

User's guide

**Ax58, Ax58S**

**AxC58 CB**

**CC-CB, CC-CB-C**

**CAN**open

DS406 - Device profile for encoder



..... **lika**

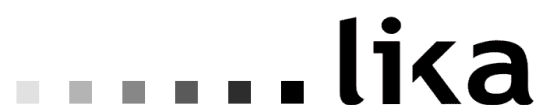
Smart encoders & actuators

This publication was produced by Lika Electronic s.r.l. 2014. All rights reserved. Tutti i diritti riservati. Alle Rechte vorbehalten. Todos los derechos reservados. Tous droits réservés.

This document and information contained herein are the property of Lika Electronic s.r.l. and shall not be reproduced in whole or in part without prior written approval of Lika Electronic s.r.l. Translation, reproduction and total or partial modification (photostat copies, film and microfilm included and any other means) are forbidden without written authorisation of Lika Electronic s.r.l.

The information herein is subject to change without notice and should not be construed as a commitment by Lika Electronic s.r.l. Lika Electronic s.r.l. reserves the right to make all modifications at any moments and without forewarning.

This manual is periodically reviewed and revised. As required we suggest checking if a new or updated edition of this document is available at Lika Electronic s.r.l.'s website. Lika Electronic s.r.l. assumes no responsibility for any errors or omissions in this document. Critical evaluation of this manual by the user is welcomed. Your comments assist us in preparation of future documentation, in order to make it as clear and complete as possible. Please send an e-mail to the following address [info@lika.it](mailto:info@lika.it) for submitting your comments, suggestions and criticisms.



# General contents

<b>User's guide</b> .....	1
General contents.....	3
Subject index.....	6
Typographic and iconographic conventions.....	7
Preliminary information.....	8
1 - Safety summary.....	9
2 - Identification.....	10
3 - Mounting instructions.....	11
3.1 Solid shaft encoders.....	11
3.1.1 Customary installation.....	11
3.1.2 With fixing clamps (LKM-386).....	11
3.1.3 With mounting bell (PF4256).....	12
3.2 Hollow shaft encoders.....	13
3.2.1 ASC58, AMC58.....	13
3.2.2 ASC59, AMC59.....	13
3.2.3 ASC60, AMC60.....	13
4 - Electrical connections.....	14
4.1 Connection cap.....	14
4.2 Ground connection.....	15
4.3 Connection cap with PGs: CC-CB.....	15
4.4 Connection cap with M12 connectors: CC-CB-C.....	16
4.5 Connection of the shield.....	16
4.6 RT Bus termination.....	17
4.7 Diagnostic LEDs.....	17
4.8 Baud rate: DIP A.....	18
4.9 Node number: DIP B.....	19
5 - Quick reference.....	20
5.1 Hardware resolution.....	20
5.1.1 Procedure to set hardware resolution.....	20
5.2 Using the default settings.....	22
6 - CANopen® interface.....	23
6.1 EDS file.....	23
6.2 Operation states.....	23
6.2.1 Initialization state.....	24
6.2.2 Pre-operational state.....	24
6.2.3 Operational state.....	24
6.2.4 Stopped state.....	24
6.3 Communication objects.....	25
6.3.1 Pre-defined connection set.....	25
6.4 NMT objects.....	26
6.5 Boot-up objects.....	26
6.6 PDO objects.....	27
<b>PDO1 Cyclic mode: cyclic transmission</b> .....	27
<b>PDO2 and PDO3 SYNC mode: synchronous transmission</b> .....	27
<b>PDO4 Cyclic mode: cyclic transmission</b> .....	27
6.7 SDO objects.....	28

6.7.1 Command.....	28
6.8 Object dictionary.....	29
6.8.1 Standard objects (DS 301).....	29
1000-00 Device type.....	29
1001-00 Error register.....	29
1003 Pre-defined error field.....	29
1005-00 COB_ID SYNC message.....	30
1008-00 Name of device manufacturer.....	30
1009-00 Hardware version.....	30
100A-00 Software version.....	30
100C-00 Guard time.....	30
100D-00 Life time factor.....	30
1010-01 Store parameters.....	31
1011-01 Restore default parameters.....	31
1014-00 COB-ID EMCY.....	32
1015-00 Inhibit time EMCY.....	32
1018 Identification object.....	32
1800 PDO1 parameters.....	32
1801 PDO2 parameters.....	34
1802 PDO3 parameters.....	35
1803 PDO4 parameters.....	36
1A00-01 PDO1 mapping parameter.....	38
1A01-01 PDO2 mapping parameter.....	38
1A02-01 PDO3 mapping parameter.....	38
1A03-01 PDO4 mapping parameter.....	38
6.8.2 Manufacturer specific objects.....	39
2104-00 Limit switch min.....	39
2105-00 Limit switch max.....	39
3000-00 Baud rate.....	39
3001-00 Node-ID.....	40
3005-00 Velocity format.....	41
3006-00 Velocity value.....	41
6.8.3 Device profile objects (DS 406).....	42
6000-00 Operating parameters.....	42
Code sequence.....	42
Scaling function.....	42
Limit switch min.....	42
Limit switch max.....	42
6001-00 Counts per revolution.....	43
6002-00 Total resolution.....	43
6003-00 Preset value.....	44
6004-00 Position value.....	45
6200-00 Cyclic time.....	45
6500-00 Operating states.....	45
Code sequence.....	45
Scaling function.....	46
Limit switch min.....	46
Limit switch max.....	46
Current operating state.....	46

6501-00 Hardware counts per revolution.....	46
6502-00 Hardware number of turns.....	46
6504-00 Supported alarms.....	47
6506-00 Supported warnings.....	47
6507-00 Profile and software version.....	47
6508-00 Operating time.....	47
6509-00 Offset value.....	47
650A-01 Manufacturer offset value.....	47
650B-00 Serial number.....	47
6.9 Warning objects.....	48
6.10 Emergency objects.....	48
6.11 Node guarding protocol.....	49
7 – Setting-up.....	50
8 – Default parameters list.....	53

# Subject index




<b>1</b>	
1000-00 Device type.....	29
1001-00 Error register.....	29
1003 Pre-defined error field.....	29
1005-00 COB_ID SYNC message.....	30
1008-00 Name of device manufacturer.....	30
1009-00 Hardware version.....	30
100A-00 Software version.....	30
100C-00 Guard time.....	30
100D-00 Life time factor.....	30
1010-01 Store parameters.....	31
1011-01 Restore default parameters.....	31
1014-00 COB-ID EMCY.....	32
1015-00 Inhibit time EMCY.....	32
1018 Identification object.....	32
1800 PDO1 parameters.....	32
1801 PDO2 parameters.....	34
1802 PDO3 parameters.....	35
1803 PDO4 parameters.....	36
1A00-01 PDO1 mapping parameter.....	38
1A01-01 PDO2 mapping parameter.....	38
1A02-01 PDO3 mapping parameter.....	38
1A03-01 PDO4 mapping parameter.....	38
<b>2</b>	
2104-00 Limit switch min.....	39
2105-00 Limit switch max.....	39
<b>3</b>	
3000-00 Baud rate.....	39
3001-00 Node-ID.....	40
3005-00 Velocity format.....	41
3006-00 Velocity value.....	41
<b>6</b>	
6000-00 Operating parameters.....	42
6001-00 Counts per revolution.....	43
6002-00 Total resolution.....	43
6003-00 Preset value.....	44
6004-00 Position value.....	45
6200-00 Cyclic time.....	45
6500-00 Operating states.....	45
6501-00 Hardware counts per revolution.....	46
6502-00 Hardware number of turns.....	46
6504-00 Supported alarms.....	47
6506-00 Supported warnings.....	47
6507-00 Profile and software version.....	47
6508-00 Operating time.....	47
6509-00 Offset value.....	47
650A-01 Manufacturer offset value.....	47
650B-00 Serial number.....	47
<b>C</b>	
COB-ID of PDO1.....	32
COB-ID of PDO4.....	36
COB-ID of the PDO2.....	34
COB-ID of the PDO3.....	35
Code sequence.....	42, 45
Current operating state.....	46
<b>F</b>	
Flash memory error.....	48
<b>I</b>	
Initialization.....	24
<b>L</b>	
Last error occurred.....	29
Limit switch max.....	42, 46
Limit switch min.....	42, 46
<b>N</b>	
Node guarding error.....	48
Number of current errors.....	29
<b>O</b>	
Operational.....	24
<b>P</b>	
Pre-operational.....	24
Previous errors occurred.....	29
Product number.....	32
<b>R</b>	
Revision number.....	32
<b>S</b>	
Scaling function.....	42, 46
Stopped.....	24
<b>T</b>	
Transmission type.....	33 e seg., 36 e seg.
<b>V</b>	
Vendor number.....	32

# Typographic and iconographic conventions

In this guide, to make it easier to understand and read the text the following typographic and iconographic conventions are used:

- parameters and objects both of the device and the interface are coloured in **ORANGE**;
- alarms are coloured in **RED**;
- states are coloured in **FUCSIA**.

When scrolling through the text some icons can be found on the side of the page: they are expressly designed to highlight the parts of the text which are of great interest and significance for the user. Sometimes they are used to warn against dangers or potential sources of danger arising from the use of the device. You are advised to follow strictly the instructions given in this guide in order to guarantee the safety of the user and ensure the performance of the device. In this guide the following symbols are used:

	This icon, followed by the word <b>WARNING</b> , is meant to highlight the parts of the text where information of great significance for the user can be found: user must pay the greatest attention to them! Instructions must be followed strictly in order to guarantee the safety of the user and a correct use of the device. Failure to heed a warning or comply with instructions could lead to personal injury and/or damage to the unit or other equipment.
	This icon, followed by the word <b>NOTE</b> , is meant to highlight the parts of the text where important notes needful for a correct and reliable use of the device can be found. User must pay attention to them! Failure to comply with instructions could cause the equipment to be set wrongly: hence a faulty and improper working of the device could be the consequence.
	This icon is meant to highlight the parts of the text where suggestions useful for making it easier to set the device and optimize performance and reliability can be found. Sometimes this symbol is followed by the word <b>EXAMPLE</b> when instructions for setting parameters are accompanied by examples to clarify the explanation.

# Preliminary information

This guide is designed to describe the technical characteristics, installation and use of the CANopen encoders of the **Ax58x series**. For any further information please refer to the product datasheet.

To make it easier to read the text, this guide is divided into two main sections.

In the first section general information concerning the safety, the mechanical installation and the electrical connection as well as tips for setting up and running properly and efficiently the unit are provided.

In the second section, entitled **CANopen Interface**, both general and specific information is given on the CANopen interface. In this section the interface features and the objects implemented in the unit are fully described.



## 1 - Safety summary

### Safety

- Always adhere to the professional safety and accident prevention regulations applicable to your country during device installation and operation;
- installation and maintenance operations have to be carried out by qualified personnel only, with power supply disconnected and stationary mechanical parts;
- device must be used only for the purpose appropriate to its design: use for purposes other than those for which it has been designed could result in serious personal and/or the environment damage;
- high current, voltage and moving mechanical parts can cause serious or fatal injury;
- failure to comply with these precautions or with specific warnings elsewhere in this manual violates safety standards of design, manufacture, and intended use of the equipment;
- Lika Electronic s.r.l. assumes no liability for the customer's failure to comply with these requirements.

### Electrical safety

- Turn off power supply before connecting the device;
- connect according to explanation in section "4 - Electrical connections";
- in compliance with the 2004/108/EC norm on electromagnetic compatibility, following precautions must be taken:
  - before handling and installing, discharge electrical charge from your body and tools which may come in touch with the device;
  - power supply must be stabilized without noise, install EMC filters on device power supply if needed;
  - always use shielded cables (twisted pair cables whenever possible);
  - avoid cables runs longer than necessary;
  - avoid running the signal cable near high voltage power cables;
  - mount the device as far as possible from any capacitive or inductive noise source, shield the device from noise source if needed;
  - to guarantee a correct working of the device, avoid using strong magnets on or near by the unit;
  - minimize noise by connecting the shield and/or the frame to ground. Make sure that ground is not affected by noise. The connection point to ground can be situated both on the device side and on user's side. The best solution to minimize the interference must be carried out by the user.

## Mechanical safety

- Install the device following strictly the information in the section "3 - Mounting instructions";
- mechanical installation has to be carried out with stationary mechanical parts;
- do not disassemble the encoder;
- do not tool the encoder or its shaft;
- delicate electronic equipment: handle with care; do not subject the device and the shaft to knocks or shocks;
- respect the environmental characteristics declared by manufacturer;
- unit with solid shaft: in order to guarantee maximum reliability over time of mechanical parts, we recommend a flexible coupling to be installed to connect the encoder and user's shaft; make sure the misalignment tolerances of the flexible coupling are respected;
- unit with hollow shaft: the encoder can be mounted directly on a shaft whose diameter has to respect the technical characteristics specified in the purchase order and clamped by means of the collar and, when requested, the anti-rotation pin.

## 2 - Identification

Device can be identified through the **ordering code** and the **serial number** printed on the label applied to its body. Information is listed in the delivery document too. Please always quote the ordering code and the serial number when reaching Lika Electronic s.r.l. for purchasing spare parts or needing assistance. For any information on the technical characteristics of the product [refer to the technical catalogue](#).



**Warning:** encoders having ordering code ending with "/Sxxx" may have mechanical and electrical characteristics different from standard and be supplied with additional documentation for special connections (Technical Info).

### 3 - Mounting instructions



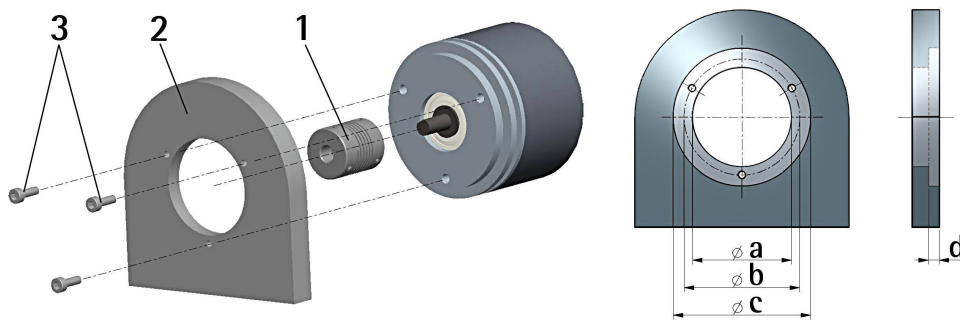
**WARNING**

Installation and maintenance operations have to be carried out by qualified personnel only, with power supply disconnected and mechanical parts absolutely in stop.

#### 3.1 Solid shaft encoders

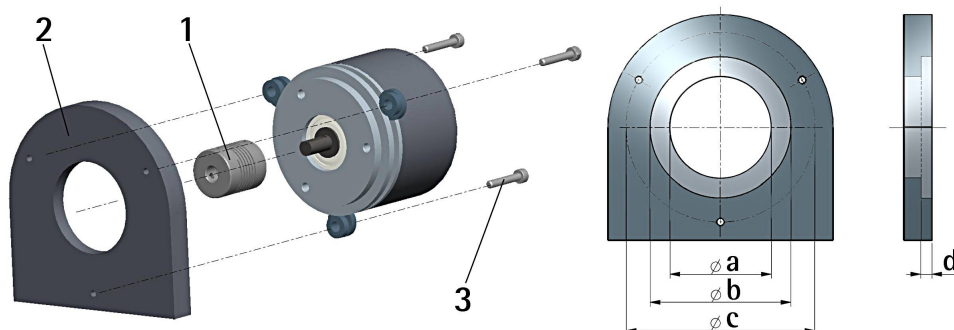
- Mount the flexible coupling **1** on the encoder shaft;
- fix the encoder to the flange **2** (or to the mounting bell) by means of screws **3**;
- secure the flange **2** to the support (or the mounting bell to the motor);
- mount the flexible coupling **1** on the motor shaft;
- make sure the misalignment tolerances of the flexible coupling **1** are respected.

##### 3.1.1 Customary installation



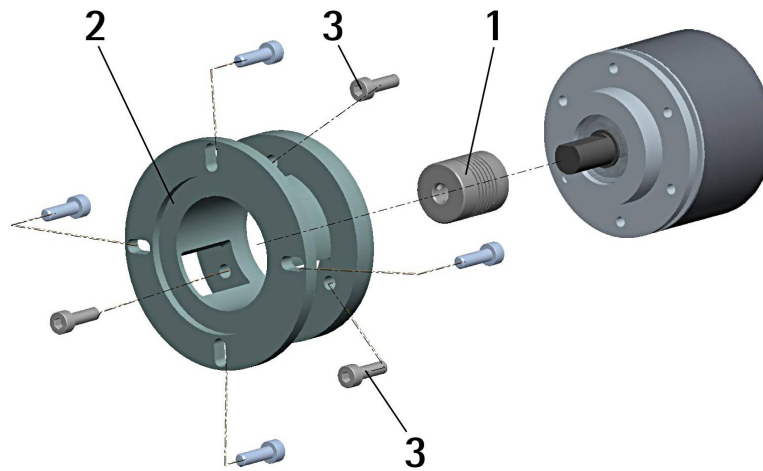
	a [mm]	b [mm]	c [mm]	d [mm]
AS58, AM58	-	42	50 F7	4
AS58S, AM58S	36 H7	48	-	-

##### 3.1.2 With fixing clamps (LKM-386)



	a [mm]	b [mm]	c [mm]	d [mm]
AS58, AM58	-	50 F7	67	4
AS58S, AM58S	36 H7	-	67	-

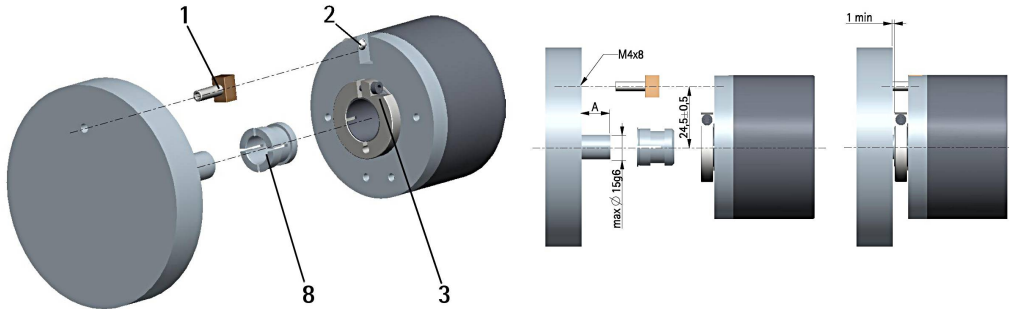
### 3.1.3 With mounting bell (PF4256)



### 3.2 Hollow shaft encoders

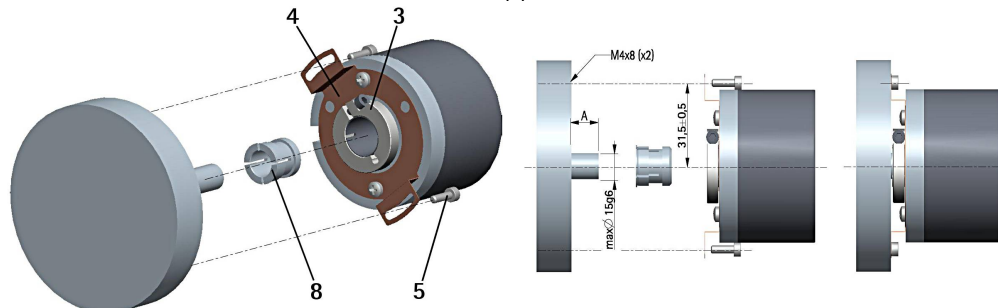
#### 3.2.1 ASC58, AMC58

- Fasten the anti-rotation pin **1** to the rear of the motor (secure it using a locknut);
- mount the encoder on the motor shaft using the reducing sleeve **8** (if supplied). Avoid forcing the encoder shaft;
- insert the anti-rotation pin **1** into the slot on the flange of the encoder; this secures it in place by grub screw **2**, preset at Lika;
- fix the collar **3** to the encoder shaft (apply threadlocker to screw **3**).



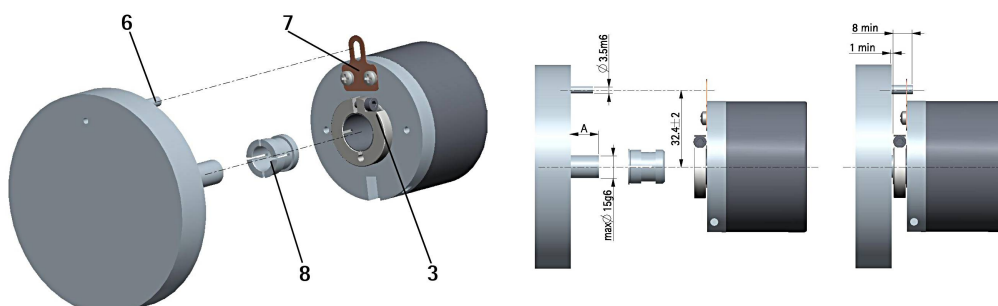
#### 3.2.2 ASC59, AMC59

- Mount the encoder on the motor shaft using the reducing sleeve **8** (if supplied). Avoid forcing the encoder shaft;
- fasten the fixing plate **4** to the rear of the motor using two M3 cylindrical head screws **5**;
- fix the collar **3** to the encoder shaft (apply threadlocker to screw **3**).



#### 3.2.3 ASC60, AMC60

- Fix the tempered pin **6** to the rear of the motor;
- mount the encoder on the motor shaft using the reducing sleeve **8** (if supplied). Avoid forcing the encoder shaft;
- make sure the anti-rotation pin **6** is inserted properly into the fixing plate **7**;
- fix the collar **3** to the encoder shaft (apply threadlocker to screw **3**).



## 4 - Electrical connections



### WARNING

Power supply must be turned off before performing any electrical connection!

### 4.1 Connection cap



### WARNING

Do not remove or mount the connection cap with power supply switched ON. Damage may be caused to internal components.

The terminal connectors for connecting the power supply and the BUS IN and BUS OUT cables (CC-CB connection cap) as well as the dip-switches meant to set the baud rate and the node ID and activate the termination resistance are located inside the encoder connection cap. Thus you must remove the connection cap to access any of them.



### NOTE

Be careful not to damage the internal components when you perform this operation.

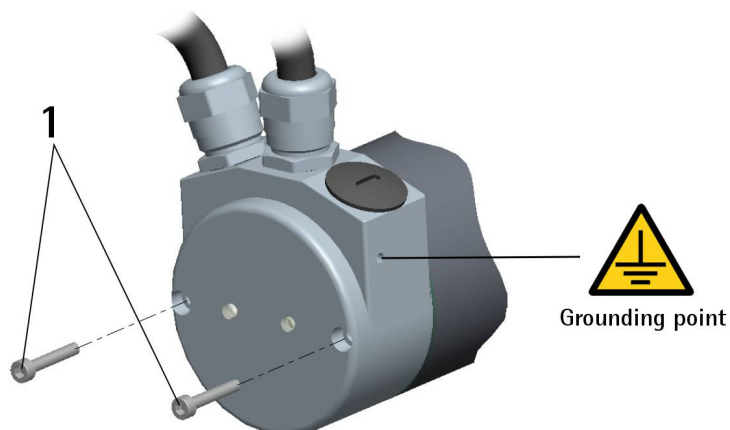
To remove the connection cap loosen the two screws **1**. Please be careful with the internal connector.

Always replace the connection cap at the end of the operation. Take care in re-connecting the internal connector. Tighten the screws **1** using a tightening torque of approx. 2.5 Nm.



### WARNING

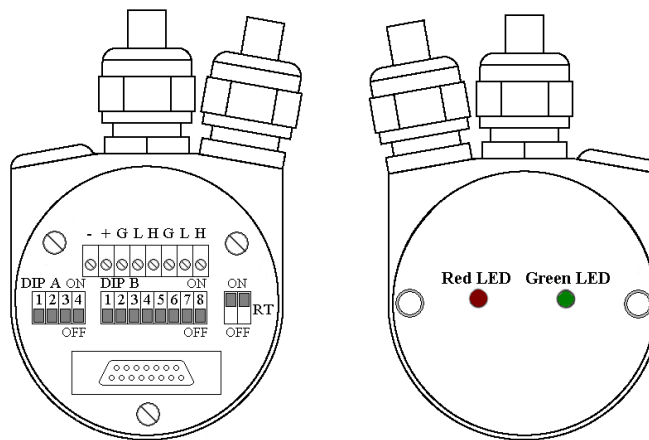
You are required to check that the encoder body and the connection cap are at the same potential before replacing the connection cap!



**4.2 Ground connection**

Minimize noise by connecting the shield and/or the connector housing and/or the frame to ground. Make sure that ground is not affected by noise. The connection point to ground can be situated both on the device side and on user's side. The best solution to minimize the interference must be carried out by the user. You are advised to provide the ground connection as close as possible to the encoder. We suggest using the ground point provided in the cap (see Figure, use 1 TCEI M3 x 6 cylindrical head screw with 2 tooth lock washers).

**4.3 Connection cap with PGs: CC-CB**



The connection cap is fitted with two PG9 cable glands for bus-IN, bus-OUT and supply voltage connections. The bus cables can be connected directly to the terminal connectors in front of each cable gland. It is recommended to use CANbus certificated cables. Core diameter should not exceed Ø 1,5 mm (0.06 inches).

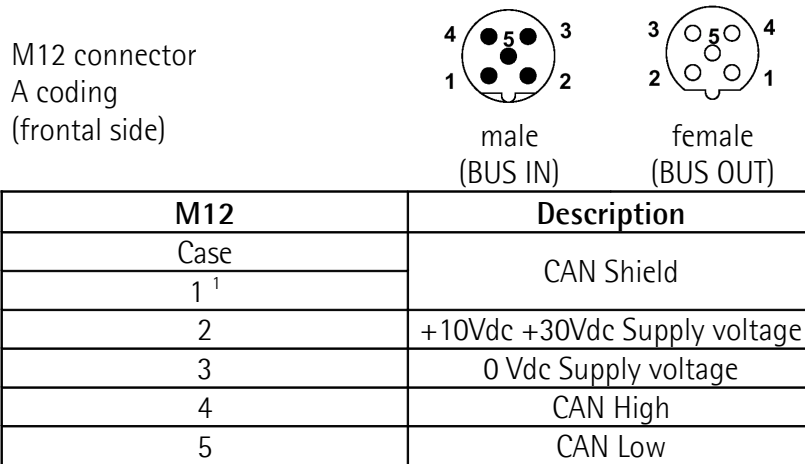
Clamp	Description
-	0 VDC Supply voltage
+	+10VDC +30VDC Supply voltage
G	CAN GND <sup>1</sup>
L	CAN Low
H	CAN High
PG	CAN Shield <sup>2</sup>

<sup>1</sup> CAN GND is the 0V reference of CAN signals, it is not connected to 0VDC supply voltage.

<sup>2</sup> Connect cable shield to cable gland.

#### 4.4 Connection cap with M12 connectors: CC-CB-C

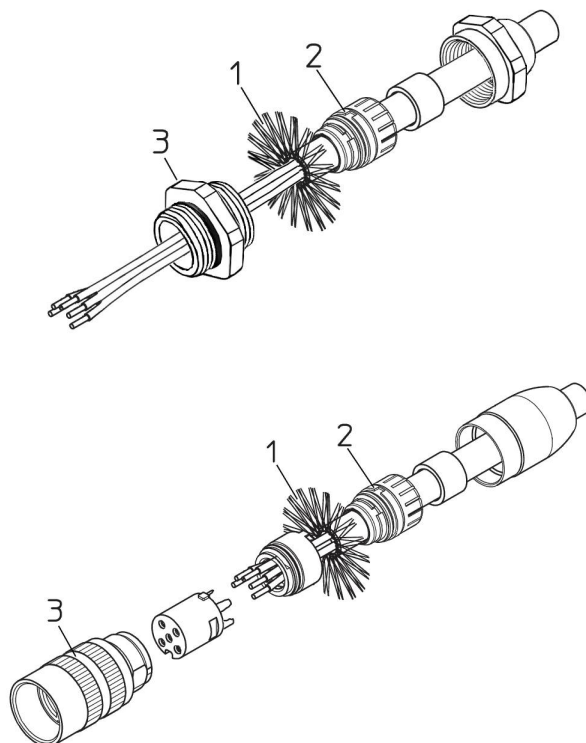
The CC-CB-C connection cap has two M12 connectors with pin-out in compliance with CANopen® standard. Therefore you can use standard CAN cables commercially available.



<sup>1</sup> CAN Shield is also connected to pin 1 to allow the connection of the shield even if the plug connector has a plastic case.

#### 4.5 Connection of the shield

Disentangle and shorten the shielding **1** and then bend it over the part **2**; finally place the ring nut **3** of the connector. Be sure that the shielding **1** is in tight contact with the ring nut **3**.





#### 4.6 RT Bus termination

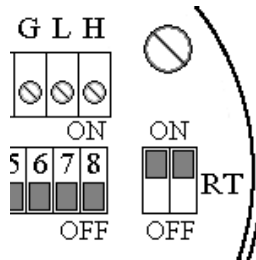


**WARNING**

Power supply must be turned off before performing this operation!

A bus termination resistance is provided in the connection cap and has to be activated as line termination in the last device of the transmission line. Use RT Switch to activate or deactivate the bus termination.

RT	Description
1 = 2 = ON	Activated: if the encoder is the last device in the transmission line
1 = 2 = OFF	Deactivated: if the encoder is not the last device in the transmission line



#### 4.7 Diagnostic LEDs

Two diagnostic LEDs located in the rear side of the connection cap are designed to show the operating or fault status of the CANopen® interface.

GREEN LED	Description
ON	Encoder is in <b>Operational</b> state
Single flash	Encoder is in <b>Stopped</b> state
Blinking	Encoder is in <b>Pre-Operational</b> state

RED LED	Description
ON	The CAN controller is switched off
Double flash	Node guarding error
Single flash	Max. number of warning errors
Blinking	Generic error or <b>Flash memory error</b>
OFF	No error

During initialization, device carries out a hardware test to check LEDs operation. Both LEDs light up.

4.8 Baud rate: DIP A



**WARNING**

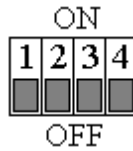
Power supply must be turned off before performing this operation!

Transmission rate can be set either via hardware or via software.

If **DIP A** bit 4 = "OFF" the bit rate is set through object **3000-00 Baud rate** in the "Object Dictionary" and can be modified using SDO messages.

If **DIP A** bit 4 = "ON" the bit rate is set by DIP A.

DIP A:



Set binary value of transmission rate considering that: ON=1, OFF=0.

bit	1 LSB	2	3 MSB	4
	$2^0$	$2^1$	$2^2$	ON/OFF

Available baud rate values:

Decimal value	Binary value	Baud rate
0	000	20 Kbit/s
1	001	50 Kbit/s
2	010	100 Kbit/s
3	011	125 Kbit/s
4	100	250 Kbit/s
5	101	500 Kbit/s (default)
6	110	800 Kbit/s
7	111	1000 Kbit/s

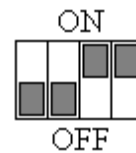


**Example**

Set 250Kbit/s:

$4_{10} = 100_2$  (binary value, see table above)

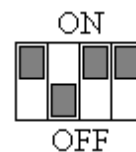
bit	1	2	3	4
	$2^0$	$2^1$	$2^2$	$2^3$
	OFF	OFF	ON	ON



Set 500Kbit/s:

$5_{10} = 101_2$  (binary value, see table above)

bit	1	2	3	4
	$2^0$	$2^1$	$2^2$	$2^3$
	ON	OFF	ON	ON



4.9 Node number: DIP B



**WARNING**

Power supply must be turned off before performing this operation!

Node number can be set both via hardware or via software.

Allowed node addresses range between 1 and 127. The default value is 1.

If all bits of **DIP B** are "OFF" node number is set through the object **3001-00 Node-ID** of the "Object Dictionary" and can be modified using SDO messages.

If one bit at least of **DIP B** is set to "ON" the node number is set through DIP B.



Set the node number in binary value: ON=1, OFF=0

bit	1	2	3	4	5	6	7	8
	LSB						MSB	not used
	2 <sup>0</sup>	2 <sup>1</sup>	2 <sup>2</sup>	2 <sup>3</sup>	2 <sup>4</sup>	2 <sup>5</sup>	2 <sup>6</sup>	

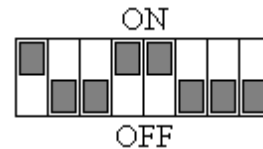


**Example**

Set node number = 25:

25<sub>10</sub> = 0001 1001<sub>2</sub> (binary value)

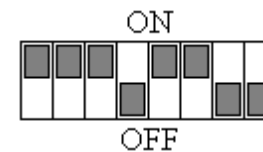
bit	1	2	3	4	5	6	7	8
	2 <sup>0</sup>	2 <sup>1</sup>	2 <sup>2</sup>	2 <sup>3</sup>	2 <sup>4</sup>	2 <sup>5</sup>	2 <sup>6</sup>	
	ON	OFF	OFF	ON	ON	OFF	OFF	OFF



Set node number = 55:

55<sub>10</sub> = 0011 0111<sub>2</sub> (binary value)

bit	1	2	3	4	5	6	7	8
	2 <sup>0</sup>	2 <sup>1</sup>	2 <sup>2</sup>	2 <sup>3</sup>	2 <sup>4</sup>	2 <sup>5</sup>	2 <sup>6</sup>	
	ON	ON	ON	OFF	ON	ON	OFF	OFF



**WARNING**

If baud rate and node number are set via software, master device has to detect the baud rate of the slave (scanning of baud rate) when the encoder is being installed. Once communication has been activated new baud rate and node number values can be set (objects **3000-00 Baud rate** and **3001-00 Node-ID**). After having set new values, transmit a **Reset node** command and then save parameters (object **1010-01 Store parameters**). To avoid conflict between Slaves, this operation has to be carried out when one only device is connected to the network.



### Step 2 - Set object 6501-00 Hardware counts per revolution

See resolution table below in the page for B0, B1, B2 and B3 values.

Master → Encoder

COB-ID	Cmd	Index			Sub	Process data			
600+ID	23	01	65	00	B0	B1	B2	B3	

Encoder → Master

COB-ID	Cmd	Index			Sub	Process data			
580+ID	60	01	65	00	00	00	00	00	

### Step 3 - Set object 6502-00 Hardware number of turns

See resolution table below in the page for B4, B5, B6 and B7 values.

Master → Encoder

COB-ID	Cmd	Index			Sub	Process data			
600+ID	23	02	65	00	B4	B5	B6	B7	

Encoder → Master

COB-ID	Cmd	Index			Sub	Process data			
580+ID	60	02	65	00	00	00	00	00	

### Step 4 - Send a Reset node command

Master → Encoder

COB-ID	Cmd	Slave ID
000	81	ID

### Step 5 - Store parameters (object 1010-01 Store parameters)

Master → Encoder

COB-ID	Cmd	Index			Sub	Process data			
600+ID	23	10	10	01	73	61	76	65	

Encoder → Master

COB-ID	Cmd	Index			Sub	Process data			
580+ID	60	10	10	01	00	00	00	00	

### Table of resolutions

Encoder type	steps/rev.				n° rev.			
	B0	B1	B2	B3	B4	B5	B6	B7
ASx58x 12/CB-xx	00	10	00	00	01	00	00	00
ASx58x 13/CB-xx	00	20	00	00	01	00	00	00
AMx58x 12/4096CB-xx	00	10	00	00	00	10	00	00
AMx58x 13/4096CB-xx	00	20	00	00	00	10	00	00

## 5.2 Using the default settings

Using the default settings provided by the manufacturer, you can switch on the device and read immediately its position.

Follow the instructions below to:

- read the device resolution;
- set the **operational** mode;
- read the current position (cyclic mode and/or sync mode).

Default Baud rate and Node-ID are:

**Baud rate = 500 Kbit/s**

**Node-ID = 1**

### Read resolution per revolution **6501-00 Hardware counts per revolution**

Master → Encoder

COB-ID	Cmd	Index	Sub	Process data			
601	40	01 65	00	-	-	-	-

Encoder → Master

COB-ID	Cmd	Index	Sub	Process data			
581	43	01 65	01	A0	A1	A2	A3

steps/rev. = ( A3<<24 | A2<<16 | A1<<8 | A0 )

### Read number of revolutions **6502-00 Hardware number of turns**

Master → Encoder

COB-ID	Cmd	Index	Sub	Process data			
601	40	02 65	00	-	-	-	-

Encoder → Master

COB-ID	Cmd	Index	Sub	Process data			
581	43	02 65	01	B0	B1	B2	B3

N. rev. = ( B3<<24 | B2<<16 | B1<<8 | B0 )

### Set cyclic time **6200-00 Cyclic time (100 ms = 64h)**

Master → Encoder

COB-ID	Cmd	Index	Sub	Process data			
600+ID	2B	00 62	00	64	00	-	-

Encoder → Master

COB-ID	Cmd	Index	Sub	Process data			
580+ID	60	00 62	00	00	00	-	-

### Set **Operational** mode

Master → Encoder

COB-ID	Cmd	Node
000	01	01

### Read the position every 100 ms

Encoder → Master

COB-ID	Byte 0	Byte 1	Byte 2	Byte 3
181	Low	...	...	High

## 6 - CANopen® interface

Lika encoders are always slave devices and comply with "Device profile for encoders", Class 2.

For any omitted information, refer to the "CiA Draft Standard 301" and "CiA Draft Standard 406" documents available at [www.can-cia.org](http://www.can-cia.org).

### 6.1 EDS file

CANopen® encoders are supplied with their own EDS file **Lika\_AxxCB\_DS406\_Vx.edb** (see enclosed documentation or click [www.lika.biz](http://www.lika.biz) > ROTARY ENCODERS > ABSOLUTE ENCODERS > CAN).

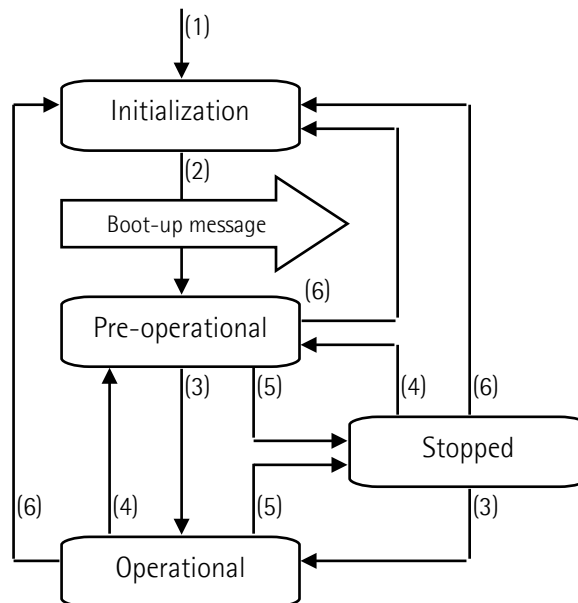
EDS file has to be installed on CANopen® master device.

**Lika\_AMxCB\_DS406\_Vx.edb**: for multiturn encoder.

**Lika\_ASxCB\_DS406\_Vx.edb**: for singleturn encoder.

### 6.2 Operation states

CANopen® devices are designed to operate using different states. Transition from one state to another is made by sending specific NMT messages (see Figure below).



(1)	Power on
(2)	Initialization carried out, boot-up message is sent automatically
(3)	NMT message: <b>Start remote node</b>
(4)	NMT message: <b>Enter pre-operational</b>
(5)	NMT message: <b>Stop remote node</b>
(6)	NMT message: <b>Reset node or Reset communication</b>

### 6.2.1 Initialization state

This is the first state the CANopen® device enters after power is turned on or after a hardware reset. As soon as the basic CANopen® device initialization is carried out, device reads and loads the parameters saved on EPROM, sends a boot-up message and then switches automatically to **Pre-operational** state.

### 6.2.2 Pre-operational state

In this state communication between master and slave is possible using SDO messages. They allow working parameters to be set. Slave cannot send PDO messages.

To switch slave device to **Operational** state master must send a **Start remote node** command using a NMT message.

### 6.2.3 Operational state

In this state slave device is active and all communication objects are available. Slave device can use the parameters available in the "Object dictionary" and is allowed to send process data using PDO messages. "Object dictionary" can be accessed using SDO messages. To switch slave device to **Pre-operational** state master must send an **Enter pre-operational** command using a NMT message.

### 6.2.4 Stopped state

In this state slave device is forced to interrupt communication with the Master (except node guarding, if active).

Communication using PDO and SDO messages is not allowed.

To switch slave device to **Pre-operational** or **Operational** state master must send the specific commands **Enter pre-operational** or **Start remote node** using a NMT message.



### 6.3 Communication objects

Four different kinds of communication messages are used in a CANopen® network:

- Network management NMT protocol: NMT protocols are used to issue state machine change commands (i.e. to start and stop the devices), detect remote device boot-ups and error conditions.
- Process Data Objects PDO protocol: used to process real time data (transmission of process data in real time).
- Service Data Objects SDO protocol: used to set and read values from the object dictionary of a remote device.
- Special Function Objects:
  - SYNC: synchronization message used by master to enable slaves devices to transmit process data (encoder position and velocity).
  - Emergency: error messages are triggered by each error event.
  - Nodeguard: used to request the state of the slave.

Relation between device states and communication objects:

	Initial.	Pre-oper.	Operat.	Stopped
NMT		X	X	X
PDO			X	
SDO		X	X	
SYNC			X	
EMCY		X	X	
Boot-up	X			
Node guard		X	X	X

#### 6.3.1 Pre-defined connection set

Master → Slave broadcast		
Type of COB (Object)	Function code (binary)	COB-ID (hex)
NMT	0000	000
SYNC	0001	080

peer-to-peer transmission		
EMERGENCY	0001	081 - 0FF
PDO 1 (tx)	0011	181 - 1FF
PDO 2 (tx)	0101	281 - 2FF
PDO 3 (tx)	0111	381 - 3FF
PDO4 (tx)	1001	481 - 4FF
SDO (tx)	1011	581 - 5FF
SDO (rx)	1100	601 - 67F
Node guard	1110	701 - 77F
Boot-up	1110	701 - 77F

The type of COB (tx or rx) is viewed from the slave device.

### 6.4 NMT objects

NMT structure:

COB-ID (11 bit)		2 CAN Data Bytes	
Func.code	Node ID	Command	Slave ID
0000	0	NMT Func.	Slave ID

If Slave ID = 00h, the NMT message is issued to all network nodes.

NMT Function:

Command	NMT Function	State node
01 hex	Start remote node	Operational
02 hex	Stop remote node	Stopped
80 hex	Enter pre-operational	Pre-operational
81 hex	Reset node	Pre-operational
82 hex	Reset communication	Pre-operational

### 6.5 Boot-up objects

Boot-up message structure:

COB-ID(hex)	1 CAN Data Bytes
700+Node ID	00

### 6.6 PDO objects

PDO (tx) messages are always made up of 4 CAN Data Bytes and are used by the encoder to transmit the position value and/or the velocity value.

PDO structure:

IDENTIFIER		4 CAN Data Bytes			
COB-ID(hex)		Byte 0	Byte 1	Byte 2	Byte 3
F.C.	Node-ID	Low	...	...	High
		position value (with PDO1, PDO2, PDO3)			
		velocity value (with PDO4)			

#### **PDO1 Cyclic mode: cyclic transmission**

Encoder uses the PDO1 message to transmit the position value cyclically, i.e. periodically and independently from the Master.

The interval between two issues is set in the object **6200-00 Cyclic time**.

To activate (or deactivate) cyclic mode it is necessary to set to 0 (or 1) the most significant bit of COB-ID used by PDO1 (object **1800 PDO1 parameters**, sub 1).

#### **PDO2 and PDO3 SYNC mode: synchronous transmission**

Transmission of position value is managed by the Master by sending a SYNC message.

SYNC message is a high-priority COB transmitted by the Master to request the position value of the encoder.

If several nodes (encoders) are connected to the network, the Master receives the position values from the slaves respecting the Node number order.

The encoder can be programmed to send a reply after a set number of SYNC messages by setting a counter.

The PDO message will be transmitted after having received the set number of SYNC messages.

For PDO2 the value of the counter must be set in object **1801 PDO2 parameters**, sub 2.

For PDO3 refer to object **1802 PDO3 parameters**, sub 2.

SYNC transmission mode can be enabled (or disabled) by setting to 0 (or 1) the most significant bit (MSB) of COB-IB used by PDO (objects **1801 PDO2 parameters** / **1802 PDO3 parameters**, sub1).

#### **PDO4 Cyclic mode: cyclic transmission**

Encoder uses the PDO4 message to transmit the velocity value cyclically, i.e. periodically and independently from the Master.

The interval between two issues is set in the object **6200-00 Cyclic time**.

To activate (or deactivate) the cyclic mode it is necessary to set to 0 (or 1) the most significant bit of COB-ID used by PDO4 (object **1803 PDO4 parameters**, sub 1).



#### **NOTE**

Several transmission modes can be active at the same time.

## 6.7 SDO objects

SDO messages are used to set and read values from the object dictionary of the encoder. These parameters are described in the "Object dictionary" section (page 29).

4 bytes at the most are used for CAN data, other 4 bytes are used for Command, Index and Sub-index fields. SDO messages are always followed by confirmation. It follows that when Master sends a SDO message to Slave, slave always sends a reply (and a warning, should an error occur).

SDO structure:

IDENTIFIER		from 4 to 8 CAN data bytes							
COB-ID(hex)		0	1	2	3	4	5	6	7
F.C.	Node-ID	Com	Index		Sub	Data			
		1byte	LSB	MSB	1byte	LSB	...	...	MSB

**Com**        command  
**Index**     parameter index  
**Sub**        parameter sub-index  
**Data**       parameter value

### 6.7.1 Command

The command byte contains the type of telegram transmitted to the CAN network.

Three types of telegram are available:

- Set: to send configuration parameters to a device;
- Req: used by master to read data from a device;
- Warnings: used by slave to send error messages to master (e.g. **Index does not exist, ...**).

Command	COB	COB type	Data length
22h	Set	M → S request	not spec.
23h	Set	M → S request	4 byte
2Bh	Set	M → S request	2 byte
2Fh	Set	M → S request	1 byte
60h	Set	S → M confirmation	0 byte
40h	Req	M → S request	0 byte
42h	Req	S → M reply	not spec.
43h	Req	S → M reply	4 byte
4Bh	Req	S → M reply	2 byte
4Fh	Req	S → M reply	1 byte
41h	Req	S → M reply segmented SDO	
80h	Warning	S → M reply	4 byte

## 6.8 Object dictionary

In the following pages the objects implemented are listed and described as follows:

### Index-subindex Object name

[data types, attribute]

- Index and subindex are expressed in hexadecimal notation.
- Attribute:
  - ro = read only access
  - rw = read and write access

Unsigned16 data type:

Process data bytes	
byte 4	byte 5
LSByte	MSByte

Unsigned32 data type:

Process data bytes			
byte 4	byte 5	byte 6	byte 7
LSByte	...	...	MSByte

### 6.8.1 Standard objects (DS 301)

#### 1000-00 Device type

[Unsigned32, ro]

Default = 0001 0196h = singleturn encoder, DS 406

0002 0196h = multiturn encoder, DS 406

#### 1001-00 Error register

[Unsigned8, ro]

Should an error occur, bit 0 in this object will be set to "1".

Default = 00h

#### 1003 Pre-defined error field

This object contains the last four errors which generated an emergency message.

- **00 Number of current errors** [Unsigned8, rw]  
(write 00h to delete the error history)
- **01 Last error occurred** [Unsigned32, ro]
- **02-04 Previous errors occurred** [Unsigned32, ro]

**1005-00 COB\_ID SYNC message**

[Unsigned32, rw]

Default = 0000 0080h

**1008-00 Name of device manufacturer**

[String, ro]

It shows the name of the device manufacturer.

Default = "LIKA SRL"

**1009-00 Hardware version**

[String, ro]

It shows the hardware version of the device.

**100A-00 Software version**

[String, ro]

It shows the software version of the device.

**100C-00 Guard time**

[Unsigned16, rw]

It contains the Guard time expressed in msec (milliseconds).

Object **100C-00 Guard time** is used in the "Node guarding protocol" controlled by the master. For more details see section "6.11 Node guarding protocol" on page 49.

Default = 0000h

**100D-00 Life time factor**

[Unsigned8, rw]

Object **100D-00 Life time factor** is used in the "Node guarding protocol" controlled by the master. For more details see section "6.11 Node guarding protocol" on page 49.

Default = 00h

**1010-01 Store parameters**

[Unsigned32, rw]

Use this object to save all parameters on non-volatile memory.

Write "save" (ASCII code in hexadecimal form) in the data bytes:

Master → Encoder

COB-ID	Cmd	Index	Sub	Data bytes
600+ID	23	10	10	01 73 61 76 65

Encoder → Master (confirmation)

COB-ID	Cmd	Index	Sub	Data bytes
580+ID	60	10	10	01 00 00 00 00

**1011-01 Restore default parameters**

[Unsig32, rw]

This object allows the operator to restore all parameters to default values (default values are set at the factory by Lika Electronic engineers to allow the operator to run the device for standard operation in a safe mode).

Write "load" (ASCII code in hexadecimal form) in the data bytes and then issue a **Reset node** command:

Master → Encoder

COB-ID	Cmd	Index	Sub	Data bytes
600+ID	23	11	10	01 6C 6F 61 64

Encoder → Master (confirmation)

COB-ID	Cmd	Index	Sub	Data bytes
580+ID	60	11	10	01 00 00 00 00

Master → Encoder (**Reset node**)

COB-ID	Cmd	Slave ID
000	81	ID

Encoder → Master (Boot-up)

COB-ID	Cmd
700+ID	00



**NOTE**

Save default values using the store parameters function (see object **1010-01 Store parameters**).

**1014-00 COB-ID EMCY**

[Unsigned32, rw]

This object defines the COB-ID used for sending emergency messages (EMCY). If the node address is set using the internal dip-switches (i.e. at least one dip-switch for setting the node has HIGH logic level = 1), when the power is turned on, this object is always forced to the default value. Otherwise, if the node address is set via software (i.e. all dip-switches for setting the node have LOW logic level = 0) it retains the set value, unless a software procedure for setting a new address is forced at power on.

Default = 0000 0080h+NodeID

**1015-00 Inhibit time EMCY**

[Unsigned16, rw]

Inhibit time of the emergency messages (EMCY) expressed in multiples of 100 µs. When set to 0, this function is disabled.

Default = 0000h

**1018 Identification object**

- **01 Vendor number** [Unsigned32, ro]
- **02 Product number** [Unsigned32, ro]
- **03 Revision number** [Unsigned32, ro]

**1800 PDO1 parameters**

PDO1 message is used by default for cyclic transmission of the position value.

For more information refer to section "6.6 PDO objects" on page 27.

See object **6200-00 Cyclic time** to set the cyclic timer.

- **01 COB-ID of PDO1** [Unsigned32, rw]

Bit number	Value	Meaning
31 (MSB)	0	PDO exists / is valid
	1	PDO does not exist / is not valid
30	0	RTR allowed on this PDO ( <b>not implemented</b> )
	1	no RTR allowed on this PDO
29	0	11-bit ID (CAN 2.0A)
	1	29-bit ID (CAN 2.0B)
28 ... 11	0	if bit 29 = 0
	X	if bit 29 = 1: bits 28-11 of 29-bit-COB-ID
10 ... 0 (LSB)	X	bits 10-0 of COB-ID

Default = 4000 0180h+NodeID (no RTR, COB-ID)





**WARNING**

It is mandatory to set the bit 30 of COB-ID to 1 (value 0 is not allowed). This means that "No RTR is allowed on the PDO".

If the node address is set using the internal dip-switches (i.e. at least one dip-switch for setting the node has HIGH logic level = 1), when the power is turned on, this object is always forced to the default value. Otherwise, if the node address is set via software (i.e. all dip-switches for setting the node have LOW logic level = 0) it retains the set value, unless a software procedure for setting a new address is forced at power on.

- **02 Transmission type** [Unsigned8, rw]

Transmission type	PDO transmission	
00h (0)	Acyclic, synchronous	not implemented
01h ... F0h (1 ... 240)	Cyclic, synchronous	implemented
F1h ... FBh (241 ... 251)	not implemented - reserved	
FCh (252)	Synchronous, RTR only	not implemented
FDh (253)	Asynchronous, RTR only	not implemented
FEh (254)	Asynchronous, manufacturer specific	implemented
FFh (255)	Asynchronous, device profile specific	not implemented

Default = FEh (cyclic transmission, see hereafter and object **6200-00 Cyclic time**)



**WARNING**

Following an attempt to set the **Transmission Type** to 0, the value is accepted but the PDO message is not sent; following an attempt to change the **Transmission Type** to any other value that is not supported by the device, an abort message (abort code = 0609 0030h: **Value range of parameter exceeded**) is generated.

If the value next to the object **6200-00 Cyclic time** ≠ 0, the PDO message is sent cyclically and the interval between two messages is the time set next to the object **6200-00 Cyclic time**; otherwise, if the value next to the object **6200-00 Cyclic time** = 0, the PDO message is not sent.

**1801 PDO2 parameters**

PDO2 message is used by default for synchronous transmission of the position value. For more information refer to section "6.6 PDO objects" on page 27.

- **01 COB-ID of the PDO2** [Unsigned32, rw]

Bit number	Value	Meaning
31 (MSB)	0	PDO exists / is valid
	1	PDO does not exist / is not valid
30	0	RTR allowed on this PDO ( <b>not implemented</b> )
	1	no RTR allowed on this PDO
29	0	11-bit ID (CAN 2.0A)
	1	29-bit ID (CAN 2.0B)
28 ... 11	0	if bit 29 = 0
	X	if bit 29 = 1: bits 28-11 of 29-bit-COB-ID
10 ... 0 (LSB)	X	bits 10-0 of COB-ID

Default = 4000 0280h+NodeID (no RTR, COB-ID)



**WARNING**

It is mandatory to set the bit 30 of COB-ID to 1 (value 0 is not allowed). This means that "No RTR is allowed on the PDO".

If the node address is set using the internal dip-switches (i.e. at least one dip-switch for setting the node has HIGH logic level = 1), when the power is turned on, this object is always forced to the default value. Otherwise, if the node address is set via software (i.e. all dip-switches for setting the node have LOW logic level = 0) it retains the set value, unless a software procedure for setting a new address is forced at power on.

- **02 Transmission type** [Unsigned8, rw]

Transmission type	PDO transmission	
00h (0)	Acyclic, synchronous	<b>not implemented</b>
01h ... F0h (1 ... 240)	Cyclic, synchronous	<b>implemented</b>
F1h ... FBh (241 ... 251)	<b>not implemented - reserved</b>	
FCh (252)	Synchronous, RTR only	<b>not implemented</b>
FDh (253)	Asynchronous, RTR only	<b>not implemented</b>
FEh (254)	Asynchronous, manufacturer specific	<b>implemented</b>
FFh (255)	Asynchronous, device profile specific	<b>not implemented</b>

Default = 01h (synchronous transmission at each SYNC command)

Position value is transmitted after the set number of SYNC commands. Interval in SYNC commands must be set next to this object **1801 PDO2 parameters**, sub 2.



**WARNING**

Following an attempt to set the **Transmission Type** to 0, the value is accepted but the PDO message is not sent; following an attempt to change the **Transmission Type** to any other value that is not supported by the device, an abort message (abort code = 0609 0030h: **Value range of parameter exceeded**) is generated.

If the value next to the object **6200-00 Cyclic time** ≠ 0, the PDO message is sent cyclically and the interval between two messages is the time set next to the object **6200-00 Cyclic time**; otherwise, if the value next to the object **6200-00 Cyclic time** = 0, the PDO message is not sent.

**1802 PDO3 parameters**

PDO3 message is used by default for synchronous transmission of the position value. For more information refer to section "6.6 PDO objects" on page 27.

- **01 COB-ID of the PDO3** [Unsigned32, rw]

Bit number	Value	Meaning
<b>31 (MSB)</b>	0	PDO exists / is valid
	1	PDO does not exist / is not valid
<b>30</b>	0	RTR allowed on this PDO ( <b>not implemented</b> )
	1	no RTR allowed on this PDO
<b>29</b>	0	11-bit ID (CAN 2.0A)
	1	29-bit ID (CAN 2.0B)
<b>28 ... 11</b>	0	if bit 29 = 0
	X	if bit 29 = 1: bits 28-11 of 29-bit-COB-ID
<b>10 ... 0 (LSB)</b>	X	bits 10-0 of COB-ID

Default = C000 0380h+NodeID (disable, no RTR)



**WARNING**

It is mandatory to set the bit 30 of COB-ID to 1 (value 0 is not allowed). This means that "No RTR is allowed on the PDO".

If the node address is set using the internal dip-switches (i.e. at least one dip-switch for setting the node has HIGH logic level = 1), when the power is turned on, this object is always forced to the default value. Otherwise, if the node address is set via software (i.e. all dip-switches for setting the node have LOW logic level = 0) it retains the set value, unless a software procedure for setting a new address is forced at power on.

- 02 Transmission type [Unsigned8, rw]

Transmission type	PDO transmission	
00h (0)	Acyclic, synchronous	not implemented
01h ... F0h (1 ... 240)	Cyclic, synchronous	implemented
F1h ... FBh (241 ... 251)	not implemented - reserved	
FCh (252)	Synchronous, RTR only	not implemented
FDh (253)	Asynchronous, RTR only	not implemented
FEh (254)	Asynchronous, manufacturer specific	implemented
FFh (255)	Asynchronous, device profile specific	not implemented

Default = 01h (synchronous transmission at each SYNC command)  
 Position value is transmitted after the set number of SYNC commands.  
 Interval in SYNC commands must be set next to this object **1802 PDO3 parameters**, sub 2.



**WARNING**

Following an attempt to set the **Transmission Type** to 0, the value is accepted but the PDO message is not sent; following an attempt to change the **Transmission Type** to any other value that is not supported by the device, an abort message (abort code = 0609 0030h: **Value range of parameter exceeded**) is generated.

If the value next to the object **6200-00 Cyclic time** ≠ 0, the PDO message is sent cyclically and the interval between two messages is the time set next to the object **6200-00 Cyclic time**; otherwise, if the value next to the object **6200-00 Cyclic time** = 0, the PDO message is not sent.

**1803 PDO4 parameters**

PDO4 is used by default for cyclic transmission of the velocity value.  
 For more information refer to section "6.6 PDO objects" on page 27.  
 See object **6200-00 Cyclic time** to set the cyclic timer.

- 01 COB-ID of PDO4 [Unsigned32, rw]

Bit number	Value	Meaning
31 (MSB)	0	PDO exists / is valid
	1	PDO does not exist / is not valid
30	0	RTR allowed on this PDO ( <b>not implemented</b> )
	1	no RTR allowed on this PDO
	0	11-bit ID (CAN 2.0A)

29	1	29-bit ID (CAN 2.0B)
28 ... 11	0	if bit 29 = 0
	X	if bit 29 = 1: bits 28-11 of 29-bit-COB-ID
10 ... 0 (LSB)	X	bits 10-0 of COB-ID

Default = C000 0480h+NodeID (no RTR, COB-ID)



**WARNING**

It is mandatory to set the bit 30 of COB-ID to 1 (value 0 is not allowed). This means that "No RTR is allowed on the PDO".

If the node address is set using the internal dip-switches (i.e. at least one dip-switch for setting the node has HIGH logic level = 1), when the power is turned on, this object is always forced to the default value. Otherwise, if the node address is set via software (i.e. all dip-switches for setting the node have LOW logic level = 0) it retains the set value, unless a software procedure for setting a new address is forced at power on.

- 02 Transmission type [Unsigned8, rw]

Transmission type	PDO transmission	
00h (0)	Acyclic, synchronous	not implemented
01h ... F0h (1 ... 240)	Cyclic, synchronous	implemented
F1h ... FBh (241 ... 251)	not implemented - reserved	
FCh (252)	Synchronous, RTR only	not implemented
FDh (253)	Asynchronous, RTR only	not implemented
FEh (254)	Asynchronous, manufacturer specific	implemented
FFh (255)	Asynchronous, device profile specific	not implemented

Default = FEh (cyclic transmission, see hereafter and object **6200-00 Cyclic time**)



**WARNING**

Following an attempt to set the **Transmission Type** to 0, the value is accepted but the PDO message is not sent; following an attempt to change the **Transmission Type** to any other value that is not supported by the device, an abort message (abort code = 0609 0030h: **Value range of parameter exceeded**) is generated.

If the value next to the object **6200-00 Cyclic time** ≠ 0, the PDO message is sent cyclically and the interval between two messages is the time set next to the object **6200-00 Cyclic time**; otherwise, if the value next to the object **6200-00 Cyclic time** = 0, the PDO message is not sent.

**NOTE**

- Transmission of PDO1, PDO2, PDO3 and PDO4 messages can be enabled (or disabled) by setting to "0" (or "1") the most significant bit (MSB) used by PDO (object **180xh**, sub1).
- Cyclic transmission or synchronous transmission can be modified by setting the object **180xh** sub 2:
  - 01h: synchronous transmission at each SYNC command;
  - 02h: synchronous transmission after two SYNC commands;
  - ...
  - FEh: cyclic transmission:
    - if **6200-00 Cyclic time**  $\neq 0$   $\rightarrow$  "cyclic transmission": cycle time is set next to the object 6200h;
    - if **6200-00 Cyclic time** = 0  $\rightarrow$  the PDO message is not sent.

**1A00-01 PDO1 mapping parameter**

[Unsig32, rw]

This object contains the mapped position value of the encoder according to DS406 device profile.

Default = 6004 0020h

**1A01-01 PDO2 mapping parameter**

[Unsig32, rw]

See object **1A00-01 PDO1 mapping parameter**.

**1A02-01 PDO3 mapping parameter**

[Unsig32, rw]

See object **1A00-01 PDO1 mapping parameter**.

**1A03-01 PDO4 mapping parameter**

[Unsig32, rw]

This object contains the mapped velocity value of the encoder according to the manufacturer profile.

Default = 3006 0020h

## 6.8.2 Manufacturer specific objects

### 2104-00 Limit switch min.

[Unsigned32, rw]

This object can be used to set the lowest software limit switch (-).

If the encoder position is below the value set in this object, bit 12 of object **6500-00 Operating states** will be set to 1.

To enable this function set bit 12 of object **6000-00 Operating parameters** to 1.

Default = 0000 0010h

### 2105-00 Limit switch max.

[Unsigned32, rw]

This object can be used to set the highest software limit switch. (+).

If the encoder position is higher than the value set in this object, bit 13 of object **6500-00 Operating states** will be set to 1.

To enable this function set bit 13 of object **6000-00 Operating parameters** to 1.

Default = 003F FFF0h

### 3000-00 Baud rate

[Unsigned8, rw]

This object is meant to set the baud rate (transmission rate) according to the following table:

Data byte	Baud rate
00h	20 Kbit/s
01h	50 Kbit/s
02h	100 Kbit/s
03h	125 Kbit/s
04h	250 Kbit/s
<b>05h</b>	<b>500 Kbit/s (default)</b>
06h	800 Kbit/s
07h	1000 Kbit/s

The bit rate is set through object **3000-00 Baud rate** only if bit 4 in DIP A is set to "OFF". If bit 4 in DIP A is set to "ON", the bit rate is set by DIP A. For any further information refer to section "4.8 Baud rate: DIP A" on page 18.

To change the baud rate value you have to:

- set the object **3000-00 Baud rate**;
- send a **Reset node** command (or **Reset communication** command);
- save the parameter;
- set the Master to the new baud rate.

Default = 05h

Master → Encoder

COB-ID	Cmd	Index	Sub	Data byte
600+ID	2F	00	30	00
				see table

Encoder → Master (confirmation)

COB-ID	Cmd	Index	Sub	Data byte
580+ID	60	00	30	00

Master → Encoder (Reset node)

COB-ID	Cmd	Slave ID
000	81	ID

Set the master device to the new baud rate:

Encoder → Master (Boot-up with new baud rate)

COB-ID	Cmd
700+ID	00



**NOTE**

To save the new Baud rate value execute the store parameters function (see object **1010-01 Store parameters**).

When the power is turned off, parameters not saved will be lost.

**3001-00 Node-ID**

[Unsigned8, rw]

This object defines the node identifier of the device.

The node number is set through the object **3001-00 Node-ID** only if all bits of DIP B are set to "OFF". If one bit at least of DIP B is set to "ON" the node number is set through DIP B. For any further information refer to section "4.9 Node number: DIP B" on page 19.

To change the Node-ID value you have to:

- set the object **3001-00 Node-ID**;
- send a **Reset node** command;
- save the parameter.

Default = 01h

Master → Encoder

COB-ID	Cmd	Index	Sub	Data byte
600+ID	2F	01	30	00
				new Node-ID

Encoder → Master (confirmation)

COB-ID	Cmd	Index	Sub	Data byte
580+ID	60	01	30	00



Master → Encoder (**Reset node**)

COB-ID	Cmd	Slave ID
000	81	old ID

Encoder → Master (Boot-up with new Node-ID)

COB-ID	Cmd
700+ID	00



**NOTE**

To save the new Node-ID value execute the store parameters function (see object **1010-01 Store parameters**).

When the power is turned off, parameters not saved will be lost.

**3005-00 Velocity format**

[Unsigned8, rw]

This attribute defines the engineering units for the velocity value.

00h = steps/s: number of steps per second (default);

01h = rpm: revolutions per minute.

**3006-00 Velocity value**

[Unsigned32, ro]

This attribute shows the current speed detected by the position sensor and calculated every 100 ms.

Value is transmitted according to the settings in the object **1803 PDO4 parameters**.

6.8.3 Device profile objects (DS 406)

6000-00 Operating parameters

[Unsigned16, rw]

Bit	Function	bit = 0	bit = 1
0	Code sequence	CW (clockwise)	CCW (counter clockwise)
1	not used		
2	Scaling function	disabled	enabled
3...11	not used		
12	Limit switch min.	disabled	enabled
13	Limit switch max.	disabled	enabled
14...15	not used		

Default = 0000h

Code sequence

It defines whether the position value output by the transducer increases when the encoder shaft rotates clockwise (CW) or counterclockwise (CCW). When **Code sequence** = 0, the position value increases when the encoder shaft rotates clockwise; on the contrary, when **Code sequence** = 1, the position value increases when the encoder shaft rotates counterclockwise. CW and CCW rotations are viewed from shaft end.

Scaling function

If disabled the device uses its own physical resolution (see objects **6501-00 Hardware counts per revolution** and **6502-00 Hardware number of turns**); if enabled it uses the resolution set in objects **6001-00 Counts per revolution** and **6002-00 Total resolution** with the following relation:

Transmitted position =

$$\frac{6001-00 \text{ Counts per revolution}}{6501-00 \text{ Hardware counts per revolution}} * \text{real position} \leq 6002-00 \text{ Total resolution}$$



WARNING

Every time you enable the scaling function and/or change the scaling values (see objects **6001-00 Counts per revolution** and **6002-00 Total resolution**) then you are required to set a new preset value (see object **6003-00 Preset value**) and finally save the new parameters (see object **1010-01 Store parameters**).

Limit switch min.

Limit switch max.

It allows to enable / disable the function of the objects **2104-00 Limit switch min.** and **2105-00 Limit switch max.**. For further information see on page 39.

**6001-00 Counts per revolution**

[Unsig32, rw]

This object sets a custom number of distinguishable steps per revolution.

Enabled if bit 2 **Scaling function** in the object **6000-00 Operating parameters** = "1".

To avoid counting errors, check that

$$\frac{\text{6501-00 Hardware counts per revolution}}{\text{6001-00 Counts per revolution}} = \text{integer value.}$$

Allowed values are equal to or lower than **6501-00 Hardware counts per revolution** (see encoder label).



**WARNING**

Every time you change the value in this object then you are required to set a new preset value (see object **6003-00 Preset value**) and finally save the new parameters (see object **1010-01 Store parameters**).

**6002-00 Total resolution**

[Unsigned32, rw]

This object sets a custom number of distinguishable steps over the total measuring range.

Enabled if bit 2 **Scaling function** in the object **6000-00 Operating parameters** = "1".

Allowed values are equal to or lower than **Total hardware resolution** (**6501-00 Hardware counts per revolution** \* **6502-00 Hardware number of turns**, see encoder label).



**WARNING**

Every time you change the value in this object then you are required to set a new preset value (see object **6003-00 Preset value**) and finally save the new parameters (see object **1010-01 Store parameters**).



**Example**

Multiturn encoder AM5812/4096CB-6 with CC-CB

Resolution is:

- Hardware counts per revolution: **6501-00 Hardware counts per revolution** = 4096 ( $2^{12}$ )
- Hardware number of turns: **6502-00 Hardware number of turns** = 4096 ( $2^{12}$ )
- Total hardware resolution: = 16777216 ( $2^{24}$ )

2048 counts/rev. \* 1024 turns are required:

- Enable scaling function: **6000-00 Operating parameters**, bit 2 = "1"
- Counts per revolution: **6001-00 Counts per revolution** = 2048 (0000 0800h)
- Total resolution: **6002-00 Total resolution** = 2048\*1024 = 2097152 (0020 0000h)



**NOTE**

We suggest values which are power of 2 ( $2^n$ : 2, 4, ..., 2048, 4096, 8192,...) to be set in the objects **6001-00 Counts per revolution** and **6002-00 Total resolution** to avoid counting errors.

If **6001-00 Counts per revolution** and/or **6002-00 Total resolution** values change, then the **6003-00 Preset value** must be updated to the new resolution. A new **6003-00 Preset value** value is also required.

**6003-00 Preset value**

[Unsigned32, rw]

This object allows to set the encoder position to a Preset value. Preset function is meant to assign a certain value to a desired physical position of the encoder shaft.

Default = 0000 0000h



**NOTE**

- If the scaling function is disabled (see bit 2 **Scaling function** in the object **6000-00 Operating parameters**), **6003-00 Preset value** must be lower than "Total hardware resolution" (**6501-00 Hardware counts per revolution** \* **6502-00 Hardware number of turns**).
- If the scaling function is enabled (see bit 2 **Scaling function** in the object **6000-00 Operating parameters**), **6003-00 Preset value** must be lower than **6002-00 Total resolution**.

### 6004-00 Position value

[Unsigned32, ro]

This object contains the position value, transmitted value is calculated considering the scaling values. Value is transmitted cyclically or synchronously according to the settings in the objects **1800 PDO1 parameters**, **1801 PDO2 parameters** and **1802 PDO3 parameters**.

### 6200-00 Cyclic time

[Unsigned16, rw]

Cyclic timer value is used in asynchronous transmission mode (**Transmission Type** = FEh) to set the interval between two PDO issues.

If the value next to this object **6200-00 Cyclic time**  $\neq 0$ , the PDO message is sent cyclically and the interval between two messages is the time set here; otherwise, if the value next to this object **6200-00 Cyclic time** = 0, the PDO message is not sent.

Value is expressed in milliseconds. See on page 27 and 32.

Default = 0000h

### 6500-00 Operating states

[Unsigned16, ro]

Bit	Function	bit = 0	bit = 1
0	<b>Code sequence</b>	Clockwise	CCW
1	not used		
2	<b>Scaling function</b>	Disabled	Enabled
3...11	not used		
12	<b>Limit switch min.</b>	posit. > <b>2104-00 Limit switch min.</b>	posit. < <b>2104-00 Limit switch min.</b>
13	<b>Limit switch max.</b>	posit. < <b>2105-00 Limit switch max.</b>	posit. > <b>2105-00 Limit switch max.</b>
14	not used		
15	<b>Current operating state</b>	<b>Stopped / Pre-operational</b>	<b>Operational</b>

#### Code sequence

It shows whether the code sequence is set to clockwise (CW) or counterclockwise (CCW). To set the code sequence to either CW or CCW you must set the bit 0 **Code sequence** of the object **6000-00 Operating parameters** to 0 / 1.

### Scaling function

It shows whether the scaling function is disabled or enabled. To disable / enable the scaling function you must set the bit 2 **Scaling function** of the object **6000-00 Operating parameters** to 0 / 1.

### Limit switch min.

If the encoder position is over the value set in the object **2104-00 Limit switch min.**, bit 12 of this object is set to "0".

If the encoder position is below the value set in the object **2104-00 Limit switch min.**, bit 12 of this object is set to "1".

To enable this function set the bit 12 **Limit switch min.** of the object **6000-00 Operating parameters** to "1".

### Limit switch max.

If the encoder position is below the value set in the object **2105-00 Limit switch max.**, bit 13 of this object is set to "0".

If the encoder position is over the value set in the object **2105-00 Limit switch max.**, bit 13 of this object is set to "1".

To enable this function set the bit 13 **Limit switch max.** of the object **6000-00 Operating parameters** to "1".

### Current operating state

It shows the current operating state of the unit. For further information on the available states see section "6.2 Operation states" on page 23.

bit 15 = 0: **Stopped** or **Pre-operational** state;

bit 15 = 1: **Operational** state.

### 6501-00 Hardware counts per revolution

[Unsigned32, ro]

This object defines the physical number of distinguishable steps each turn given by the hardware.

To set a custom resolution see object **6001-00 Counts per revolution**.

### 6502-00 Hardware number of turns

[Unsig16, ro]

This object defines the physical number of distinguishable turns given by the hardware.

**Total hardware resolution = 6501-00 Hardware counts per revolution \* 6502-00 Hardware number of turns.**

To set a custom number of turns see objects **6001-00 Counts per revolution** and **6002-00 Total resolution**.

**6504-00 Supported alarms**

[Unsigned16, ro]

Default = 0000h (Alarms not supported).

**6506-00 Supported warnings**

[Unsigned16, ro]

Default = 0000h (Warnings not supported).

**6507-00 Profile and software version**

[Unsig32, ro]

Contains the profile and software version.

Profile version for encoders = 3.1

Software version = 1.1

Default = 0301 0101h

**6508-00 Operating time**

[Unsigned32, ro]

Default = FFFF FFFFh (not used)

**6509-00 Offset value**

[Integer32, ro]

This object contains the Offset value. This value is the difference between the physical position of the encoder and the position relating to the preset value (see object **6003-00 Preset value**).

Default = 0000 0000h

**650A-01 Manufacturer offset value**

[Integer32, ro]

This object contains the manufacturer offset value. This is the difference between the physical zero position of the encoder (zero set mechanically) and the zero position set by the manufacturer (zero set via software).

Default = 0000 0000h

**650B-00 Serial number**

[Unsigned32, ro]

Default = FFFF FFFFh (not used)

**NOTE**

To save new parameters execute the store parameters function (see object **1010-01 Store parameters**).

When the power is turned off or in case of **Reset node** and **Restore node** commands, parameters not saved are lost.

### 6.9 Warning objects

For the complete list and meaning of the warning messages please refer to "SDO abort codes" section in "CiA Draft Standard 301" document available at [www.can-cia.org](http://www.can-cia.org).

### 6.10 Emergency objects

Emergency (EMCY) objects are issued by the device when an internal error occurs.

EMCY structure:

IDENTIFIER	CAN Data Byte			
	0	1	2	3...7
COB-ID(hex)				
see object <b>1014-00 COB-ID EMCY</b>	Error code		Error Sub-register	Specific code
	LSB	MSB	01	00...00

Available error codes:

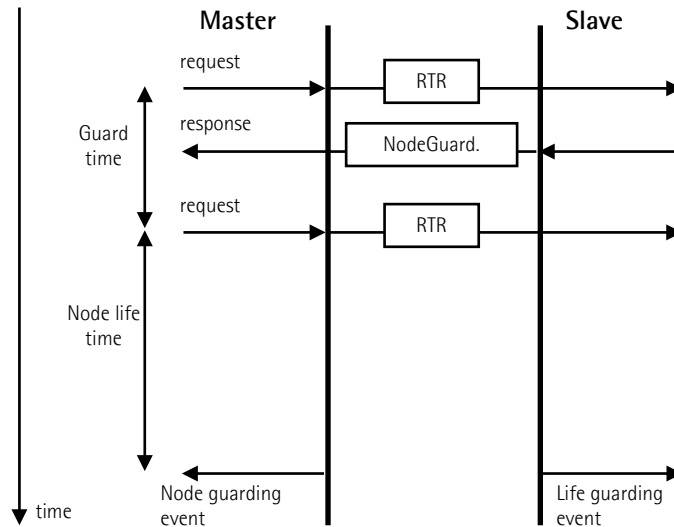
1000h = **Node guarding error**

5530h = **Flash memory error**



### 6.11 Node guarding protocol

At system boot the "Node guarding protocol" is disabled; this protocol is enabled automatically as soon as the master device sends a RTR message (Remote Transmission Request) the first time.



**100C-00 Guard time:** interval between two RTR messages.

**Node life time:** maximum time available for the encoder to receive a RTR message.

**Node life time = 100C-00 Guard time \* 100D-00 Life time factor.**

"Node guarding" is enabled if **Node life time** ≠ 0.

If the slave does not receive a RTR message before the **Node life time** has expired, it warns activating a "Life Guarding Event". Furthermore the red LED starts flashing so indicating the Node guarding error, objects **1001-00 Error register** and **1003 Pre-defined error field** are updated and an error message is sent.

To reset the error send a **Reset node** command.

## 7 – Setting-up

Here following are some examples of transmission between Master and Slave devices.

A generic "ID" value is used to indicate the encoder address; Master address is always 0. All values are expressed in hexadecimal notation.

### Set Operational, Pre-operational state

NMT message	Master → Slave		
	COB-ID	Cmd	Node
<b>Operational:</b>	000	01	ID
<b>Pre-operational:</b>	000	80	ID

### Set resolution per revolution **6001-00 Counts per revolution** ( $2^{10}=0000\ 0400h$ )

Master → Encoder (Set request)

COB-ID	Cmd	Index	Sub	Process data				
600+ID	23	01	60	00	00	40	00	00

Encoder → Master (Set confirmation)

COB-ID	Cmd	Index	Sub	Process data				
580+ID	60	01	60	00	00	00	00	00

### Set total resolution **6002-00 Total resolution** ( $2^{24}=0100\ 0000h$ )

Master → Encoder (Set request)

COB-ID	Cmd	Index	Sub	Process data				
600+ID	23	02	60	00	00	00	00	01

Encoder → Master (Set confirmation)

COB-ID	Cmd	Index	Sub	Process data				
580+ID	60	02	60	00	00	00	00	00

### Set operating parameters **6000-00 Operating parameters**

(Code sequence: CW, Scaling function: enabled, Limit switch min. / Limit switch max.: disabled)

Master → Encoder (Set request)

COB-ID	Cmd	Index	Sub	Process data				
600+ID	2B	00	60	00	04	00	-	-

Encoder → Master (Set confirmation)

COB-ID	Cmd	Index	Sub	Process data			
580+ID	60	00	60	00	00	-	-

Set the preset value **6003-00 Preset value** (preset = 1000 = 03E8h)

Master → Encoder (Set request)

COB-ID	Cmd	Index	Sub	Process data				
600+ID	23	03	60	00	E8	03	00	00

Encoder → Master (Set confirmation)

COB-ID	Cmd	Index	Sub	Process data				
580+ID	60	03	60	00	00	00	00	00

Set Sync counter **1801 PDO2 parameters**, sub 2 (n = 5 = 05h)

Master → Encoder (Set request)

COB-ID	Cmd	Index	Sub	Process data				
600+ID	2F	01	18	02	05	-	-	-

Encoder → Master (Set confirmation)

COB-ID	Cmd	Index	Sub	Process data				
580+ID	60	01	18	02	00	-	-	-

Disable Sync mode **1801 PDO2 parameters**, sub 1

Read COB-ID used by PDO2:

Master → Encoder (Req request)

COB-ID	Cmd	Index	Sub	Process data				
600+ID	40	01	18	01	-	-	-	-

Encoder → Master (Req reply)

COB-ID	Cmd	Index	Sub	Process data				
580+ID	43	01	18	01	B0	B1	B2	B3

COB-ID used by PDO2 = ( B3<<24 ) | ( B2<<16 ) | ( B1<<8 ) | B0 )

set the most significant bit to 1:

B3 |= 0x80;

Set new COB-ID used by PDO2 (**1801 PDO2 parameters**, sub 1):

Master → Encoder (Set request)

COB-ID	Cmd	Index	Sub	Process data				
600+ID	23	01	18	01	B0	B1	B2	B3

Encoder → Master (Set confirmation)

COB-ID	Cmd	Index	Sub	Process data				
580+ID	60	01	18	01	00	00	00	00

### Enable Cyclic mode

Set cyclic time **6200-00 Cyclic time** (100 ms = 64h)

Master → Encoder (Set request)

COB-ID	Cmd	Index	Sub	Process data				
600+ID	2B	00	62	00	64	00	-	-

Encoder → Master (Set confirmation)

COB-ID	Cmd	Index	Sub	Process data				
580+ID	60	00	62	00	00	00	-	-

Read COB-ID used by PDO1 (**1800 PDO1 parameters**, sub 1):

Master → Encoder (Req request)

COB-ID	Cmd	Index	Sub	Process data				
600+ID	40	00	18	01	-	-	-	-

Encoder → Master (Req reply)

COB-ID	Cmd	Index	Sub	Process data				
580+ID	43	00	18	01	B0	B1	B2	B3

COB-ID used by PDO1 = ( B3<<24 | B2<<16 | B1<<8 | B0 )

set the most significant bit to 0:

B3 &= 0x7F;

Set new COB-ID used by PDO1 (**1800 PDO1 parameters**, sub 1):

Master → Encoder (Set request)

COB-ID	Cmd	Index	Sub	Process data				
600+ID	23	00	18	01	B0	B1	B2	B3

Encoder → Master (Set confirmation)

COB-ID	Cmd	Index	Sub	Process data				
580+ID	60	00	18	01	00	00	00	00



#### NOTE

To save new parameters execute the store parameters function (see object **1010-01 Store parameters**).

When the power is turned off or in case of **Reset node** and **Restore node** commands, parameters not saved will be lost.

## 8 – Default parameters list

Default values are expressed in hexadecimal notation.

Parameters list	Default values		
1000-00 Device type	0x00010196 0x00020196		
1001-00 Error register	0x00		
1003 Pre-defined error field	-		
1005-00 COB_ID SYNC message	0x00000080		
1008-00 Name of device manufacturer	LIKA SRL*		
1009-00 Hardware version	-		
100A-00 Software version	-		
100C-00 Guard time	0x0000		
100D-00 Life time factor	0x00		
1014-00 COB-ID EMCY	NODEID+0x00000080		
1015-00 Inhibit time EMCY	0x0000		
1018 Identification object	-		
1800 PDO1 parameters, sub 1	NODEID+0x40000180		
1800 PDO1 parameters, sub 2	0xFE		
1801 PDO2 parameters, sub 1	NODEID+0x40000280		
1801 PDO2 parameters, sub 2	0x01		
1802 PDO3 parameters, sub 1	NODEID+0xC0000380		
1802 PDO3 parameters, sub 2	0x01		
1803 PDO4 parameters, sub 1	NODEID+0xC0000480		
1803 PDO4 parameters, sub 2	0xFE		
1A00-01 PDO1 mapping parameter	0x60040020		
1A01-01 PDO2 mapping parameter	0x60040020		
1A02-01 PDO3 mapping parameter	0x60040020		
1A03-01 PDO4 mapping parameter	0x36000020		
2104-00 Limit switch min.	0x00000010		
2105-00 Limit switch max.	0x003FFFF0		
3000-00 Baud rate	0x05		
3001-00 Node-ID	0x01		
3005-00 Velocity format	0x00		
6000-00 Operating parameters	0x0000		
6001-00 Counts per revolution	-		
6002-00 Total resolution	-		
6003-00 Preset value	0x00000000		
6200-00 Cyclic time	0x0000		
6500-00 Operating states	0x0000		
6504-00 Supported alarms	0x0000		
6506-00 Supported warnings	0x0000		
6507-00 Profile and software version	0x03010101		
6508-00 Operating time	0xFFFFFFFF		
6509-00 Offset value	0x00000000		
650A-01 Manufacturer offset value	0x00000000		
650B-00 Serial number	0xFFFFFFFF		

\* Text string

Page intentionally left blank

Page intentionally left blank

