8
OD Index [16bit]	Sub-Index [8bit]	Data length in bit [8bit]

Table 28 – Mapping entry organization

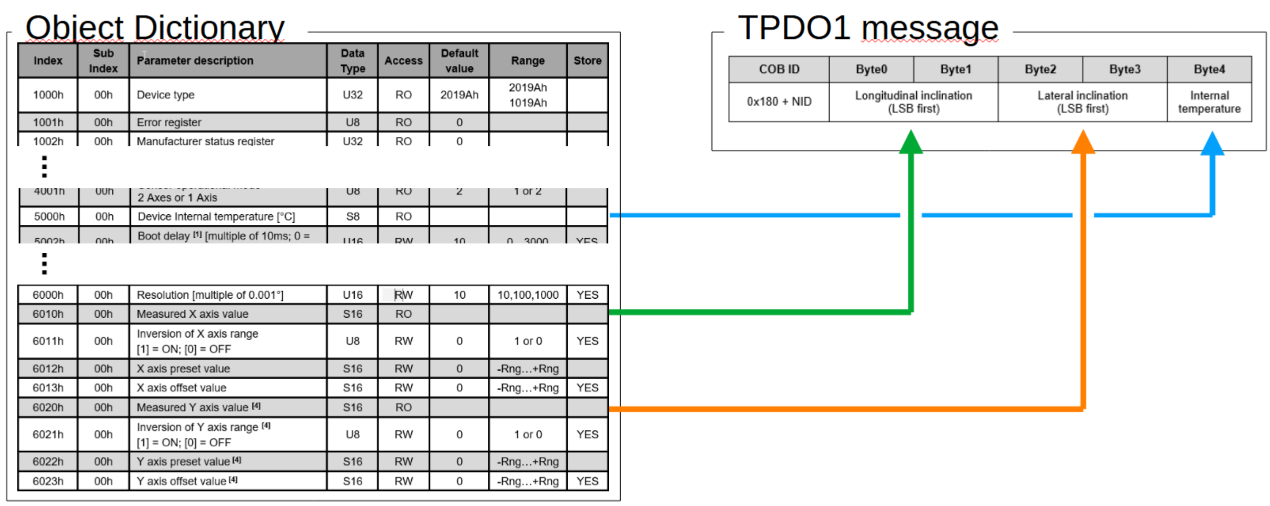


Table 29 - example of a dual-axis OIAC2 inclinometer process data values mapping in TPDO1 (with axis swap OFF)

12 MANUFACTURER-SPECIFIC OBJECTS SPECIFICATIONS

12.1 2000h - Node identifier (NID)

Sub Index	00h
Data Type	Unsigned 8bit
Access	Read and Write
Default value	09h
Range	01h – 7Fh
Write Effect	after save all + restart
NVM saved	after save all

This object holds the device's Node Identifier (NID) in the CANopen network. A new NID value will take effect only after the new configuration is saved to permanent memory (refer to par.9.2) and the node is reset. During the device initialization phase, all the COB-IDs for the various communication objects are recalculated based on the NID value previously stored in permanent memory.

The NID value cannot be reset to its default using the *Restore All Default Parameters* option (see par. 9.3); it must always be changed manually or by using the *Restore Default NID value* object (see par.12.3) at index 2002h.

12.2 2001h - Bit rate

Sub Index	00h
Data Type	Unsigned 8bit
Access	Read and Write
Default value	01h
Range	00h – 03h
Write Effect	after save all + restart
NVM saved	after save all

This object allows to select from the various bit rate values supported by the OIAC2 inclinometer. A new bit rate value becomes effective only after the new configuration is saved to permanent memory (see par.9.2) and the node is reset. During the device's initialization phase, the CAN interface is set to operate with the bit rate previously stored in permanent memory.

A CANopen device can support multiple CAN bit rates. The supported bit rates and their corresponding selection values are listed in the table below.

Object value	0	1	2	3

Bit Rate [kbit/s]	125	250	500	1000

Table 30 – selection value and corresponding bit rate values supported by OIAC2

The bit rate cannot be restored to its default value using the restore all default parameters object (see par. 9.3); it must always be changed manually.

12.3 2002h – Restore default NID value

Sub Index	00h
Data Type	Unsigned 32bit
Access	Read and Write
Default value	01h
Range	“load” 6Ch 6Fh 61h 64h
Write Effect	reboot

This object allows the restoringe of the device's default node identifier (NID) within the CANopen network. To prevent accidental restoration of the default NID, the process of writing to permanent memory is initiated only when a specific signature is provided. This signature is the word *load*, represented as 64h 61h 6Fh 65h. Upon receiving the *load* signature the device restores the default NID value in its permanent memory and confirms the success of the process with a SDO download response frame. After the default NID is restored, it is necessary to reset the sensor to make the default NID effective. If an incorrect signature is written to index 1011h, sub-index 01h, the device will not restore any parameters and will respond with a SDO abort frame containing the abort code: 0800 0020h.

** IMPORTANT ** If the device has a custom NID that differs from 09h (the default OIAC2 NID value), the restore default NID command will revert to the requested custom NID value.

12.4 3000h - Digital filter

Sub Index	00h
Data Type	Unsigned 16bit
Access	Read and Write
Default value	04h
Range	0000h – 0009h
Write Effect	immediate
NVM saved	after save all

This object configures the digital filter (moving average) for the OIAC2 inclinometers. The digital filter helps reduce noise in the measured angle values and allows for adjustments to the sensor's response time based on user applications. Heavy filtering stabilizes the measured angle values but slows down the sensor's response to changes in inclination. In contrast, light filtering enables a quicker sensor response to inclination changes but results in less stable measured angle values. The filter level can be adjusted by setting the value at index 3000h, sub-index 00h. The default value of 04h corresponds to a response time of 250 ms. If set to 00h, the filter is deactivated, and the step response time is 200 ms due to the internal hardware and signal conditioning circuitry of the OIAC2.
This setting applies to both single-axis and two-axes inclinometers. It is advisable to conduct practical tests to determine the optimal filter configuration for your specific needs.

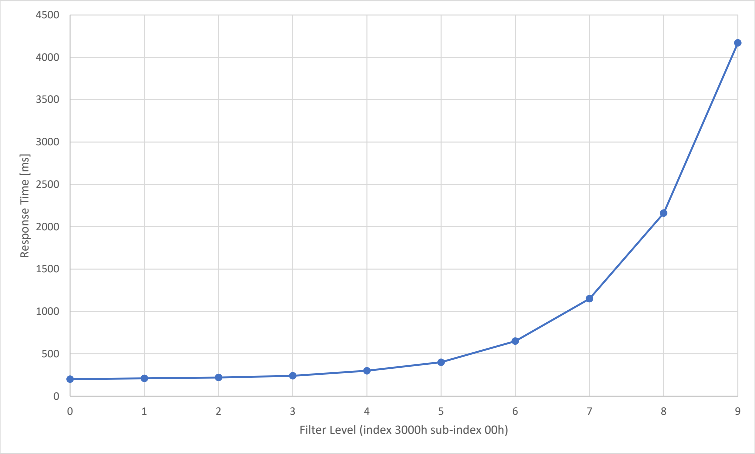


Figure 11 - Response time vs filtered samples

12.5 3002h – Single-axis data format

Sub Index	00h
Data Type	Unsigned 8bit
Access	Read and Write
Default value	00h
Range	00h – 01h
Write Effect	immediate
NVM saved	after save all

This object is applicable **only to single-axis model**. It modifies the format of the measured angle values that are available for SDO upload read access at index 6010h and transmitted with TPDO1 frames.

If this parameter is set to zero, the measured angle values will range from 0 degrees to 360 degrees (unsigned 16-bit). If it is set to one, the measured angles will range from -180 degrees to +180deg (signed 16-bit).

** EXAMPLE ** Let's consider a single-axis OIAC2 with a resolution of 0.01 degrees. The default single-axis data format is set to zero, which means the data range is 0-360 degrees. The measured angle value read from index 6010h is 80E0h, which converts to 32992 in decimal. By applying the resolution of 0.01 degrees, the measured angle is then 329.92 degrees.

Now, let's examine the same single-axis OIAC2 measuring the same angle, but this time with the single-axis data format set to one. In this case, the data format ranges from -180 to +180 degrees. The measured angle value read from index 6010h is F440h, which converts to -3008 in decimal. Applying the resolution of 0.01 degrees the measured angle is -30.08 degrees.

12.6 3003h – Axis swap in TPDO1

Sub Index	00h
Data Type	Unsigned 8bit
Access	Read and Write
Default value	00h
Range	00h – 01h
Write Effect	immediate
NVM saved	after save all

This object is **only applicable to dual-axes model**. It modifies the order of measured angle values in the TPDO1 frames. If the value in the axis swap object is set to zero, this feature is deactivated (this is the default setup), and the angle values in the TPDO1 frame are organized as described in paragraph 7.1.

To activate the axis swap option, set the value to "1" in the axis swap object, save the new configuration, and restart the OIAC2. This will invert the axis order in the TPDO1 frame message.

12.7 4000h – Measure range

Sub Index	01h – X-Axis
Data Type	Unsigned 16bit
Access	Read and Write
Default value	Dependent on the axis number and data format
Range	Dependent on the axis number and data format
Write Effect	Immediate
NVM saved	after save all

Sub Index	02h – Y-Axis
Data Type	Unsigned 16bit
Access	Read and Write
Default value	4Bh → 75deg
Range	01h – 5Ah
Write Effect	immediate
NVM saved	after save all

This part outlines the operational measure ranges for the X and Y axes of OIAC2. Please note that the Y-axis measure range is applicable only for dual-axes inclinometers.

The values stored in sub-indexes 01h and 02h are unsigned 16-bit values, represented in whole degrees. If the measured angle exceeds the specified range, the OIAC2 will transmit an EMCY message frame indicating which axis is in an out-of-range condition.

The sub-index 01h is applicable to both single-axis and dual-axis models, whereas the sub-index 02h is applicable only to single-axis models. The following table lists the default values and value ranges for sub-index 01h, depending on the device type (single-axis or dual-axis) and the data format (specific to single-axis devices).

Type	**Data format**	**Valid for**	**Default**	**Min**	**Max**
Dual-axis		X-axis Y-axis	75	0	90
Single-axis	[0…360] – Data Format 0	X-Axis	360	0	360
	[-180…+180] – Data Format 1		180	0	180

Table 31: Default, minimum, and maximum measuring ranges for OIAC2 types and their configurations

12.7.1 Measure range for dual-axis devices

The measure range of the dual-axes OIAC2 tilt angle is centred at zero position. This zero position is the sum of the OIAC2 physical tilt angle and the offset values in objects 6013h and 6023h. The maximum value for the measuring range is 90 degrees. If the OIAC2 detects a tilt above +90deg or below -90deg, it clamps the measured value to the nearest limit.

Do not exceed the measuring range value (index 4000h) using the preset (index 6012h for X and 6013h for Y) or the offset (index 6012h for X and 6022h for Y).

12.7.2 Measure range for single-axis devices

The measuring range of the single-axes OIAC2 tilt angle is centred at zero position. This zero position is the sum of the OIAC2's physical tilt angle and the offset value found in object 6013h. Depending on the data format, the maximum measuring range is either 360 for format 0 or 180 degrees for format 1.

12.8 4001h - Sensor operational mode

Sub Index	00h
Data Type	Unsigned 8bit
Access	Read Only
Range	01h – 02h

This object reports the operational mode of the OIAC2. A value of 01h means a single-axis device. A value of 02h means a dual-axes device.

12.9 5000h -Internal temperature

Sub Index	00h
Data Type	Signed 8bit
Access	Read Only
Range	00h – FFh (-128…+127)

This object provides the internal temperature of the OIAC2, measured in Celsius degrees. It is represented by a signed 8-bit value in two's complement.

12.10 5002h - Boot delay

Sub Index	00h
Data Type	Unsigned 16bit
Access	Read and Write
Default value	0Ah
Range	00h – BB8h
Write Effect	after save all + restart
NVM saved	after save all

This object introduces a startup delay for the OIAC2; it may be useful when the network master takes longer to start than the OIAC2 itself. The boot delay allows for postponing the transmission of the boot-up message frame. Users can set the delay in multiples of 10ms, ranging from 0 (OFF) to 3000 (30 seconds).

12.11 5003h – CAN error control register

Sub Index	00h
Data Type	Unsigned 8bit
Access	Read and Write
Default value	00h
Range	00h – 07h
Write Effect	after save all + restart
NVM saved	after save all

This object permits disabling one or all of the error condition.

Bit meaning:

- bit.0 - disable CAN ERROR BUSOFF
- bit.1 - disable CAN ERROR WARNING
- bit.2 - disable CAN ERROR OVERRUN

12.12 5004h – Auto operational

Sub Index	00h
Data Type	Unsigned 8bit
Access	Read and Write
Default value	00h
Range	00h – 01h
Write Effect	after save all + restart
NVM saved	after save all

When activated, this object makes the OIAC2 switches to the NMT operational state automatically. To use this feature, you must enable the boot delay timer and the TPDO1 event timer (index 1800h, sub-index 05h). Once enabled, the sensor starts sending TPDO1 frames right after the boot delay time.

12.13 5005h – X axis acceleration

Sub Index	00h
Data Type	Unsigned 16bit
Access	Read Only
Range	00h – 1000h

This object contains the measured acceleration along the sensor's X-axis as an unsigned 12-bit number, ranging from 0 to 4096. At 0g, the typical offset value is 2048 +/-150 LSB. The typical sensitivity is 820 LSB/g. The acceleration data is unfiltered and sampled at a rate of 1 kilosample per second. The axis swap options do not affect this object.

12.14 5006h – Y axis acceleration

Sub Index	00h
Data Type	Unsigned 16bit
Access	Read Only
Range	00h – 1000h

This object contains the measured acceleration along the sensor's Y-axis as an unsigned 12-bit number, ranging from 0 to 4096. At 0g, the typical offset value is 2048 +/-150 LSB. The typical sensitivity is 820 LSB/g. The acceleration data is unfiltered and sampled at a rate of 1 kilosample per second. The axis swap options do not affect this object.

12.15 5544h – Reserved

This object is used by the manufacturer for calibrating and configuring sensors.

12.16 5555h – Reserved

This object is used by the manufacturer for calibrating and configuring sensors.

13 DEVICE PROFILE OBJECTS SPECIFICATIONS

13.1 6000h - Resolution

Sub Index	00h
Data Type	Unsigned 16bit
Access	Read and Write
Default value	0Ah
Range	0Ah or 64h or 3E8h
Write Effect	immediate
NVM saved	after save all

This object holds the resolution for the angle data values stored in objects at indexes 6010h, 6012h, 6013h, and 6020h, 6022h, 6023h. The resolution value is expressed in thousandths of a degree [0.001deg].

Resolution			Resolution object value	
			[dec]	[hex]
Entire degree	1	deg	1000	03E8
Tenths of degree	0,1	deg	100	0064
Cents of degree	0,01	deg	10	000A
Thousandths of degree	0,001	deg	NOT SUPPORTED	

Table 32 - admitted resolutions values for OIAC2 inclinometers

To get the angular values with the desired resolution from the data values read from object 6010h or 6020h, use the following equations:

$$X_{AXIS}Angle[deg] = Value_{6010h} * \frac{Value_{6000h}}{1000}$$

$$Y_{AXIS}Angle[deg] = Value_{6020h} * \frac{Value_{6000h}}{1000}$$

These equations also apply to the data in the TPDO1 message frame.

13.2 6010h – X-axis value

Sub Index	00h
Data Type	Signed 16bit
Access	Read Only
Range	{resolution.dep.}

This object holds the measured X-axis angle value. The value is a 16-bit signed integer in 2's complement, stored in object 6010h. The value in this object depends on the resolution (6000h), offset value (6013h), and sign inversion object (6011h). You can access the X-axis value using the SDO protocol in any device state, as well as through TPDO1 when in the operational state.

The X-axis inclination value (6010h) is calculated as follows:

$$X_{AXIS\ 6010h}\ Angle\ Value = Physically\ measured\ inclination + X_{OFFSET\ 6013h} Value$$

13.3 6011h – X-axis sign inversion

Sub Index	00h
Data Type	Unsigned 8bit
Access	Read and Write
Default value	00h
Range	00h or 01h
Write Effect	immediate
NVM saved	after save all

13.3.1 Dual axis OIAC2 X-axis sign inversion

You can invert the sign of the X-axis measured angle writing 01h into object 6011h's sub-index 00h. By default, sign inversion is off. The tilt value's sign matches the direction of the axis arrows on the OIAC2 label.

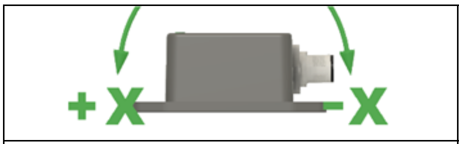
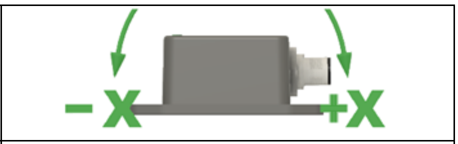
	
X-Axis inversion **OFF** [0]	X-Axis inversion **ON** [1]

Table 33: Dual-axis OIAC2 X-Axis angle sign with inversion OFF and ON

13.3.2 Single axis OIAC2 X-axis sign inversion

The single-axis OIAC2 inclinometer's inversion object at index 6011h behaves based on the data format set at index 3002h (see par.12.5).

13.3.2.1 Data format 0

If the data format object 3002h is set to 0, the measured tilt angle can range from 0 degrees to 360 degrees. This angle is available as a 16-bit unsigned integer, meaning it has no sign. Due to this, the sign inversion object reverses the sense of rotation around the X axis. See the pictures in Table 34 for a simpler understanding.

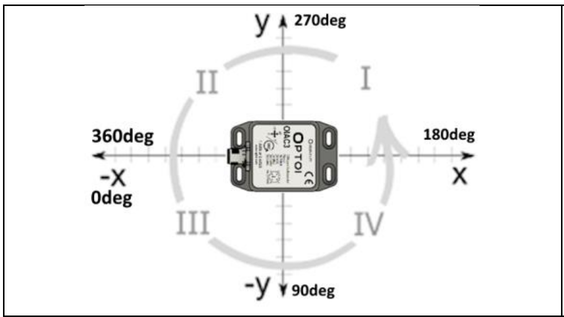
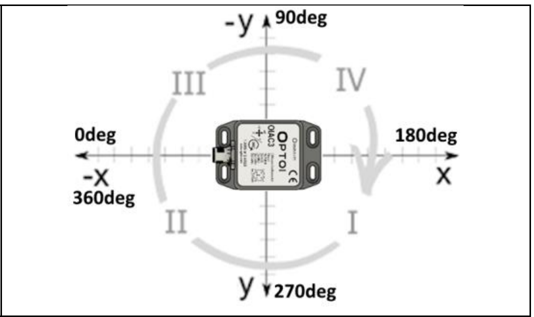
	
X-Axis inversion **OFF** [0]	X-Axis inversion **ON** [1]

Table 34 - Single axis OIAC2 with data format 0 [0…360] X-axis inversion effect

** EXAMPLE ** An inclination of 35 degrees is read from the device. The value at index 3002h is 00h, which means the range is from 0 to 360 degrees. To invert the X axis just write 01h into index 6011h (it's currently 00h), then the new inclination is 325 degrees.

13.3.2.2 Data format 1

If the data format object 3002h is set to 1, the measured tilt angle can range from -180 degrees to +180 degrees. This angle is available as a 16-bit signed integer, meaning it has the sign. Due to this, the sign inversion object inverts the sign of the measured angle. See the pictures in Table 35 for a simpler understanding.

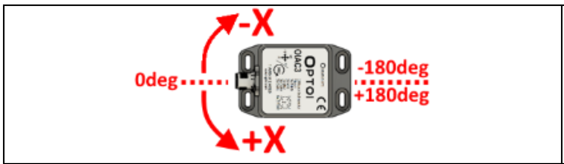
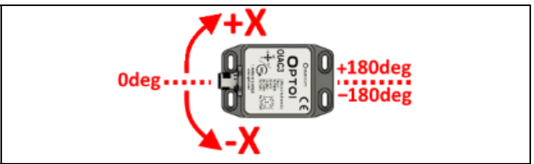
	
X-Axis inversion **OFF** [0]	X-Axis inversion **ON** [1]

Table 35: Single axis OIAC2 with data format 1 [±180] X-axis sign with inversione OFF and ON

13.4 6012h – X-axis preset

Sub Index	00h
Data Type	Signed 16bit
Access	Read and Write
Default value	0000h
Range	{OIAC2 type & cfg.dep.}
Write Effect	immediate

The offset can be written in the offset object at index 6013h sub-index 00h (see par.13.5). Alternatively, you can write the requested angle in the preset value object at index 6012h sub-index 00h when the device is at a specific tilt angle. The preset value is the desired angle the sensor should measure when positioned at a specific inclination. The preset must be written as a signed 16-bit number in 2's complement form in index 6012h, sub-index 00h. The preset value must have the same resolution as the measured angle. The resolution value is available at index 6000h, sub-index 00h (see par.13.1). The preset value is subtracted from the absolute measured angle, and the result is stored as the offset value in index 6013h sub-index 00h. The preset value is commonly used to implement an *auto-zero* command.

** EXAMPLE ** Let's consider an OIAC2 tilted at an inclination of +13.00deg on the X-axis. It is requested to read a zero angle when the device is tilted at +7.00deg. The preset value you have to write at index 6012h is +6deg.

** EXAMPLE ** Let's consider an OIAC2 tilted at an inclination of 3.27deg. It is requested to read a zero angle when the OIAC2 is in this position. Then, the preset value to be written at index 6012h is zero.

** IMPORTANT **

Before writing to the X-axis preset value object, always clear the X-axis offset value object at index 6013h. If the offset value is not zero, the new offset value will be calculated incorrectly.

13.5 6013h – X-axis offset

Sub Index	00h
Data Type	Signed 16bit
Access	Read and Write
Default value	0000h
Range	{OIAC2 type & cfg.dep.}
Write Effect	immediate
	after save all

The X-axis offset value must be written to object 6013h, subindex 00h. Alternatively, you can set the offset using the preset value object at index 6013h, subindex 00h (see par.13.5). The X-axis offset value must be a signed 16-bit number in 2's complement form. This value is added to the absolute measured X-axis tilt angle, and the result is available in the X-axis angle object at index 6010h, subindex 00h.

The measured X angle at 6010h is the sum of the acquired absolute angle and the X offset at 6013h:

$$X_{AXIS\ 6010h}\ Angle\ Value = Physically\ measured\ inclination + X_{OFFSET\ 6013h} Value$$

13.6 6020h – Y-axis value

Sub Index	00h
Data Type	Signed 16bit
Access	Read Only
Range	{resolution.dep.}0h

This object holds the measured Y-axis angle value. The value is a 16-bit signed integer in 2's complement, stored in object 6020h. The value in this object depends on the resolution (6000h), offset value (6023h), and sign inversion object (6021h). You can access the Y-axis value using the SDO protocol in any device state, as well as through TPDO1 when in the operational state.

The X-axis inclination value (6020h) is calculated as follows:

$$X_{AXIS\ 6020h}\ Angle\ Value = Physically\ measured\ inclination + X_{OFFSET\ 6023h} Value$$

** IMPORTANT **
the Y-axis value object is available only for dual axis OIAC2 inclinometers

13.7 6021h – Y-axis sign inversion

Sub Index	00h
Data Type	Unsigned 8bit
Access	Read and Write
Default value	00h
Range	00h or 01h
Write Effect	immediate
NVM saved	after save all

You can invert the sign of the Y-axis measured angle by writing 01h into object 6011h's sub-index 00h. By default, sign inversion is off. The tilt value's sign matches the direction of the axis arrows on the OIAC2 label.

	
Y-Axis inversion **OFF** [0]	Y-Axis inversion **ON** [1]

Table 36: Dual-axis OIAC2 X-Axis angle sign with inversion OFF and ON

** IMPORTANT **
the Y-axis sign inversion object is available only for dual axis OIAC2 inclinometers

13.8 6022h – Y-axis preset

Sub Index	00h
Data Type	Signed 16bit
Access	Read and Write
Default value	0000h
Range	{OIAC2 type & cfg.dep.}
Write Effect	immediate

The offset can be written in the offset object at index 6023h sub-index 00h (see par.13.9). Alternatively, you can write the requested angle in the preset value object at index 6022h sub-index 00h when the device is at a specific tilt angle. The preset value is the desired angle the sensor should measure when positioned at a specific inclination. The preset must be written as a signed 16-bit number in 2's complement form in index 6022h, sub-index 00h. The preset value must have the same resolution as the measured angle. The resolution value is available at index 6000h, sub-index 00h (see par.13.1). The preset value is subtracted from the absolute measured angle, and the result is stored as the offset value in index 6023h sub-index 00h. The preset value is commonly used to implement an "auto-zero command".

** IMPORTANT **

Before writing to the Y-axis preset value object, always clear the Y-axis offset value object at index 6023h. If the offset value is not zero, the new offset value will be calculated incorrectly.

** IMPORTANT **

the Y-axis preset object is available only for dual axes OIAC2 inclinometers

13.9 6023h – Y-axis offset

Sub Index	00h
Data Type	Signed 16bit
Access	Read and Write
Default value	0000h
Range	{OIAC2 type & cfg.dep.}
Write Effect	immediate
	after save all

The Y-axis offset value must be written to object 6023h, subindex 00h. Alternatively, you can set the offset using the preset value object at index 6022h, subindex 00h (see par.13.8). The Y-axis offset value must be a signed 16-bit number in 2's complement form. This value is added to the absolute measured Y-axis tilt angle, and the result is available in the Y-axis angle object at index 6020h, subindex 00h (see par.13.6).

The measured X angle at 6020h is the sum of the acquired absolute angle and the X offset at 6023h:

$$Y_{AXIS\ 6020h}\ Angle\ Value = Physically\ measured\ inclination + Y_{OFFSET\ 6023h} Value$$

** IMPORTANT **

the Y-axis preset object is available only for dual axis OIAC2 inclinometers

14 EMERGENCY MESSAGES (EMCY)

When an error occurs, an emergency message, known as an EMCY frame, is triggered and sent to the bus with high priority. Once the emergency situation is resolved, the device sends an "Error reset" message. The structure of an EMCY frame consists of the COB-ID (specified at index 1014h, see par. 11.11) plus the Node ID, followed by the Emergency object that contains all the error-related information.

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3	Byte 4	Byte 5	Byte 6	Byte 7
80h+NID	Emergency error Code (Table 38)		Error register	Manufacturer specific error field				
				Communication error	Device error	00h	00h	00h

Table 37 - Emergency object frame structure

Error Code	Error description
0000h	Error reset or no more error present
1030h	Generic error
4200h	Device temperature error
5000h	Self-test error or CRC memory error
5010h	Sensor error on X axis
5020h	Sensor error on Y axis
8110h	Receive\Transmit buffer overflow
8120h	CAN warning limit reached
8130h	Node guard event occurred
8140h	Recover from Bus-off

Table 38 - Emergency error codes

15 STATUS LED

The two-color LED complies with CiA DR-303-3 specifications. Green LED is used as run LED indicator and red LED is used as error LED. The tables below describe all LED configurations.

RUN LED	LED state	LED state description
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	OFF	The device is switched off
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	Single flash	The device is in Stop mode
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	Blinking	The device is in Pre-operational mode
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	ON	The device is in Operational mode

Table 39 - Run LED configurations

ERROR LED	LED state	LED state description
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	OFF	The device is in working conditions
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	Single flash	CAN warning limit reached
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	Double flash	Loss of Guarding-master detected
¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤ ¤	ON	The device is in state Bus-Off

Table 40 - Error LED configurations