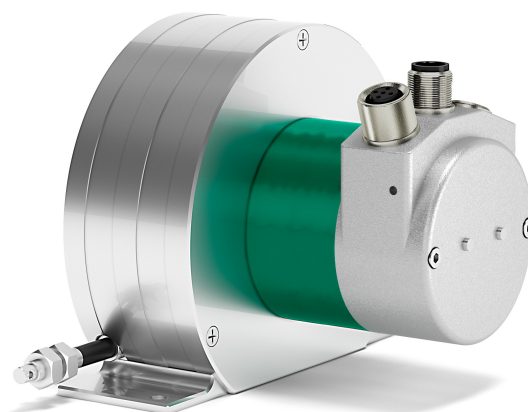


## SFAM1-05000-FD SFAM2-10000-FD



**DeviceNet™**

- 5000 mm (196.85") & 10000 (393.7") mm draw-wire encoder
- Integrated 25 bit absolute multiturn encoder
- Programmable resolution down to 24  $\mu$ m
- Cable and M12 connectors options
- DeviceNet interface, "Group 2 only server" devices

### Suitable for the following models:

- SFAM1-05000-FD2-08192-RM12
- SFAM1-05000-FD2-08192-RPG
- SFAM2-10000-FD2-08192-RM12
- SFAM2-10000-FD2-08192-RPG

### General Contents

Preliminary information	10
Mounting instructions	14
Electrical connections	18
Quick reference (using RSNetWorx)	27
DeviceNet interface	48
Setup	80
Default parameters table	84

This publication was produced by Lika Electronic s.r.l. 2023. 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.

The logo for Lika Electronic s.r.l. features the word "lika" in a bold, lowercase, sans-serif typeface. The letters are black and have a modern, clean appearance.

# General contents

User's guide.....	1
General contents.....	3
Subject index.....	7
Typographic and iconographic conventions.....	9
Preliminary information.....	10
<b>1 Safety summary.....</b>	<b>11</b>
1.1 Safety.....	11
1.2 Electrical safety.....	11
1.3 Mechanical safety.....	12
<b>2 Identification.....</b>	<b>13</b>
<b>3 Mounting instructions.....</b>	<b>14</b>
3.1 Overall dimensions.....	14
3.2 Mounting instructions.....	15
3.3 Useful information.....	16
3.4 Maintenance.....	17
<b>4 Electrical connections.....</b>	<b>18</b>
4.1 Connection cap.....	18
4.2 Connection cap with PG gland (cable output).....	20
4.3 Connection cap with M12 connectors.....	21
4.4 Ground connection.....	22
4.5 Connection of the shield.....	22
4.6 Setting the baud rate: DIP A.....	23
4.7 Setting the node address: DIP B.....	24
4.8 Setting the RT bus termination.....	25
4.9 Diagnostic LEDs.....	26
<b>5 Quick reference (using RSNetWorx).....</b>	<b>27</b>
5.1 Import EDS file.....	27
5.2 Adding a node to the project.....	29
5.3 Encoder parameters configuration.....	29
5.4 Saving the parameters with RSNetWorx.....	33
5.5 Using the scanner module.....	34
5.6 Parameters.....	37
<b>1 Position.....</b>	<b>37</b>
<b>2 Position sensor type.....</b>	<b>38</b>
<b>3 Counting direction.....</b>	<b>38</b>
<b>4 Scaling function control.....</b>	<b>39</b>
<b>5 Measuring units/rev.....</b>	<b>39</b>
<b>6 Total measuring range.....</b>	<b>41</b>
<b>7 Preset value.....</b>	<b>43</b>
<b>8 Physical resolution/revolution.....</b>	<b>44</b>
<b>9 Physical number of revolutions.....</b>	<b>45</b>
<b>10 Alarms.....</b>	<b>45</b>
<b>11 Supported alarms.....</b>	<b>45</b>
Flash memory error.....	45
Machine data not valid.....	45

Bus off.....	45
12 Offset value.....	46
13 Velocity value.....	46
14 Velocity format.....	46
15 Dip switch.....	46
16 Data Tx in POLL mode.....	46
17 Data Tx in COS/Cyclic mode.....	47
18 Alarm flags.....	47
19 Delta for COS.....	47
20 Auto-save parameters.....	47
<b>6 DeviceNet interface.....</b>	<b>48</b>
6.1 EDS file.....	48
6.2 Communication messages.....	48
6.3 I/O Messages (Msg group 1).....	50
6.4 Explicit Messages (Msg group 2).....	51
6.4.1 Duplicate MAC-ID check.....	52
6.4.2 Error messages.....	53
No error.....	53
Resource unavailable.....	53
Invalid parameter value.....	53
Service not supported.....	53
Invalid attribute value.....	53
Already in requested mode / state.....	53
Object state conflict.....	53
Attribute not settable.....	53
Device state conflict.....	53
Not enough data.....	53
Attribute not supported.....	53
Too much data.....	53
Object does not exist.....	54
Invalid parameter.....	54
6.4.3 Explicit Messages Connection.....	55
6.4.4 Message data transfer.....	56
6.5 Object dictionary.....	57
6.5.1 Class 01h: Identity Object.....	58
01-01-01 Vendor-ID.....	58
01-01-02 Device type.....	58
01-01-03 Product code.....	58
01-01-04 Revision.....	58
01-01-05 Status.....	58
Owned.....	58
Configured.....	58
Extended device status.....	59
Minor recoverable fault.....	59
Minor unrecoverable fault.....	59
Major recoverable fault.....	59
Major unrecoverable fault.....	59
01-01-06 Serial number.....	59
01-01-07 Product name.....	60
6.5.2 Class 03h: DeviceNet Object.....	61

03-01-01 Node Address.....	61
03-01-02 Baud rate.....	61
03-01-03 Bus-off interrupt.....	61
03-01-04 Bus-off counter.....	61
03-01-05 Allocation information.....	62
6.5.3 Class 05h: Connection Object.....	63
05-Inst-01 Connection status.....	63
05-Inst-02 Instance type.....	63
05-Inst-03 TransportClass_trigger.....	63
05-Inst-04 Produced connection ID.....	63
05-Inst-05 Consumed connection ID.....	63
05-Inst-06 Initial comm. Characteristics.....	63
05-Inst-07 Produced connection size.....	64
05-Inst-08 Consumed connection size.....	64
05-Inst-09 Expected packet rate.....	64
05-Inst-0C Watchdog time-out action.....	64
05-Inst-0D Produced connection path length.....	64
05-Inst-0E Produced connection path.....	64
05-Inst-0F Consumed connection path length.....	64
05-Inst-10 Consumed connection path.....	64
05-Inst-11 Production inhibit time.....	65
6.5.4 Class 23h: Position Sensor Object.....	66
23-01-03 Position value.....	66
23-01-0B Device type.....	68
23-01-0C Code sequence.....	68
23-01-0E Scaling function control.....	68
23-01-10 Resolution per revolution.....	69
23-01-11 Total measuring range.....	70
23-01-13 Preset value.....	73
23-01-14 Delta for COS.....	75
23-01-18 Velocity value.....	75
23-01-19 Velocity format.....	75
23-01-2A Hardware counts per revolution.....	75
23-01-2B Hardware number of turns.....	76
23-01-2C Alarms.....	76
23-01-2D Supported alarms.....	76
Flash memory error.....	76
Machine data not valid.....	76
Bus off.....	76
23-01-2E Alarm flags.....	77
23-01-33 Offset.....	77
23-01-65 Dip switch.....	77
23-01-66 Data Tx in Poll mode.....	77
23-01-67 Data Tx in COS/Cyclic mode.....	77
23-01-68 Auto-save parameters.....	78
6.5.5 Class 2Bh: Acknowledge Handler Object.....	79
2B-01-01 Acknowledge timer.....	79
2B-01-02 Retry Limit.....	79
2B-01-03 Connection Instance of COS.....	79

<b>7 Setup.....</b>	<b>80</b>
7.1 Read and set parameters.....	80
7.1.1 P2P Master/Slave connection (without I/O msg).....	80
7.1.2 Reading the position 23-01-03 Position value (in the "Object dictionary").....	80
7.1.3 Setting 23-01-13 Preset value.....	81
7.1.4 Saving the parameters.....	81
7.1.5 Closing the Master/Slave connection.....	82
7.2 Setting the Cyclic mode without velocity.....	82
7.2.1 P2P Master/Slave connection.....	82
7.2.2 Setting 05-Inst-09 Expected packet rate for Cyclic mode (milliseconds).....	82
<b>8 Default parameters table.....</b>	<b>84</b>

# Subject index

## O

01-01-01 Vendor-ID.....	58
01-01-02 Device type.....	58
01-01-03 Product code.....	58
01-01-04 Revision.....	58
01-01-05 Status.....	58
01-01-06 Serial number.....	59
01-01-07 Product name.....	60
03-01-01 Node Address.....	61
03-01-02 Baud rate.....	61
03-01-03 Bus-off interrupt.....	61
03-01-04 Bus-off counter.....	61
03-01-05 Allocation information.....	62
05-Inst-01 Connection status.....	63
05-Inst-02 Instance type.....	63
05-Inst-03 TransportClass_trigger.....	63
05-Inst-04 Produced connection ID.....	63
05-Inst-05 Consumed connection ID.....	63
05-Inst-06 Initial comm. Characteristics.....	63
05-Inst-07 Produced connection size.....	64
05-Inst-08 Consumed connection size.....	64
05-Inst-09 Expected packet rate.....	64
05-Inst-0C Watchdog time-out action.....	64
05-Inst-0D Produced connection path length...64	
05-Inst-0E Produced connection path.....	64
05-Inst-0F Consumed connection path length..64	
05-Inst-10 Consumed connection path.....	64
05-Inst-11 Production inhibit time.....	65

## 2

23-01-03 Position value.....	66
23-01-0B Device type.....	68
23-01-0C Code sequence.....	68
23-01-0E Scaling function control.....	68
23-01-10 Resolution per revolution.....	69
23-01-11 Total measuring range.....	70
23-01-13 Preset value.....	73
23-01-14 Delta for COS.....	75
23-01-18 Velocity value.....	75
23-01-19 Velocity format.....	75
23-01-2A Hardware counts per revolution.....	75
23-01-2B Hardware number of turns.....	76
23-01-2C Alarms.....	76
23-01-2D Supported alarms.....	76
23-01-2E Alarm flags.....	77
23-01-33 Offset.....	77

23-01-65 Dip switch.....	77
23-01-66 Data Tx in Poll mode.....	77
23-01-67 Data Tx in COS/Cyclic mode.....	77
23-01-68 Auto-save parameters.....	78
2B-01-01 Acknowledge timer.....	79
2B-01-02 Retry Limit.....	79
2B-01-03 Connection Instance of COS.....	79

## A

Alarm flags.....	47
Alarms.....	45
Already in requested mode / state.....	53
Attribute not settable.....	53
Attribute not supported.....	53
Auto-save parameters.....	47

## B

Bus off.....	45, 76
--------------	--------

## C

Configured.....	58
Counting direction.....	38

## D

Data Tx in COS/Cyclic mode.....	47
Data Tx in POLL mode.....	46
Delta for COS.....	47
Device state conflict.....	53
Dip switch.....	46

## E

Extended device status.....	59
-----------------------------	----

## F

Flash memory error.....	45, 76
-------------------------	--------

## I

Invalid attribute value.....	53
Invalid parameter.....	54
Invalid parameter value.....	53

## M

Machine data not valid.....	45, 76
Major recoverable fault.....	59
Major unrecoverable fault.....	59
Measuring units/rev.....	39
Minor recoverable fault.....	59
Minor unrecoverable fault.....	59

## **N**

No error.....	53
Not enough data.....	53

## **O**

Object does not exist.....	54
Object state conflict.....	53
Offset value.....	46
Owned.....	58

## **P**

Physical number of revolutions.....	45
Physical resolution/revolution.....	44
Position.....	37
Position sensor type.....	38
Preset value.....	43

## **R**

Resource unavailable.....	53
---------------------------	----

## **S**

Scaling function control.....	39
Service not supported.....	53
Supported alarms.....	45

## **T**

Too much data.....	53
Total measuring range.....	41

## **V**




Velocity format.....	46
Velocity value.....	46

# 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 **GREEN**;
- 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 provide the most complete and exhaustive information the operator needs to correctly and safely install and operate the **SFAM1-05000 and SFAM2-10000 absolute draw-wire encoders with DeviceNet interface**.

The cable pulling mechanism integrates a 13 x 12 bit absolute multiturn encoder (13 bits = singleturn resolution = 8,192 cpr; 12 bits = 4,096 revolutions).

SFAM1-05000 / SFAM2-10000 cable-pulling encoder is aimed at speed and position measurements and controls in a variety of industrial applications through the movement of a **5,000 mm (196.85") or 10,000 mm (393.7")** stainless steel wire. The typical back and forth travel of the moving equipment causes the wire to reel and unreel and thus the linear movement to be converted into a rotary motion detected by the encoder which is coupled to the drum.

The stroke per turn is always 200 mm (7.874"), the maximum number of turns is 25 for SFAM1-05000 and 50 for SFAM2-10000.

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

In the first section some 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 SFAM1-5000 / SFAM2-10000 cable-actuated encoder are provided.

In the second section, entitled **DeviceNet Interface**, you can find detailed information on the DeviceNet interface. In this section the interface features and the objects implemented in the unit are fully described.

# 1 Safety summary



## 1.1 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;
- warning ! Do not use in explosive or flammable areas;
- 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 assumes no liability for the customer's failure to comply with these requirements.



## 1.2 Electrical safety

- Turn off the power supply before connecting the device;
- connect according to the explanation in the "4 - Electrical connections" section on page 18;
- in compliance with the 2014/30/EU norm on electromagnetic compatibility, the 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 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. Provide the ground connection as close as possible to the encoder. We suggest using the ground point provided in the cap, use one TCEI M3 x 6 cylindrical head screw with two tooth lock washers.



### 1.3 Mechanical safety

- Install the device following strictly the information in the "3 - Mounting instructions" section on page 14;
- mechanical installation has to be carried out with stationary mechanical parts;
- do not disassemble the encoder;
- do not tool the encoder;
- 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;
- we suggest installing the unit providing protection means against waste, especially swarf as turnings, chips, or filings; should this not be possible, please make sure that adequate cleaning measures are in place in order to prevent the wire from jamming;
- to avoid failures, never exceed the maximum measuring length and prevent the wire from tangling up;
- never release the wire freely, always help the wire wind properly: risk of personal injury and/or equipment damage;
- always keep the wire aligned not to damage the equipment;
- the stroke per turn of the draw-wire unit is 200 mm (7.874").

## 2 Identification

Device can be identified through the **order 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 order code and the serial number when reaching Lika Electronic 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 order code ending with "/Sxxx" may have mechanical and electrical characteristics different from standard and be supplied with additional documentation for special connections (Technical info).

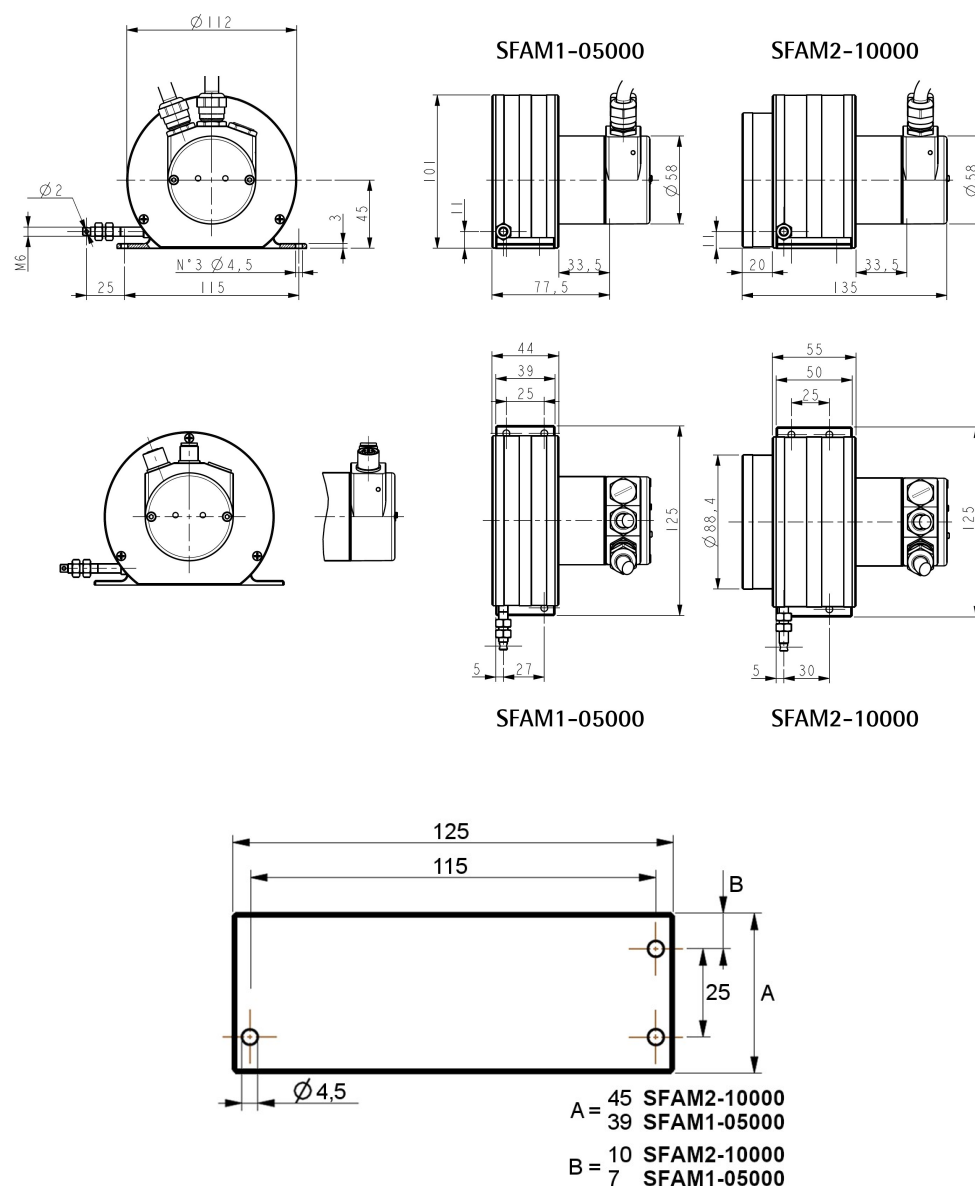
### 3 Mounting instructions



#### WARNING

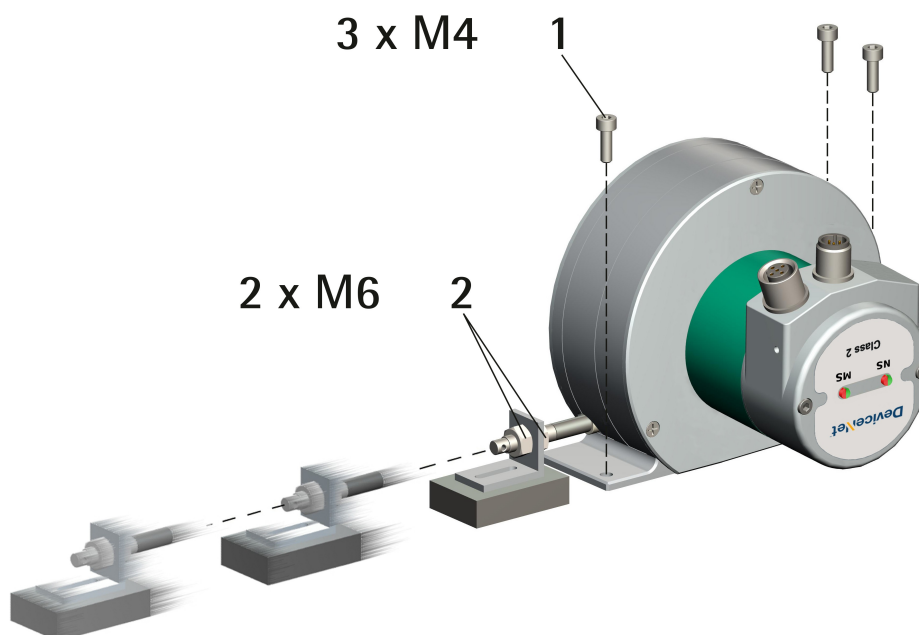
Installation, electrical connection and maintenance operations must be carried out by qualified personnel only, with power supply disconnected. Mechanical components must be in stop.

#### 3.1 Overall dimensions



Values are expressed in mm

### 3.2 Mounting instructions



- Fasten the draw-wire unit onto a fixed support using **three M4 screws 1**;
- remove the safety wire that pins the end of the measuring wire during transport;
- fix the end of the measuring wire to the moving unit using the provided **M6 nuts 2**.

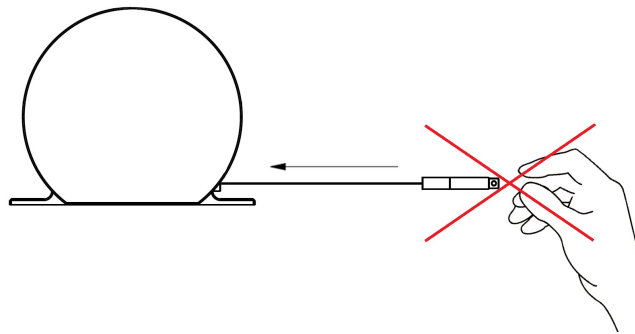


#### WARNING

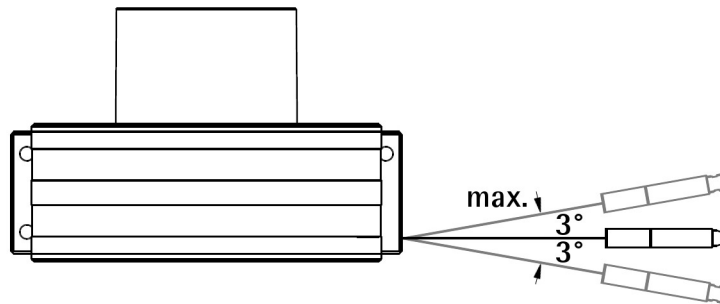
We suggest installing the unit providing protection means against waste, especially swarf as turnings, chips, or filings; should this not be possible, please make sure that adequate cleaning measures are in place in order to prevent the wire from jamming.

To avoid irreparable failures, never exceed the maximum measuring length and prevent the wire from tangling up.

Never release the wire freely, always help the wire wind properly: risk of personal injury and/or equipment damage.



Always keep the wire aligned not to damage the equipment (maximum deviation: 3°).



### 3.3 Useful information

If you want to know the **maximum measuring length** and the **physical linear resolution** of the draw-wire encoder please refer to the order code. The stroke per turn is always 200 mm (7.874"), the maximum number of turns is 25 for SFAM1-05000 and 50 for SFAM2-10000.



#### EXAMPLE 1

SFAM1-05000-FD2-08192-RPG using the physical resolution (**23-01-0E Scaling function control** = 00)

Stroke per turn of the drum = 200 mm (7.874")

**23-01-2A Hardware counts per revolution**, physical resolution per turn = 13 bits = 8,192 cpr

**23-01-2B Hardware number of turns**, number of physical revolutions = 12 bits = 4,096 revolutions

Total physical resolution = 25 bits = 33,554,432 information

Physical linear resolution = 0.024 mm = 24 µm

Max. number of turns of the drum = 25

Max. measuring length = 5,000 mm (196.85")

Number of information = 204,800



#### EXAMPLE 2

SFAM2-**10000**-FD2-**08192**-RM12 using a custom resolution (**23-01-0E Scaling function control** = 01)

Stroke per turn of the drum = 200 mm (7.874")

**23-01-2A Hardware counts per revolution**, physical resolution per turn = 13 bits = 8,192 cpr

**23-01-2B Hardware number of turns**, number of physical revolutions = 12 bits = 4,096 revolutions

Custom resolution per turn = **23-01-10 Resolution per revolution** = 2,000 cpr (example)

**23-01-11 Total measuring range** = 8,192,000 information (example)

$$\text{Custom number of encoder revolutions} = \frac{\text{23-01-11 Total measuring range}}{\text{23-01-10 Resolution per revolution}} = 4,096$$

Linear resolution = 0.1 mm = 100 µm

Max. number of turns of the drum = 50

Max. measuring length = 10,000 mm (393.7")

Number of information = 100,000

### 3.4 Maintenance

The measuring system does not need any particular maintenance; anyway it has to be handled with the utmost care as any delicate electronic equipment. From time to time we recommend the following operations:

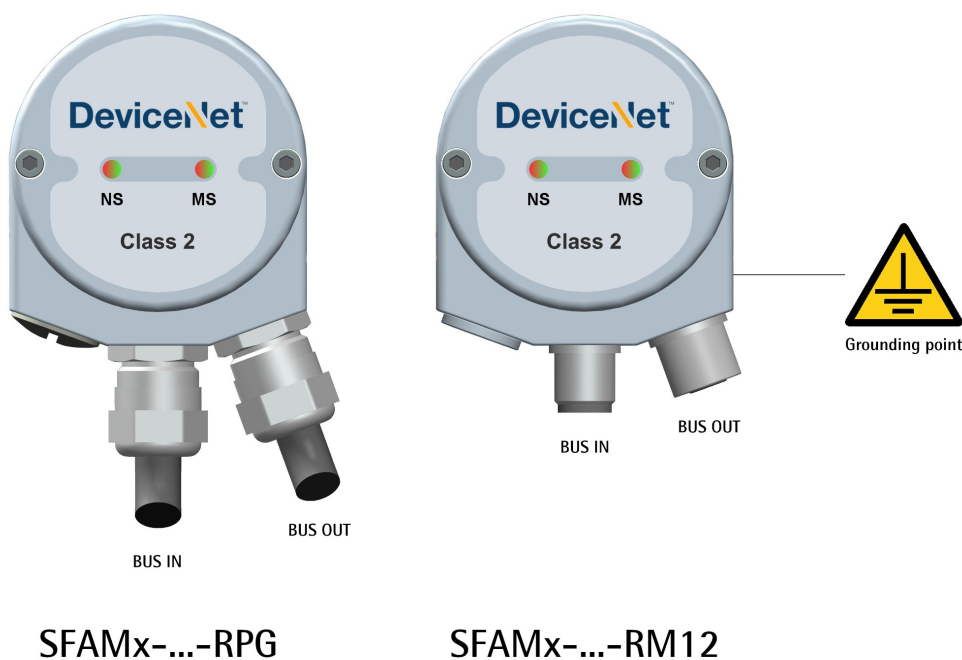
- the unit and the wire have to be cleaned regularly using a soft and clean cloth to remove dust, chips, moisture etc.; do not use oil to clean the wire.

## 4 Electrical connections



### WARNING

Installation, electrical connection and maintenance operations must be carried out by qualified personnel only, with power supply disconnected. Mechanical components must be in stop.



### 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 (PG cable version) as well as the DIP switches meant to set the node ID and the baud rate and activate the termination resistance (PG cable version and M12 connector version) are located inside the encoder connection cap. Thus you must remove the connection cap to access any of them.



### NOTE

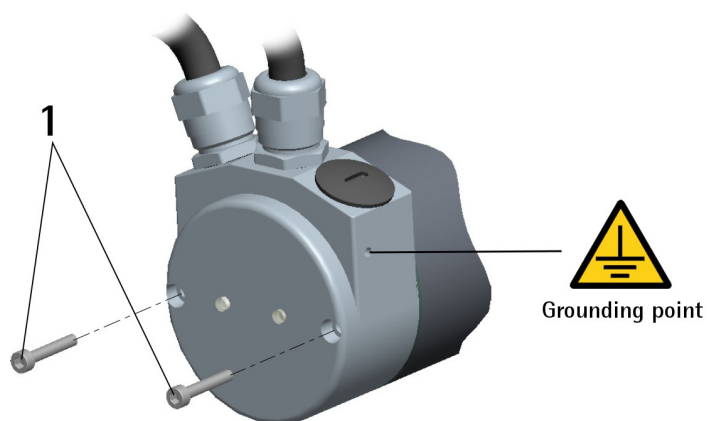
Be careful not to damage the internal components when you perform this operation.

To remove the connection cap loosen the two M3 screws **1**. Please be careful when you disconnect the internal connector.

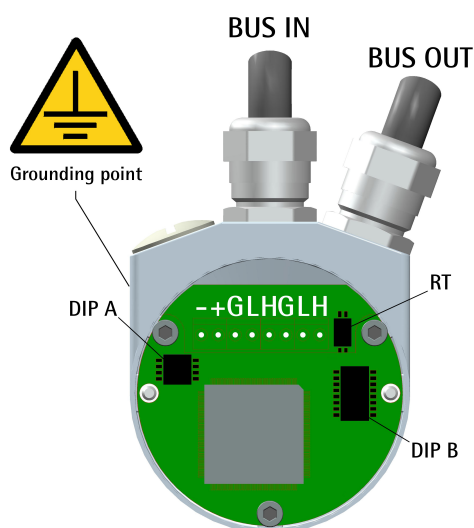
Always replace the connection cap at the end of the operation. Take care in re-connecting the internal connector. Tighten the screws **1** using a tightening torque of approx. 2.5 Nm.

**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 Connection cap with PG gland (cable output)



Cable output versions (...-RPG order code) are equipped with two PG9 cable glands for BUS IN and BUS OUT connections as well as for power supply. The bus cables can be connected directly to the terminal connectors located by each cable gland. You can use either cable for power supply.

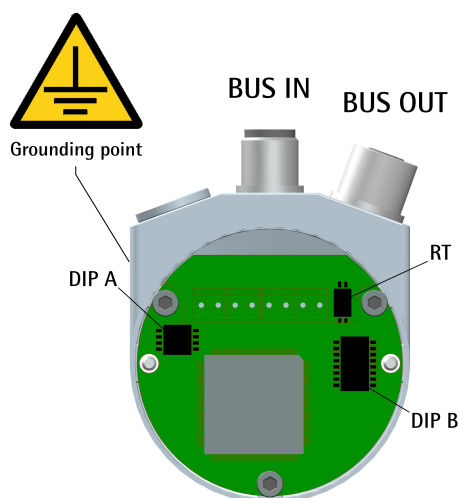
We recommend DeviceNet certificated cables to be used. Core diameter should not exceed Ø 1.5 mm (0.06 inches).

Terminal connector	Description
-	0Vdc 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.

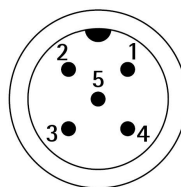
<sup>2</sup> Connect the cable shield to the cable gland.

### 4.3 Connection cap with M12 connectors

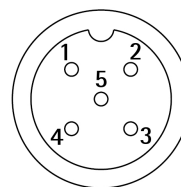


Connector output versions (...-RM12 order code) are equipped with two M12 connectors with pin-out in compliance with DeviceNet standard. Therefore you can use standard DeviceNet cordsets and patchcords commercially available. For a complete list of the available cordsets and patchcords please refer to the product datasheet ("Accessories" list).

M12 5-pin connector  
A coding  
(frontal side)



male  
(BUS IN)



female  
(BUS OUT)

M12	Description
Case	CAN Shield
1 <sup>1</sup>	
2	+10Vdc +30Vdc power supply voltage
3	0Vdc power supply voltage
4	CAN High
5	CAN Low

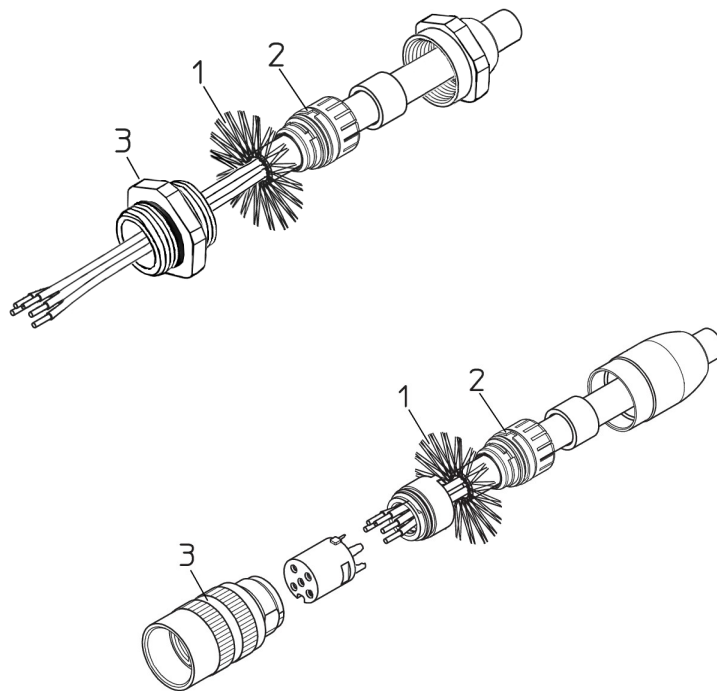
<sup>1</sup> CAN Shield is also connected to pin 1 to allow the connection of the shield even if the plug connector has a plastic case.

#### 4.4 Ground connection

To minimize noise connect properly the shield and/or the connector housing and/or the frame to ground. Connect properly the cable shield to ground on user's side. Lika's EC- pre-assembled cables are fitted with shield connection to the connector ring nut in order to allow grounding through the body of the device. Lika's E- connectors have a plastic gland, thus grounding is not possible. If metal connectors are used, connect the cable shield properly as recommended by the manufacturer. Anyway make sure that ground is not affected by noise. It is recommended to provide the ground connection as close as possible to the device. We suggest using the ground point provided in the cap (see the Figures, use one TCEI M3 x 6 cylindrical head screw with two tooth lock washers).

#### 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**.



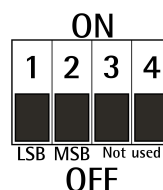
## 4.6 Setting the baud rate: DIP A



### WARNING

Power supply must be turned off before performing this operation!

The baud rate must be set using **DIP A** DIP switches:



Switch off the unit and set the binary value of the transmission rate considering that: ON = 1, OFF = 0.

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

Table of the available baud rate values:

Binary value	Baud rate
00	125 Kbit/s
01	250 Kbit/s
10	500 Kbit/s (default)



### NOTE

Bits 3 and 4 must be always set to OFF.

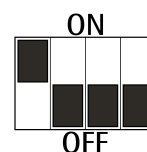


### EXAMPLE

Set the baud rate to 250 Kbit/s:

**01** (binary value, see the table above)

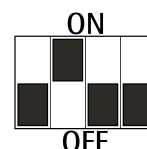
bit	1	2	3	4
	ON	OFF	OFF	OFF



Set the baud rate to 500 Kbit/s:

**10** (binary value, see the table above)

bit	1	2	3	4
	OFF	ON	OFF	OFF



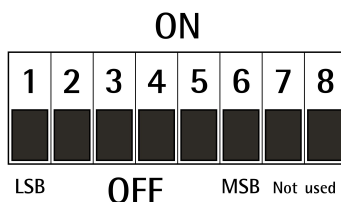
## 4.7 Setting the node address: DIP B



### WARNING

Power supply must be turned off before performing this operation!

The node number must be set via hardware using DIP B DIP switches. Allowed addresses range between 0 and 63. **The default address is 1.**



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

bit	1	2	3	4	5	6	7	8
	LSB					MSB	not used	not used
	$2^0$	$2^1$	$2^2$	$2^3$	$2^4$	$2^5$		

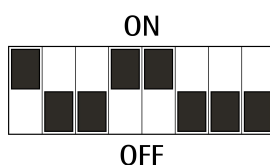


### EXAMPLE

Set the node address = 25:

$25_{10} = 0001\ 1001_2$  (binary value)

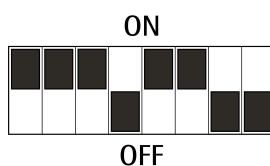
bit	1	2	3	4	5	6	7	8
	$2^0$	$2^1$	$2^2$	$2^3$	$2^4$	$2^5$		
	ON	OFF	OFF	ON	ON	OFF	OFF	OFF



Set the node address = 55:

$55_{10} = 0011\ 0111_2$  (binary value)

bit	1	2	3	4	5	6	7	8
	$2^0$	$2^1$	$2^2$	$2^3$	$2^4$	$2^5$		
	ON	ON	ON	OFF	ON	ON	OFF	OFF



## 4.8 Setting the RT bus termination



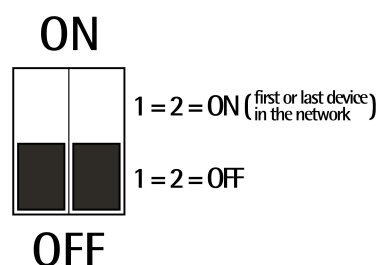
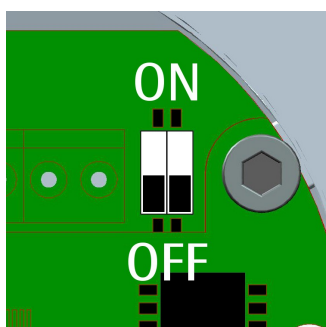
### WARNING

Power supply must be turned off before performing this operation!

A bus termination resistor is provided inside the connection cap and must be activated as line termination if the encoder is at the ends of the transmission line (i.e. it is either the first or the last device in the transmission line).

Use RT Switch to activate or deactivate the bus termination.

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



## 4.9 Diagnostic LEDs

Two diagnostic LEDs located in the connection cap are designed to show the operating or fault status of the DeviceNet interface and the system as well.

**NS Led** Network Status LED: this LED shows the status of the communication network.

**MS Led** Module Status LED: this LED shows the status of the device.

NS LED	Description
OFF	Not powered or not on-line
<b>Red</b> ON	Encoder communication failure
Flashing <b>red</b>	Connection time-out
<b>Green</b> ON	Device on-line and connected
Flashing <b>green</b>	Device on-line but not connected
Flashing <b>green / red</b>	Communication error

MS LED	Description
OFF	No power supply
<b>Red</b> ON	Unrecoverable fault (see <b>01-01-05 Status</b> attribute on page 58)
Flashing <b>red</b>	Recoverable fault, may need replacing (see <b>01-01-05 Status</b> attribute on page 58)
<b>Green</b> ON	Device operational
Flashing <b>green</b>	Device on-line but not connected (standby)
Flashing <b>green / red</b>	Device self test

## 5 Quick reference (using RSNetWorx)

### 5.1 Import EDS file

DeviceNet draw-wire encoders are supplied with their own EDS file SFA.eds, it can be downloaded at the address [www.lika.biz](http://www.lika.biz) > **PRODUCTS** > **DRAW-WIRE ENCODERS**.

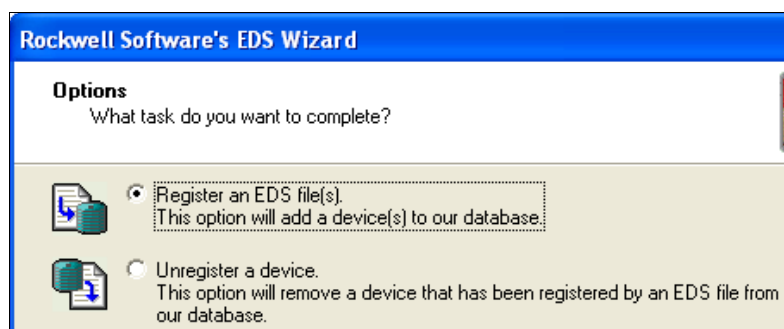
EDS file must be installed in the Master device.

In this section the installation and configuration of the device using **Rockwell Automation's RSNetWorx** program are described.

On the menu bar of the **RSNetWorx** window, select the **Tools** menu and then press the **EDS Wizard...** command.



In the **EDS Wizard** window select **Register an EDS file(s)** option and then press **Next >** button.



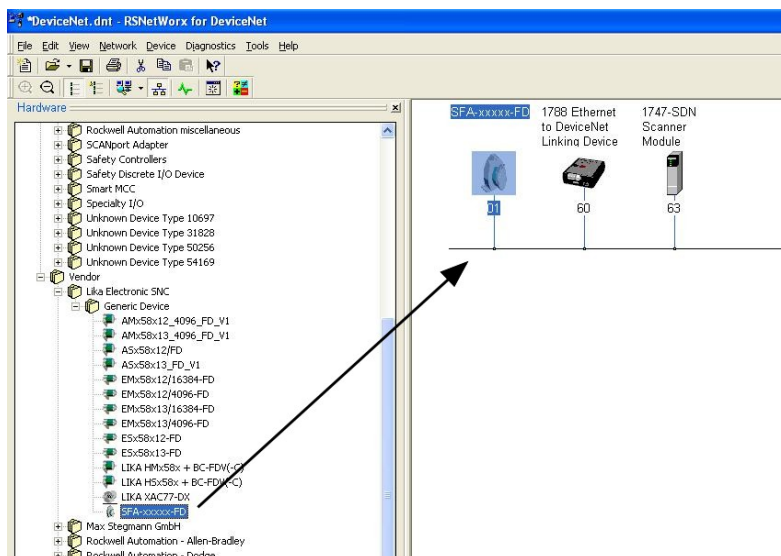
Press the **Browse...** button to browse through the folders and select the .eds file to be installed; finally press the **Next >** button to continue.



Then, follow the remaining steps to complete the EDS wizard.

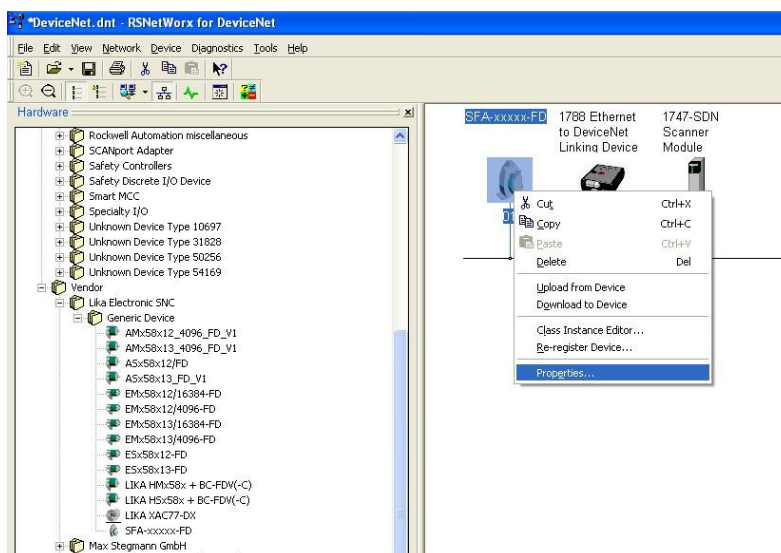
## 5.2 Adding a node to the project

In the **Hardware** pane of the **RSNetworx** window, open the directory tree and select DeviceNet\Vendor\Lika Electronic SNC\Generic Device; drag the required module SFA-xxxxx-FD to the **Graph** tabbed page on the right.

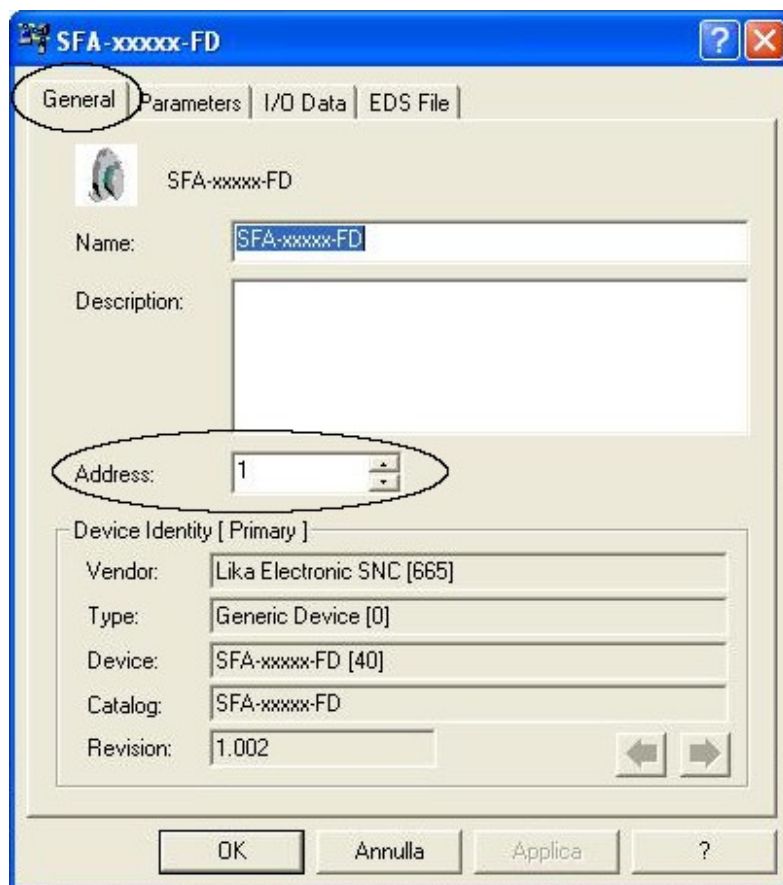


## 5.3 Encoder parameters configuration

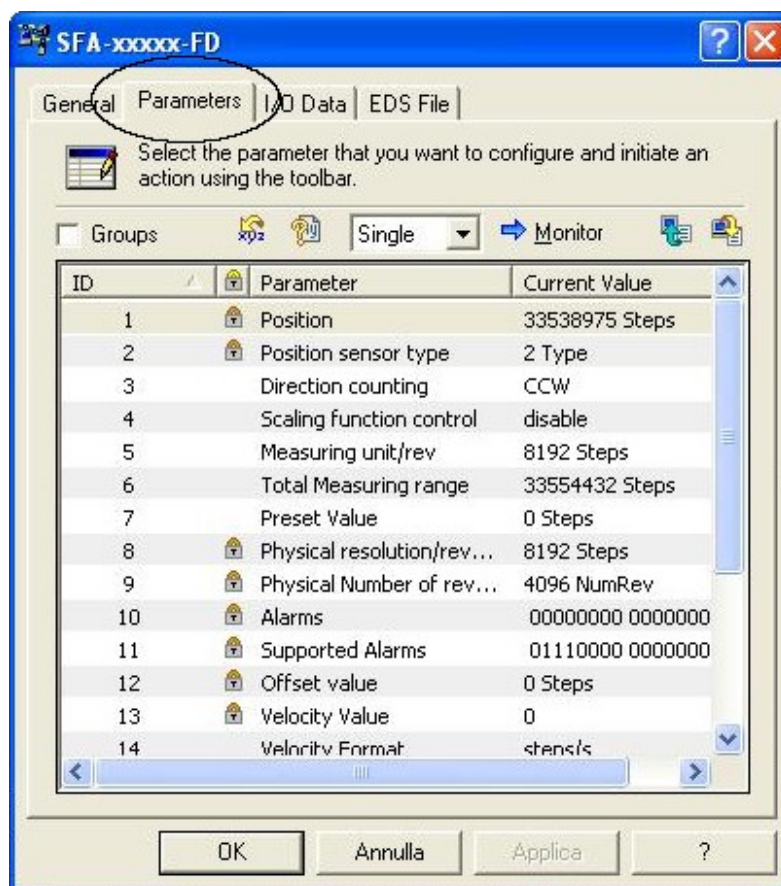
Double-click the encoder icon in the **Graph** tabbed page and open the **Properties** window. Or right-click the encoder icon and press the **Properties...** command in the menu.



Open the **General** tabbed page of the **Properties** window and set the node address in the **Address** box.

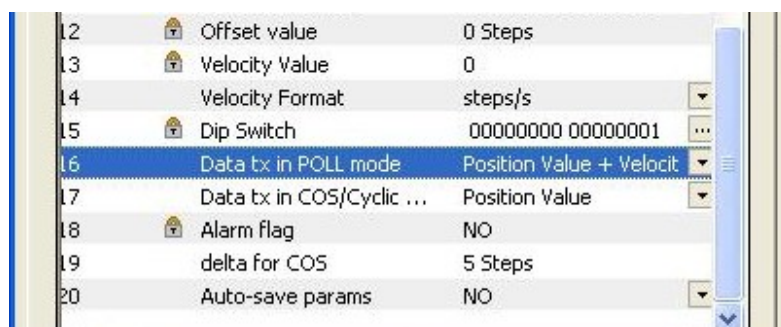


Open the **Parameters** page of the **Properties** window if you need to read, set and save the encoder parameters.



Please pay particular attention to **16 Data Tx in POLL mode**, **17 Data Tx in COS/Cyclic mode** and **20 Auto-save parameters** attributes.

**16 Data Tx in POLL mode** and **17 Data Tx in COS/Cyclic mode** attributes must be set in accordance with the scanner module settings (see the "5.5 Using the scanner module" section on page 34).





### EXAMPLE

Parameter 16:

**16 Data Tx in POLL mode** = "Position Value + Velocity"

Parameter 17:

**17 Data Tx in COS/Cyclic mode** = "Position Value"

Set the scanner module parameters accordingly:

For any further information on the scanner module parameters refer to the "5.5 Using the scanner module" section on page 34.



Click the **Upload** button to upload and read the current configuration.



Click the **Download** button to send the new parameters to the encoder.



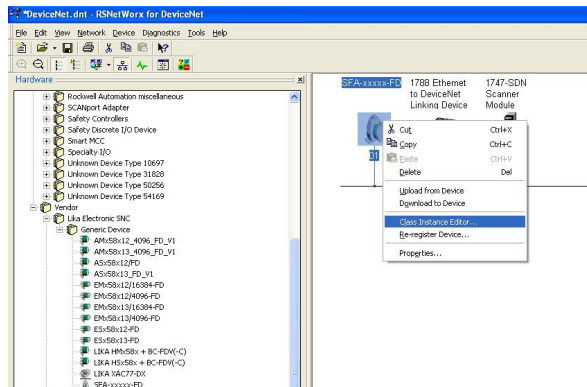
### NOTE

If **20 Auto-save parameters** = YES, the new parameters values are saved automatically; otherwise if **20 Auto-save parameters** = NO, you must start manually the "save parameters" procedure; see the following "5.4 Saving the parameters with RSNetWorx" section.

## 5.4 Saving the parameters with RSNetWorx

**Properties** window allows you to set device parameters. After this operation is carried out, data is stored in the RAM memory only. In case of "Reset node", "Node restore" or power off, parameters will be lost.

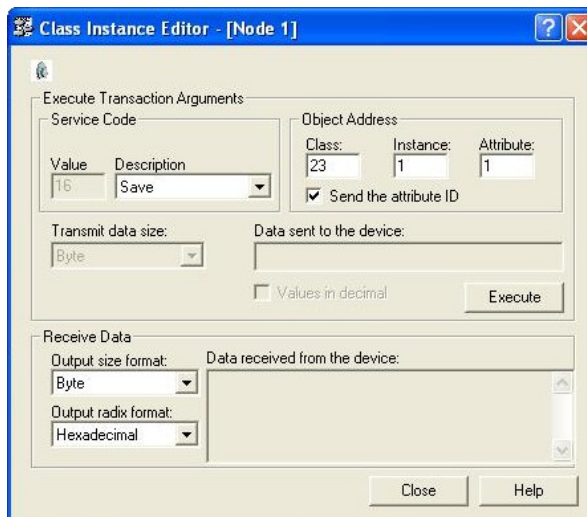
To save the new parameters permanently you must start the CLASS INSTANCE EDITOR; to do this right-click the encoder icon in the **Graph** tabbed page and select the **Class Instance Editor...** command.



In the **Class Instance Editor** window enter values expressed in hexadecimal notation as shown in the Figure. When setting up is carried out, press the **Execute** button: data will be downloaded to the encoder and saved.

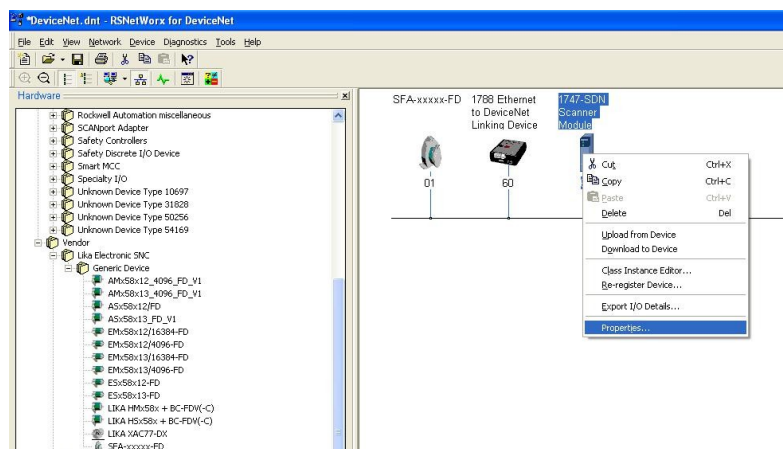
Soon afterwards, **The execution was completed** message will appear to confirm that data has been downloaded and saved properly.

Close the **Class Instance Editor** window pressing the **Close** button.

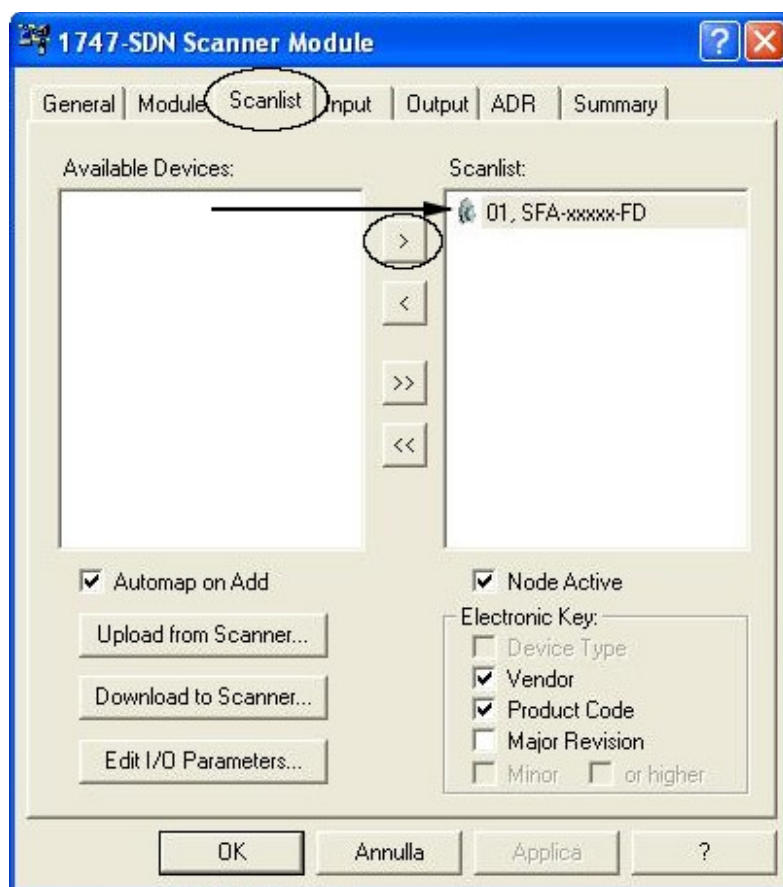


## 5.5 Using the scanner module

To manage I/O data using the scanner module, right-click the scanner icon and open the **Properties** window; if requested, upload the configuration.



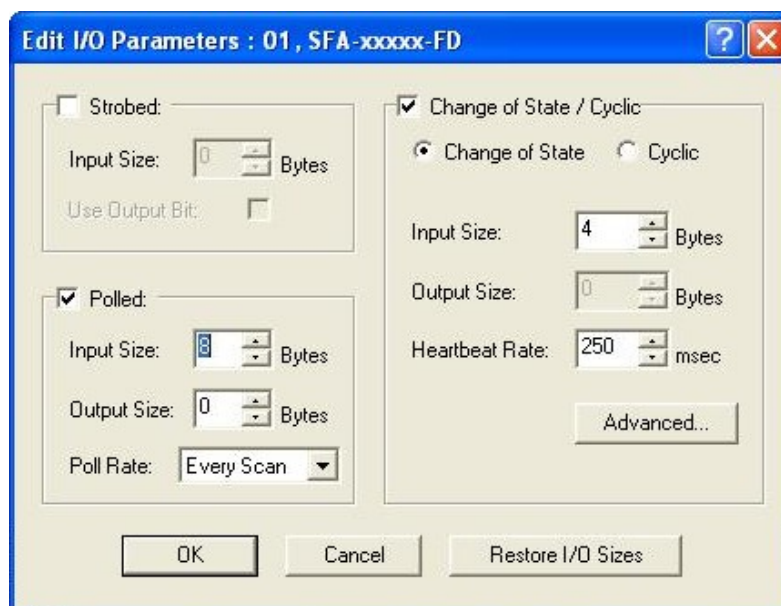
Open the **Scanlist** tabbed page. Select Lika device in the **Available Devices** pane on the left and move it to the **Scanlist** pane on the right pressing the > button. Now double-click the item in the **Scanlist** pane to open the **Edit I/O parameters** window.



Now, set the **Input Size** value in both panes **Polled** and **Change of State / Cyclic** according to **16 Data Tx in POLL mode** and **17 Data Tx in COS/Cyclic mode** values respectively (see the encoder parameters in the "5.6 Parameters" section on page 37).

If "Position value" is set: **Input Size** = 4 bytes

If "Position value + velocity" is set: **Input Size** = 8 bytes



Click the **OK** button to confirm the parameters.

The list of variables is available in the **Input** page.

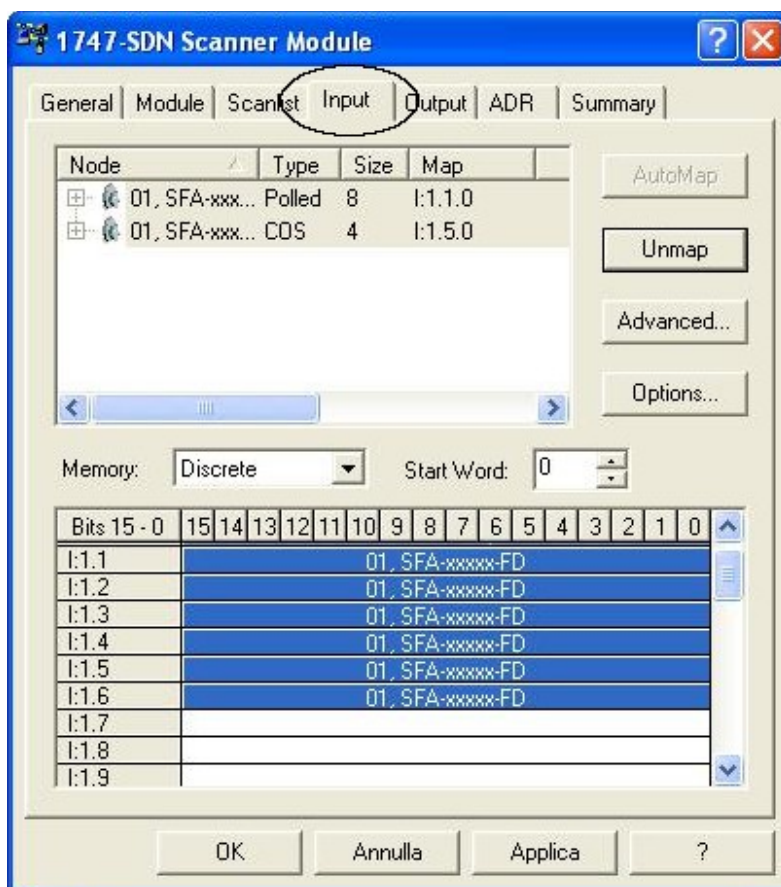


# EXAMPLE

In the following example:

**16 Data Tx in POLL mode** = Position value + velocity;

**17 Data Tx in COS/Cyclic mode** = Position value.



Polled messages:

Variables	Bytes	Input
1.1	1	Position value
	2	
1.2	3	Position value
	4	
1.3	5	Velocity value
	6	
1.4	7	Velocity value
	8	

COS messages:

Variables	Bytes	Input
1.5	1	Position value
	2	
1.6	3	Position value
	4	

## 5.6 Parameters

### 1 Position

This attribute shows the current absolute position detected by the position sensor. The output value is scaled according to the scaling attributes, see the [4 Scaling function control](#) attribute.



#### WARNING

Please note that the position value issued by the encoder is expressed in counts; thus you have then to convert the number of counts into a linear measuring unit.

To convert the position value into millimetres (mm) or micrometres (µm) you have to multiply the number of information by the linear resolution of the encoder expressed in millimetres or micrometres.

To know the linear resolution of the encoder please consider that **the stroke per turn of the drum is 200 mm (7.874")**.

The linear resolution results from the following calculation:

$$\text{Linear resolution} = \frac{\text{Stroke per turn of the drum mm}}{\text{Singleturn resolution cpr}}$$

If you want to know the linear position value you will need to multiply the transmitted position value by the linear resolution.

**Linear position value** = transmitted position \* linear resolution



#### NOTE

Please note that the encoder's linear resolution can be read also in the order code next to the rotary resolution. Refer to the product datasheet.



#### EXAMPLE 1

Let's suppose that we are using the physical resolution of the SFAM1-05000-FD2-08192-RPG draw-wire encoder (the [4 Scaling function control](#) attribute = 0).

The physical singleturn resolution of the measuring device is 8,192 cpr (= 0.024 mm, see the order code in the product datasheet).

As stated, the linear resolution results from the following calculation:

$$\text{Linear resolution} = \frac{\text{Stroke per turn of the drum mm}}{\text{Singleturn resolution cpr}}$$

$$\text{Linear resolution} = \frac{200 \text{ mm}}{8,192 \text{ cpr}} = 0.024 \text{ mm} = 24 \text{ }\mu\text{m}$$

Let's say that the transmitted position value is 123.

Thus the linear position value will be as follows:

**Linear position value** = transmitted position \* linear resolution

$$\text{Linear position value} = 123 * 0.024 = 2.952 \text{ mm} = 2,952 \text{ }\mu\text{m}$$



#### EXAMPLE 2

Let's suppose that we are using the SFAM1-05000-FD2-08192-RPG draw-wire encoder. The singleturn resolution is set to the custom value of 4,000 cpr (**5 Measuring units/rev** = 4000). The transmitted position value is 1,569.

The linear resolution can be easily calculated as follows:

$$\text{Linear resolution} = \frac{200 \text{ mm}}{4,000 \text{ cpr}} = 0.05 \text{ mm} = 50 \text{ }\mu\text{m}$$

Thus the linear position value will be as follows:

$$\text{Linear position value} = 1,569 * 0.05 = 78.45 \text{ mm} = 78,450 \text{ }\mu\text{m}$$

### 2 Position sensor type

Type of device.

2: Multiturn absolute rotary encoder.

Default = 2 (min. = 2, max. = 2)

### 3 Counting direction

**3 Counting direction** attribute sets whether the position value output by the encoder increases (count up information) when you rewind the wire (0) or when you pull the wire out (1, default).

0: rewinding the wire the position will increase;

**1: pulling the wire out the position will increase.**

Default = 1 (min. = 0, max. = 1)



#### WARNING

Every time you change the **3 Counting direction** then you are required to set a new preset value (see the **7 Preset value** attribute) and finally save the new parameters (see the "5.4 Saving the parameters with RSNetWorx" section on page 33).

#### 4 Scaling function control

If this attribute is disabled (OFF = 00), the device uses the physical resolution values (see the **8 Physical resolution/revolution** and **9 Physical number of revolutions** attributes); if it is enabled (ON = 01), it uses the custom resolution set in the **5 Measuring units/rev** and **6 Total measuring range** attributes with the following relation:

Transmitted position =

$$\frac{\text{5 Measuring units/rev}}{\text{8 Physical resolution/revolution}} * \text{real position} \leq \text{6 Total measuring range}$$

Default = 0 (min. = 0, max. = 1)



#### WARNING

When you enable the scaling function (**4 Scaling function control** = 1), please enter scaled values next to the **5 Measuring units/rev** and **6 Total measuring range** attributes that are consistent with the physical values.



#### WARNING

Every time you enable the scaling function and/or change the scaling values (see the **5 Measuring units/rev** and **6 Total measuring range** attributes) then you are required to set a new preset value (see the **7 Preset value** attribute) and finally save the new parameters (see the "5.4 Saving the parameters with RSNetWorx" section on page 33).

#### 5 Measuring units/rev



#### WARNING

This attribute is active only if the **4 Scaling function control** attribute is set to "1"; otherwise it is ignored and the system uses the physical values (**8 Physical resolution/revolution** and **9 Physical number of revolutions**) to calculate the position information.

This attribute sets a custom number of distinguishable steps per revolution (custom singleturn resolution).

To avoid counting errors, check that

$$\frac{\text{8 Physical resolution/revolution}}{\text{5 Measuring units/rev}} = \text{integer value.}$$

You are allowed to set whatever integer value less than or equal to the **maximum number of physical steps per revolution** (see the hardware counts

per revolution in the encoder identification label and the **8 Physical resolution/revolution** attribute).

Default = 8,192 (min. = 1, max. = 8,192)



#### WARNING

When you set a new value next to the **5 Measuring units/rev** attribute, please always check also the **6 Total measuring range** attribute value and be sure that the resulting number of revolutions complies with the **Hardware number of revolutions** of the device (4,096 revolutions, see the **9 Physical number of revolutions** attribute).

Let's suppose that the encoder is programmed as follows:

**5 Measuring units/rev**: 8,192 cpr

**6 Total measuring range** =  $33\,554\,432_{10} = 8,192 \text{ (cpr)} * 4,096 \text{ (rev.)}$

Let's set a new singleturn resolution, for instance: **5 Measuring units/rev** = 360 cpr.

If we do not change the **6 Total measuring range** value at the same time, we will get the following result:

$$\text{Number of revolutions} = \frac{33\,554\,432 \text{ (6 Total measuring range)}}{360 \text{ (5 Measuring units/rev)}} = 93,206.755...$$

As you can see, the encoder is required to carry out more than 93,000 revolutions, this cannot be as the hardware number of revolutions is, as stated, 4,096. When this happens, the encoder falls into an error signalling the faulty condition through the diagnostic LEDs (see on page 26).



#### WARNING

When you enable the scaling function (**4 Scaling function control** = 01), please enter scaled values next to the **5 Measuring units/rev** and **6 Total measuring range** attributes that are consistent with the physical values. In the case of inconsistent values, the system will warn about the wrong parametrization and fault condition by means of the dedicated attributes and visually by means of the diagnostic LEDs.



#### WARNING

Every time you change the scaled values (see the **5 Measuring units/rev** and **6 Total measuring range** attributes), then you are required to set a new preset value (see the **7 Preset value** attribute).

## 6 Total measuring range



### WARNING

This attribute is active only if the **4 Scaling function control** attribute is set to "1"; otherwise it is ignored and the system uses the physical values (**8 Physical resolution/revolution** and **9 Physical number of revolutions**) to calculate the position information.

This attribute sets a custom number of distinguishable steps over the total measuring range. In other words, this attribute allows to set the length of the travel the encoder has to measure expressed in number of distinguishable steps (number of information). The total resolution of the encoder results from the product of **5 Measuring units/rev** by the required **Number of revolutions**.

You are allowed to set whatever integer value less than or equal to the **overall hardware resolution** (see the encoder identification label). The overall hardware resolution results from:

**8 Physical resolution/revolution \* 9 Physical number of revolutions.**

We recommend the **Number of revolutions** to be set to a power of 2.

The set **Number of revolutions** results from the following calculation:

$$\text{Number of revolutions} = \frac{\text{6 Total measuring range}}{\text{5 Measuring units/rev}}$$

Setting the **Number of revolutions** to a value which is a power of 2 is meant to avoid problems when using the device in endless operations requiring the physical zero to be overstepped. If you set the **Number of revolutions** which is not a power of 2, a counting error is generated before the physical zero.

Default = 33 554 432 (min. = 1, max. = 33 554 432)



### WARNING

When you set a new value next to the **6 Total measuring range** attribute, please always check also the **5 Measuring units/rev** attribute value and be sure that the resulting number of revolutions complies with the Hardware number of revolutions of the device (4,096 revolutions).

Let's suppose that the encoder is programmed as follows:

**5 Measuring units/rev**: 8,192 cpr

**6 Total measuring range** = 33 554 432<sub>10</sub> = 8,192 (cpr) \* 4,096 (rev.)

Let's set a new total resolution, for instance: **6 Total measuring range** = 360.

As the **6 Total measuring range** must be greater than or equal to the **5 Measuring units/rev**, the above setting is not allowed.



#### EXAMPLE

We install the following draw-wire encoder: **SFAM1-05000-FD2-08192-RM12**.

The physical values are:

Stroke per turn of the drum = 200 mm (7.874")

Physical resolution per turn = 13 bits = 8,192 cpr

Max. number of physical revolutions = 4,096 revolutions

Total physical resolution = 25 bits = 33 554 432 information

Physical linear resolution = 0.024 mm = 24 µm

Max. number of turns of the drum = 25

Max. measuring length = 5,000 mm (196.85")

Number of information = 204,800

Let's suppose that we need a tenth of a millimetre linear resolution in the specific installation.

- Enable the scaling function: **4 Scaling function control** attribute = 1
- Custom resolution per turn = **5 Measuring units/rev** = 2,000 cpr
- Linear resolution = 0.1 mm = 100 µm

$$\text{Linear resolution} = \frac{\text{Stroke per turn mm}}{\text{5 Measuring units/rev}} = \frac{200 \text{ mm}}{2,000 \text{ cpr}} = 0.1 \text{ mm}$$

The custom number of revolutions can be as the physical number of revolutions:

$$\text{Custom number of encoder revolutions} = \frac{\text{6 Total measuring range}}{\text{5 Measuring units/rev}} = 4,096$$

- **6 Total measuring range** = 8 192 000



#### NOTE

Please note that if you set a preset along the path, when the encoder moves back and cross the zero, the value immediately after 0 will be 8 192 000 - 1, i.e. 8 191 999.

←

...	8,191,997	8,191,998	8,191,999	0	1	2	...
-----	-----------	-----------	-----------	---	---	---	-----



#### EXAMPLE

Using the values in the previous example let's suppose that the travel in the application is 2 m long. As the stroke per turn is 200 mm you need 10 revolutions to cover the travel length.

- **6 Total measuring range** = **5 Measuring units/rev** \* custom number of revolutions = 2,000 \* 10 = 20,000

In fact:

$$\text{Custom number of encoder revolutions} = \frac{6 \text{ Total measuring range}}{5 \text{ Measuring units/rev}} = 10$$

In this case you will obtain several 20,000 information sections following each other all along the whole measuring length. The position information will be from 0 to 19,999; then again from 0 to 19,999 and so on.

...	19,997	19,998	19,999	0	1	2	...	19,997	19,998	19,999	0	1	2	...
← max. measuring length →														



#### NOTE

To avoid counting errors we recommend values which are the power of 2 ( $2^n$ : 2, 4, ..., 2048, 4096, 8192,...) to be set next to the **5 Measuring units/rev** and **6 Total measuring range** attributes.



#### WARNING

If you have set the preset, when you change the value next to **5 Measuring units/rev** and/or **6 Total measuring range** attribute, then you must check the value in the **7 Preset value** attribute and execute the preset operation.

#### 7 Preset value

This attribute allows to set the encoder position to a Preset value. The Preset function is meant to assign a desired value to a physical position of the encoder (i.e. a position in the travel of the wire). The chosen physical position will get the value set next to this item and all the previous and following positions will get a value according to it. This function can be useful, for instance, when the zero position of the encoder and the zero position of the axis need to match.

The preset value will be set for the position of the encoder (i.e. the position of the wire) in the moment when the preset value is transmitted. We suggest setting the preset value when the encoder is in stop.

Default = 0 (min. = 0, max. = 33 554 431)



#### EXAMPLE

Let's take a look at the following example to better understand the preset function and the meaning and use of the related attributes: **7 Preset value** and **12 Offset value**.

The encoder position which is transmitted results from the following calculation:

**Transmitted value** = **read position** (it does not matter whether the position is physical or scaled) + **7 Preset value** - **12 Offset value**.

If you never set the **7 Preset value** and you never performed the preset setting, then the transmitted value and the read position are necessarily the same as **7 Preset value** = 0 and **12 Offset value** = 0.

When you set the **7 Preset value** and then execute the preset setting, the system saves the current encoder position in the **12 Offset value** attribute. It follows that the transmitted value and the **7 Preset value** are the same as **read position** - **12 Offset value** = 0; in other words, the value set next to the **7 Preset value** attribute is paired with the current position of the encoder as you wish.

For example, let's assume that the value "50" is set next to the **7 Preset value** attribute and you execute the preset setting when the encoder position is "1000". In other words, you want to receive the value "50" when the encoder reaches the position "1000".

We will obtain the following information sequence:

**Transmitted value** = **read position** (= "1000") + **7 Preset value** (= "50") - **12 Offset value** (= "1000") = 50.

The following transmitted value will be:

**Transmitted value** = **read position** (= "1001") + **7 Preset value** (= "50") - **12 Offset value** (= "1000") = 51.

And so on.



#### NOTE

- If the scaling function is disabled (see the **4 Scaling function control** attribute), the **7 Preset value** must be less than or equal to the "**Total hardware resolution**" - 1 (**8 Physical resolution/revolution** \* **9 Physical number of revolutions** - 1).
- If the scaling function is enabled (see the **4 Scaling function control** attribute), the **7 Preset value** must be less than or equal to the **6 Total measuring range** - 1.



#### WARNING

Check the value in the **7 Preset value** attribute and perform the preset operation every time you change the **3 Counting direction** or the scaled values (**5 Measuring units/rev** and/or **6 Total measuring range** attributes).

### 8 Physical resolution/revolution



#### WARNING

This attribute is active only if the **4 Scaling function control** attribute is set to "=0"; otherwise it is ignored and the system uses the custom values (**5**

Measuring units/rev and 6 Total measuring range) to calculate the position information.

This attribute is intended to show the number of physical distinguishable steps per each revolution provided by the hardware (physical singleturn resolution).

If you want to set a custom singleturn resolution see the 5 Measuring units/rev attribute on page 39.

Default = 8,192

## 9 Physical number of revolutions



### WARNING

This attribute is active only if the 4 Scaling function control attribute is set to "=0"; otherwise it is ignored and the system uses the custom values (5 Measuring units/rev and 6 Total measuring range) to calculate the position information.

This attribute is intended to show the number of physical revolutions provided by the hardware (number of physical revolutions).

The **Total hardware resolution** results from 8 Physical resolution/revolution \* 9 Physical number of revolutions.

If you want to set a custom number of revolutions see the 5 Measuring units/rev and 6 Total measuring range attributes on page 39 ff.

Default = 4,096

## 10 Alarms

An alarm is set when a bit indicating a fault is set to true (high). See the 11 Supported alarms values.

Default = 0 (min. = 0, max. = 32 767)

## 11 Supported alarms

This attribute contains information on the supported alarms.

Bits 0 ... 11	Reserved	
Bit 12	Flash memory error	Internal error, it cannot be restored.
Bit 13	Machine data not valid	One or more parameters are not valid, set proper values to restore normal work condition.
Bit 14	Bus off	This bit is intended to warn that the communication in the bus network has broken (cable disconnected? Voltage drop? ...).

Default = 0 (min. = 0, max. = 32 767)

## 12 Offset value

As soon as you activate the preset, the current position value of the encoder is saved on this attribute. The offset value is then used in the preset function in order to calculate the encoder position value to be transmitted. To zero set the value in this attribute you must upload the factory default values (see the 15h Restore Service code on page 66).

For any further information on the preset function and the meaning and use of the related attributes **7 Preset value** and **12 Offset value** please refer to page 43.

Default = 0 (min. = 0, max. = 33 444 431)

## 13 Velocity value

This attribute shows the current speed detected by the position sensor and calculated every 100 ms. The speed can be expressed in number of steps per second or in revolutions per minute (see the **14 Velocity format** attribute).

## 14 Velocity format

This attribute sets the engineering units for the velocity value (**13 Velocity value**).

steps/s (0): number of steps per second (default value);

rpm (1): revolutions per minute.

Default = 0 (min. = 0, max. = 1)

## 15 Dip switch

This attribute shows the status of the dip switches.

Dip switches are located in the connection cap and used to set the baud rate and the MAC-ID (see the sections "4.6 Setting the baud rate: DIP A" and "4.7 Setting the node address: DIP B").

bits 0 ... 5 = MAC-ID

bits 6 and 7 = not used

bits 8 and 9 = Baud Rate

bits 10 ... 15 = not used

## 16 Data Tx in POLL mode

This parameter is meant to set which input data is sent when the **Polled** mode connection is enabled.

Allowed values: Position value (0, default value)

Position value + velocity (1)

Default = 0 (min. = 0, max. = 1)

### 17 Data Tx in COS/Cyclic mode

This parameter is meant to set which input data is sent when the **Change Of State** (COS) / **Cyclic** mode connection is enabled.

Allowed values:      Position value (0, default value)  
                                  Position value + velocity (1)

Default = 0 (min. = 0, max. = 1)

### 18 Alarm flags

It shows whether a fault occurred and an alarm has been activated, see the **11 Supported alarms** attribute.

NO (0)                  No active alarm  
 YES (1)                Active alarms

Default = 0 (min. = 0, max. = 1)

### 19 Delta for COS

When the **Change Of State** (COS) mode connection is enabled, this parameter sets the minimum gap between two I/O messages. Values too low could cause the Bus network to saturate.

Default = 5 (min. = 0, max. = 262 143)

### 20 Auto-save parameters

NO (0): New parameters are not saved on EPROM automatically. To save new values Class Instance Editor must be used (default value; see the "5.4 Saving the parameters with RSNetWorx" section on page 33).

YES (1): When a new parameter is set, the encoder waits 5 seconds and then saves automatically all the new values on EPROM, but only if the received value has been changed.

Default = 0 (min. = 0, max. = 1)

## 6 DeviceNet interface

Lika draw-wire encoders are "Group 2 only server" devices and do not support UCMM messages.

For any omitted information refer to the "Open DeviceNet Vendor Association" documents or visit [www.odva.org](http://www.odva.org).

### 6.1 EDS file

DeviceNet draw-wire encoders are supplied with their own EDS file SFA.eds, it can be downloaded at the address [www.lika.biz](http://www.lika.biz) > **PRODUCTS** > **DRAW-WIRE ENCODERS**.

EDS file must be installed in the Master device.

### 6.2 Communication messages

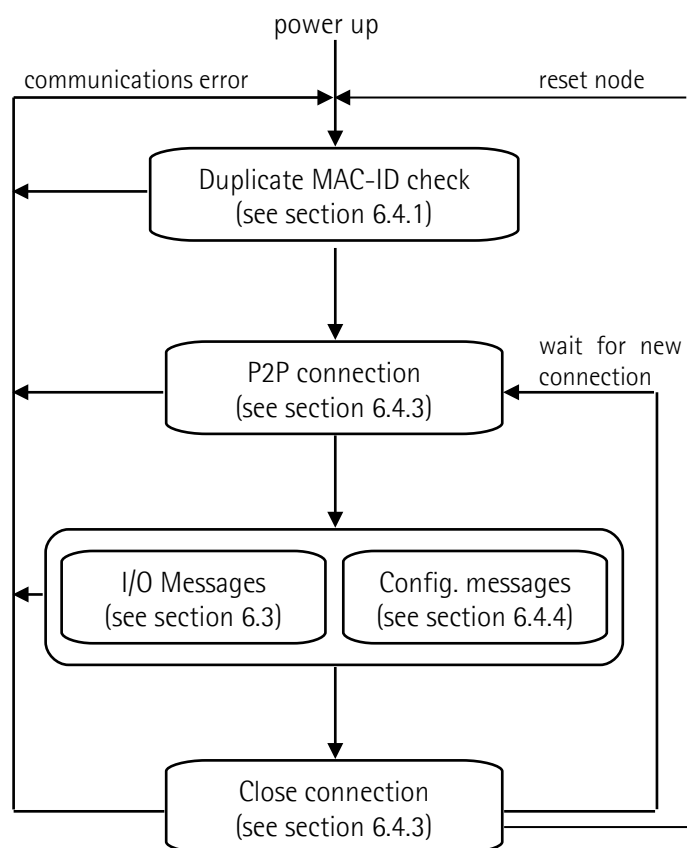
Structure of the communication messages:

CAN-ID	CAN Data bytes
11 bit	from 3 to 8 bytes

Lika DeviceNet draw-wire encoders support two kinds of communication messages:

- **I/O messages:** I/O messages are high priority messages. Their meaning is explained in the "Message-ID". Slave device uses these messages to send process data. For further information see the "6.3 I/O Messages (Msg group 1)" section on page 50.
- **Explicit messages:** Explicit messages are low priority messages. They are used to start and close communications between Master and Slave, to read and write Slave configurations and to send diagnostic messages. For further information see the "6.4 Explicit Messages (Msg group 2)" section on page 51.

Structure of Master-Slave communication:



**NOTE**

Lika encoders accept only one P2P connection at a time. Simultaneous or multiple P2P connections are rejected.

### 6.3 I/O Messages (Msg group 1)

I/O messages are used by Lika devices to send position and velocity values. The "Message-ID" shows how the message is transmitted.

Position:

CAN-ID			4 CAN Data bytes			
10	9...6	5...0	Byte 0	Byte 1	Byte 2	Byte 3
0	Msg ID	Source ID	$2^7...2^0$	$2^{15}...2^8$	$2^{23}...2^{16}$	$2^{31}...2^{24}$
			Low	...	...	High

Position and velocity:

CAN-ID			8 CAN Data bytes							
10	9...6	5...0	Position				Velocity			
0	Msg ID	Source ID	B 0	B 1	B 2	B 3	B 4	B 5	B 6	B 7
			Lo	...	...	Hi	Lo	...	...	Hi

**Msg-ID:**

1100<sub>2</sub>: Slave's I/O Multicast Poll Response Msg

1101<sub>2</sub>: Slave's I/O Change Of State or Cyclic Msg

1111<sub>2</sub>: Slave's I/O Poll Rsp/COS/Cyclic Ack Msg

**Source-ID:** device address.

**Byte 0 ... Byte 3:** position value.

Available Slave transmission modes:

- **Polled:** the Master device sends a polled request and the Slave device replies sending its position value.  
If the expected packet rate of polled mode is null (see [05-Inst-09 Expected packet rate](#) attribute), communication is always active and time-out status is never triggered; otherwise if the Slave does not receive any message from the Master within the time set in the [05-Inst-09 Expected packet rate](#) attribute, then the communication is closed (time-out status).
- **Cyclic:** the Slave sends an "I/O message" cyclically. The time between two subsequent transmissions is set in the [05-Inst-09 Expected packet rate](#) attribute of the cyclic mode (see the "6.5.3 Class 05h: Connection Object" section on page 63).
- **Change Of State:** the Slave sends an "I/O message" every time its status changes or at every heartbeat rate.

The Slave can use one of the afore-mentioned "I/O messages" transmission modes only if a peer-to-peer (P2P) connection is arranged with the Master (using an "Explicit Messages Connection", see the "6.4.3 Explicit Messages Connection" section on page 55).

## 6.4 Explicit Messages (Msg group 2)

These messages are used to:

- perform a duplicate MAC-ID check (see the "6.4.1 Duplicate MAC-ID check" section on page 52);
- send error messages (see the "6.4.2 Error messages" section on page 53);
- arrange an explicit message connection: peer-to-peer (P2P) between Master and Slave (see the "6.4.3 Explicit Messages Connection" section on page 55);
- arrange a message data transfer: send/receive data configuration (see the "6.4.4 Message data transfer" section on page 56).

CAN-ID table of explicit message:

bit	10	9	8	7	6	5	4	3	2	1	0
	1	0	MAC-ID						Msg-ID		

**MAC-ID:** device address.

### Msg-ID:

000<sub>2</sub>: Master's I/O Bit-Strobe Command Msg

001<sub>2</sub>: Master's I/O Multicast Poll Command Msg

010<sub>2</sub>: Master's I/O COS / Cyclic Acknowledge Msg

011<sub>2</sub>: Slave's Explicit / Unconnected Response Msg

100<sub>2</sub>: Master's Connected Explicit Request Msg

101<sub>2</sub>: Master's I/O Poll Command / COS

110<sub>2</sub>: Group 2 only Unconnected Explicit Request Msg

111<sub>2</sub>: Duplicate MAC-ID Check Msg

### 6.4.1 Duplicate MAC-ID check

This function is used to check, when a device is connected to the DeviceNet network, whether its serial number, node-ID and vendor-ID are single (univocal) in the network.

Message structure:

CAN-ID											
bit	10	9	8	7	6	5	4	3	2	1	0
	1	0	MAC-ID						1	1	1

7 CAN Data bytes								
Byte	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	R/R	port number						
1	vendor-ID - Low byte							
2	vendor-ID - High byte							
3	serial number - Low byte							
4	serial number							
5	serial number							
6	serial number - High byte							

**R/R bit** = 0: duplicate MAC-ID check request. This message is sent by a device to all nodes in the network.

1: duplicate MAC-ID check response. This message is sent back to the sender by the node in the network having the same MAC-ID.

**Port number** = 0: device that allows only one P2P connection.

When serial number, node-ID and vendor-ID are not single (univocal) in the network, an error message is generated and the device that sent the "duplicate MAC-ID check request" switches to "Unrecoverable fault" status (MS LED = red, see the "4.9 Diagnostic LEDs" section and the **01-01-05 Status** attribute).

### 6.4.2 Error messages

These messages are meant to warn about device faults.

Message structure:

Byte	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	F=0	0	Master MAC-ID					
1	R/R=1	Service Code = 14h						
2	General error code							
3	Specific service error code							

#### General error codes

General Status Code (in hex)	Status name	Description of Status
00h	<b>No error</b>	Service was successfully performed by the object specified.
02h	<b>Resource unavailable</b>	Resources needed for the object to perform the requested service were unavailable.
03h	<b>Invalid parameter value</b>	A parameter associated with the request was invalid. This code is used when a parameter does not meet the requirements of this specification and/or the requirements defined in an Application Object Specification.
08h	<b>Service not supported</b>	The requested service was not implemented or was not defined for this Object Class/Instance.
09h	<b>Invalid attribute value</b>	Invalid attribute data detected
0Bh	<b>Already in requested mode / state</b>	The object is already in the mode/state being requested by the service.
0Ch	<b>Object state conflict</b>	The object cannot perform the requested service in its current mode/state.
0Eh	<b>Attribute not settable</b>	A request to modify a non-modifiable attribute was received.
10h	<b>Device state conflict</b>	The device's current mode/state prohibits the execution of the requested service.
13h	<b>Not enough data</b>	The service did not supply enough data to perform the specified operation.
14h	<b>Attribute not supported</b>	The attribute specified in the request is not supported.
15h	<b>Too much data</b>	The service supplied more data than

		was expected.
16h	<b>Object does not exist</b>	The object specified does not exist in the device.
20h	<b>Invalid parameter</b>	A parameter associated with the request was invalid. This code is used when a parameter does not meet the requirements of this specification and/or the requirements defined in an Application Object Specification.

For any further information on general error codes refer to the publication "The CIP Networks Library. Volume I. Common Industrial Protocol (CIP™)".

### 6.4.3 Explicit Messages Connection

These messages are meant to open and close P2P connections between Master and Slave. This is necessary to allow the Slave both to communicate process data using "I/O Messages" and communicate or change parameters using "Data transfer" messages.

Open connection request:

Byte	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	F=0	0	Master MAC-ID					
1	R/R=0	Service Code = 4Bh						
2	Class ID = 03h							
3	Instance ID = 01h							
4	Allocation Choice *							
5	Master MAC-ID							

Open connection response:

Byte	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	F=0	0	Master MAC-ID					
1	R/R=1	Service Code = 4Bh						
2	General error code							

Close connection request:

Byte	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	F=0	0	Master MAC-ID					
1	R/R=0	Service Code = 4Ch						
2	Class ID = 03h							
3	Instance ID = 01h							
4	Release choice = Allocation Choice *							

Close connection response:

Byte	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	F=0	0	Master MAC-ID					
1	R/R=1	Service Code = 4Ch						
2	General error code							

\* Allocation choice:

bit 7	Reserved	bit 3	Multi polled
bit 6	Ack suppression	bit 2	Bit strobe
bit 5	Cyclic	bit 1	Polled
bit 4	Change of status	bit 0	Explicit message



#### EXAMPLE

Allocation choice =

01h: used to read and write configuration parameters without "I/O Messages".

03h: activate **Polled** communication.

61h: activate **Cyclic** communication without ACK.

#### 6.4.4 Message data transfer

These messages are meant to set, read, save or restore configuration parameters.

Data transfer request:

Byte	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	F=0	0	Master MAC-ID					
1	R/R=0	Service Code						
2	Class ID							
3	Instance ID							
4	Attribute ID							
5...7	Data byte							

Data transfer response:

Byte	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	F=0	0	Master MAC-ID					
1	R/R=1	Service Code						
2...7	Data byte							

Service code, Class ID, Instance ID and Attribute ID are meant to set the type and content of data bytes. They must be specified according to the "Object dictionary" (see the "6.5 Object dictionary" section on page 57).

## 6.5 Object dictionary

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

### Class-Instance-Attribute Object name

[var, access]

- Class, instance and attribute are expressed in hexadecimal notation.
- Var: data type variable
  - USINT: unsigned single integer 8 bit (1 byte)
  - UINT: unsigned integer 16 bit (2 bytes)
  - UDINT: unsigned double integer 32 bit (4 bytes)
  - WORD: 16 bit specify (2 bytes)
- Access type:
  - ro = read only access
  - rw = read and write access



### NOTE

All data bytes are sent from least significant byte (LSB) to most significant byte (MSB).



### EXAMPLE

UDINT (4 data bytes):

Byte	bit7	bit6	bit5	bit4	bit3	bit2	bit1	bit0
0	F=0	0	Master MAC-ID					
1	R/R=1	Service Code						
2	Data byte - Low byte							
3	Data byte							
4	Data byte							
5	Data byte - High byte							

### 6.5.1 Class 01h: Identity Object

This object provides identification and general information on the device.

Supported **Service codes**:

0Eh = Get\_Attribute\_Single: used to read the connection class attribute value.

Supported **Instances**: 01h = Explicit Message

#### 01-01-01 Vendor-ID

[UINT, ro]

Identification of the vendor by its own number.

Default = 0299h

#### 01-01-02 Device type

[UINT, ro]

Default = 00h: general type device.

#### 01-01-03 Product code

[UINT, ro]

Identification of a particular product of an individual vendor.

#### 01-01-04 Revision

[UINT, ro]

Hardware and software revisions, the Identity Object represents:

LSByte	MSByte
Major revision	Minor revision

#### 01-01-05 Status

[WORD, ro]

This attribute represents the current status of the device. Its value changes as the state of the device changes. Bit definitions are as follows:

Bit(s)	Called	Definition
0	<b>Owned</b>	TRUE indicates the device (or an object within the device) has an owner. Within the Master/Slave paradigm the setting of this bit means that the Predefined Master/Slave Connection Set has been allocated to a Master. Outside the Master/Slave paradigm the meaning of this bit is TBD.
1	Reserved	Reserved, shall be 0
2	<b>Configured</b>	TRUE indicates that the application of the

		device has been configured to do something different than the "out-of-box" default. This shall not include configuration of the communications.																								
3	Reserved	Reserved, shall be 0																								
4-7	Extended device status	Bits are defined as follows: <table><tr><td>0000</td><td>Self-Testing or Unknown</td></tr><tr><td>0</td><td>Firmware Update in Progress</td></tr><tr><td>0</td><td>At least one faulted I/O connection</td></tr><tr><td>0</td><td>No I/O connections established</td></tr><tr><td>0</td><td>Non-Volatile Configuration bad</td></tr><tr><td>0</td><td>Major Fault – either bit 10 or bit 11 is true (1)</td></tr><tr><td>0</td><td>At least one I/O connection in run mode</td></tr><tr><td>0</td><td>At least one I/O connection established, all in idle mode</td></tr><tr><td>1 0 0 0</td><td>Reserved, shall be 0</td></tr><tr><td>1 0 0 1</td><td></td></tr><tr><td>1 0 1 0 ...</td><td>Vendor/Product specific</td></tr><tr><td>1 1 1 1</td><td></td></tr></table>	0000	Self-Testing or Unknown	0	Firmware Update in Progress	0	At least one faulted I/O connection	0	No I/O connections established	0	Non-Volatile Configuration bad	0	Major Fault – either bit 10 or bit 11 is true (1)	0	At least one I/O connection in run mode	0	At least one I/O connection established, all in idle mode	1 0 0 0	Reserved, shall be 0	1 0 0 1		1 0 1 0 ...	Vendor/Product specific	1 1 1 1	
0000	Self-Testing or Unknown																									
0	Firmware Update in Progress																									
0	At least one faulted I/O connection																									
0	No I/O connections established																									
0	Non-Volatile Configuration bad																									
0	Major Fault – either bit 10 or bit 11 is true (1)																									
0	At least one I/O connection in run mode																									
0	At least one I/O connection established, all in idle mode																									
1 0 0 0	Reserved, shall be 0																									
1 0 0 1																										
1 0 1 0 ...	Vendor/Product specific																									
1 1 1 1																										
8	Minor recoverable fault	TRUE indicates that the device detected a problem with itself, which is thought to be recoverable. The problem does not cause the device to go into one of the faulted states.																								
9	Minor unrecoverable fault	TRUE indicates that the device detected a problem with itself, which is thought to be unrecoverable. The problem does not cause the device to go into one of the faulted states.																								
10	Major recoverable fault	TRUE indicates that the device detected a problem with itself, which caused the device to go into the "Major Recoverable Fault" state.																								
11	Major unrecoverable fault	TRUE indicates that the device detected a problem with itself, which caused the device to go into the "Major Unrecoverable Fault" state.																								
12-15	Reserved	Reserved, shall be 0																								

For any further information on status instance attribute refer to the publication "The CIP Networks Library. Volume I. Common Industrial Protocol (CIP™)".

#### 01-01-06 Serial number

[UDINT, ro]

This attribute is a number used in conjunction with the Vendor ID to form a unique identifier for each device on any CIP network.

**01-01-07 Product name**

[SHORT\_STRING, ro]

Default = 05 4C 49 4B 41 20h = "LIKA"

### 6.5.2 Class 03h: DeviceNet Object

This DeviceNet Object is meant to provide the configuration and status of the physical node connected to the DeviceNet network.

Supported **Service code**:

0Eh = Get\_Attribute\_Single: used to read connection class attribute value.

10h = Set\_Attribute\_Single: used to write connection class attribute value.

Supported **Instance**: 01h = Explicit Message

#### 03-01-01 Node Address

[USINT, ro]

This attribute contains the MAC-ID of the device.

To set the node address see the "4.7 Setting the node address: DIP B" section on page 24.

#### 03-01-02 Baud rate

[USINT, ro]

This attribute shows the set baud rate.

Binary value	Baud rate
00	125 Kbit/s
01	250 Kbit/s
10	500 Kbit/s (default)

To set the baud rate see the "4.6 Setting the baud rate: DIP A" section on page 23.

#### 03-01-03 Bus-off interrupt

[BOOL, ro]

This attribute sets how the device has to act after a bus-off event.

00: when a bus-off event occurs, device resets and hold the bus-off (reset) status.

01: when a bus-off event occurs, device resets and then tries to get the previous communication status.

Default = 01h

#### 03-01-04 Bus-off counter

[USINT, ro]

This attribute shows the number of triggered bus-off events.

Default = 00h

**03-01-05 Allocation information**

[UINT, ro]

- 1<sup>st</sup> byte: "Allocation choice": indicates which "Predefined Master/Slave Connections" is active.
- 2<sup>nd</sup> byte: "Master's MAC-ID" contains the MAC-ID of the device that has allocated the "Predefined Master/Slave Connection". The FFh value means that the "Predefined Master/Slave Connection" has not been allocated (no communication active).

### 6.5.3 Class 05h: Connection Object

The Connection Class allocates and manages the internal resources associated to both "I/O Messages" and "Explicit Messaging Connections".

Supported **Service code**:

0Eh = Get\_Attribute\_Single: used to read connection class attribute value.

10h = Set\_Attribute\_Single: used to write connection class attribute value.

Supported **Instance**: 01h = Explicit Message

02h = **Polled**

04h = **Change Of State (COS) / Cyclic**

#### 05-Inst-01 Connection status

[USINT, ro]

This attribute defines the current status of the Connection instance. Value = 03 means that the connection has been configured correctly.

#### 05-Inst-02 Instance type

[USINT, ro]

00h: Explicit messaging

01h: I/O messaging

#### 05-Inst-03 TransportClass\_trigger

[BYTE, ro]

It defines whether this is a producing only, consuming only, or both producing and consuming connection.

#### 05-Inst-04 Produced connection ID

[UINT, ro]

This is the value that will be specified in the CAN Identifier Field when this Connection transmits.

#### 05-Inst-05 Consumed connection ID

[UINT, ro]

This is the CAN Identifier Field value that is associated to the messages this Connection Object receives.

#### 05-Inst-06 Initial comm. Characteristics

[USINT, ro]

Defines the Message Group(s) the sent and received messages are related to.

Default = 21h

#### 05-Inst-07 Produced connection size

[UINT, ro]

Maximum number of bytes transmitted in this connection.

#### 05-Inst-08 Consumed connection size

[UINT, ro]

Maximum number of bytes received in this connection.

#### 05-Inst-09 Expected packet rate

[UINT, rw]

This attribute is meant to set the time between two subsequent "I/O message" transmissions (Transmission Trigger Timer) and the Inactivity/Watchdog Timer.

#### 05-Inst-0C Watchdog time-out action

[USINT, ro]

This attribute defines the action the Connection Object should perform when the Inactivity/Watchdog Timer gap has expired.

Default = 01h: the Connection Class automatically closes the connection after the Inactivity/Watchdog Timer gap has expired.

#### 05-Inst-0D Produced connection path length

[UINT, ro]

This attribute specifies the number of bytes the **05-Inst-0E Produced connection path** attribute consists of.

#### 05-Inst-0E Produced connection path

[EPATH, ro]

This attribute is fitted with a byte stream which defines the Application Object(s) whose data is to be produced by this Connection Object.

#### 05-Inst-0F Consumed connection path length

[UINT, ro]

This attribute specifies the number of bytes the **05-Inst-10 Consumed connection path** attribute consists of.

#### 05-Inst-10 Consumed connection path

[EPATH, ro]

This attribute is fitted with a byte stream which defines the Application Object(s) whose data is to be received by this Connection Object.

**05-Inst-11 Production inhibit time**

[UINT, ro]

Default = 00h: no inhibit time

**NOTE**

To save the parameters execute the "Save parameters" function (see "7.1.4 Saving the parameters" in the "7.1 Read and set parameters" section on page 81).

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

#### 6.5.4 Class 23h: Position Sensor Object

This class is meant to describe the objects used by the device to calculate the transmitted position values.

Supported **Service code**:

- 05h = Reset: resets all parameter values to the factory default values (without saving them on flash memory).
- 0Eh = Get\_Attribute\_Single: used to read connection class attribute value.
- 10h = Set\_Attribute\_Single: used to write connection class attribute value.
- 15h = Restore: restores all parameter values from flash memory (without saving them).
- 16h = Save: saves all parameters to non-volatile memory.

Supported **Instance**: 01h = Explicit Message

#### 23-01-03 Position value

[UDINT, ro]

This attribute represents the absolute position detected by the position sensor. The output value is scaled according to the scaling attributes, see the [23-01-0E Scaling function control](#) attribute.

Reading the position [23-01-03 Position value](#), see also on page 80.

Master → Encoder

CAN-ID	5 Data byte						
404+(ID<<3)	Ms	SC	CI	Ins	At		
	00	0E	23	01	03		

Encoder → Master

CAN-ID	6 Data byte						
403+(ID<<3)	Ms	SC	position				
	00	8E	Low	...	...	High	



#### WARNING

Please note that the position value issued by the encoder is expressed in counts; thus you have then to convert the number of counts into a linear measuring unit.

To convert the position value into millimetres (mm) or micrometres (µm) you have to multiply the number of information by the linear resolution of the encoder expressed in millimetres or micrometres.

To know the linear resolution of the encoder please consider that **the stroke per turn of the drum is 200 mm / 7.874"**.

The linear resolution results from the following calculation:

$$\text{Linear resolution} = \frac{\text{Stroke per turn of the drum mm}}{\text{Singleturn resolution cpr}}$$

If you want to know the linear position value you will need to multiply the transmitted position value by the linear resolution.

$$\text{Linear position value} = \text{transmitted position} * \text{linear resolution}$$



#### NOTE

Please note that the encoder's linear resolution can be read also in the order code next to the rotary resolution. Refer to the product datasheet.



#### EXAMPLE 1

Let's suppose that we are using the physical resolution of the SFAM1-05000-FD2-08192-RPG draw-wire encoder (the **23-01-0E Scaling function control** attribute = 0).

The physical singleturn resolution of the measuring device is 8,192 cpr (= 0.024 mm, see the order code in the product datasheet).

As stated, the linear resolution results from the following calculation:

$$\text{Linear resolution} = \frac{\text{Stroke per turn of the drum mm}}{\text{Singleturn resolution cpr}}$$

$$\text{Linear resolution} = \frac{200 \text{ mm}}{8,192 \text{ cpr}} = 0.024 \text{ mm} = 24 \text{ }\mu\text{m}$$

Let's say that the transmitted position value is 123.

Thus the linear position value will be as follows:

$$\text{Linear position value} = \text{transmitted position} * \text{linear resolution}$$

$$\text{Linear position value} = 123 * 0.024 = 2.952 \text{ mm} = 2,952 \text{ }\mu\text{m}$$



#### EXAMPLE 2

Let's suppose that we are using the SFAM1-05000-FD2-08192-RPG draw-wire encoder. The singleturn resolution is set to the custom value of 4,000 cpr (**23-01-10 Resolution per revolution** = 4000). The transmitted position value is 1,569.

The linear resolution can be easily calculated as follows:

$$\text{Linear resolution} = \frac{200 \text{ mm}}{4,000 \text{ cpr}} = 0.05 \text{ mm} = 50 \text{ }\mu\text{m}$$

Thus the linear position value will be as follows:

**Linear position value** =  $1,569 * 0.05 = 78.45 \text{ mm} = 78,450 \text{ }\mu\text{m}$

### 23-01-0B Device type

[UINT, ro]

Type of device.

0002h: multiturn absolute rotary encoder.

### 23-01-0C Code sequence

[BOOL, rw]

**23-01-0C Code sequence** attribute sets whether the position value output by the encoder increases (count up information) when you rewind the wire (00) or when you pull the wire out (01, default).

00: rewinding the wire the position will increase;

**01: pulling the wire out the position will increase.**



#### WARNING

Every time you change the **23-01-0C Code sequence** then you are required to set a new preset value (see the **23-01-13 Preset value** attribute) and finally save the new parameters (execute the "Save parameters" function, see "7.1.4 Saving the parameters" in the "7.1 Read and set parameters" section on page 81).

### 23-01-0E Scaling function control

[BOOL, rw]

If this attribute is disabled (OFF = 00), the device uses the physical resolution values (see the **23-01-2A Hardware counts per revolution** and **23-01-2B Hardware number of turns** attributes); if it is enabled (ON = 01), it uses the custom resolution set in the **23-01-10 Resolution per revolution** and **23-01-11 Total measuring range** attributes with the following relation:

Transmitted position =

$$\frac{\text{23-01-10 Resolution per revolution}}{\text{23-01-2A Hardware counts per revolution}} * \text{real position} \leq \text{23-01-11 Total measuring range}$$

Default = 00h



#### WARNING

When you enable the scaling function (**23-01-0E Scaling function control** = 01), please enter scaled values next to the **23-01-10 Resolution per**

**revolution** and **23-01-11 Total measuring range** attributes that are consistent with the physical values.



#### WARNING

Every time you enable the scaling function and/or change the scaling values (see the **23-01-10 Resolution per revolution** and **23-01-11 Total measuring range** attributes) then you are required to set a new preset value (see the **23-01-13 Preset value** attribute) and finally save the new parameters (execute the "Save parameters" function, see "7.1.4 Saving the parameters" in the "7.1 Read and set parameters" section on page 81).

#### 23-01-10 Resolution per revolution

[UDINT, rw]



#### WARNING

This attribute is active only if the **23-01-0E Scaling function control** attribute is set to "=01"; otherwise it is ignored and the system uses the physical values (**23-01-2A Hardware counts per revolution** and **23-01-2B Hardware number of turns**) to calculate the position information.

This attribute sets a custom number of distinguishable steps per revolution (custom singleturn resolution).

To avoid counting errors, check that

$$\frac{\text{23-01-2A Hardware counts per revolution}}{\text{23-01-10 Resolution per revolution}} = \text{integer value.}$$

You are allowed to set whatever integer value less than or equal to the **maximum number of physical steps per revolution** (see the hardware counts per revolution in the encoder identification label and the **23-01-2A Hardware counts per revolution** attribute).

Default = 0000 2000h (8,192 cpr)



#### WARNING

When you set a new value next to the **23-01-10 Resolution per revolution** attribute, please always check also the **23-01-11 Total measuring range** attribute value and be sure that the resulting number of revolutions complies with the **Hardware number of revolutions** of the device (4,096 revolutions, see the **23-01-2B Hardware number of turns** attribute).

Let's suppose that the encoder is programmed as follows:

**23-01-10 Resolution per revolution**: 8,192 cpr

**23-01-11 Total measuring range** =  $33\,554\,432_{10} = 8,192 \text{ (cpr)} * 4,096 \text{ (rev.)}$

Let's set a new singleturn resolution, for instance: **23-01-10 Resolution per revolution** = 360 cpr.

If we do not change the **23-01-11 Total measuring range** value at the same time, we will get the following result:

$$\text{Number of revolutions} = \frac{33\,554\,432 \text{ (23-01-11 Total measuring range)}}{360 \text{ (23-01-10 Resolution per revolution)}} = 93,206.755...$$

As you can see, the encoder is required to carry out more than 93,000 revolutions, this cannot be as the hardware number of revolutions is, as stated, 4,096. When this happens, the encoder falls into an error signalling the faulty condition through the diagnostic LEDs (see on page 26).



#### WARNING

When you enable the scaling function (**23-01-0E Scaling function control** = 01), please enter scaled values next to the **23-01-10 Resolution per revolution** and **23-01-11 Total measuring range** attributes that are consistent with the physical values. In the case of inconsistent values, the system will warn about the wrong parametrization and fault condition by means of the dedicated attributes and visually by means of the diagnostic LEDs.



#### WARNING

Every time you change the scaled values (see the **23-01-10 Resolution per revolution** and **23-01-11 Total measuring range** attributes), then you are required to set a new preset value (see the **23-01-13 Preset value** attribute).

#### 23-01-11 Total measuring range

[UDINT, rw]



#### WARNING

This attribute is active only if the **23-01-0E Scaling function control** attribute is set to "01"; otherwise it is ignored and the system uses the physical values (**23-01-2A Hardware counts per revolution** and **23-01-2B Hardware number of turns**) to calculate the position information.

This attribute sets a custom number of distinguishable steps over the total measuring range. In other words, this attribute allows to set the length of the travel the encoder has to measure expressed in number of distinguishable steps (number of information). The total resolution of the encoder results from the

product of **23-01-10 Resolution per revolution** by the required **Number of revolutions**.

You are allowed to set whatever integer value less than or equal to the **overall hardware resolution** (see the encoder identification label). The overall hardware resolution results from:

**23-01-2A Hardware counts per revolution** \* **23-01-2B Hardware number of turns**.

We recommend the **Number of revolutions** to be set to a power of 2.

The set **Number of revolutions** results from the following calculation:

$$\text{Number of revolutions} = \frac{\text{23-01-11 Total measuring range}}{\text{23-01-10 Resolution per revolution}}$$

Setting the **Number of revolutions** to a value which is a power of 2 is meant to avoid problems when using the device in endless operations requiring the physical zero to be overstepped. If you set the **Number of revolutions** which is not a power of 2, a counting error is generated before the physical zero.

Default = 0200 0000h (33 554 432)



#### WARNING

When you set a new value next to the **23-01-11 Total measuring range** attribute, please always check also the **23-01-10 Resolution per revolution** attribute value and be sure that the resulting number of revolutions complies with the Hardware number of revolutions of the device (4,096 revolutions).

Let's suppose that the encoder is programmed as follows:

**23-01-10 Resolution per revolution**: 8,192 cpr

**23-01-11 Total measuring range** = 33 554 432<sub>10</sub> = 8,192 (cpr) \* 4,096 (rev.)

Let's set a new total resolution, for instance: **23-01-11 Total measuring range** = 360.

As the **23-01-11 Total measuring range** must be greater than or equal to the **23-01-10 Resolution per revolution**, the above setting is not allowed.



#### EXAMPLE

We install the following draw-wire encoder: **SFAM1-05000-FD2-08192-RM12**.

The physical values are:

Stroke per turn of the drum = 200 mm (7.874")

Physical resolution per turn = 13 bits = 8,192 cpr

Max. number of physical revolutions = 4,096 revolutions

Total physical resolution = 25 bits = 33 554 432 information

Physical linear resolution = 0.024 mm = 24 µm

Max. number of turns of the drum = 25

Max. measuring length = 5,000 mm (196.85")

Number of information = 204,800

Let's suppose that we need a tenth of a millimetre linear resolution in the specific installation.

- Enable the scaling function: **23-01-0E Scaling function control** attribute = 01
- Custom resolution per turn = **23-01-10 Resolution per revolution** = 2,000 cpr
- Linear resolution = 0.1 mm = 100 µm

$$\text{Linear resolution} = \frac{\text{Stroke per turn mm}}{\text{23-01-10 Resolution per revolution}} = \frac{200 \text{ mm}}{2,000 \text{ cpr}} = 0.1 \text{ mm}$$

The custom number of revolutions can be as the physical number of revolutions:

$$\text{Custom number of encoder revolutions} = \frac{\text{23-01-11 Total measuring range}}{\text{23-01-10 Resolution per revolution}} = 4,096$$

- **23-01-11 Total measuring range** = 8 192 000



#### NOTE

Please note that if you set a preset along the path, when the encoder moves back and cross the zero, the value immediately after 0 will be 8 192 000 - 1, i.e. 8 191 999.

...	8,191,997	8,191,998	8,191,999	0	1	2	...
-----	-----------	-----------	-----------	---	---	---	-----



#### EXAMPLE

Using the values in the previous example let's suppose that the travel in the application is 2 m long. As the stroke per turn is 200 mm you need 10 revolutions to cover the travel length.

- **23-01-11 Total measuring range** = **23-01-10 Resolution per revolution** \* custom number of revolutions = 2,000 \* 10 = 20,000

In fact:

$$\text{Custom number of encoder revolutions} = \frac{\text{23-01-11 Total measuring range}}{\text{23-01-10 Resolution per revolution}} = 10$$

In this case you will obtain several 20,000 information sections following each other all along the whole measuring length. The position information will be from 0 to 19,999; then again from 0 to 19,999 and so on.

...	19,997	19,998	19,999	0	1	2	...	19,997	19,998	19,999	0	1	2	...
← max. measuring length →														



#### NOTE

To avoid counting errors we recommend values which are a power of 2 ( $2^n$ : 2, 4, ..., 2048, 4096, 8192,...) to be set next to the **23-01-10 Resolution per revolution** and **23-01-11 Total measuring range** attributes.



#### WARNING

If you have set the preset, when you change the value next to **23-01-10 Resolution per revolution** and/or **23-01-11 Total measuring range** attribute, then you must check the value in the **23-01-13 Preset value** attribute and execute the preset operation.

#### 23-01-13 Preset value

[UDINT, rw]

This attribute allows to set the encoder position to a Preset value. The Preset function is meant to assign a desired value to a physical position of the encoder (i.e. a position in the travel of the wire). The chosen physical position will get the value set next to this item and all the previous and following positions will get a value according to it. This function can be useful, for instance, when the zero position of the encoder and the zero position of the axis need to match.

The preset value will be set for the position of the encoder (i.e. the position of the wire) in the moment when the preset value is transmitted. We suggest setting the preset value when the encoder is in stop.

Default = 0000 0000h

Setting **23-01-13 Preset value**, see also on page 80.

Master → Encoder (1<sup>st</sup> message)

CAN-ID	8 Data byte							
404+(ID<<3)	Ms	FC	SC	CI	Ins	At	preset	
	80	00	10	23	01	13	Low	...

Encoder → Master (1<sup>st</sup> message received)

CAN-ID	3 Data byte							
403+(ID<<3)	Ms	SC	Err					
	80	C0	00					

Master → Encoder (2<sup>nd</sup> message)

CAN-ID	4 Data byte							
404+(ID<<3)	Ms	FC	preset					
	80	81	...	Hi				

Encoder → Master (2<sup>nd</sup> message received)

CAN-ID	3 Data byte						
403+(ID<<3)	Ms	SC	Err				
	80	C1	00				

Encoder → Master (parameter value accepted)

CAN-ID	3 Data byte						
403+(ID<<3)	Ms	SC	Err				
	00	90	00				



### EXAMPLE

Let's take a look at the following example to better understand the preset function and the meaning and use of the related attributes: **23-01-13 Preset value** and **23-01-33 Offset**.

The encoder position which is transmitted results from the following calculation:

**Transmitted value** = **read position** (it does not matter whether the position is physical or scaled) + **23-01-13 Preset value** - **23-01-33 Offset**.

If you never set the **23-01-13 Preset value** and you never performed the preset setting, then the transmitted value and the read position are necessarily the same as **23-01-13 Preset value** = 0 and **23-01-33 Offset** = 0.

When you set the **23-01-13 Preset value** and then execute the preset setting, the system saves the current encoder position in the **23-01-33 Offset** attribute. It follows that the transmitted value and the **23-01-13 Preset value** are the same as **read position** - **23-01-33 Offset** = 0; in other words, the value set next to the **23-01-13 Preset value** attribute is paired with the current position of the encoder as you wish.

For example, let's assume that the value "50" is set next to the **23-01-13 Preset value** attribute and you execute the preset setting when the encoder position is "1000". In other words, you want to receive the value "50" when the encoder reaches the position "1000".

We will obtain the following information sequence:

**Transmitted value** = **read position** (= "1000") + **23-01-13 Preset value** (= "50") - **23-01-33 Offset** (= "1000") = 50.

The following transmitted value will be:

**Transmitted value** = **read position** (= "1001") + **23-01-13 Preset value** (= "50") - **23-01-33 Offset** (= "1000") = 51.

And so on.



#### NOTE

- If the scaling function is disabled (see the [23-01-0E Scaling function control](#) attribute), the [23-01-13 Preset value](#) must be less than or equal to the "Total hardware resolution" - 1 ([23-01-2A Hardware counts per revolution](#) \* [23-01-2B Hardware number of turns](#) - 1).
- If the scaling function is enabled (see the [23-01-0E Scaling function control](#) attribute), the [23-01-13 Preset value](#) must be less than or equal to the [23-01-11 Total measuring range](#) - 1.



#### WARNING

Check the value in the [23-01-13 Preset value](#) attribute and perform the preset operation every time you change the [23-01-0C Code sequence](#) or the scaled values ([23-01-10 Resolution per revolution](#) and/or [23-01-11 Total measuring range](#) attributes).

#### 23-01-14 Delta for COS

[UDINT, rw]

When the **Change Of State** (COS) mode connection is enabled, this parameter sets the minimum interval between two I/O messages. Values too low could cause the Bus network to saturate.

Default = 05h

#### 23-01-18 Velocity value

[UDINT, ro]

This attribute represents the current speed detected by the position sensor and calculated every 100 ms. The speed can be expressed in number of steps per second or in revolutions per minute (see the following [23-01-19 Velocity format](#) attribute).

#### 23-01-19 Velocity format

[USINT, rw]

This attribute identifies the engineering units for the velocity value. See the previous [23-01-18 Velocity value](#) attribute.

00: steps/s      number of steps per second  
01: rpm          revolutions per minute

#### 23-01-2A Hardware counts per revolution

[UDINT, ro]



#### WARNING

This attribute is active only if the [23-01-0E Scaling function control](#) attribute is set to "00"; otherwise it is ignored and the system uses the custom values

([23-01-10 Resolution per revolution](#) and [23-01-11 Total measuring range](#)) to calculate the position information.

This attribute is intended to show the number of physical distinguishable steps per each revolution provided by the hardware (physical singleturn resolution). If you want to set a custom singleturn resolution see the [23-01-10 Resolution per revolution](#) attribute on page 69.

#### 23-01-2B Hardware number of turns

[UINT, ro]



#### WARNING

This attribute is active only if the [23-01-0E Scaling function control](#) attribute is set to "00"; otherwise it is ignored and the system uses the custom values ([23-01-10 Resolution per revolution](#) and [23-01-11 Total measuring range](#)) to calculate the position information.

This attribute is intended to show the number of physical revolutions provided by the hardware (number of physical revolutions).

The **Total hardware resolution** results from [23-01-2A Hardware counts per revolution](#) \* [23-01-2B Hardware number of turns](#).

If you want to set a custom number of revolutions see the [23-01-10 Resolution per revolution](#) and [23-01-11 Total measuring range](#) attributes on page 69 ff.

#### 23-01-2C Alarms

[WORD, ro]

An alarm is set when a bit indicating a fault is set to true (high). See the following [23-01-2D Supported alarms](#) values.

#### 23-01-2D Supported alarms

[WORD, ro]

This attribute contains information on the supported alarms.

Bits 0 ... 11	Reserved	
Bit 12	<b>Flash memory error</b>	Internal error, the alarm cannot be restored.
Bit 13	<b>Machine data not valid</b>	One or more parameters are not valid, set proper values to restore normal work condition.
Bit 14	<b>Bus off</b>	This bit is intended to warn that the communication in the bus network has

		broken (cable disconnected? Voltage drop? ...).
--	--	---

### 23-01-2E Alarm flags

[BOOL, ro]

It indicates that a fault occurred and an alarm has been triggered, see the previous [23-01-2D Supported alarms](#) attribute.

### 23-01-33 Offset

[DINT, ro]

As soon as you activate the preset, the current position value of the encoder is saved on this attribute. The offset value is then used in the preset function in order to calculate the encoder position value to be transmitted. To zero set the value in this attribute you must upload the factory default values (see the 15h Restore Service code on page 66).

For any further information on the preset function and the meaning and use of the related attributes [23-01-13 Preset value](#) and [23-01-33 Offset](#) please refer to page 73.

### 23-01-65 Dip switch

[UINT, ro]

This attribute shows the status of the DIP switches.

Dip switches are located in the connection cap and used to set the baud rate and the MAC-ID (see the sections "4.6 Setting the baud rate: DIP A" and "4.7 Setting the node address: DIP B").

bits 0 ... 5 = MAC-ID  
bits 6 and 7 = not used  
bits 8 and 9 = Baud Rate  
bits 10 ... 15 = not used

### 23-01-66 Data Tx in Poll mode

[USINT, rw]

This attribute is meant to set which input data is sent when **Polled** mode connection is enabled.

**00**: Position value (default)  
**01**: Position value + velocity

### 23-01-67 Data Tx in COS/Cyclic mode

[USINT, rw]

This parameter is meant to set which input data is sent when the **Change Of State** (COS) / **Cyclic** mode connection is enabled.

**00**: Position value (default)

01: Position value + velocity

### 23-01-68 Auto-save parameters

[BOOL, rw]

- 0 = No:** new parameters are not saved on EPROM automatically. To save new values, the Class Instance Editor must be used (default value; see the "5.4 Saving the parameters with RSNetWorx" section on page 33).
- 1 = Yes:** when new parameters are set, the encoder waits 5 seconds and then saves automatically all new values on EPROM, but only if the received value has been changed.



#### NOTE

To save new parameters execute the "Save parameters" function (see "7.1.4 Saving the parameters" in the "7.1 Read and set parameters" section on page 81).

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

### 6.5.5 Class 2Bh: Acknowledge Handler Object

This class is meant to manage the receipt of acknowledgement messages.

Supported **Service code**:

0Eh = Get\_Attribute\_Single: used to read connection class attribute value.

10h = Set\_Attribute\_Single: used to write connection class attribute value.

Supported **Instance**: 01h = Explicit Message

#### 2B-01-01 Acknowledge timer

[UINT, rw]

After a Master request, the Slave waits for the set acknowledgement time before sending back a reply.

Default = 10h: 16 ms

#### 2B-01-02 Retry Limit

[USINT, rw]

Number of Ack Timeouts after which the application informs the Master that a RetryLimit\_Reached event occurred.

#### 2B-01-03 Connection Instance of COS

[UINT, rw]

Connection Instance which contains the path where the Ack Handler events will be notified.

## 7 Setup

Here follow some examples of parameters reading and setting; data exchange between Master and Slave devices is highlighted. A generic "ID" value is used to indicate the encoder address; Master address is always assumed to be 0.

All values are written in hexadecimal notation.

Abbreviations: (ID<<3): 3 bits left logic shift

Ms: Master ID

FC: Fragmented message counter

SC: Service Code

Cl: Class

Ins: Instance

AC: Allocation Choice

At: Attribute

Err: Error code

### 7.1 Read and set parameters

#### 7.1.1 P2P Master/Slave connection (without I/O msg)

Master → Encoder (request)

CAN-ID	6 Data byte							
406+(ID<<3)	Ms	SC	Cl	Ins	AC	Ms		
	00	4B	03	01	01	00		

Encoder → Master (response)

CAN-ID	3 Data byte							
403+(ID<<3)	Ms	SC	Err					
	00	CB	00					

#### 7.1.2 Reading the position **23-01-03 Position value** (in the "Object dictionary")

Master → Encoder

CAN-ID	5 Data byte							
404+(ID<<3)	Ms	SC	Cl	Ins	At			
	00	0E	23	01	03			

Encoder → Master

CAN-ID	6 Data byte							
403+(ID<<3)	Ms	SC	position					
	00	8E	Low	...	...	High		

### 7.1.3 Setting 23-01-13 Preset value

Master → Encoder (1<sup>st</sup> message)

CAN-ID	8 Data byte							
404+(ID<<3)	Ms	FC	SC	Cl	Ins	At	preset	
	80	00	10	23	01	13	Low	...

Encoder → Master (1<sup>st</sup> message received)

CAN-ID	3 Data byte							
403+(ID<<3)	Ms	SC	Err					
	80	C0	00					

Master → Encoder (2<sup>nd</sup> message)

CAN-ID	4 Data byte							
404+(ID<<3)	Ms	FC	preset					
	80	81	...	Hi				

Encoder → Master (2<sup>nd</sup> message received)

CAN-ID	3 Data byte							
403+(ID<<3)	Ms	SC	Err					
	80	C1	00					

Encoder → Master (parameter value accepted)

CAN-ID	3 Data byte							
403+(ID<<3)	Ms	SC	Err					
	00	90	00					

### 7.1.4 Saving the parameters

Master → Encoder

CAN-ID	5 Data byte							
404+(ID<<3)	Ms	SC	Cl	Ins	At			
	00	16	23	01	01			

Encoder → Master

CAN-ID	3 Data byte							
403+(ID<<3)	Ms	SC	Err					
	00	96	00					

### 7.1.5 Closing the Master/Slave connection

Master → Encoder

CAN-ID	5 Data byte							
406+(ID<<3)	Ms	SC	CI	Ins	AC			
	00	4C	03	01	01			

Encoder → Master

CAN-ID	3 Data byte							
403+(ID<<3)	Ms	SC	Err					
	00	CC	00					

## 7.2 Setting the Cyclic mode without velocity

### 7.2.1 P2P Master/Slave connection

Master → Encoder (request)

CAN-ID	6 Data byte							
406+(ID<<3)	Ms	SC	CI	Ins	AC	Ms		
	00	4B	03	01	61	00		

Encoder → Master (response)

CAN-ID	3 Data byte							
403+(ID<<3)	Ms	SC	Err					
	00	CB	00					

### 7.2.2 Setting 05-Inst-09 Expected packet rate for Cyclic mode (milliseconds)

Master → Encoder

CAN-ID	7 Data byte							
404+(ID<<3)	Ms	SC	CI	Ins	At	Ctime		
	00	10	05	04	09	Low	Hi	

Encoder → Master

CAN-ID	3 Data byte							
403+(ID<<3)	Ms	SC	Err					
	00	90	00					

From now on, the encoder sends I/O messages at the interval set in the **05-Inst-09 Expected packet rate** attribute until the connection will be closed or the **05-Inst-09 Expected packet rate** attribute will be modified.

Encoder → Master

CAN-ID	4 Data byte							
340+ID	position							
	Low	...	...	Hi				

## 8 Default parameters table

Parameter list	Default value		
3 Counting direction	1 = count up information when pulling the cable out		
4 Scaling function control	0 = DISABLED		
5 Measuring units/rev	8192		
6 Total measuring range	33 554 432		
7 Preset value	0		
14 Velocity format	0 = STEPS/S		
16 Data Tx in POLL mode	0 = POSITION VALUE		
17 Data Tx in COS/Cyclic mode	0 = POSITION VALUE		
19 Delta for COS	5		
20 Auto-save parameters	0 = NO		

This page intentionally left blank

This page intentionally left blank

This page intentionally left blank

Document release	Release date	Description	HW	SW	EDS file version
1.0	19.04.2018	1st issue	1.0	1.002	V1
1.1	02.03.2023	New product name, new order code, minor amendments, general revision	1.0	1.002	V1



Dispose separately

**lika**

**Lika Electronic**

Via S. Lorenzo, 25 • 36010 Carrè (VI) • Italy

Tel. +39 0445 806600

Fax +39 0445 806699



info@lika.biz • www.lika.biz