lika

Smart encoders & actuators

User's guide

HM58 PN
HM58 PY

Programmable encoder with parallel interface

- Programmable multiturn absolute encoder
- Parallel interface, 28 bit max. position information
- 32 bit max. resolution (262,144 cpr x 16,384 rev.)
- Programmable scaling and TEACH-IN function
- Solid and hollow shaft versions
- Free-of-charge programming software

Suitable for the following models:
- HM58-18-14-PN2-...
- HM58-18-14-PY2-...
- HM58S-18-14-PN2-...
- HM58S-18-14-PY2-...
- HMC58-18-14-PN2-...
- HMC58-18-14-PY2-...
- HMC59-18-14-PN2-...
- HMC59-18-14-PY2-...
- HMC60-18-14-PN2-...
- HMC60-18-14-PY2-...

Table of Contents

Lika Electronic • Tel. +39 0445 806600 • info@lika.biz • www.lika.biz

lika

General contents

Subject Index

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 Lika device and 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 **WARNING**, 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 **NOTE**, 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 **EXAMPLE** 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 **absolute multiturn programmable encoder equipped with parallel interface model HM58 PY (parallel output with Push-Pull driver) and HM58 PN (parallel output with NPN o. c. driver)**.

HM58 PY-PN programmable encoder from Lika Electronic is designed to maximize customization and versatility and really allows the operator to get a complete parametrization and the better configuration as well as the absolute position information. In this way it can perfectly suit specific needs in a good many applications and machines. The operator can configure not only the parallel interface by setting for instance the output code and the logic level as well as the counting direction; but also to program the desired resolution. Please note that in this encoder the maximum number of physical information per revolution is 262,144 (2^{18}), while the maximum number of physical revolutions is 16,384 (2^{14}). However the overall resolution (total counts) you are allowed to set is 268,435,456 (2^{28}). It follows that you must arrange the counts per revolution and the number of revolutions so that the overall resolution is not greater than 268,435,456 information.
HM58 PY-PN programmable encoder is available in both hollow and solid shaft versions.

The parametrization and set up of the HM58 PY-PN programmable encoder is achieved through a software expressly developed and released by Lika Electronic. The program is supplied for free and can be installed in any PC fitted with a Windows operating system (Windows XP or later). It allows the operator to set the working parameters of the device and monitor whether the device is running properly. Communication is established using **RS-232 serial cable and port**. The software tool is compatible with the whole series of HM- and EM- programmable encoders: programmable encoder with SSI serial output (...-PS2-... code), programmable encoder with parallel output (...-PN2-... and ...-PY2-... codes), programmable encoder with analogue output (...-PA2-... code).

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 unit are provided.
In the second section, entitled **Software tool and configuration parameters**, the programming interface and the parameters 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 "Electrical connection" section on page 17;
- the wires of unused output signals must be cut at different lengths and insulated singularly;
- connect Preset, Counting direction, and Latch inputs to 0Vdc if not used;
  - to set the Preset, connect Preset input to +Vdc for 100 μs at least, then disconnect +Vdc; normally Preset voltage must be at 0Vdc; preset must be performed after Counting direction; we suggest performing the preset while the encoder is in stop;
  - Counting direction: CW increasing count (viewed from the shaft side) = connect to 0Vdc; CCW increasing count = connect to +Vdc;
  - Latch: 0Vdc = deactivated; +Vdc = activated (information is "frozen");
- Tri-state: +Vdc = deactivated (encoder standard work mode); 0Vdc = activated (encoder outputs disabled);
- in compliance with 2014/30/EU norm on electromagnetic compatibility, the following precautions must be taken:

- before handling and installing the equipment, 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.

1.3 Mechanical safety

- Install the device following strictly the information in the "Mechanical installation" section on page 12;
- mechanical installation has to be carried out with stationary mechanical parts;
- do not disassemble the unit;
- do not tool the unit or its shaft;
- delicate electronic equipment: handle with care; do not subject the device and the shaft to knocks or shocks;
- respect the environmental characteristics of the product;
- unit with solid shaft: in order to guarantee the maximum reliability over time of the mechanical parts, we recommend a flexible coupling to be installed to connect the encoder and the installation shaft; make sure the misalignment tolerances of the flexible coupling are met;
- unit with hollow shaft: the encoder can be mounted directly on a shaft whose diameter has to respect the technical characteristics specified in the purchase order and clamped by means of the collar and the fixing plate into which an anti-rotation pin has to be inserted.

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 <u>refer to the technical catalogue</u>.

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 Mechanical installation

⚠️ **WARNING**
Installation has to be carried out by qualified personnel only, with power supply disconnected and mechanical parts compulsorily in stop.

3.1 Encoder with solid shaft

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

3.1.1 Customary installation
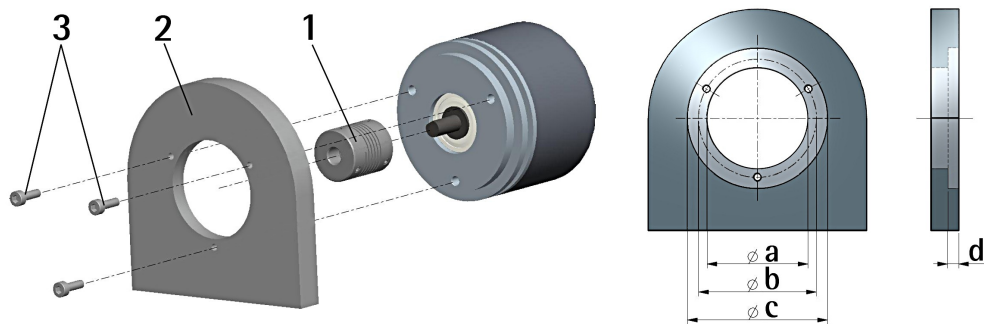
	a [mm]	b [mm]	c [mm]	d [mm]
HM58 PY-PN	-	42	50 F7	4
HM58S PY-PN	36 H7	48	-	-

3.1.2 Installation using fixing clamps (optional kit code LKM 386)
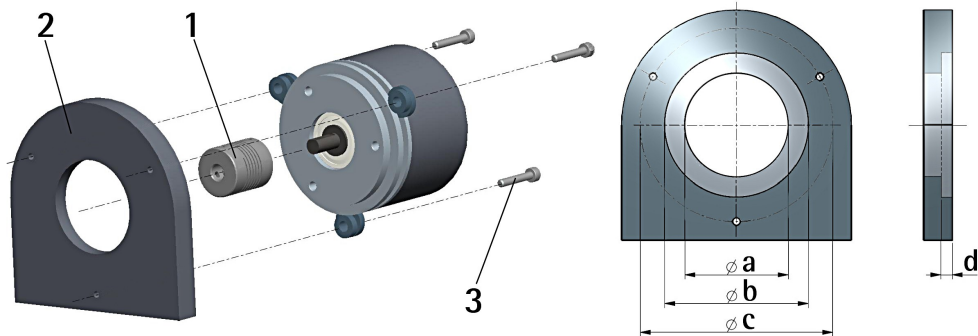
	a [mm]	b [mm]	c [mm]	d [mm]
HM58 PY–PN	–	50 F7	67	4
HM58S PY–PN	36 H7	–	67	–

3.1.3 Installation using a mounting bell (optional kit code PF4256)
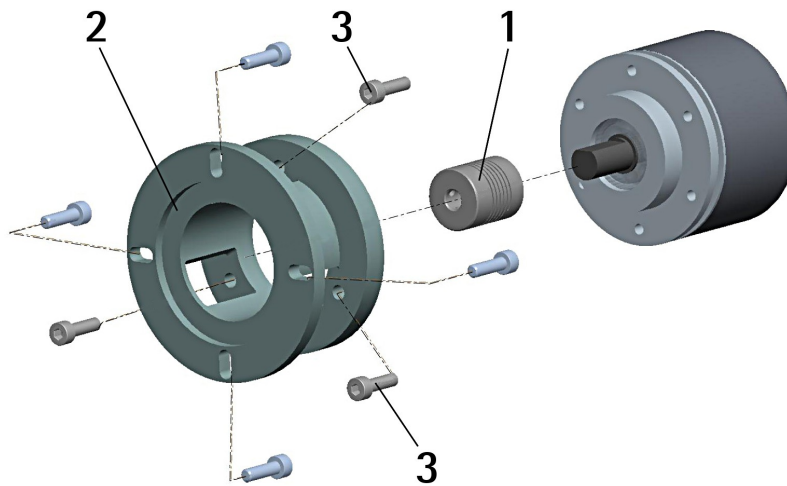
NOTE

In order to guarantee reliability over time of the encoder mechanical parts, we recommend a flexible coupling to be installed between the encoder and the motor shaft. Make sure the misalignment tolerances of the flexible coupling are respected.

3.2 Encoder with hollow shaft

3.2.1 HMC58 PY-PN installation using the antirotation pin

- Fasten the anti-rotation pin **1** to the rear of the motor (secure it using a locknut);
- mount the encoder on the motor shaft using the reducing sleeve **8** (if supplied). Avoid forcing the encoder shaft;
- insert the anti-rotation pin **1** into the slot on the flange of the encoder; this secures it in place by grub screw **2**, preset at Lika;
- fix the collar **3** to the encoder shaft.
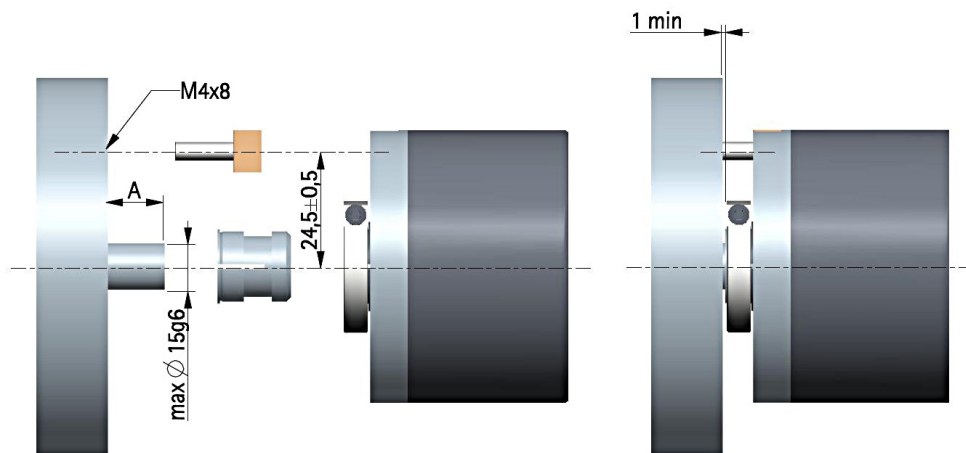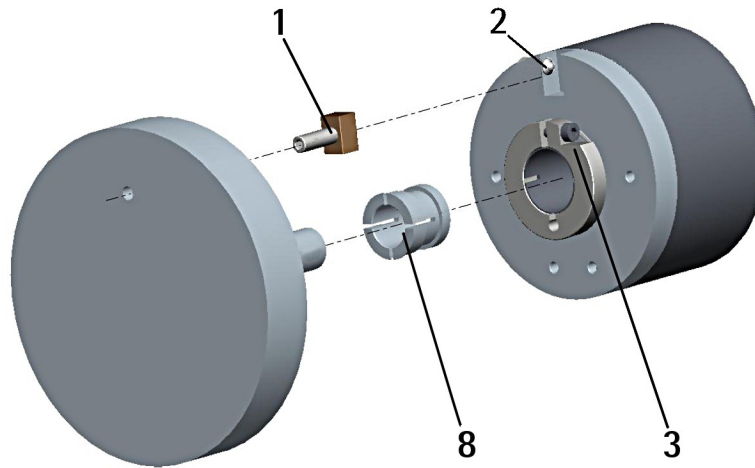
A = min. 8, max. 18 mm

3.2.2 HMC59 PY-PN installation using the fixing plate

- Remove the antirotation pin **1** (see the Figure in the previous page);
- mount the encoder on the motor shaft using the reducing sleeve **8** (if supplied). Avoid forcing the encoder shaft;
- fasten the fixing plate **4** to the rear of the motor using two M3 x 8 cylindrical head screws **5**;
- fix the collar **3** to the encoder shaft.

A = min. 8, max. 18 mm
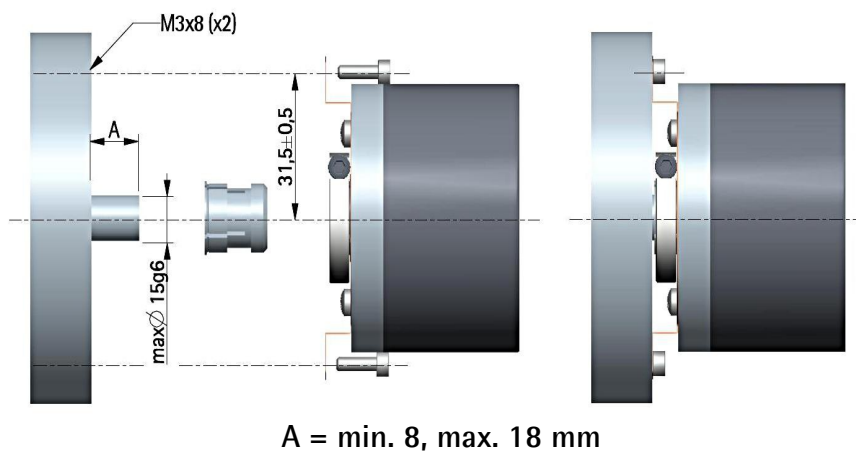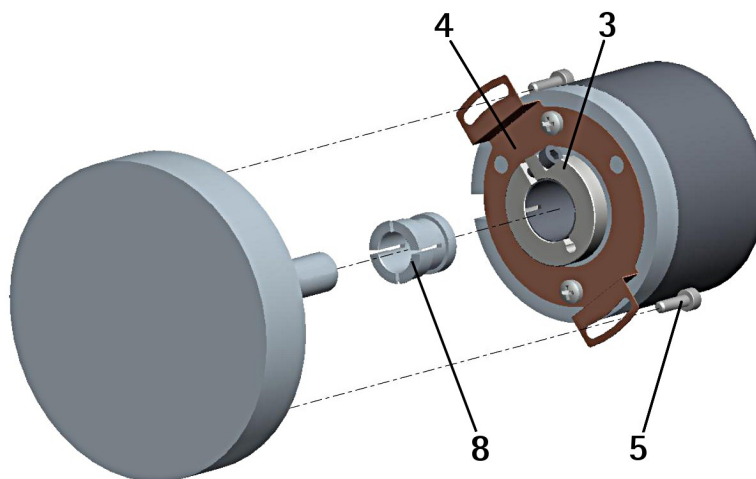
3.2.3 HMC60 PY-PN installation using the antirotation pin and the fixing plate

- Remove the antirotation pin **1** (see the Figure on page 14);
- fix the tempered pin **6** to the rear of the motor;
- mount the encoder on the motor shaft using the reducing sleeve **8** (if supplied). Avoid forcing the encoder shaft;
- make sure the anti-rotation pin **6** is inserted properly into the fixing plate **7**;
- fix the collar **3** to the encoder shaft.
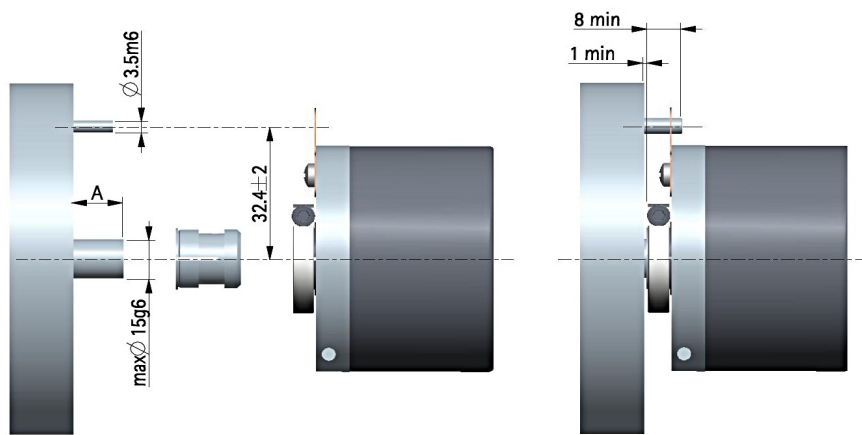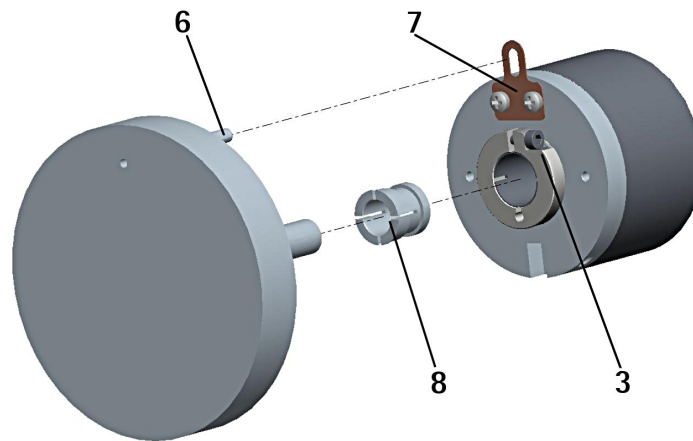
A = min. 8, max. 18 mm

NOTE

You are strongly advised not to carry out any mechanical operations (drilling, milling, etc.) on the encoder shaft. This could cause serious damages to the internal parts and an immediate warranty loss. Please contact our technical personnel for the complete availability of "custom made" shafts.

4 Electrical connection

WARNING
Electrical connection has to be carried out by qualified personnel only, with power supply disconnected and mechanical parts compulsorily in stop.

WARNING
If wires of unused signals come in contact, irreparable damage could be caused to the device. Please insulate them singularly.

4.1 Cable and connector connections

Function	A40 cable	MIL 41-pin
Bit 1 LSB	Brown	A
Bit 2	Red	B
Bit 3	Pink	C
Bit 4	Yellow	D
Bit 5	Green	E
Bit 6	Blue	F
Bit 7	Violet	G
Bit 8	Grey	H
Bit 9	White	J
Bit 10	Black	K
Bit 11	Brown_Red	L
Bit 12	White_Red	M
Bit 13	Blue_Red	N
Bit 14	Pink_Grey	P
Bit 15	White_Yellow	R
Bit 16	Brown_Green	S
Bit 17	White_Green	T
Bit 18	Yellow_Brown	U
Bit 19	White_Blue	V
Bit 20	Brown_Blue	W
Bit 21	White_Pink	X
Bit 22	White_Grey	Y
Bit 23	Pink_Brown	Z
Bit 24	Grey_Brown	a

Bit 25	Brown_Black	b
Parity bit / Bit 28 MSB	White_Black	c
Preset / Offset	Grey_Green	d
Latch	Yellow_Grey	e
TD RS-232	Pink_Green	f
RD RS-232	Yellow_Pink	g
0Vdc RS-232	Green_Blue	h
Tri-state	Yellow_Blue	i
Bit 26	Yellow_Black	n
Bit 27	Pink_Blue	p
Counting direction	Grey_Red	r
+10Vdc +30Vdc	Pink_Red	s
0Vdc	Grey_Blue	t
Shield	Shield	Case

4.2 A40 cable specifications

Model:	LIKA A40 Li-YCY cable
Cross section:	40 x 0.14 mm² (26 AWG)
Jacket:	PVC, flame retardant
Shield:	Tinned copper braid, coverage > 85%
Outer diameter:	10.2 mm ±0.1 mm / 0.40" ±0.0039"
Min. bend radius:	min. 50 mm / 1.97", static application min. 100 mm / 3.94", dynamic application
Work temperature:	-40°C +80°C / -40°F +176°F, static application -5°C +80°C / +23°F +176°F, dynamic application
Conductor resistance:	< 250 Ω/Km

4.3 MIL 41-pin connector specifications
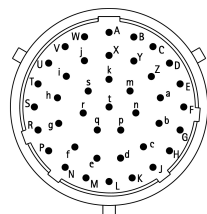
MIL 41-pin connector
Male
Clockwise

4.4 Connection of the shield

For signals transmission always use shielded cables. The cable shielding must be connected properly to the metal ring nut **3** of the connector in order to ensure a good earthing through the frame of the device. To do this 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**.
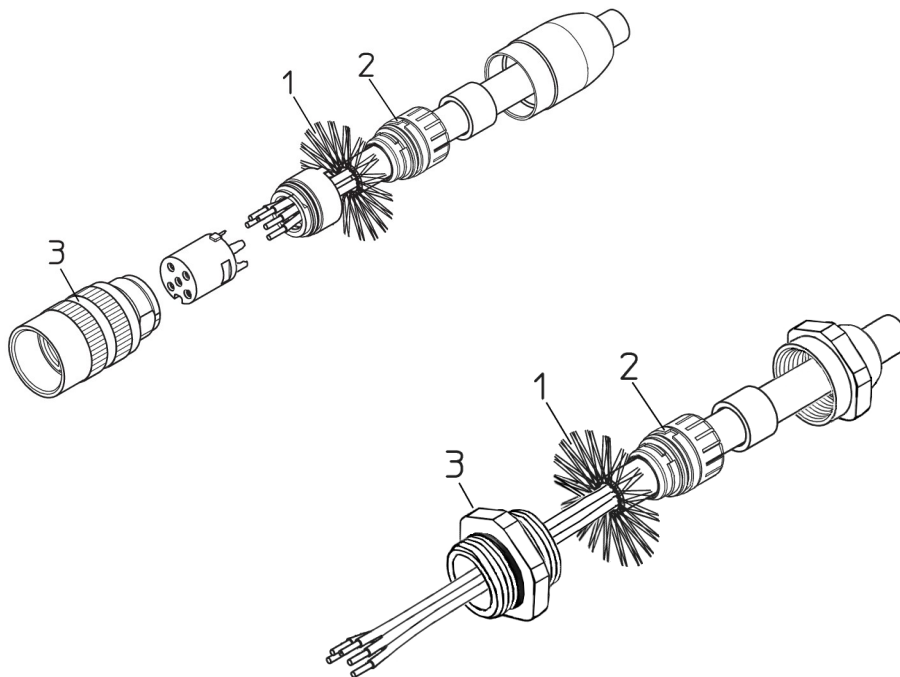
4.5 Ground connection

Minimize noise by connecting the shield and/or the connector housing and/or the frame to ground. Make sure that ground is not affected by noise. The connection point to ground can be situated both on the device side and on user's side. The best solution to minimize the interference must be carried out by the user. You are advised to provide the ground connection as close as possible to the encoder.

4.6 Parity bit

If the parity bit error detection is enabled (see the **Enable parity bit** parameter in the **ENCODER CONFIGURATION** box), the value of this bit will have either the HIGH or the LOW logic level depending on the value of the bits in the word and the type of parity bit you have set next to the **Select parity bit** item in the same box. The parity bit is used for checking in a very basic way whether errors

occur during transmission. The parity bit is the simplest form of error detecting code and offers the advantage that only one bit is required for error detection. On the other hand it only indicates that a parity error occurred in the transmission and there is no way to determine which particular bit is corrupted. Furthermore the parity bit is only guaranteed to detect an odd number of bit errors. If an even number of bits have errors, the parity bit records the correct number of ones, even though the data is corrupt. It can be either even or odd (see the **Select parity bit** parameter). When even parity bit is set (**Select parity bit** = EVEN), the sum of the bits having HIGH logic level (=1) in the character has to be even; therefore if, for instance, the sum of the bits having logic level 1 in the character is 3, parity bit 1 will be added before transmission in order to have an even number of bits, namely 4 bits, having logic level 1; on the contrary if the sum is 4, parity bit 0 will be added as an even number of bits already have logic level 1. The reverse when odd parity bit is set (**Select parity bit** = ODD). In this case the sum of the bits having HIGH logic level (=1) in the character has to be odd.

4.7 Preset / Offset

Preset / Offset input is active only when the **Enable preset / offset** software function in the **ENCODER CONFIGURATION** box is enabled (see on page 45); otherwise the hardware function is disabled. Furthermore it holds two different functions depending on the value of the **Select preset / offset** parameter in the **ENCODER CONFIGURATION** box whether it is set to PRESET or OFFSET. In the first case (**Select preset / offset** = PRESET) the input is used to set the preset (**Preset** parameter in the box PRESET / OFFSET, see on page 32); while in the second case (**Select preset / offset** = OFFSET) it is used to set the offset (**Offset** parameter in the box PRESET / OFFSET, see on page 32). For detailed information on the preset and offset functions refer to the "5.3.2 PRESET / OFFSET box" section on page 32.
The output information can be forced through a command sent to the PRESET input by a PLC or via a button. To set the preset stop the encoder in the desired position and then connect Preset input to +Vdc for 100 μs at least.
After this, the position output by the encoder in that point will be the one set (and then loaded to the encoder) next to the **Preset** parameter in the box PRESET / OFFSET, see on page 32). By default the preset value is 0. Connect Preset input to 0Vdc if not used. To set the preset, connect Preset input to +Vdc for 100 μs at least, then disconnect +Vdc. Normally voltage must be at 0Vdc or floating. We suggest performing the preset setting when the encoder is in stop.

WARNING
Check the **Preset** and **Offset** parameters and activate the functions (either through the input Preset / Offset or by using the **Activate preset** or **Activate**

offset commands) every time you change the value next to the Counts per revolution and/or Number of revolutions parameters when the Programming mode: Scaling option is set; or Counts to be displayed when the Programming mode: Start / Stop option is set; or Counts to be displayed when the Programming mode: Prog option is set.

4.8 Latch

The Latch circuit allows to temporarily store the position information that the encoder is providing to output in a defined moment. The position information will be available until the latch signal is deactivated. This can be very useful when, for instance, the transmission speed of the encoder is faster than the data acquisition of the interface electronics: in this case information is "frozen" until data is fully acquired.
Connect the Latch input to +Vdc to activate the Latch function and temporarily "frozen" the output position information; on the contrary, connect it to 0Vdc to deactivate this function.

WARNING
In the **ENCODER CONFIGURATION** box (see on page 46) the Enable Latch item is available. Please note that this parameter allows to activate / deactivate the Latch software function; the hardware Latch input is independent from the Latch software function and available anyway.

WARNING
When the Latch input is activated, the position information is temporarily "frozen" at output, not in the Position field. So if you move the encoder shaft, the transmitted position information is kept "frozen" at output, while it is updated as usual in the Position field.

4.9 Tri-state

Tri-state circuit allows the transmission of the output position information from the encoder to be interrupted. In this way, for instance, it is possible to have several encoders connected in parallel (multiplexing) but a single control unit and shared circuitry; information about the position of each encoder can be acquired with an updating time equal to the duration of the multiplexing cycle.
Connect the Tri-state input to +Vdc to run the encoder is a standard mode (outputs are normally activated); connect it to 0Vdc to enable the function and thus disable the outputs.

4.10 Counting direction

The counting direction circuit is designed to reverse the counting direction. In other words it allows the count up also when the shaft is rotating counter-clockwise. Connect the Counting direction input to 0Vdc if not used. Connect the counting direction input to 0Vdc to have an increasing count (count up information) when the encoder is turning clockwise (CW, viewed from the shaft side); connect the Counting direction input to +Vdc to have an increasing count (count up information) when the encoder is turning counter-clockwise (CCW).

WARNING
The Counting direction can be set also through the programming tool. The Positive counting direction parameter in the CONFIGURATION page allows the operator to choose the CW (STANDARD) and CCW (INVERTED) options. When the counting direction is set to CW (STANDARD) -Positive counting direction = CW (STANDARD)-, if the Counting direction input has LOW logic level (0Vdc) the encoder will provide the increasing count when the encoder is turning clockwise (and the decreasing count when the encoder is turning counter-clockwise); on the contrary if the Counting direction input has HIGH logic level (+Vdc) the encoder will provide the increasing count when the encoder is turning counter-clockwise (and the decreasing count when the encoder is turning clockwise). When the CCW (INVERTED) option is set -Positive counting direction = CCW (INVERTED)-, if the Counting direction input has LOW logic level (0Vdc) the encoder will provide the increasing count when the encoder is turning counter-clockwise (and the decreasing count when the encoder is turning clockwise); on the contrary if the Counting direction input has HIGH logic level (+Vdc) the encoder will provide the increasing count when the encoder is turning clockwise (and the decreasing count when the encoder is turning counter-clockwise).

WARNING
After having set the new counting direction it is necessary to set also the preset / offset functions.

4.11 RS-232 serial connection

This HM58 PY-PN programmable encoder is equipped with a serial interface for the parametrization and set up of the device. A software tool designed to program the encoder via RS-232 interface is available from **Lika's web site**. Before starting the program, connect the device to the personal computer through a serial port. Should the personal computer not be equipped with a serial port, you must install a USB / RS-232 converter, easily available in the market.

The serial cable must be connected as shown in the table below. See also the "Electrical connection" section on page 17.

Function	A40 cable	MIL 41-pin	Sub-D 9-pin female	Function
TD	Pink_Green	f	2	RD
RD	Yellow_Pink	g	3	TD
0Vdc RS-232	Green_Blue	h	5	0Vdc

Please always make sure that the RD of the ENCODER is cross-wired to the TD of the PC while the TD is cross-wired to the RD.

An optional kit fitted with RS-232/USB adapter for the communication between the encoder and the PC is available.
The order codes are as follows:

- **IF92**   converter

                         +
- **EC-HM58PY-M41**   MIL 41 pin mating cordset for IF92

NOTE
Encoders with A40 cable output require IF92 converter only.

Please note that the configuration parameters of the serial port have fixed values so the user cannot change them.
They are:

RS-232 serial port settings	Value
Baud rate	9600
Byte size	8
Parity	None
Stop bits	1
Flow control	None

4.12 Recommended circuit, parallel output with PNP o. c. drive (PN code)
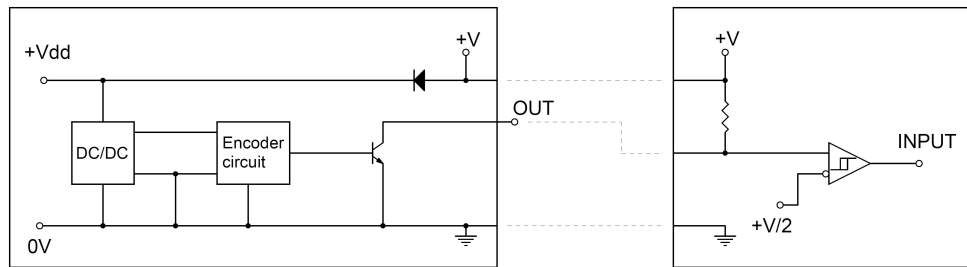
4.13 Recommended circuit, parallel output with Push-Pull drive (PY code)
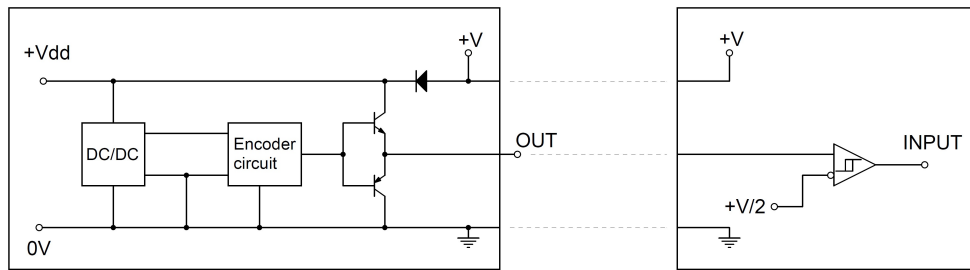
5 Software tool and configuration parameters

5.1 Configuring the device using the software tool

HM58 PY-PN encoder with parallel interface is supplied with a software expressly developed and released by Lika Electronic in order to easily programme and configure the device. The software tool is compatible with the whole series of HM- and EM- programmable encoders from Lika Electronic: programmable encoder with SSI serial output (...-PS2-... code), programmable encoder with parallel output (...-PN2-... and ...-PY2-... codes), programmable encoder with analogue output (...-PA2-... code). It allows the operator to set the working parameters of the device and monitor whether the device is running properly. The program is supplied for free and can be installed in any PC fitted with a Windows operating system (Windows XP or later). Communication is established using **RS-232 serial cable and port**. The executable file **LIKA_PROGRAMMABLE_ENCODER_Vxxxx.EXE** is available from **Lika's web site**. The program is designed to be installed simply by copying the executable file to the desired location and there is **no installation** process. To launch it just double-click the file icon. To close the program press the **Exit** button in the **File** menu of the **CONFIGURATION** page.

WARNING
Please be aware that the following compatibilities between the hardware-software version of the device and the software version of the executable file have to be respected compulsorily.

Compatibility	HW-SW	EXE FILE
	1-1	Up to ...

NOTE
Before starting the program, connect the device to the personal computer through a serial port. The serial interface of the SSI programmable encoder is a RS-232 type connector. Should the personal computer not be equipped with a serial port, you must install a USB / RS-232 converter, easily available in the market.
The serial cable must be connected as shown in the "4.11 RS-232 serial connection" section on page 22.

5.1.1 Available pages and commands

In each page of the Lika Electronic configuration tool software there are the **left navigation bar** and the usual **menu bar** on the top.

In the left navigation bar the following buttons are available:
- **Identification** button: it allows to enter the **IDENTIFICATION** page in order to search and activate the communication with the device connected through the serial port; see on page 26;
- **Configuration** button: it allows to enter the page where the configuration parameters are located; see on page 31;
- **Load / save configuration** button: it enters the page where the current configuration of the encoder can be saved; or a previously saved configuration can be loaded; see on page 59.

5.2 IDENTIFICATION page

When you start the program by double-clicking the **LIKA_PROGRAMMABLE_ENCODER_Vxxxx.EXE** executable file, the **IDENTIFICATION** page is displayed.

First of all this page allows the operator to choose the language used to display texts and items in the user interface. Click the **Italian flag** icon to choose the Italian language; click the **UK flag** icon to choose the English language; select the **Japanese flag** icon to choose the Japanese language.

If you click the www.lika.biz link below in the page, you enter the Lika Electronic web site page where the technical documentation and information on the device as well as the downloadable files for installing the software tool are available.

At this stage the buttons in the left navigation bar as well as the commands in the menu bar are not available; furthermore the fields in the **ENCODER** box show a question mark ("?") as the connected device has not been identified yet.

Before starting the identification process please make sure the unit is connected properly to the personal computer and the power is on. The software will force the settings of the personal computer serial port to the values required by the serial port of the connected device, thus it is not necessary to set the parameters of the pc serial port in the Windows **Device Manager** menu first.

For any information on the characteristics of the serial connection please refer to the "4.11 RS-232 serial connection" section on page 22.

WARNING
Make sure that one device only is connected to the personal computer and switched on when you start a device searching process!

Now you are ready to establish the connection to the encoder: press the **DEVICE SEARCH** button in the **COM PORT** box. During the connection attempt a blue arrow appears next to the **DEVICE SEARCH** button.

If the connection is established properly, a green tick (√) appears next to the **DEVICE SEARCH** button and the fields in the **ENCODER** box are filled with information relating to the connected device. Furthermore the buttons in the left navigation bar and the commands in the menu bar become active.

On the contrary, if the program is not able to establish a connection to the encoder, a red X (X) appears next to the **DEVICE SEARCH** button.

Should this happen, please check the unit is connected properly and the power is on. Make sure that the power supply voltage is correct. Also check that the serial cable characteristics meet the requirements described in the "4.11 RS-232 serial connection" section on page 22.

WARNING
Make sure that one device only is connected to the personal computer and switched on when you start a device searching process!

5.2.1 COM PORT box

In the **COM PORT** box the following functions are available:

Device search
After having connected a device to the serial port of your personal computer and turned it on, press this button to start establishing a communication and allow the program to detect it. During the connection attempt a blue arrow appears next to the button. If the connection is established properly, a green tick

(√) appears next to the **DEVICE SEARCH** button and the fields in the **ENCODER** box are filled with information relating to the connected device. Furthermore the buttons in the left navigation bar and the commands in the menu bar become active.

On the contrary, if the program is not able to establish a connection to the encoder, a red X (**X**) appears next to the **DEVICE SEARCH** button.

Should this happen, please check the unit is connected properly and the power is on. Make sure that the power supply voltage is correct. Also check that the serial cable characteristics meet the requirements described in the "4.11 RS-232 serial connection" section on page 22.

WARNING

Make sure that one device only is connected to the personal computer and switched on when you start a device searching process!

Port

It shows the list of the serial ports available in the personal computer. It is not necessary either to choose the serial port where the unit is connected or to set the parameters of the pc serial port in the Windows **Device Manager** menu first as the program automatically scroll through the available ports and search in each one for a device which is connected and on. Furthermore it forces the settings of the personal computer serial port to the values required by the serial port of the connected device. For any information on the characteristics of the serial connection please refer to the "4.11 RS-232 serial connection" section on page 22.

5.2.2 ENCODER box

In the **ENCODER** box the following functions are available:

Device type

As soon as the connection is established properly and the device is detected, the main characteristics of the unit are shown respecting the structure of the order code. They are: device series (HM or EM series); singleturn resolution - multiturn resolution (expressed in bits; for example: 18-14), output type (PN: programmable encoder with parallel output and NPN driver; PY: programmable encoder with parallel output and Push-Pull driver; PS: programmable encoder with SSI serial output; PA: programmable encoder with analogue output).

Serial number

As soon as the connection is established properly and the device is detected, the univocal serial number of the device is shown next to this item.

Output type

As soon as the connection is established properly and the device is detected, the type of output of the device is shown next to this item: PARALLEL, SSI, or ANALOGUE.

Version

As soon as the connection is established properly and the device is detected, the **software** and **hardware versions** of the device are shown next to this item.

Read parameters

This button allows the operator to force a new read of the device parameters. During the connection attempt a blue arrow appears next to the button. If the program is not able to establish a connection to the encoder, the fields in the **ENCODER** box and listed above show a question mark ("?").

Load default parameters

Default parameters are set at the factory by Lika Electronic engineers to allow the operator to run the device for standard operation in a safe mode. As soon as you press this button the default parameters are uploaded and activated. All parameters which have been set previously are overwritten, thus previously set values are lost. The complete list of machine data and relevant default parameters preset by Lika Electronic engineers is available on page 67.

WARNING

As soon as you press this button, all parameters which have been set previously are overwritten, thus previously set values are lost!

5.3 CONFIGURATION page

When you press the **CONFIGURATION** button in the left navigation page you enter the **CONFIGURATION** page. This page allows the operator to start programming and monitoring the unit. The following window will appear:
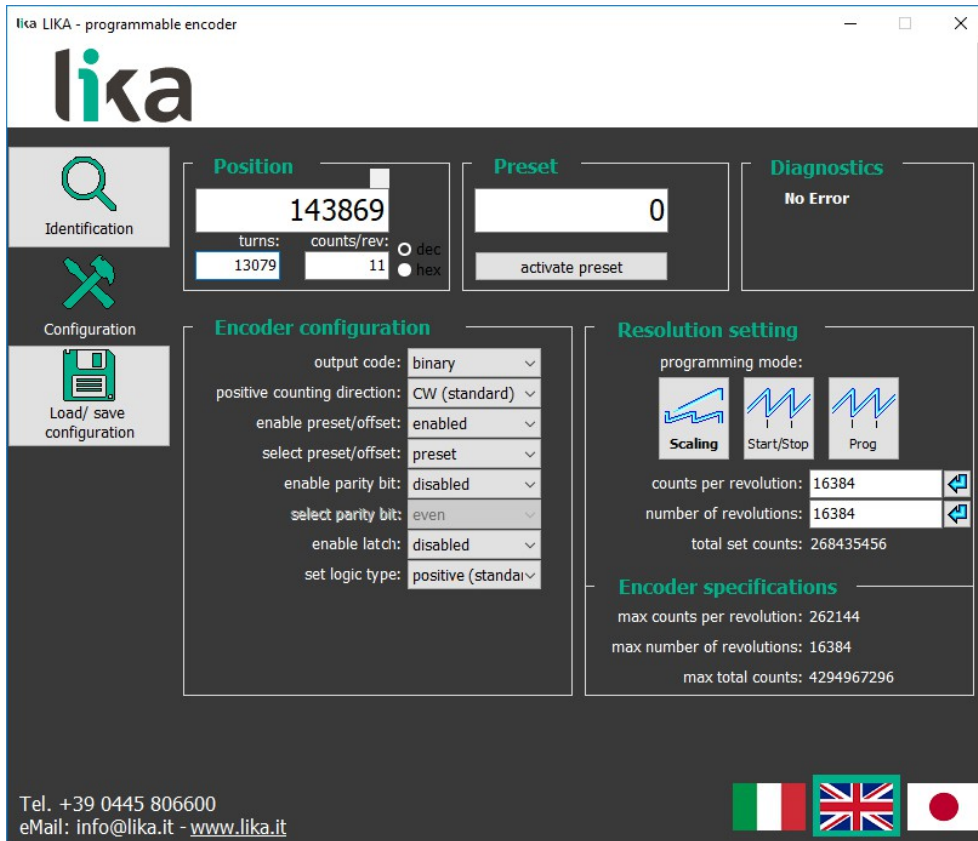
5.3.1 POSITION box

In the **POSITION** box at the top left of the page the following functions are available. The cyclical movement of the blue arrows located at the top right of the **Position** field is intended to show the proper communication with the device.

Position

It shows the current position information of the connected device; if enabled, the set scaling is applied. The value is expressed in counts. The value can be represented in either decimal or hexadecimal format, click the DEC or HEX option buttons in this box to choose the desired display format.

Turns

It shows the number of revolutions already performed by the encoder. As soon as the number of revolutions set next to the **Number of revolutions** parameter (multiturn resolution) is achieved, the counter zero sets and then restarts again the increasing count. The value can be represented in either decimal or hexadecimal format, click the DEC or HEX option buttons in this box to choose the desired display format.

Counts/rev

It shows the current position information in the revolution. As soon as the number of information per revolution (namely, counts per revolution, cpr) set next to the **Counts per revolution** parameter (singleturn resolution) is achieved, the counter restarts again the increasing count. The value is expressed in counts. The value can be represented in either decimal or hexadecimal format, click the DEC or HEX option buttons in this box to choose the desired display format.

5.3.2 PRESET / OFFSET box

The functions in this box are activated and thus available only if the **Enable preset / offset** parameter in the **ENCODER CONFIGURATION** box is set to ENABLED; on the contrary, if DISABLED is set, they are not available. Furthermore they have a different meaning depending on the value of the **Select preset / offset** parameter in the **ENCODER CONFIGURATION** box whether it is set to PRESET or OFFSET. In the first case (**Select preset / offset** = PRESET) the functions in this box are used to set the preset, thus both the box and the button are labelled consequently; while in the second case (**Select preset / offset** = OFFSET) the functions in this box are used to set the offset, thus both the box and the button are labelled consequently.

Preset

This function is available only if the **Enable preset / offset** parameter in the **ENCODER CONFIGURATION** box is set to ENABLED and the **Select preset / offset** parameter in the **ENCODER CONFIGURATION** box is set to PRESET.
This parameter allows the operator to enter the preset value.
The Preset function is meant to assign a value to a desired physical position of the encoder. 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 is useful, for example, 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 (the current position is displayed next to the **Position**

item in the **POSITION** box) in the moment when the preset value is activated. To activate the preset, stop the encoder in the desired position and then press the **Activate preset** button in this box (otherwise connect the Preset / Offset input as explained on page 20).
Default = 0 (min. = 0, max. = 4294967296)

EXAMPLE
Here follows an example useful to better understand the preset function and the meaning of the related parameters and commands: **Preset** and **Activate preset**.

To know the position information which will be transmitted by the encoder, first of all we must calculate the so-called **stored preset value**.
The stored preset value is equal to the encoder position read in the moment when the **Activate preset** command is sent minus the value set next to the **Preset** parameter.
For instance, if you set the value "=50" next to the **Preset** parameter and the position of the encoder when the **Activate preset** command is sent is "=1,000", then the stored preset value will be: 1,000 – 50 = 950.

Thus the encoder position information to be transmitted will result from the following calculation:

if current encoder position ≥ stored preset value, then:
Transmitted position information = current encoder position – stored preset value.

Otherwise, if current encoder position < stored preset value, then:
Transmitted position information = current encoder position - stored preset value + overall set resolution (**Total set counts** in **Programming mode: Scaling**).

If a **Preset** value has been never set previously and anyway the **Activate preset** command has been never sent, the transmitted position information and the current position information necessarily match as **Preset** = 0 and stored preset value = 0.
When you enter a **Preset** value and then execute the **Activate preset** command, the system saves the so-called stored preset value. It follows that the transmitted position information and the **Preset** value match as the current encoder position – stored presed value = **Preset**; in other words, for the desired encoder position the value set next to the **Preset** item is output and displayed.
For instance, let's suppose that we set the value "50" next to the **Preset** item and then we execute the **Activate preset** command after having stopped the encoder at position "1,000" (**Position** = 1,000; after having sent the command by pressing the **Activate preset** button, **Position** = 50; **Turns** = 0; **Counts/rev** =

50). So we need the encoder to output the value "50" when it reaches the real position "1,000". The stored preset value will be "950".

Thus it follows that:

Transmitted position information = **current encoder position** (="1,000") - stored preset value (="950") = **50**.

The following transmitted position information will be:

Transmitted position information = **current encoder position** (="1,001") - stored preset value (="950") = **51**.

...

Transmitted position information = **current encoder position** (="998") - stored preset value (="950") = **48**.

Transmitted position information = **current encoder position** (="999") - stored preset value (="950") = **49**.

And so on.

Please pay attention to the following condition! As previously stated, if current encoder position < stored preset value, then:

Transmitted position information = **current encoder position** - **stored preset value** + **overall set resolution**.

For instance, if the overall set resolution is 2,048 (**Total set counts** in **Programming mode: Scaling** = 2,048), when the current encoder position is lower than "950", it will be as follows:

Transmitted position information = **current encoder position** (="948") - stored preset value (="950") + overall set resolution (2,048) = **2,046**.

Transmitted position information = **current encoder position** (="949") - stored preset value (="950") + overall set resolution (2,048) = **2,047**.

Transmitted position information = **current encoder position** (="950") - stored preset value (="950") = **0**.

And so on.

NOTE

- When **Programming mode: Scaling** is set, then **Preset** must be less than or equal to **Total set counts** (**Counts per revolution** ∗ **Number of revolutions**).
- When **Programming mode: Start / Stop** is set, then **Preset** must be less than or equal to **Counts to be displayed**).
- When **Programming mode: Prog** is set, then **Preset** must be less than or equal to **Counts to be displayed**).

If you set a greater value the **Value not valid, preset greater than the maximum position** error message is displayed.

WARNING
Check the **Preset** parameter value and then execute the **Activate preset** command every time you enter new values next to the following parameters: **Counts per revolution** and/or **Number of revolutions** in **Programming mode: Scaling**; **Counts to be displayed** in **Programming mode: Start / Stop**; and **Counts to be displayed** in **Programming mode: Prog**.

Activate preset
This function is active only when the **Enable preset / offset** parameter in the **ENCODER CONFIGURATION** box is set to ENABLED and the **Select preset / offset** parameter in the **ENCODER CONFIGURATION** box is set to PRESET.
Press this button to activate the preset function. The preset value (it is set next to the **Preset** item above) will be set for the position of the encoder in the moment when the preset value is activated. To activate the preset, stop the encoder in the desired position and then press this **Activate preset** button. The chosen physical position will get the value set next to the **Preset** item and all the previous and following positions will get a value according to it. For any further information on the preset function and the meaning of the related parameters and commands **Preset** and **Activate preset** refer to page 32.

WARNING
The Preset / Offset hardware input works in parallel with the **Activate preset** command. For any information on using the Preset / Offset input refer to the "4.7 Preset / Offset" section on page 20.

Offset
This function is available only if the **Enable preset / offset** parameter in the **ENCODER CONFIGURATION** box is set to ENABLED and the **Select preset / offset** parameter in the **ENCODER CONFIGURATION** box is set to OFFSET.
This parameter allows the operator to enter the offset value.
The offset function is meant to assign a value to a desired physical position of the encoder so that the output position information will be shifted according to the value set next to this **Offset** item. The number of transmitted values will match the set resolution (**Total set counts**), but the output information will range between the **Offset** value (minimum value) and the sum of the set resolution + the **Offset** value (maximum value). The offset value will be set for the position of the encoder (the current position is displayed next to the **Position** item in the **POSITION** box) in the moment when the offset value is activated. To activate the offset, stop the encoder in the desired position and

then press the **Activate offset** button in this box (otherwise connect the Preset / Offset input as explained on page 20).

EXAMPLE
Here follows an example useful to better understand the offset function and the meaning of the related parameters and commands: **Offset** and **Activate offset**.

To know the position information which will be transmitted by the encoder, first of all we must calculate the so-called **stored preset value** (see the explanation for the **Preset** parameter on page 32); the **Preset** value is always "=0" in the moment when we press the **Activate offset** command.
The stored preset value is equal to the encoder position read in the moment when the **Activate offset** command is sent minus the preset value; as stated, when we activate the offset function, the preset value is always 0.
For instance, if the position of the encoder when the **Activate offset** command is sent is "=0", then the stored preset value will be: 0 – 0 = 0.

Thus the encoder position information to be transmitted will result from the following calculation:

Transmitted position information = current encoder position – stored preset value + Offset value.

Here is a positive example concerning the so-called "Shifted Gray code" or "Gray Excess code"; we are going to deal with it in the following "Shifted Gray code or Gray Excess code" section; let's suppose that the overall set resolution is "=360" (**Total set counts** in **Programming mode: Scaling** = 360).

Thus in the example it follows that:
Transmitted position information = current encoder position (="0") - **stored preset value** (="0") + **Offset** value (="76") = **76**.
The following transmitted position information will be:
Transmitted position information = current encoder position (="1") - **stored preset value** (="0") + **Offset** value (="76") = **77**.
...
Transmitted position information = current encoder position (="358") - **stored preset value** (="0") + **Offset** value (="76") = **434**.
Transmitted position information = current encoder position (="359") - **stored preset value** (="0") + **Offset** value (="76") = **435**.
And so on.

Shifted Gray code or Gray Excess code

The offset function is very useful when we need to set the values for the so-called "Shifted Gray code" or "Gray Excess code".

As Binary code, the Gray code (for any information refer to the Output code parameter on page 42) represents the information through the base-2 number system; but the binary strings are ordered so that two successive values differ in only one bit; and this property is preserved even at transition from the last to the first position of the sequence, provided that the number of information (namely, the resolution) is a power of 2.

When the number of information (resolution) is not a power of 2, the main feature of the Gray code (i.e. successive values differ in exactly one bit) is lost at transition from the last position of the sequence back to the first position. This could lead to the same measuring errors affecting the Binary code. Let's suppose, for instance, that an encoder has a resolution of 360 information per turn. The sequence of the Gray code is shown in the following Figures (HIGH logic level is represented in black).

Decimal	...	356	357	358	359	0	1	2	3	...
Gray	...	111010110	111010111	111010101	111010100	000000000	000000001	000000011	000000010	...

As you can easily see in the Figures, more than one bit changes from position 359 to position 0 and this disagrees with the main rule of the Gray code. To compensate for this "hitch", the so-called "Shifted Gray" or "Gray Excess" code has to be used. This is a method intended to calculate a new sequence of Gray code values so that the Gray code rule is always respected.

The Δ value (first output information in the "Shifted Gray" code instead of using 0) results from the following algorithm:

$$\Delta = \frac{\text{Integral Gray} - \text{Shifted Gray}}{2}$$

where:

- Integral Gray: it represents the number of counts expressed in power of 2 immediately higher than the resolution involved (in the example: $512 = 2^9$);
- Shifted Gray: it represents the resolution involved (even numbers only; 360 in the example).

So:

$$\Delta = \frac{512 - 360}{2} = 76$$

It follows that the first output information (instead of 0) will be 76, while the last position will be 360 + (Δ - 1) = 435. In this way shifting from 435 back to 76 does not cause the main feature of the Gray code to be lost, as shown in the following Figure.

Bit 1 (LSB)		▓			▓	
Bit 2		▓	▓	▓	▓	
Bit 3						
Bit 4		▓	▓	▓		
Bit 5						
Bit 6		▓	▓	▓	▓	
Bit 7		▓	▓	▓	▓	
Bit 8						
Bit 9 (MSB)		▓	▓			
Decimal	...	434	435	76	77	...
Position	...	358	359	0	1	...

First position
76

Last position
435

Total counts = 360

360°
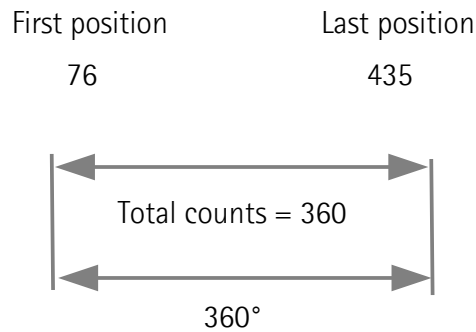
Please note that:

- the initial count value is not zero (in the example the first output value is 76);
- the final count value does not coincide with the max. value of the information requested (in the example, the last value is 435, while the number of requested information is 360).

See also the **EXAMPLE** in the explanation concerning the Offset parameter on page 35.

NOTE
The following relation must be always considered to know the maximum value which is allowed in this parameter:
Offset value ≤ Max total counts (4,294,967,296) – set overall resolution (Total set counts or Counts to be displayed).
If you set a greater value the **Value not valid, offset greater than the maximum value allowed** error message is displayed.

WARNING
Check the Offset parameter value and then execute the Activate offset command every time you enter new values next to the following parameters: Counts per revolution and/or Number of revolutions in Programming mode: Scaling; Counts to be displayed in Programming mode: Start / Stop; and Counts to be displayed in Programming mode: Prog.

Default = 0 (min. = 0, max. = ≤ Max total counts (4,294,967,296) – set overall resolution (Total set counts or Counts to be displayed).)

Activate offset

This function is active only when the Enable preset / offset parameter in the **ENCODER CONFIGURATION** box is set to ENABLED and the Select preset / offset parameter in the **ENCODER CONFIGURATION** box is set to OFFSET.

Press this button to activate the offset function. The offset value (it is set next to the Offset item above) will be used to calculate the information to be output for the position of the encoder in the moment when the offset value is activated. To activate the offset, stop the encoder in the desired position and then press this Activate offset button. The chosen physical position will get the value according to the Offset item and all the previous and following positions will get a value according to it. For any further information on the offset function and the meaning of the related parameters and commands Offset and Activate offset refer to page 35.

WARNING

The Preset / Offset hardware input works in parallel with the Activate offset command. For any information on using the Preset / Offset input refer to the "4.7 Preset / Offset" section on page 20.

5.3.3 DIAGNOSTICS box

This box is designed to show the error messages which could be triggered while programming the encoder.
Here are the available error messages.

Encoder travel not valid: counts to be displayed greater than physical resolution between start and stop

When programming the encoder using the START / STOP mode (see Programming mode: Start / Stop on page 52), if the travel exceeds the maximum encoder resolution as set by the operator, then this error message is displayed.

Encoder travel not valid: unable to reach the minimum stroke

When programming the encoder using the START / STOP mode (see Programming mode: Start / Stop on page 52), if the start position (**1 – Start teaching**) and final position (**2 – Stop teaching**) match (i.e., you do not change the axis position during programming process and press both the buttons when the axis is in the same exact position), then this error message is displayed.

No error

There are not active errors.

5.3.4 ENCODER CONFIGURATION box

Output code

The absolute encoder provide the absolute position information in a code format. Thus the output information can be represented in **BINARY**, **GRAY**, or **BCD code**. Select the required code format in the drop-down box.

BINARY CODE

In the base-2 number system only two digits are used: 0 and 1. For this reason it is the base of data processing in computing and electronic systems where 0 and 1 can be represented, electronically, as states "off" and "on". Binary code represents the information through the base-2 number system. It is very efficient, but has a great disadvantage from the point of view of the measuring equipment: more than one digit at a time often changes in consecutive positions. Because of the variations caused by gate delays, line impedances, etc. transitions do not occur simultaneously and this could result in erroneous readings and therefore lead to significant measuring errors. Consider, for instance, the following sequence:

Decimal	0	1	2	3	4	5	6	7
Binary (4 bits)	0000	0001	0010	0011	0100	0101	0110	0111
Gray (4 bits)	0000	0001	0011	0010	0110	0111	0101	0100

From position 1 to 2 (decimal), the last two digits both change in the binary representation ($0 \rightarrow 1$; $1 \rightarrow 0$). The same happens from position 3 to 4, where the last three digits all change ($0 \rightarrow 1$; $1 \rightarrow 0$; $1 \rightarrow 0$). During operation, a sensing error of the detection elements could result in a positioning command error. For instance, let's suppose the following wrong sequence: 0001 $\rightarrow$ 0011 $\rightarrow$ 0010 (position $1 \rightarrow 3 \rightarrow 2$ instead of $1 \rightarrow 2 \rightarrow 3$). Motion controller would command position 1, then position 3 and then a reverse motion to position 2.

NOTE

Please consider that Lika encoders are designed to ensure that only consecutive position values are always transmitted.

Binary code 4-bit representation (high logic level is represented in black)

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Bit 1 (LSB)		■		■		■		■		■		■		■		■
Bit 2			■	■			■	■			■	■			■	■
Bit 3					■	■	■	■					■	■	■	■
Bit 4 (MSB)									■	■	■	■	■	■	■	■
Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

GRAY CODE

As binary code, also Gray code, patented by Bell Laboratories researcher Frank Gray in 1947, represents the information through the base-2 number system; but the binary strings are ordered so that two successive values differ in only one bit (see the table here below); and this property is preserved even at transition from the last to the first position of the sequence, provided that the number of information (namely, the resolution) is a power of 2 (otherwise, we must exploit the property of the so-called "Shifted Gray" code or "Gray Excess" code, see the "Shifted Gray code or Gray Excess code" section on page 37 where the Offset function is explained).This is a safer method than binary code, because any sequence where more than one bit at a time changes can be easily and immediately detected by the control unit. Anyway, the maximum error is always the value of the LSB.

It was originally called "reflected binary code" as it "may be built up from the conventional binary code by a sort of reflection process". The Gray sequence in base 2 (G_2) in fact is constructed starting from the operation of reflecting the strings in base 1 (G_1); and the Gray sequence in base 3 (G_3) is constructed starting from the operation of reflecting the strings in base 2 (G_2); and so on.

See the Figure. To construct the strings in base 2 (G_2), you must reflect the two values in base 1 (G_1), namely 0 and 1. Then prefix 0 to the normal G_1 strings (above the line) and prefix 1 to the reflected G_1 strings (below the line). To have the strings in base 3 (G_3), you must reflect the values in base 2 (G_2), namely 00, 01, 11 and 10. Then prefix 0 to the normal G_2 strings (above the line) and prefix 1 to the reflected G_2 strings (below the line). And so on, in a recursive and iterative way.

Gray code 4-bit representation (high logic level is represented in black)

Bit 1 (LSB)		■	■			■	■			■	■			■	■	
Bit 2			■	■	■	■					■	■	■	■		
Bit 3					■	■	■	■					■	■	■	■
Bit 4 (MSB)									■	■	■	■	■	■	■	■
Decimal	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15

BCD CODE

BCD code (the acronym for binary-coded decimal) represents the decimal numbers using for each digit the corresponding 4-bit binary value. This means that only binary numbers from 0 (0000) to 9 (1001) are used. For instance: decimal number 123_{10} is represented as: 0001 0010 0011 (1111011 in binary code). The main advantage of the BCD code is its straight relation with the decimal code which allows to easily convert the binary numbers into decimal numbers and vice versa. On the other hand it requires more memory. BCD code is more and more rarely used, in particular in outdated electronics and spare parts now.

Default = binary

WARNING

Make sure that the Master and the Slave are configured in the same way.

Positive counting direction

It allows to set whether the position information output by the encoder increases when the shaft rotates clockwise or counter-clockwise. Clockwise and counter-clockwise rotations are viewed from the shaft. It is possible to choose the following options: CW (STANDARD) and CCW (INVERTED). When the counting direction is set to CW (STANDARD) -**Positive counting direction** = CW (STANDARD)-, if the Counting direction input (see on page 22) has LOW logic level (0Vdc) the encoder will provide the increasing count when the encoder is turning clockwise (and the decreasing count when the encoder is turning counter-clockwise); on the contrary if the Counting direction input has HIGH logic level (+Vdc) the encoder will provide the increasing count when the encoder is turning counter-clockwise (and the decreasing count when the encoder is turning clockwise). When the option CCW (INVERTED) is set -**Positive counting direction** = CCW (INVERTED)-, if the Counting direction input has LOW logic level (0Vdc) the encoder will provide the increasing count when the encoder is turning counter-clockwise (and the decreasing count when the encoder is turning clockwise); on the contrary if the Counting direction input has HIGH logic level (+Vdc) the encoder will provide the increasing count when the encoder is turning clockwise (and the decreasing count when the encoder is turning counter-clockwise). For any information on the Counting direction input refer to the "Electrical connection" section on page 17 and in particular to the "4.10 Counting direction" section on page 22.
Default = CW (standard)

WARNING
After having set the new counting direction it is necessary to set also the preset / offset functions.

Enable preset / offset

It enables / disables the preset / offset functions. When this parameter is set to ENABLED, the functions available in the **PRESET / OFFSET** box are active and can be set; otherwise they are disabled. For any information on the preset and offset functions refer to the "5.3.2 PRESET / OFFSET box" section on page 32.
Default = enabled

Select preset / offset

This item is available only if the **Enable preset / offset** parameter is set to ENABLED. It allows to activate either the preset function (**Select preset / offset** = PRESET) or the offset function (**Select preset / offset** = OFFSET) by using the items available in the **PRESET / OFFSET** box. After having enabled the preset / offset functions (**Enable preset / offset** = ENABLED), this item allows to activate either the preset function or the offset function. The parameters in the **PRESET / OFFSET** box will have a different meaning depending on the value of this parameter whether it is set to PRESET or OFFSET. In the first case (**Select preset / offset** = PRESET) the functions in the box are used to set the preset, thus both the box and the button are labelled consequently; while in the second case (**Select preset / offset** = OFFSET) the functions in the box are used to set the offset, thus both the box and the button are labelled consequently. For any information on the preset and the offset functions refer to the "5.3.2 PRESET / OFFSET box" section on page 32.
Default = preset

Enable parity bit

It enables / disables the use of the parity bit for checking in a very basic way whether errors occur during transmission. The parity bit is the simplest form of error detecting code and offers the advantage that only one bit is required for error detection. On the other hand it only indicates that a parity error occurred in the transmission and there is no way to determine which particular bit is corrupted. Furthermore the parity bit is only guaranteed to detect an odd number of bit errors. If an even number of bits have errors, the parity bit records the correct number of ones, even though the data is corrupt. It can be either even or odd (see the following **Select parity bit** parameter). When even parity bit is set (**Select parity bit** = EVEN), the sum of the bits having HIGH logic level (=1) in the character has to be even; therefore if, for instance, the sum of the

bits having logic level 1 in the character is 3, parity bit 1 will be added before transmission in order to have an even number of bits, namely 4 bits, having logic level 1; on the contrary if the sum is 4, parity bit 0 will be added as an even number of bits already have logic level 1. The reverse when odd parity bit is set (**Select parity bit** = ODD). In this case the sum of the bits having HIGH logic level (=1) in the character has to be odd. See also the following **Select parity bit** parameter.

For further information on the parity bit refer to the "4.6 Parity bit" section on page 19.

Default = disabled

Select parity bit

This parameter is available only if **Enable parity bit** = ENABLED. It sets whether the parity bit has to be even or odd. For complete information on the parity bit refer to the explanation of the previous parameter **Enable parity bit**.

For further information on the parity bit refer to the "4.6 Parity bit" section on page 19.

Default = even

WARNING

Make sure that the Master and the Slave are configured in the same way.

Enable Latch

WARNING

Please note that this **Enable Latch** parameter allows to activate / deactivate the Latch <u>software</u> function; the <u>hardware</u> Latch input (see the "4.8 Latch" section on page 21) is independent from the Latch software function and available anyway.

This parameter allows the operator to enable / disable the <u>software</u> Latch function. Choose the desired option (ENABLED / DISABLED) in the drop-down box.

The Latch function allows to temporarily store the information that the encoder is providing to output in a defined moment. Information will be available until the **Enable Latch** parameter is deactivated. This can be very useful when, for instance, the transmission speed of the encoder is faster than the data acquisition of the interface electronics: in this case information is "frozen" until data is fully acquired.

To activate the Latch function and temporarily "frozen" the output position information set the **Enable Latch** parameter to ENABLE; on the contrary, to

deactivate this function and make the position information normally available set the **Enable Latch** parameter to DISABLE.

For further information on the Latch signal refer also to the "Electrical connection" section on page 17 and in particular to the "4.8 Latch" section on page 21.

Default = disabled

WARNING

When the Latch input is activated (**Enable Latch** = ENABLE), the position information is temporarily "frozen" at output, not in the **Position** field. So if you move the encoder shaft, the transmitted position information is kept "frozen" at output, while it is updated as usual in the **Position** field.

Set logic type

It allows the operator to choose whether the logic type is "positive" -POSITIVE (STANDARD) option- or "negative" -NEGATIVE (INVERTED) option-.

As previously stated (see the **Output code** parameter on page 42), the available output codes (Binary, Gray, or BCD code) represent the information through the base-2 number system; thus the binary information, namely the bits, can acquire only two values. In the base-2 number system in fact only two digits are used: 0 and 1. For this reason it is the base of data processing in computing and electronic systems where 0 and 1 can be represented, electronically, as states "off" and "on" respectively.

It is a convention that 0 is represented by the "off" status, i.e. the logical LOW level (the lower of two voltage levels) and, depending on the case, it has from time to time a different meaning: zero, deactivated, false, etc.; and, on the contrary, 1 is represented by the "on" status, i.e. the logical HIGH level (the higher of two voltage levels) and, depending on the case, it has from time to time a different meaning: one, activated, true, etc. This is the so-called "positive logic". According to the "positive logic", for instance, a signal is ACTIVE – HIGH.

On the other hand, in the "negative logic" the meaning of the values is reversed. Thus the "off" status, i.e. the logical LOW level (the lower of two voltage levels) indicates from time to time one, activated, true, etc.; and, on the contrary, the "on" status, i.e. the logical HIGH level (the higher of two voltage levels) indicates from time to time zero, deactivated, false, etc. In this second case we can state that in the "negative logic" a signal is ACTIVE – LOW.

Choose the desired option -POSITIVE (STANDARD) or NEGATIVE (INVERTED)- in the drop-down box.

Default = positive (standard)

5.3.5 RESOLUTION SETTING box

The three graphic buttons available in the **RESOLUTION SETTING** box allows to programme the encoder work mode and to set its resolution in the way that best suits the requirements of the specific application. The functions available in the box are different for each programming mode.
The available programming modes are:

- **Programming mode: Scaling**: it allows to set the number of information per revolution and the number of revolutions needed by the specific application, see the following page.
- **Programming mode: Start / Stop**: TEACH IN, it allows to programme the encoder by considering directly the physical travel of the application you need to measure and control, see on page 52.
- **Programming mode: Prog**: it allows to programme the encoder when you exactly know the number of encoder information (counts) for the physical travel of the application you need to measure and control, see on page 55.

Programming mode: Scaling
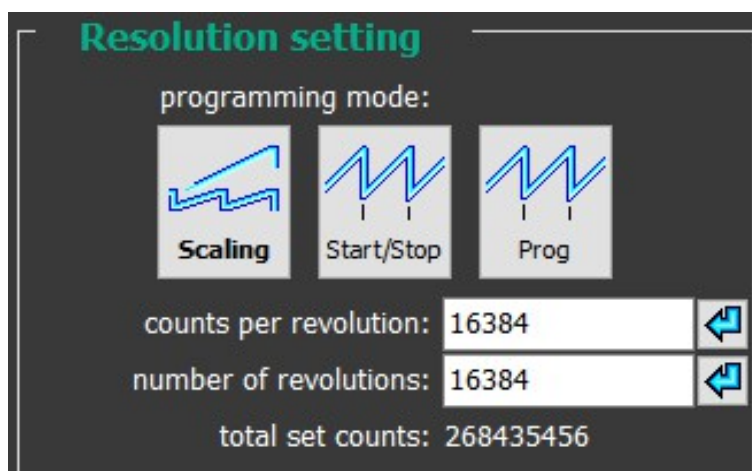
Press this button to enter the SCALING programming mode and activate the relevant functions.
The SCALING programming mode allows to set the number of information per revolution (**Counts per revolution** parameter) and the number of revolutions (**Number of revolutions** parameter) according to the requirements of the specific application. Thanks to this characteristic the HM58 PY-PN programmable encoder can be interchangeable with the whole series of parallel encoders manufactured by Lika Electronic.

Counts per revolution

This parameter is available only when the **Programming mode: Scaling** is enabled.

This parameter sets the number of information per revolution needed by the application (singleturn resolution). Enter the needed value and then press the confirmation button on the right.

You are allowed to set whatever integer value less than or equal to the **number of physical information per revolution** or **hardware counts** (**Max counts per revolution** in the **ENCODER SPECIFICATIONS** box) as the encoder is able to manage dynamically the resolutions which are not a power of 2 (see the "5.6 "Red zone" compensation" section on page 63). If you set a value greater than the number of physical information per revolution, then the **Value higher than the hardware resolution, please set a lower value** error message is displayed. For any information on the number of physical information per revolution and the number of physical revolutions refer to the encoder specifications listed in the **ENCODER SPECIFICATIONS** box below.

Default = 262144 (min. = 1, max. = 262144)

NOTE

Please note that in this encoder the maximum number of physical information per revolution is 262,144 (2^{18}), while the maximum number of physical revolutions is 16,384 (2^{14}), see the **ENCODER SPECIFICATIONS** box on page 56. However the overall resolution (total counts) you are allowed to set is 268,435,456 (2^{28}). It follows that you must arrange the counts per revolution and the number of revolutions so that the overall resolution (**Total set counts**) is not greater than 268,435,456 information.

For instance: if we set the maximum singleturn resolution, i.e. 262,144 cpr (2^{18}), then we can set max. 1,024 revolutions (2^{10}) as $2^{18} + 2^{10} = 2^{28}$. On the other hand, if we set the maximum number of revolutions, i.e. 16,384 revolutions (2^{14}), then we can set max. 16,384 cpr (max. singleturn resolution = 2^{14}) as $2^{14} + 2^{14} = 2^{28}$. And so on, in a whole variety of combinations.

If you enter a wrong resolution value, no matter whether singleturn resolution or number of revolution, so that the overall number of information is greater than 2^{28}, then the **Attention: the maximum transmitted position requires a number of bits greater than the available one** error message is displayed.

NOTE

If the preset / offset functions are active, after having set a new value next to this **Counts per revolution** item, then you must check the value in the **Preset** / **Offset** parameter and activate it by sending the relevant **Activate preset** / **Activate offset** command).

Number of revolutions

This parameter is available only when the **Programming mode: Scaling** is enabled.

This parameter sets the number of revolutions needed by the application (multiturn resolution). Enter the needed value and then press the confirmation button on the right.

You are allowed to set whatever integer value less than or equal to the **number of physical revolutions** (**Max number of revolutions** in the **ENCODER SPECIFICATIONS** box) as the encoder is able to manage dynamically the resolutions which are not a power of 2 (see the "5.6 "Red zone" compensation" section on page 63). If you set a value greater than the number of physical revolutions, then the **Value higher than the hardware resolution, please set a lower value** error message is displayed.

For any information on the number of physical information per revolution and the number of physical revolutions refer to the encoder specifications listed in the **ENCODER SPECIFICATIONS** box below.

Default = 16384 (min. = 1, max. = 16384)

NOTE

Please note that in this encoder the maximum number of physical information per revolution is 262,144 (2^{18}), while the maximum number of physical revolutions is 16,384 (2^{14}), see the **ENCODER SPECIFICATIONS** box on page 56. However the overall resolution (total counts) you are allowed to set is 268,435,456 (2^{28}). It follows that you must arrange the counts per revolution and the number of revolutions so that the overall resolution (**Total set counts**) is not greater than 268,435,456 information.

For instance: if we set the maximum singleturn resolution, i.e. 262,144 cpr (2^{18}), then we can set max. 1,024 revolutions (2^{10}) as $2^{18} + 2^{10} = 2^{28}$. On the other hand, if we set the maximum number of revolutions, i.e. 16,384 revolutions (2^{14}), then we can set max. 16,384 cpr (max. singleturn resolution = 2^{14}) as $2^{14} + 2^{14} = 2^{28}$. And so on, in a whole variety of combinations.

If you enter a wrong resolution value, no matter whether singleturn resolution or number of revolution, so that the overall number of information is greater than 2^{28}, then the **Attention: the maximum transmitted position requires a number of bits greater than the available one** error message is displayed.

NOTE

If the preset / offset functions are active, after having set a new value next to this **Number of revolutions** item, then you must check the value in the **Preset** / **Offset** parameter and activate it by sending the relevant **Activate preset** / **Activate offset** command).

EXAMPLE

HM58 PY **18-14** multiturn encoder.

Physical resolution (refer to the specifications listed in the **ENCODER SPECIFICATIONS** box):

- **Max counts per revolution**   = 262,144 info per turn = 18 bits (2^{18})
- **Max number of revolutions**   = 16,384 revolutions = 14 bits (2^{14})
- **Max total counts**   = 4,294,967,296 = 32 bits (2^{32})

You need to set a **11-bit singleturn resolution** (**Counts per revolution = 2,048**); and the **Number of revolutions = 1,024** (10 bits), **Total set counts** $\leq 2^{28}$:

- Activate **Programming mode: Scaling**
- Set the number of information per revolution: **Counts per revolution = 2,048** (2^{11})
- Set the number of revolutions: **Number of revolutions = 1,024** (2^{10})

In the **Total set counts** field below the overall programmed resolution will be displayed: 2,097,152 (2,048 * 1,024).

Total set counts

This parameter is available only when the **Programming mode: Scaling** is enabled.

It shows the overall programmed resolution which results from the number of information per revolution **Counts per revolution** multiplied by the number of revolutions **Number of revolutions**, as set in the two previous parameters.

NOTE

Please note that in this encoder the maximum number of physical information per revolution is 262,144 (2^{18}), while the maximum number of physical revolutions is 16,384 (2^{14}), see the **ENCODER SPECIFICATIONS** box on page 56. However the overall resolution (total counts) you are allowed to set is 268,435,456 (2^{28}). It follows that you must arrange the counts per revolution and the number of revolutions so that the overall resolution (**Total set counts**) is not greater than 268,435,456 information.

For instance: if we set the maximum singleturn resolution, i.e. 262,144 cpr (2^{18}), then we can set max. 1,024 revolutions (2^{10}) as $2^{18} + 2^{10} = 2^{28}$. On the other hand, if we set the maximum number of revolutions, i.e. 16,384 revolutions (2^{14}), then we can set max. 16,384 cpr (max. singleturn resolution = 2^{14}) as $2^{14} + 2^{14} = 2^{28}$. And so on, in a whole variety of combinations.

If you enter a wrong resolution value, no matter whether singleturn resolution or number of revolution, so that the overall number of information is greater

---

than 2^{28}, then the **Attention: the maximum transmitted position requires a number of bits greater than the available one** error message is displayed.

Programming mode: Start / Stop

Press this button to enter the START / STOP programming mode and activate the relevant functions.

The START / STOP programming mode (TEACH IN) allows to programme the encoder by considering directly the physical travel of the application you need to measure and control; the set output range (the overall information to be output which is set next to the **Counts to be displayed** item) is defined over the travel of the specific application and is comprised between the ends of the axis: the origin of the axis, i.e. the point where you press the **1 – Start teaching** button, and the end of the axis, i.e. the point where you press the **2 – Stop teaching** button; this is then repeated cyclically.

To programme the encoder using the START / STOP programming mode proceed as follows:

- set the number of information you need for measuring and controlling your axis travel next to the **Counts to be displayed** item;
- move to the origin of the axis travel you need to measure; in other words, move the axis to the point where it starts the travel, i.e. to the 0 point;
- now press the **1 – Start teaching** button; at this stage the display fields in the **POSITION** box are "frozen" and not available;
- move to the end of the axis travel you need to measure; in other words, move the axis to the final point where it stops the travel;
- now press the **2 – Stop teaching** button.

EXAMPLE 1

Here follows a first example useful to better understand how to programme the encoder using the START / STOP programming mode.

Let's suppose you need to measure and control the <u>linear travel</u> of an axis; the length of the travel is known, it is 3 metres. We need the information to be displayed with a resolution of thousandths of a metre (i.e. millimetres resolution).

So, let's proceed as follows:

- set the number of information you need to display next to the **Counts to be displayed** item; as the travel is 3 metres long and we need a thousandths resolution (millimetres), then set **Counts to be displayed** = 3,000;
- now move the axis to the 0 point; in other words, move it to the point where the travel starts; in this point the displayed position value will be "0";
- press the **1 – Start teaching** button;
- move the axis to the end of the travel; in other words, move it to the point where the travel stops; in this point the displayed position value will be "2,999";
- press the **2 – Stop teaching** button to close the programming procedure.

EXAMPLE 2

Here follows a second example useful to better understand how to programme the encoder using the START / STOP programming mode.

Let's suppose you need to measure and control the <u>angular movement</u> of a device; the magnitude of the displacement is known, it is 90°, but we need the position to be displayed with an angular resolution of one arcsecond.

So, let's proceed as follows:

- set the number of information you need to display next to the **Counts to be displayed** item; as the magnitude of the movement is 90° and we need the information to be displayed in arcseconds, we must set: **Counts to be displayed** = 324,000 (90 * 60 * 60);
- now move the axis to the 0 point; in other words, move it to the point where the travel starts; in this point the displayed position value will be "0";
- press the **1 – Start teaching** button;
- move the axis to the end of the travel; in other words, move it to the point where the travel stops; in this point the displayed position value will be "323,999";
- press the **2 – Stop teaching** button to close the programming procedure.

For instance, when the position of the axis will be "7,201" (**Position** = 7,201), then the angular position of the device will be 2° 0' 1".

Counts to be displayed

This parameter is available only when the **Programming mode: Start / Stop** is enabled.

It sets the overall information to be output for the travel comprised between the ends of the axis: the origin of the axis, i.e. the point where you press the **1 – Start teaching** button, and the end of the axis, i.e. the point where you press the **2 – Stop teaching** button; in other words it sets the output range of the information to be displayed.

Default = 4294967296 (min. = 1, max. = 4294967296)

NOTE

Please note that in this encoder the maximum number of physical information per revolution is 262,144 (2^{18}), while the maximum number of physical revolutions is 16,384 (2^{14}), see the **ENCODER SPECIFICATIONS** box on page 56. However the overall resolution (total counts) you are allowed to set is 268,435,456 (2^{28}).

1 – Start teaching

This button is available only when the **Programming mode: Start / Stop** is enabled.

After having moved the axis to the start position (0 position), i.e. the point where the travel we need to measure starts, press this button in order to bind the first value of the output range to the physical position you have moved the axis to (axis origin). As soon as you press this button, the display fields in the **POSITION** box are "frozen" and not available.

2 – Stop teaching

This button is available only when the **Programming mode: Start / Stop** is enabled.

After having moved the axis to the end position, i.e. the point where the travel we need to measure stops, press this button in order to bind the last value of the output range to the physical position you have moved the axis to (axis end).

WARNING

- If the start position (**1 – Start teaching**) and final position (**2 – Stop teaching**) match (i.e., you do not change the axis position during programming process and press both the buttons when the axis is in the same exact position), then the **Encoder travel not valid: unable to reach the minimum stroke** error message is displayed.
- if the travel exceeds the maximum encoder resolution as set by the operator, then the **Encoder travel not valid: counts to be displayed greater than physical resolution between start and stop** error message is displayed.

- If you set the start position, move the axis in a direction and then reverse the movement of the axis so going back over the start position, then the error message is displayed.

Programming mode: Prog
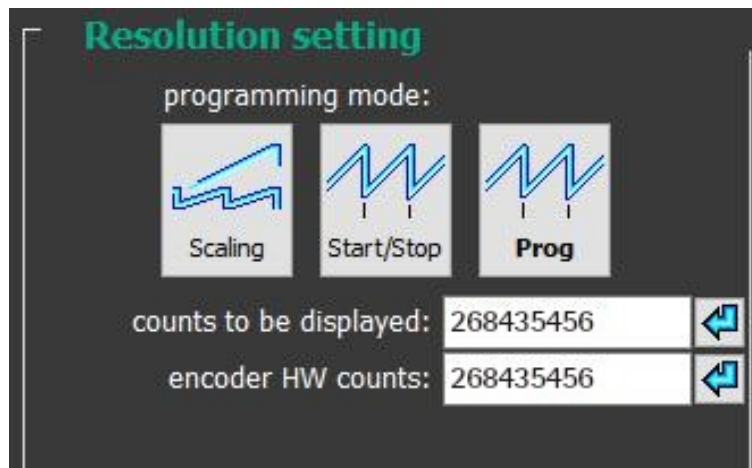
Press this button to enter the PROG programming mode and activate the relevant functions.

The PROG programming mode allows to programme the encoder when you exactly know the number of encoder information (counts) for the physical travel of the application you need to measure and control. The length of the travel you need to measure and control must be set next to the **Encoder HW counts** item and is expressed in counts. While the output range to be displayed must be set next to the **Counts to be displayed** item. Information is repeated cyclically.

EXAMPLE

Here follows an example useful to better understand how to programme the encoder using the PROG programming mode.

Let's suppose we need to measure and control the angular displacement of a device; the magnitude of the displacement is 27°. We can easily calculate the number of encoder information corresponding to a 27 degree rotation. If the maximum number of information per turn (i.e. in the 360 degrees) is 262,144 (the **Max counts per revolution** field in the **ENCODER SPECIFICATIONS** box = 262,144), we can set the following ratio: 262,144 : 360 = x : 27 => 262,144 * 27 / 360 = 19,661. So the number of encoder information in the 27 degree rotary displacement will be 19,661.

Now let's proceed as follows:

- set the number of information you need to display next to the **Counts to be displayed** item; let's suppose we need to express the displacement in degrees; thus set: **Counts to be displayed** = 27;
- now enter the number of encoder information corresponding to a 27 degree rotation, thus set: **Encoder HW counts** = 19,661.

Counts to be displayed

This parameter is available only when the **Programming mode: Prog** is enabled. It allows to enter the number of information you need to display for the travel whose length is set next to the following **Encoder HW counts** item, in other words it sets the output range to be displayed.
Default = 4294967296 (min. = 1, max. = 4294967296)

Encoder HW counts

This parameter is available only when the **Programming mode: Prog** is enabled. It sets the length of the displacement you need to measure and control expressed in encoder information (counts). When needed, calculate the portions of a turn by considering the **Max counts per revolution**. For instance: if the displacement is 1.8 turn long, enter: 262,144 * 1.8 = 471,859 counts.
Default = 4294967296 (min. = 1, max. = 4294967296)

5.3.6 ENCODER SPECIFICATIONS box

Encoder specifications
max counts per revolution: 262144
max number of revolutions: 16384
max total counts: 4294967296
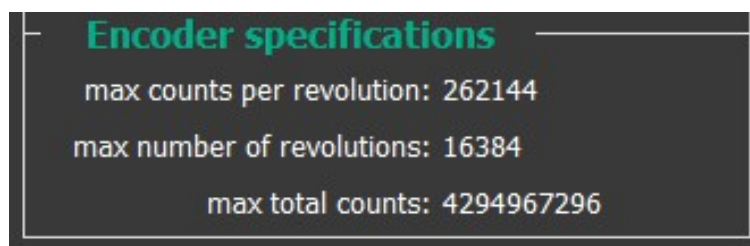
Max counts per revolution

It displays the number of physical information per revolution the device is able to output (namely, the singleturn resolution: 262,144 cpr -counts per revolution-, i.e. 2^{18}).

If you need to set a custom singleturn resolution refer to the **Counts per revolution** parameter on page 49.

NOTE

Please note that in this encoder the maximum number of physical information per revolution is 262,144 (2^{18}), while the maximum number of physical revolutions is 16,384 (2^{14}). However the overall resolution (total counts) you are allowed to set is 268,435,456 (2^{28}). It follows that you must arrange the counts per revolution and the number of revolutions so that the overall resolution (Total set counts) is not greater than 268,435,456 information.

For instance: if we set the maximum singleturn resolution, i.e. 262,144 cpr (2^{18}), then we can set max. 1,024 revolutions (2^{10}) as $2^{18} + 2^{10} = 2^{28}$. On the other hand, if we set the maximum number of revolutions, i.e. 16,384 revolutions (2^{14}), then we can set max. 16,384 cpr (max. singleturn resolution = 2^{14}) as $2^{14} + 2^{14} = 2^{28}$. And so on, in a whole variety of combinations.

If you enter a wrong resolution value, no matter whether singleturn resolution or number of revolution, so that the overall number of information is greater than 2^{28}, then the **Attention: the maximum transmitted position requires a number of bits greater than the available one** error message is displayed.

Max number of revolutions

It displays the number of physical revolutions the device is able to output (16,384 turns, i.e. 2^{14}).

If you need to set a custom multiturn resolution refer to the **Number of revolutions** parameter on page 50.

NOTE

Please note that in this encoder the maximum number of physical information per revolution is 262,144 (2^{18}), while the maximum number of physical revolutions is 16,384 (2^{14}). However the overall resolution (total counts) you are allowed to set is 268,435,456 (2^{28}). It follows that you must arrange the counts per revolution and the number of revolutions so that the overall resolution (Total set counts) is not greater than 268,435,456 information.

For instance: if we set the maximum singleturn resolution, i.e. 262,144 cpr (2^{18}), then we can set max. 1,024 revolutions (2^{10}) as $2^{18} + 2^{10} = 2^{28}$. On the other hand, if we set the maximum number of revolutions, i.e. 16,384 revolutions (2^{14}), then we can set max. 16,384 cpr (max. singleturn resolution = 2^{14}) as $2^{14} + 2^{14} = 2^{28}$. And so on, in a whole variety of combinations.

If you enter a wrong resolution value, no matter whether singleturn resolution or number of revolution, so that the overall number of information is greater than 2^{28}, then the **Attention: the maximum transmitted position requires a number of bits greater than the available one** error message is displayed.

Max total counts

It displays the overall physical resolution of the device (4,294,967,296, i.e. 2^{32}).

The overall physical resolution results from the following calculation:

Max total counts = Max counts per revolution * Max number of revolutions.

If you need to set a custom overall resolution refer to the parameters **Counts per revolution** on page 49 and **Number of revolutions** on page 50.

NOTE

Please note that in this encoder the maximum number of physical information per revolution is 262,144 (2^{18}), while the maximum number of physical revolutions is 16,384 (2^{14}). However the overall resolution (total counts) you are allowed to set is 268,435,456 (2^{28}). It follows that you must arrange the counts per revolution and the number of revolutions so that the overall resolution (**Total set counts**) is not greater than 268,435,456 information.

For instance: if we set the maximum singleturn resolution, i.e. 262,144 cpr (2^{18}), then we can set max. 1,024 revolutions (2^{10}) as $2^{18} + 2^{10} = 2^{28}$. On the other hand, if we set the maximum number of revolutions, i.e. 16,384 revolutions (2^{14}), then we can set max. 16,384 cpr (max. singleturn resolution = 2^{14}) as $2^{14} + 2^{14} = 2^{28}$. And so on, in a whole variety of combinations.

If you enter a wrong resolution value, no matter whether singleturn resolution or number of revolution, so that the overall number of information is greater than 2^{28}, then the **Attention: the maximum transmitted position requires a number of bits greater than the available one** error message is displayed.

5.4 LOAD / SAVE CONFIGURATION page

When you press the **LOAD / SAVE CONFIGURATION** button in the left navigation page you enter the **LOAD / SAVE CONFIGURATION** page.
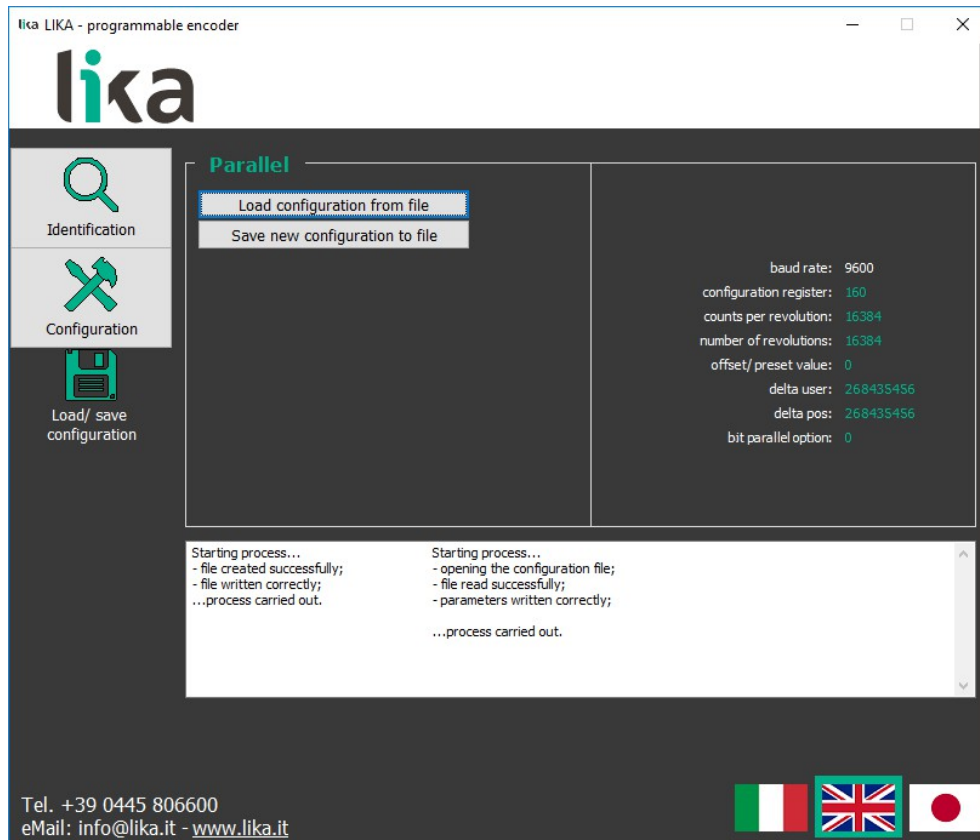
This page allows the operator to save the current encoder configuration to a .dat file; or to load a previously saved configuration. The current encoder configuration is listed in the pane on the right.

5.4.1 Saving the encoder configuration to a file

To save the current encoder configuration just press the **SAVE NEW CONFIGURATION TO FILE** button. As soon as you press the button the **Save as** dialogue box appears on the screen and the "Starting process…" message appears on the display box below in the page. The operator must type the .dat file name and specify the path where the file has to be located. When you press the **SAVE** button to confirm, the dialogue box closes. If the process is carried out properly, then the "- file created successfully; - file written correctly; …process

carried out" message appears on the display box. Otherwise the **Error creating the file** warning message appears on the display box.

5.4.2 Loading the encoder configuration from a file

When you need to load an encoder configuration you have previously saved to a file, press the **LOAD CONFIGURATION FROM FILE** button. As soon as you press the button, the **Open** dialogue box appears on the screen and the "Starting process..." message appears on the display box below in the page. The operator must select the folder and then the file where the configuration has been saved. When you press the **OPEN** button to confirm, the dialogue box closes. If the process is carried out properly, then the "- opening the configuration file; - file read successfully; - parameters written correctly; ...process carried out" message appears on the display box. Otherwise the **Error reading the file** warning message appears on the display box.

5.5 Error messages

(in alphabetical order)

Attention: the maximum transmitted position requires a number of bits greater than the available one

The operator entered a singleturn resolution value or a number of revolutions which cause the overall set resolution to be greater than 2^{28}. In this encoder the maximum number of physical information per revolution is 262,144 (2^{18}), while the maximum number of physical revolutions is 16,384 (2^{14}), see the **ENCODER SPECIFICATIONS** box on page 56. However the overall resolution (**Total set counts**) you are allowed to set is 268,435,456 (2^{28}). It follows that you must arrange the counts per revolution and the number of revolutions so that the overall resolution is less than 268,435,456 information.

For instance: if we set the maximum singleturn resolution, i.e. 262,144 cpr (2^{18}), then we can set max. 1,024 revolutions (2^{10}) as $2^{18} + 2^{10} = 2^{28}$. On the other hand, if we set the maximum number of revolutions, i.e. 16,384 revolutions (2^{14}), then we can set max. 16,384 cpr (max. singleturn resolution = 2^{14}) as $2^{14} + 2^{14} = 2^{28}$. And so on, in a whole variety of combinations.

If you enter a wrong resolution value, no matter whether singleturn resolution or number of revolution, so that the overall number of information is greater than 2^{28}, then this error message is displayed.

Counts to be displayed must be lower than the hardware counts of the encoder

While configuring the encoder in the PROG programming mode (see on page 55), the operator has set next to the **Counts to be displayed** item a value higher than the number of encoder information which defines the application displacement as set next to the **Encoder HW counts** parameter.

Error creating the file

If an error occurs while saving the encoder configuration to file (see the "5.4.1 Saving the encoder configuration to a file" section on page 59), then this error message is displayed.

Error reading the file

If an error occurs while uploading an encoder configuration file (see the "5.4.2 Loading the encoder configuration from a file" section on page 60), then this error message is displayed.

Invalid character

You entered an invalid character or comma (such as , \ / : * ? " < > |). All the fields only allow integer values.

Value higher than the hardware resolution, please set a lower value

The operator entered next to the **Counts per revolution** parameter in the **RESOLUTION SETTING** box a value greater than the number of physical information per revolution (see **Max counts per revolution** in the **RESOLUTION SETTING** box); or: he entered next to the **Number of revolutions** parameter in the **RESOLUTION SETTING** box a value greater than the number of physical revolutions (see **Max number of revolutions** in the **RESOLUTION SETTING** box).

Value not valid, offset greater than the maximum value allowed

The operator entered next to the **Offset** parameter in the **OFFSET** box a value greater than the one resulting from the following calculation:

Max total counts (4,294,967,296) – set overall resolution (**Total set counts** or **Counts to be displayed**).
Please enter a lower offset value.

**Value not valid, preset greater than the maximum position**
The operator has entered next to the **Preset** parameter in the **PRESET** box a value greater than the set overall resolution.

- When **Programming mode: Scaling** is set, then the **Preset** must be less than or equal to the **Total set counts** (**Counts per revolution** * **Number of revolutions**).
- When **Programming mode: Start / Stop** is set, then the **Preset** must be less than or equal to the **Counts to be displayed**).
- When **Programming mode: Prog** is set, then the **Preset** must be less than or equal to the **Counts to be displayed**).

5.6 "Red zone" compensation

Most of the encoders suffer the problem of the so-called "red zone". This problem occurs when the overall resolution (i.e. the number of information per revolution * the number of revolutions) is not a power of 2.

When this problem arises, the device must operate within the "red zone" for a certain number of positions. The size of the "red zone" is variable. To calculate it we must subtract the overall set resolution from the overall physical resolution of the device as many times as until the difference is less than the overall set resolution. When the encoder crosses the limit of the last value in the overall physical resolution, a counting error occurs, i.e. a jump in the position count. The problem is represented graphically in the following Figure.
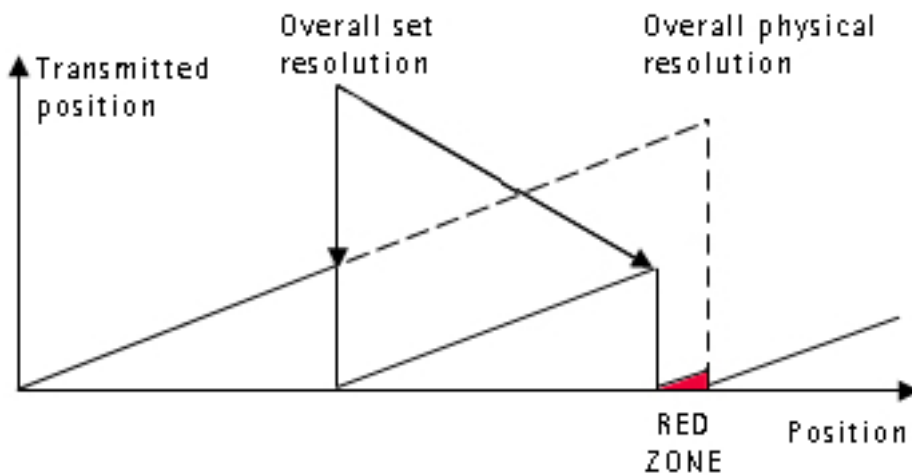
EXAMPLE

18–14 multiturn encoder.

Physical resolution:
- **Singleturn physical resolution**   = 262,144 counts/turn = 18 bits (2^{18})
- **Number of physical revolutions**   = 16,384 revolutions = 14 bits (2^{14})
- **Overall physical resolution**   = 4,294,967,296 = 32 bits (2^{32})

Set values:
- **Information per revolution**   = 262,144 = 2^{18}
- **Number of revolutions**   = 100 = it is NOT a power of 2
- **Overall set resolution**   = 26,214,400 = it is NOT a power of 2

This can be proved easily:

$$\frac{\textbf{Overall physical resolution}}{\textbf{Overall set resolution}} = \frac{4{,}294{,}967{,}296}{26{,}214{,}400} = 163.84$$

It follows that for 22,020,096 positions (4,294,967,296 – 26,214,400 * 163 = 22,020,096), i.e. for 84 revolutions, the encoder will work within the limits of the so-called "red zone". After the position 22,020,095 (i.e. at the end of the "red zone") a position error (namely, a "jump" in the position count) would happen as the following position would be "0". See the Figure in the previous page; please note that the cycle defined by the overall set resolution is not carried out twice within the overall physical resolution as shown in the Figure, but 163 times.

This encoder from Lika Electronic was designed to dynamically compensate the problem of the "red zone". An algorithm developed by Lika Electronic in fact allows to recognize the "red zone" so adding a compensation; in this way at the counting overflow where the max. physical resolution point is located, namely 4,294,967,296, a counting error does not occur.
The matter is overcome as shown in the following Figure.
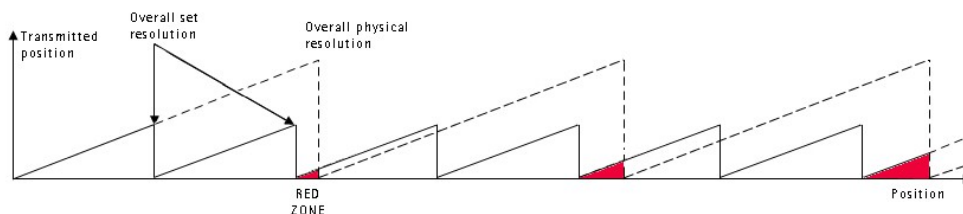
As you can easily understand, the transmitted position is consistent although the encoder is working within the limits of the "red zone".

WARNING
Please mind the following limitation.
It concerns the number of revolutions you are allowed to carry out while the encoder is switched off; if this value is exceeded, the compensation of the "red zone" is lost as well as the counting position.
This value is variable and depends on the set number of revolutions ($N_{umber\ of\ revolutions}$, see the **Number of revolutions** parameter) and has to be calculated in two different ways depending on whether the set number of revolutions is greater than or less than 8,192.

CASE 1

If the set number of revolutions ($N_{umber of revolutions}$) is $\leq$ **8192**, then the maximum number of revolutions you will be allowed to carry out while the encoder is switched off ($N_{max revolutions}$) will be as follows:

$$N_{max revolutions} = \frac{N_{cycles}}{2} * N_{umber of revolutions}$$

where

$$N_{cycles} = \text{INTEGER} \left(\frac{\text{Number of physical revolutions}}{N_{umber of revolutions}} \right) = \text{INTEGER} \left(\frac{2^{14}}{N_{umber of revolutions}} \right)$$

CASE 2

Otherwise, if the set number of revolutions ($N_{umber of revolutions}$) is > **8192**, then the maximum number of revolutions you will be allowed to carry out while the encoder is switched off ($N_{max revolutions}$) will be as follows:

$$N_{max revolutions} = SIZE_{red zone}$$

where

$$SIZE_{red zone} = \text{RESIDUAL} \left(\frac{\text{Number of physical revolutions}}{N_{umber of revolutions}} \right) = \text{RESIDUAL} \left(\frac{2^{14}}{N_{umber of revolutions}} \right)$$

Thus, if we use the values already reported in the example above in this paragraph, we will have as follows:

Physical resolution:
- **Singleturn physical resolution** = 262,144 counts per turn = 18 bits (2^{18})
- **Number of physical revolutions** = 16,384 revolutions = 14 bits (2^{14})
- **Overall physical resolution** = 4,294,967,296 = 32 bits (2^{32})

Set values:
- **Counts per turn** = 262,144 = 2^{18}
- **Number of revolutions** = 100 = it is NOT a power of 2
- **Overall set resolution** = 26,214,400 = it is NOT a power of 2

The set number of revolutions ($N_{umber\ of\ revolutions}$) is $\leq$ **8192**, thus $N_{max\ revolutions}$ will result from the following calculations:

$$N_{cycles} = \text{INTEGER}\left(\frac{2^{14} = 16{,}384}{100}\right) = 163\ (163.84)$$

So:

$$N_{max\ revolutions} = \frac{N_{cycles}}{2} * N_{umber\ of\ revolutions} = \frac{163}{2} * 100 = \mathbf{8{,}150}$$

From the example above it follows that the number of revolutions you are allowed to carry out while the encoder is switched off will be 8,150.

6   Default parameters list

Parameters list	Default value		
Preset	0		
Offset	0		
Output code	binary		
Positive counting direction	CW (standard)		
Enable preset / offset	enabled		
Select preset / offset	preset		
Enable parity bit	disabled		
Select parity bit	even		
Enable Latch	disabled		
Set logic type	positive (standard)		
Programming mode: Scaling	Scaling		
Counts per revolution PPR	16,384		
Number of revolutions R	16,384		
Total set counts P	268,435,456		
Counts to be displayed in Programming mode: Start / Stop	4,294,967,296		
Counts to be displayed in Programming mode: Prog	4,294,967,296		
Encoder HW counts P in Programming mode: Prog	4,294,967,296		

Document release	Release date	Description	HW	SW	Interface
1.0	02.10.2013	First issue	1	1	V1210
1.1	25.01.2016	Latch function description updated, general review	1	1	V1210
1.2	17.02.2021	MIL 41 picture updated, information about IF92 kit added, information about new programming software tool, general review	1	1	V1911, V2105, V3.6
1.3	09.03.2023	New order code, minor amendments, general revision, interface updated	1	1	V3.7

This device is to be supplied by a Class 2 Circuit or Low-Voltage Limited Energy or Energy Source not exceeding 30 Vdc. Refer to the order code for supply voltage rate.
Ce dispositif doit être alimenté par un circuit de Classe 2 ou à très basse tension ou bien en appliquant une tension maxi de 30Vcc. Voir le code de commande pour la tension d'alimentation.

Dispose separately