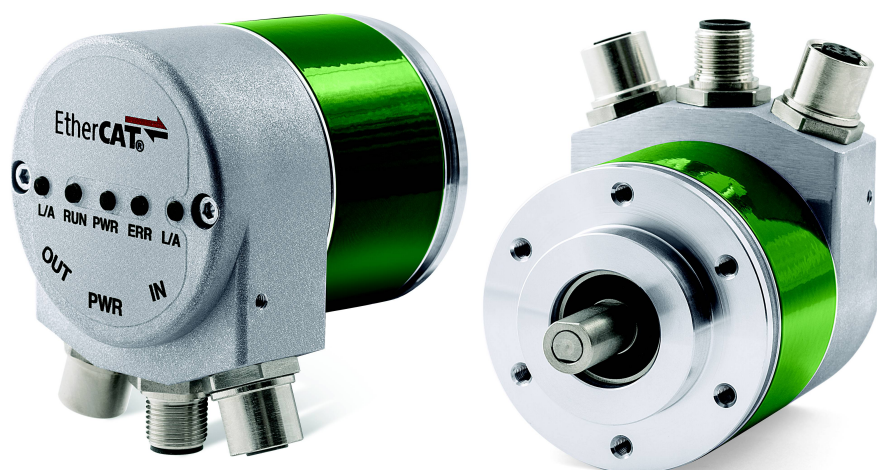


User's guide

EM58 EC

HS58, HM58 EC

Ether**CAT**® 



..... **lika**

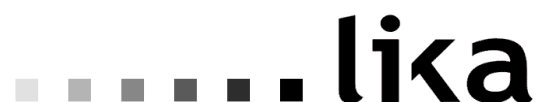
Smart encoders & actuators

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

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

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

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



General contents

User's guide.....	1
General contents.....	3
Subject Index.....	6
Typographic and iconographic conventions.....	7
Preliminary information.....	8
1 - Safety summary	9
Safety.....	9
Electrical safety.....	9
Mechanical safety.....	9
2 - Identification	10
3 - Mechanical installation	11
3.1 Solid shaft encoders.....	11
3.1.1 Customary installation.....	11
3.1.2 Installation using fixing clamps (code LKM-386).....	12
3.1.3 Installation using a mounting bell (code PF4256).....	12
3.2 Hollow shaft encoders.....	13
3.2.1 EMC58, HSC58, HMC58.....	13
3.2.2 EMC59, HSC59, HMC59.....	14
3.2.3 EMC60, HSC60, HMC60.....	15
4 - Electrical connections	16
4.1 M12 connectors.....	16
4.2 Ground connection.....	17
4.3 Diagnostic LEDs.....	17
5 - Quick reference (TwinCAT)	18
5.1 System configuration using TwinCAT software system from Beckhoff.....	18
5.1.1 Setting the Network Card.....	18
5.1.2 Add new I/O modules (Boxes).....	20
5.2 Setting the communication mode.....	25
5.2.1 Synchronous with SM3.....	25
5.2.2 Synchronous with DC (SYNCO).....	26
5.3 Process Data Objects.....	27
5.4 COE Object Dictionary.....	28
5.5 Online Data.....	29
5.6 EEPROM upgrade.....	30
5.7 Firmware upgrade.....	35
6 - EtherCAT® interface	39
6.1 Basic Information on EtherCAT® Protocol.....	39
6.1.1 Data transfer.....	40
6.1.2 ISO/OSI Layer model.....	40
6.1.3 Topology.....	41
6.1.4 Line Termination.....	42
6.1.5 Addressing.....	42
6.1.6 Communication mode.....	43
FreeRun.....	43
Synchronous with SM3.....	44
Synchronous with DC SYNCO.....	45

6.1.7 EtherCAT State Machine (ESM).....	46
6.1.8 Slave configuration.....	47
6.1.9 Timing and Synchronization.....	48
Sync Manager.....	48
Buffered Mode (3-Buffer Mode).....	48
Mailbox Mode (1-Buffer Mode).....	48
6.2 CANopen Over EtherCAT (CoE).....	49
6.2.1 XML file.....	49
6.2.2 Communication messages.....	50
6.2.3 Process Data Objects (PDO).....	51
6.2.4 Service Data Objects (SDO).....	51
6.2.5 Object dictionary.....	52
Standard objects.....	52
1000-00 Device type.....	52
1008-00 Manufacturer Device Name.....	52
1009-00 Hardware version.....	52
100A-00 Software version.....	53
1010-01 Store parameters.....	53
1011-01 Restore default parameters.....	53
1018 Identification object.....	54
01 Vendor number.....	54
02 Product number.....	54
03 Revision number.....	54
04 Serial number.....	54
1A00-01 PDO mapping parameter.....	54
1C00 Sync Manager Communication Type.....	54
01 SM MailBox Receive (SM0).....	54
02 SM MailBox Send (SM1).....	54
03 SM PDO output (SM2).....	54
04 SM PDO input (SM3).....	54
1C12-00 Sync Manager RxPDO Assigned.....	54
1C13-01 Sync Manager TxPDO Assigned.....	54
1C33 Sync Manager input parameter.....	55
01 Sync Type.....	55
02 Cycle time.....	55
03 Shift Time.....	55
04 Sync modes supported.....	55
05 Minimum cycle time.....	55
06 Calc and Copy time.....	55
Device profile objects.....	56
6000-00 Operating parameters.....	56
Code sequence.....	56
Scaling function.....	56
6001-00 Counts per revolution.....	56
6002-00 Total resolution.....	57
6003-00 Preset value.....	58
6004-00 Position value.....	59
6500-00 Operating status.....	59
Code sequence.....	59
Scaling function.....	60
6501-00 Hardware counts per revolution.....	60

6502-00 Hardware number of turns.....	60
6503-00 Errors.....	60
6504-00 Supported errors.....	60
6505-00 Warnings.....	60
6506-00 Supported warnings.....	61
6509-00 Offset value.....	61
6.2.6 SDO Abort codes.....	62
6.2.7 Emergency Error Codes.....	62
6.2.8 AL Status Error Codes.....	62
7 – Default parameters list.....	64

Subject Index




1	
1000-00 Device type.....	52
1008-00 Manufacturer Device Name.....	52
1009-00 Hardware version.....	52
100A-00 Software version.....	53
1010-01 Store parameters.....	53
1011-01 Restore default parameters.....	53
1018 Identification object.....	54
1A00-01 PDO mapping parameter.....	54
1C00 Sync Manager Communication Type.....	54
1C12-00 Sync Manager RxPDO Assigned.....	54
1C13-01 Sync Manager TxPDO Assigned.....	54
1C33 Sync Manager input parameter.....	55
6	
6000-00 Operating parameters.....	56
6001-00 Counts per revolution.....	56
6002-00 Total resolution.....	57
6003-00 Preset value.....	58
6004-00 Position value.....	59
6500-00 Operating status.....	59
6501-00 Hardware counts per revolution.....	60
6502-00 Hardware number of turns.....	60
6503-00 Errors.....	60
6504-00 Supported errors.....	60
6505-00 Warnings.....	60
6506-00 Supported warnings.....	61
6509-00 Offset value.....	61
C	
Calc and Copy time.....	55
Code sequence.....	56, 59
Cycle time.....	55
D	
Diagnostic data.....	62
H	
Hardware error.....	62
I	
INIT.....	46
M	
Minimum cycle time.....	55
O	
OPERATIONAL.....	46
P	
PRE-OPERATIONAL.....	46
Product number.....	54
R	
Revision number.....	54
S	
SAFE-OPERATIONAL.....	46
Scaling function.....	56, 60
Serial number.....	54
Shift Time.....	55
SM MailBox Receive (SM0).....	54
SM MailBox Send (SM1).....	54
SM PDO input (SM3).....	54
SM PDO output (SM2).....	54
Sync modes supported.....	55
Sync Type.....	55
V	
Vendor number.....	54

Typographic and iconographic conventions

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

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

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

	This icon, followed by the word 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 describe the technical characteristics, installation and use of the EtherCAT encoders of the **EMx58x series** and **Hx58x series**. For any further information please refer to the product datasheet.

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

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

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

1 - Safety summary

Safety

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

Electrical safety

- Turn off power supply before connecting the device;
- connect according to explanation in the section "4 - Electrical connections";
- in compliance with the 2004/108/EC norm on electromagnetic compatibility, following precautions must be taken:
 - before handling and installing, discharge electrical charge from your body and tools which may come in touch with the device;
 - power supply must be stabilized without noise, install EMC filters on device power supply if needed;
 - always use shielded cables (twisted pair cables whenever possible);
 - avoid cables runs longer than necessary;
 - avoid running the signal cable near high voltage power cables;
 - mount the device as far as possible from any capacitive or inductive noise source, shield the device from noise source if needed;
 - to guarantee a correct working of the device, avoid using strong magnets on or near by the unit;
 - minimize noise by connecting the shield and/or the 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.



Mechanical safety

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

2 - Identification

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



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

3 - Mechanical installation



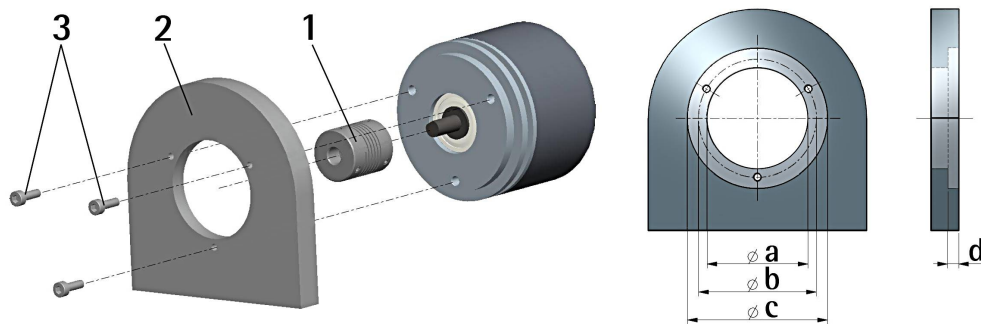
WARNING

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

3.1 Solid shaft encoders

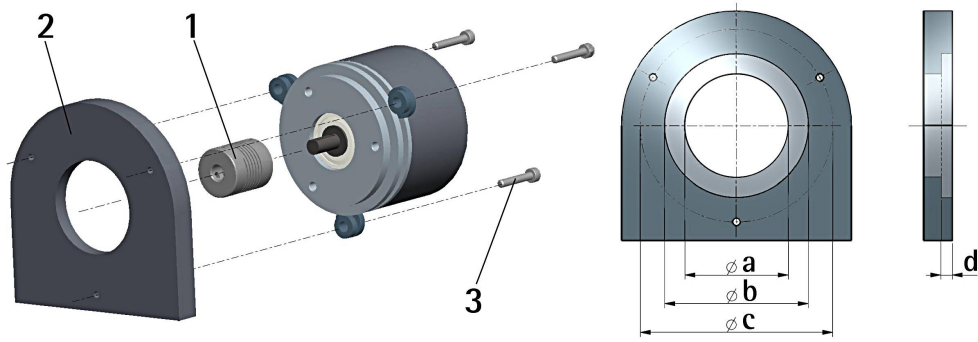
- Mount the flexible coupling **1** on the encoder shaft;
- fix the encoder to the flange **2** (or to the mounting bell) by means of the screws **3**;
- secure the flange **2** to the support (or the mounting bell to the motor);
- mount the flexible coupling **1** on the motor shaft;
- make sure the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

3.1.1 Customary installation



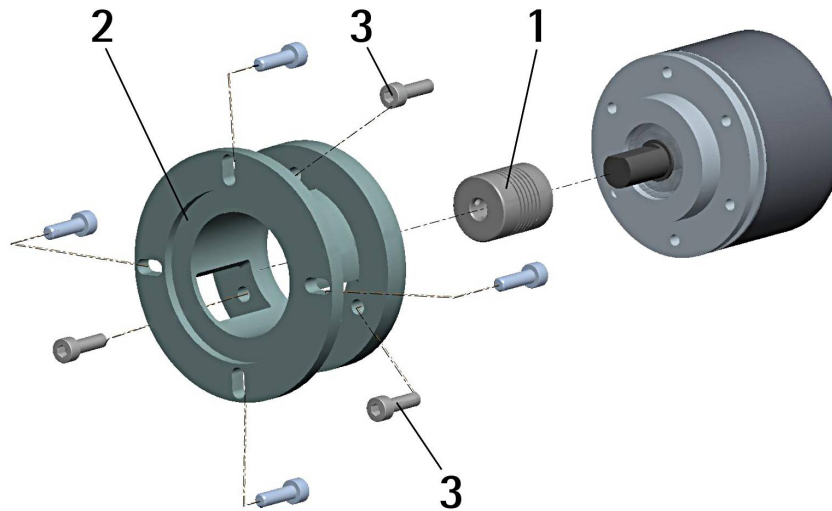
	a [mm]	b [mm]	c [mm]	d [mm]
EM58, HS58, HM58	-	42	50 F7	4
EM58S, HS58S, HM58S	36 H7	48	-	-

3.1.2 Installation using fixing clamps (code LKM-386)



	a [mm]	b [mm]	c [mm]	d [mm]
EM58, HS58, HM58	-	50 F7	67	4
EM58S, HS58S, HM58S	36 H7	-	67	-

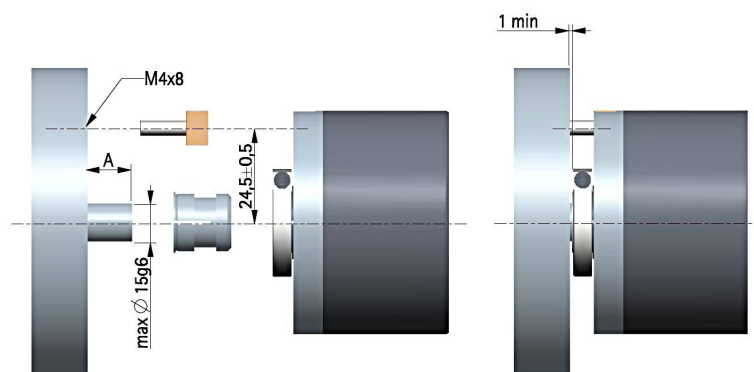
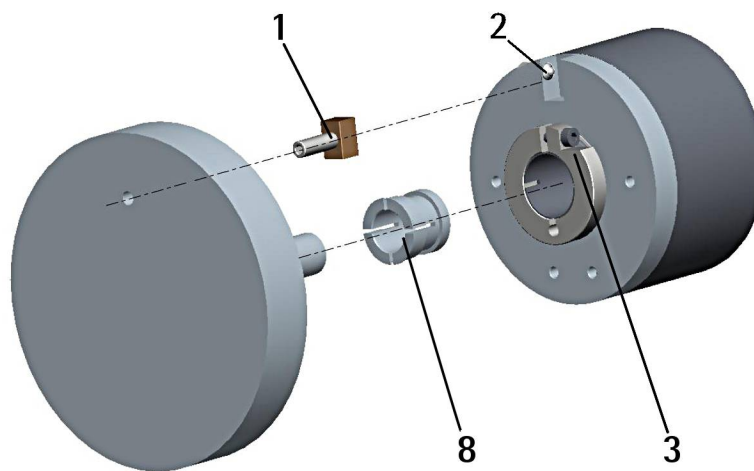
3.1.3 Installation using a mounting bell (code PF4256)



3.2 Hollow shaft encoders

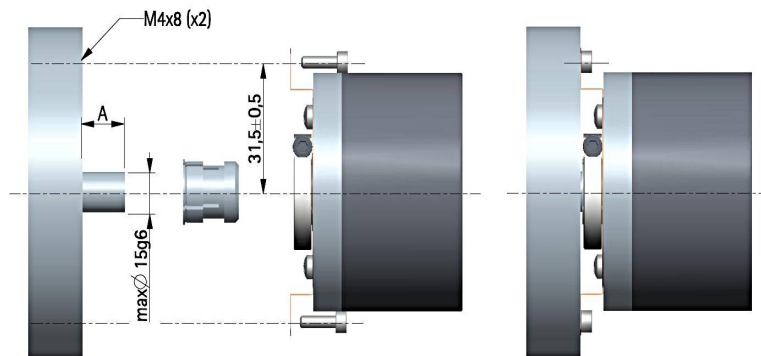
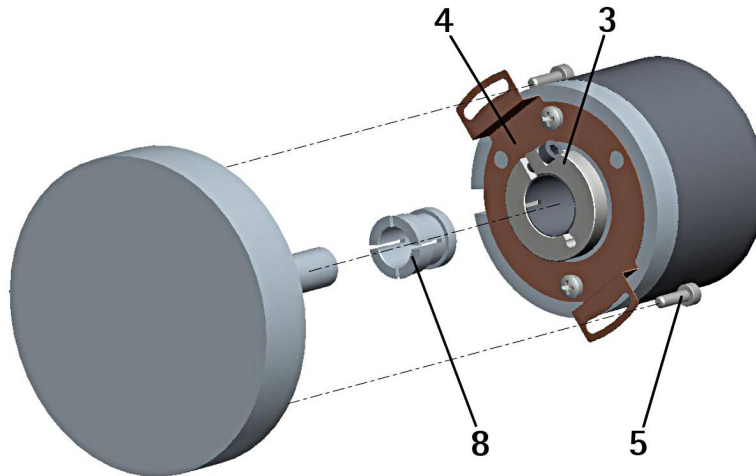
3.2.1 EMC58, HSC58, HMC58

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



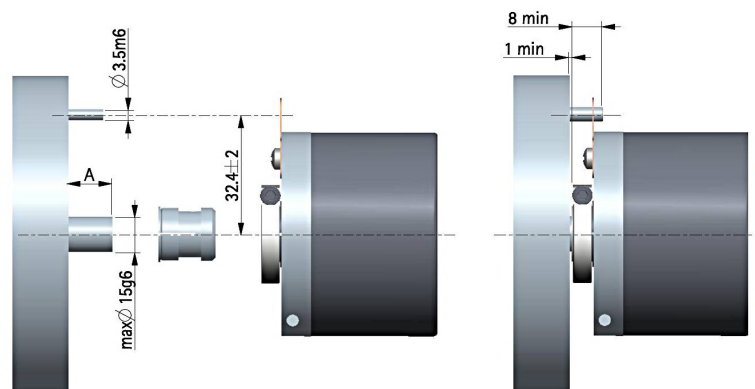
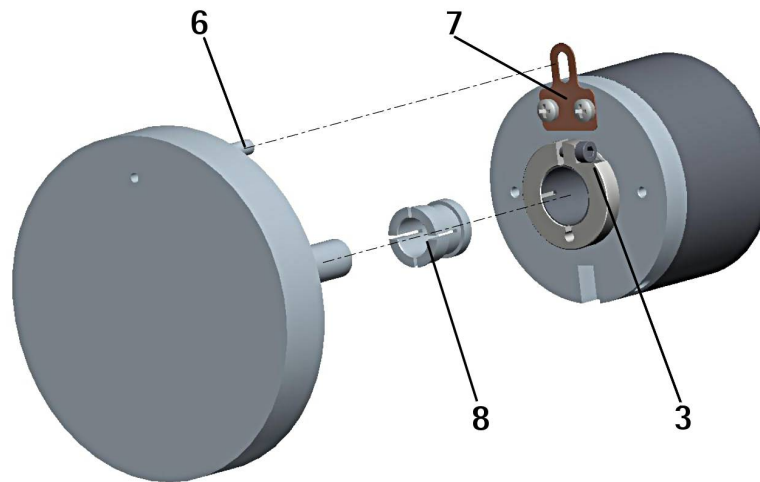
3.2.2 EMC59, HSC59, HMC59

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



3.2.3 EMC60, HSC60, HMC60

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

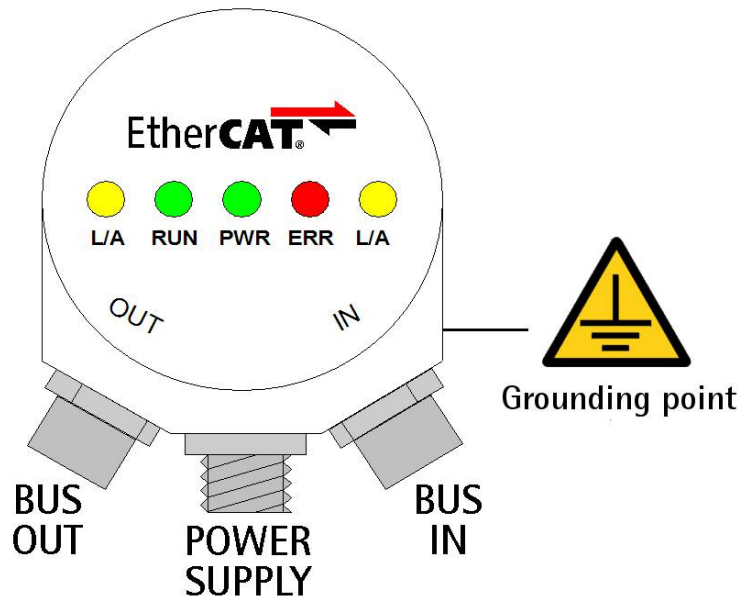


4 - Electrical connections



WARNING

Power supply must be turned off before performing any electrical connection!
Do not remove the connection cap of the encoder. Damage may be caused to internal components.



4.1 M12 connectors

The connection cap is fitted with three M12 connectors with pin-out in compliance with the EtherCAT® standard. Therefore you can use standard EtherCAT cables commercially available.

Input (IN) and output (OUT) connectors are not interchangeable! IN connector must be placed in the direction of the EtherCAT Master.

M12 (frontal side)	BUS IN & BUS OUT	POWER
	<p>D coding female</p>	<p>A coding male</p>

Pin	Description	Description
1	Tx Data +	+10VDC +30VDC
2	Rx Data +	n.c.
3	Tx Data -	0 VDC
4	Rx Data -	n.c.

n.c.= not connected.

4.2 Ground connection

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

4.3 Diagnostic LEDs

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

Link/ Activity (yellow)	it shows the status of the physical link and the activity in this link
OFF	condition: port closed, link: YES, activity: N.A.
FLICKERING	condition: port open, link: YES, activity: YES
ON	condition: port open, link: YES, activity: NO

RUN (green)	it shows the status of the EtherCAT State Machine (ESM)
OFF	encoder is in INIT status
BLINKING	encoder is in PRE-OPERATIONAL status
SINGLE FLASH	encoder is in SAFE-OPERATIONAL status
ON	encoder is in OPERATIONAL status

PWR (green)	it shows the power supply status
OFF	encoder is switched OFF
ON	encoder is switched ON

ERR (red)	it shows the error status
OFF	no error
FLICKERING	error on loading parameters from flash memory at start-up
FLICKERING	encoder parameters have not been saved correctly on flash memory
BLINKING	invalid configuration
SINGLE FLASH	local error (see ETG1000.6)
DOUBLE FLASH	watchdog timeout
ON	memory error and ESC controller not active

5 - Quick reference (TwinCAT)

5.1 System configuration using TwinCAT software system from Beckhoff

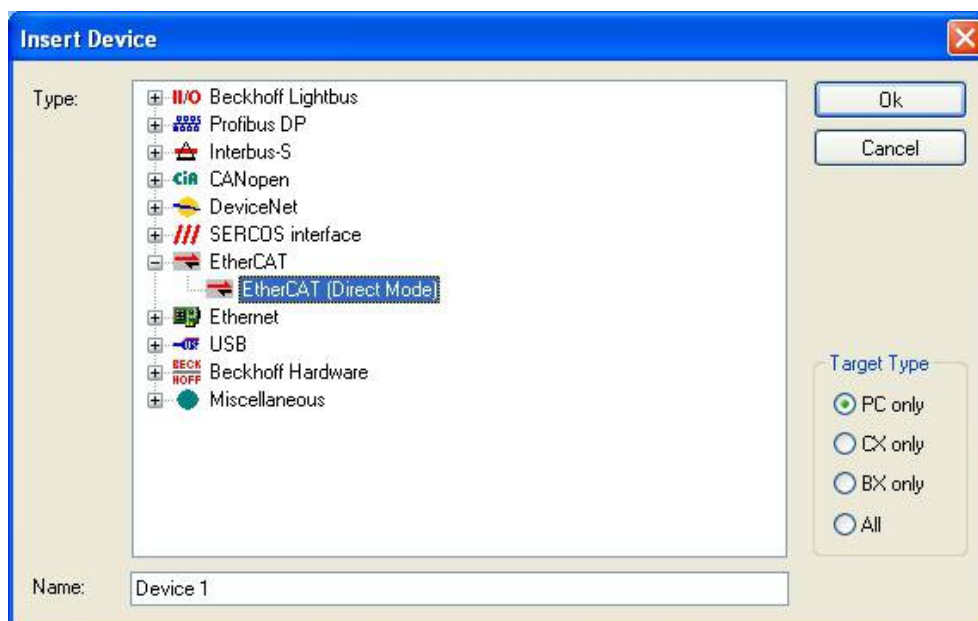
5.1.1 Setting the Network Card

Launch **TwinCAT System Manager**.

In the left pane of the main window extend the devices tree and select the **I/O Devices** item; right-click the **I/O Devices** item and then press the **Append Device...** command.



In the **Insert Device** window select the **EtherCAT (Direct Mode)** item and confirm pressing the **OK** button.

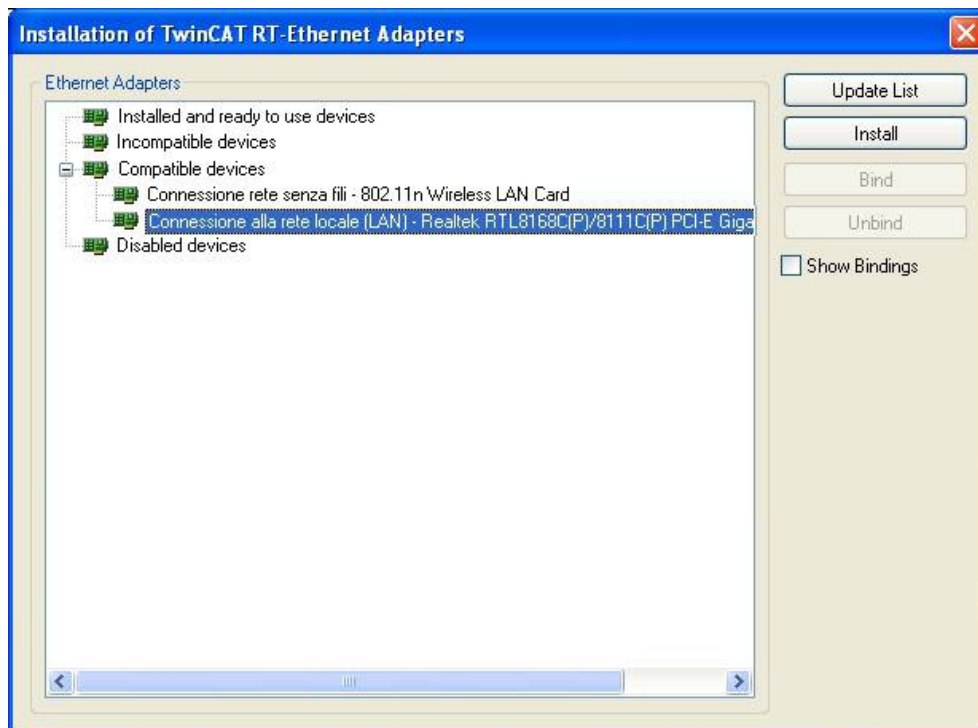


If a network card has been already installed properly, the following window will appear showing the list of the installed devices.



Select the network card you want to use and then confirm the choice pressing the **OK** button.

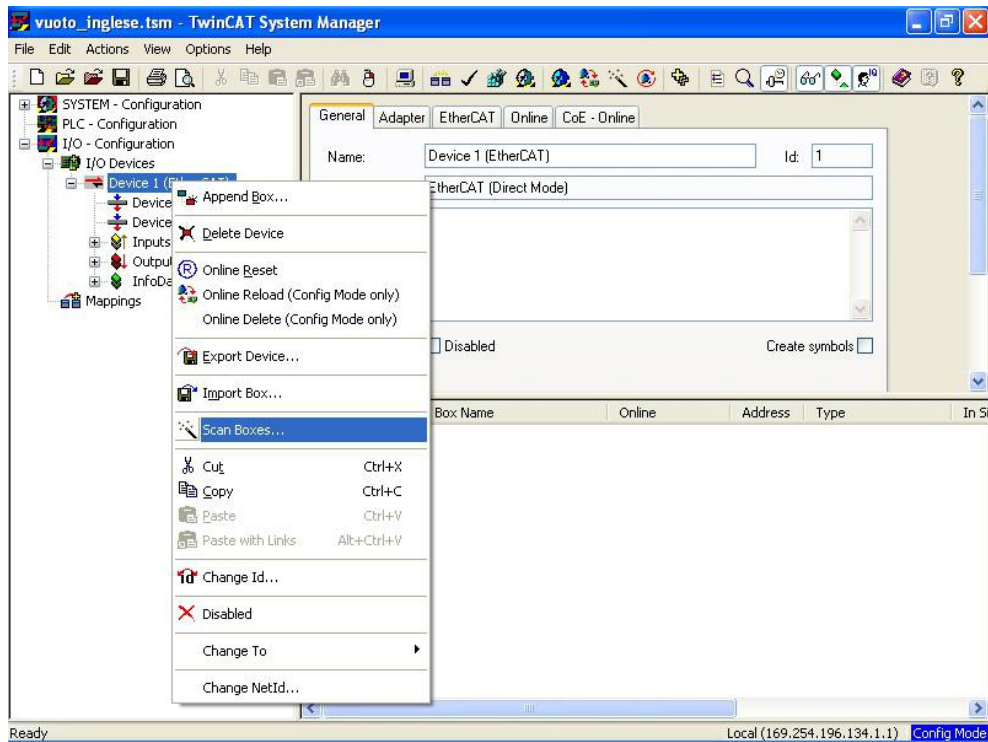
If there are no network cards installed, you must install one before proceeding. To do this, on the menu bar of the **TwinCAT System Manager** main window, select the **Options** menu and then press the **Show Real Time Ethernet Compatible Devices...** command.



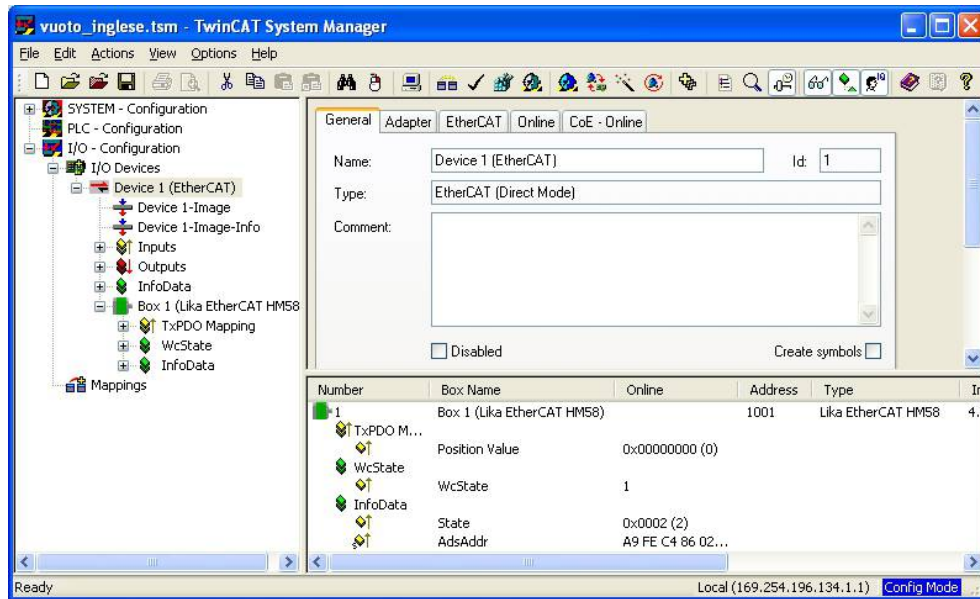
The **Installation of TwinCAT RT – Ethernet Adapter** window will appear. Now select **Compatible Devices** item and choose the network card you want to install; finally press the **Install** button to confirm your choice.

5.1.2 Add new I/O modules (Boxes)

If devices are connected to the network and switched ON, right-click the **Device 1 (EtherCAT)** item in the left pane of the **TwinCAT System Manager** main window and press the **Scan Boxes...** command.



At the end of the process some information will be listed in the right page as in Figure here below.



If devices are not already connected to the network it is necessary to use the XML file supplied with the encoder: **Lika_Ex58_Hx58_EC_Vx.xml** (see at www.lika.biz > **ROTARY ENCODERS** > **ABSOLUTE ENCODERS** > **ETHERCAT**). At the same address you can also download the most recent files to update both the firmware and the EEPROM of the encoder. For older versions of the files please contact Lika Electronic's After Sales Service Dept. For any information on the firmware upgrade procedure refer to the section "5.7 Firmware upgrade" on page 35. For any information on the EEPROM upgrade procedure refer to the section "5.6 EEPROM upgrade" on page 30).

**WARNING**

Before installing the XML file please check that it is compatible with the firmware version and the EEPROM version of the device; the XML file version, the firmware version and the EEPROM version must always comply. For example: if the firmware version is H1_S4 (Hardware version: 1; Software version: 4), it is mandatory that the EEPROM version is S4, therefore you must then install the XML file version V4. For a description of the EEPROM upgrade procedure please refer to the section "5.6 EEPROM upgrade" on page 30. For a description of the firmware upgrade procedure please refer to the section "5.7 Firmware upgrade" on page 35.

**WARNING**

It is mandatory that in an EtherCAT network all devices are provided with the same version of the firmware, EEPROM and XML file. So when you need to replace an old encoder installed in your network, then you must either upgrade all the encoders in the network to the last version compatible with the new encoder; or you must downgrade the new encoder to the older version compatible with the encoders already installed in the network.

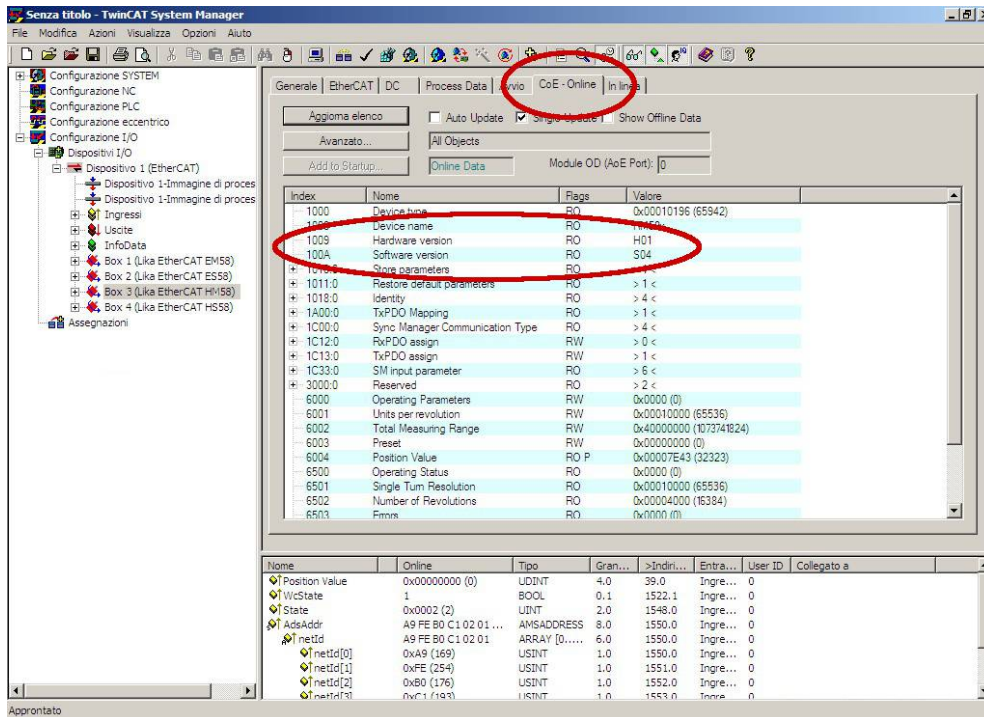
**WARNING**

H- series has been implemented starting from version V1.
E- series has been implemented starting from version V4.

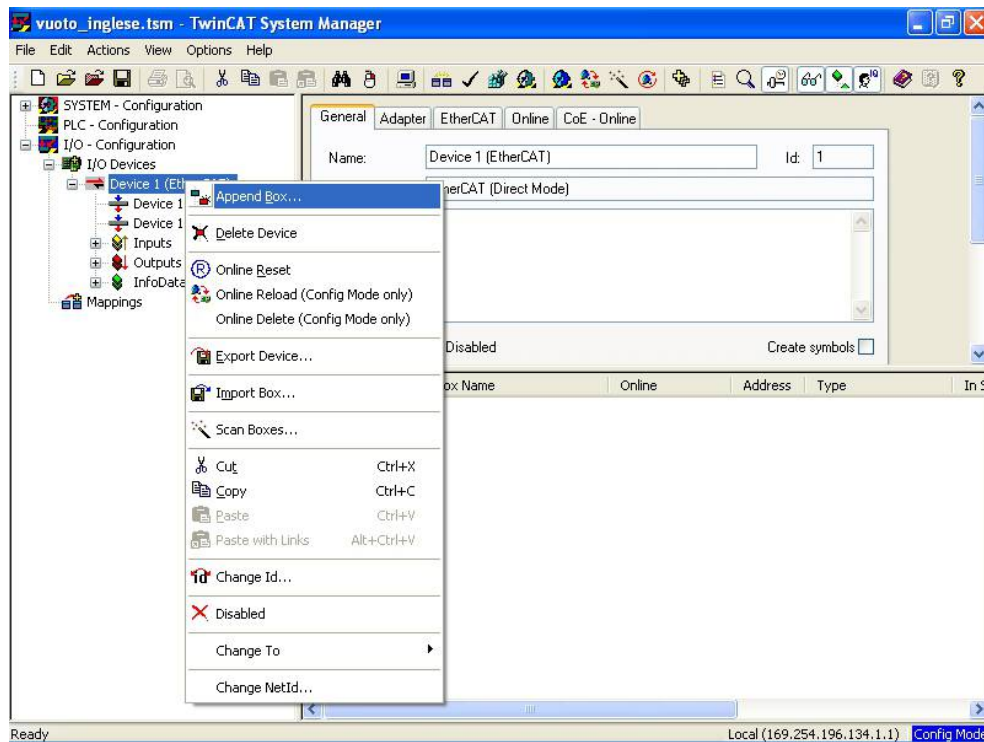
**WARNING**

In the H- series all versions are compatible for upgrade and downgrade except version V1.
In the E- series all versions are compatible for upgrade and downgrade starting from version V4.

If you want to know the firmware version of a device, press the **Box (Lika EtherCAT EM58, ES58, HM58 or HS58)** item in the left pane of the **TwinCAT System Manager** main window: some tabbed pages for configuring and managing the device will appear in the right pane. Enter the **CoE - Online** page and refer to the indexes Hardware version and Software version.



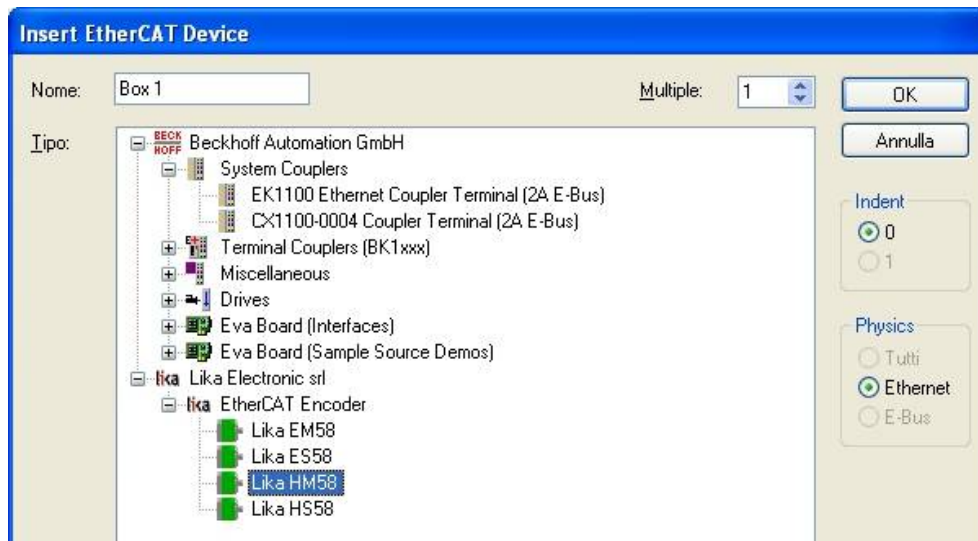
Right-click the **Device 1 (EtherCAT)** item in the left pane of the TwinCAT System Manager main window and press the **Append Box...** command.



In the **Insert EtherCAT Device** window that appears select **Lika Electronic srl** and then **EtherCAT Encoder** items; now choose from the list the encoder you want to install:

- Lika EM58: multi-turn encoder E- series;
- Lika ES58: single-turn encoder E- series;
- Lika HM58: multi-turn encoder H- series;
- Lika HS58: single-turn encoder H- series.

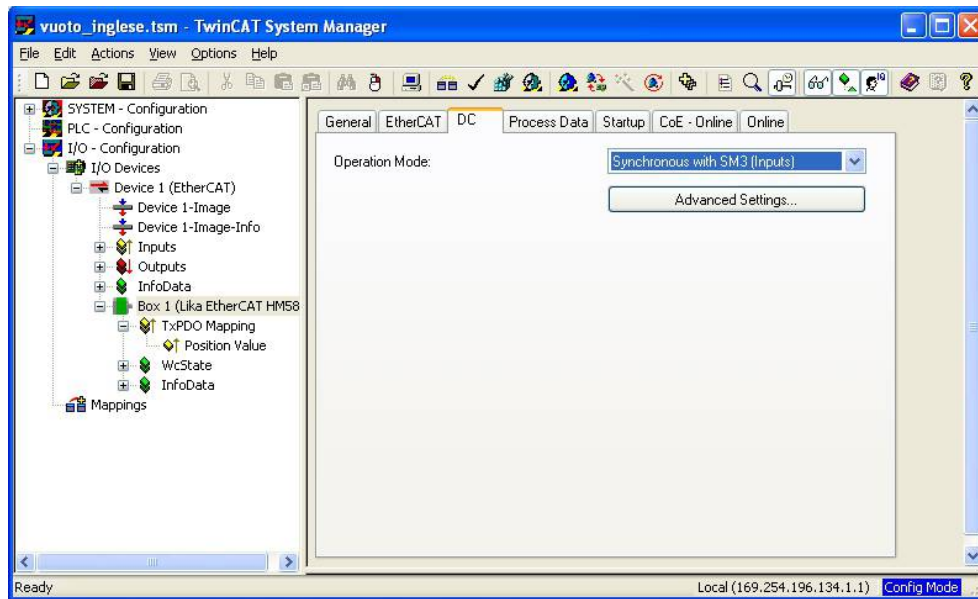
Press the **OK** button to confirm your choice.



5.2 Setting the communication mode

5.2.1 Synchronous with SM3

In the left pane of the **TwinCAT System Manager** main window press the **Box** (**Lika EtherCAT EM58, ES58, HM58 or HS58**) item: some tabbed pages for configuring and managing the device will appear in the right pane. Enter **DC** page. Select the **Synchronous with SM3 (Inputs)** option in the **Operation Mode** box.

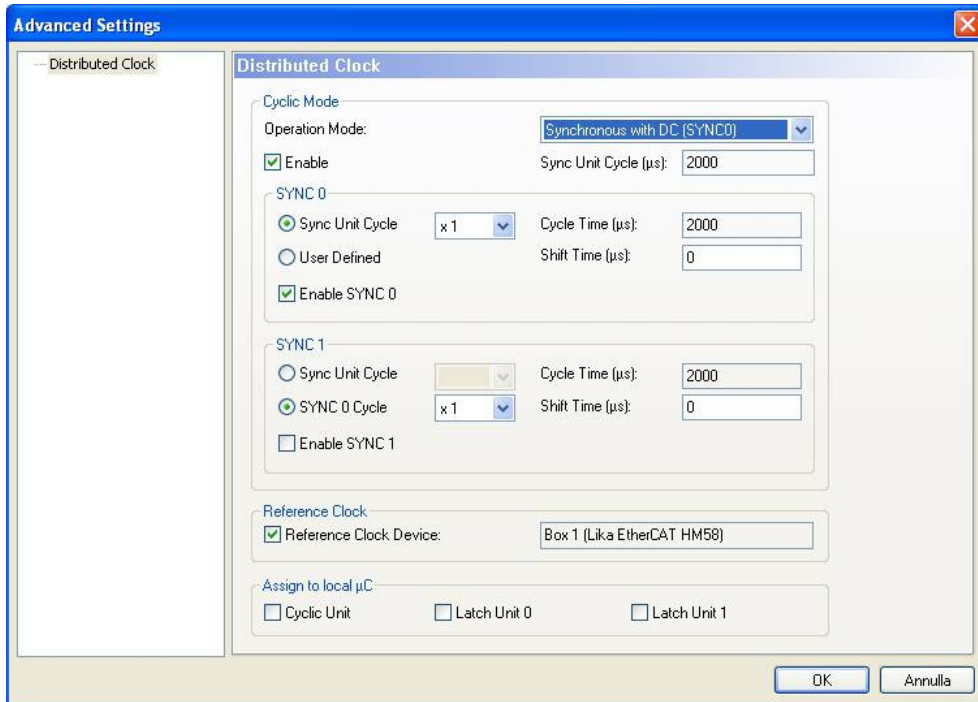


For any further information on the Synchronous with SM3 operation mode please refer to the section "Synchronous with SM3" on page 44 and to the object **1C33 Sync Manager input parameter** on page 55.

5.2.2 Synchronous with DC (SYNC0)

In the left pane of the **TwinCAT System Manager** main window press the **Box (Lika EtherCAT EM58, ES58, HM58 or HS58)** item: some tabbed pages for configuring and managing the device will appear in the right pane. Enter **DC** page.

Select the **Synchronous with DC (SYNC0)** option in the **Operation Mode** box. Then press the **Advanced Settings...** button.

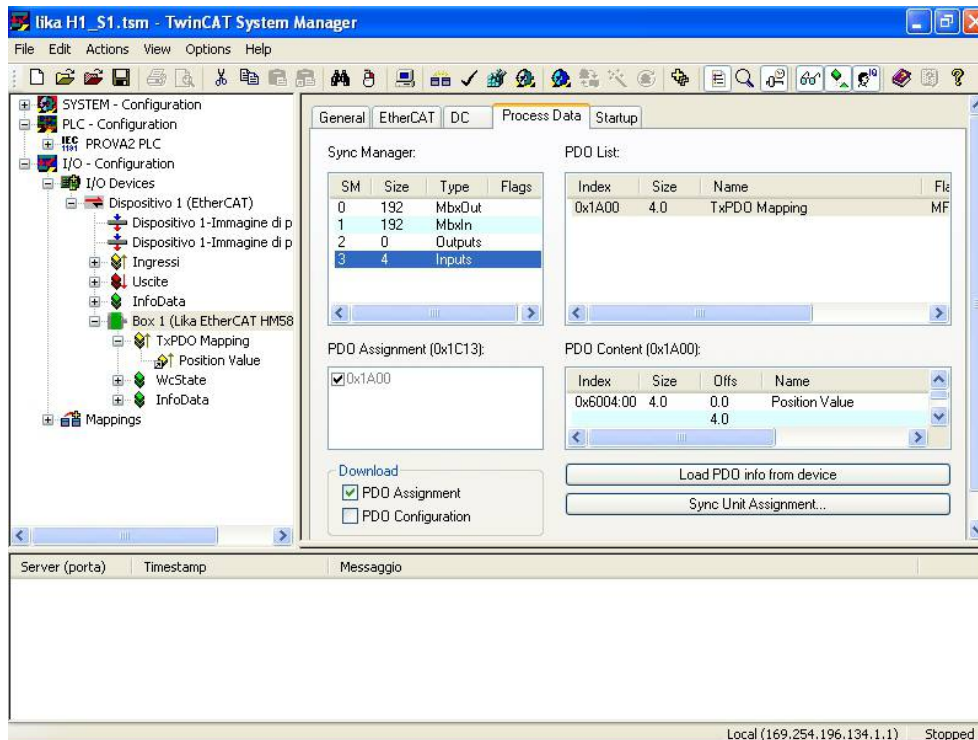


The **Advanced Settings** window will appear. In the section tabbed **SYNC 0** set the cycle time next to the **Sync Unit Cycle** box; sync time is calculated as multiple (or sub-multiple) of the value set in the **Sync Unit Cycle (µs)** item right above.

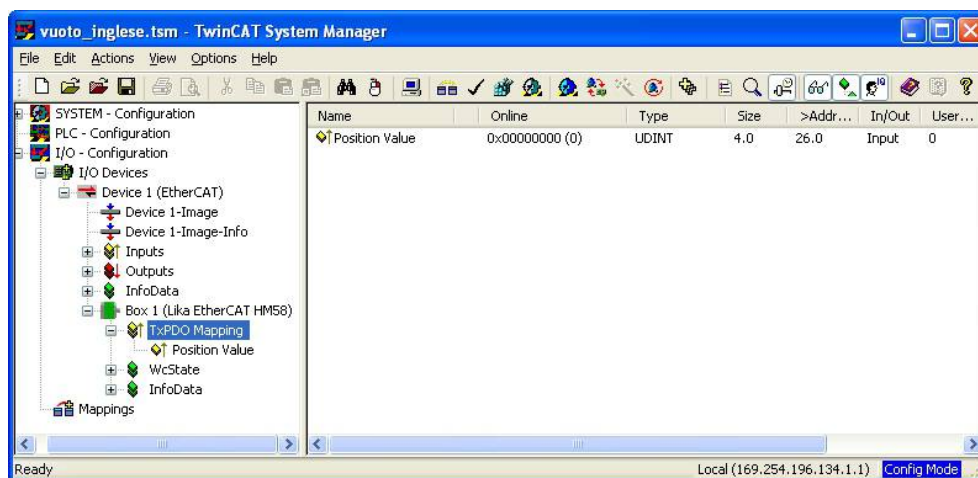
For any further information on the Synchronous with DC operation mode please refer to the section "Synchronous with DC SYNC0" on page 45 and to the object **1C33 Sync Manager input parameter** on page 55.

5.3 Process Data Objects

In the left pane of the **TwinCAT System Manager** main window press the **Box (Lika EtherCAT EM58, ES58, HM58 or HS58)** item: some tabbed pages for configuring and managing the device will appear in the right pane. Enter the **Process Data** page. In this page process data objects (PDO Mapping) are shown.

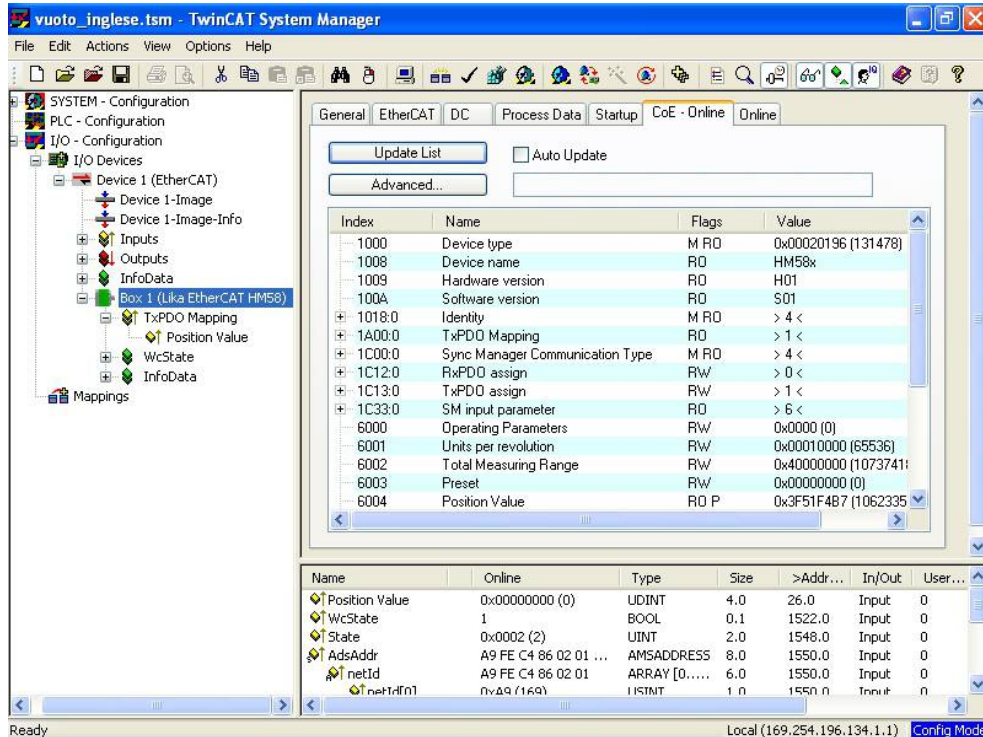


Process data objects can be displayed also by pressing the **TxPDO Mapping** item in the left pane of the **TwinCAT System Manager** main window; data is listed in the right pane.

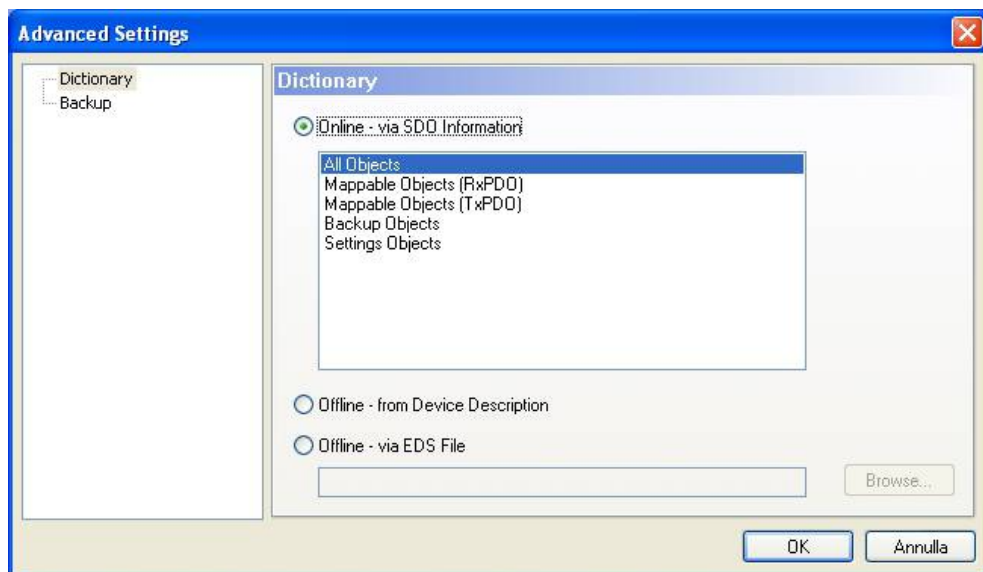


5.4 COE Object Dictionary

In the left pane of the **TwinCAT System Manager** main window press the **Box (Lika EtherCAT EM58, ES58, HM58 or HS58)** item: some tabbed pages for configuring and managing the device will appear in the right pane. Enter the **CoE - Online** page. In this page the objects dictionary is shown.



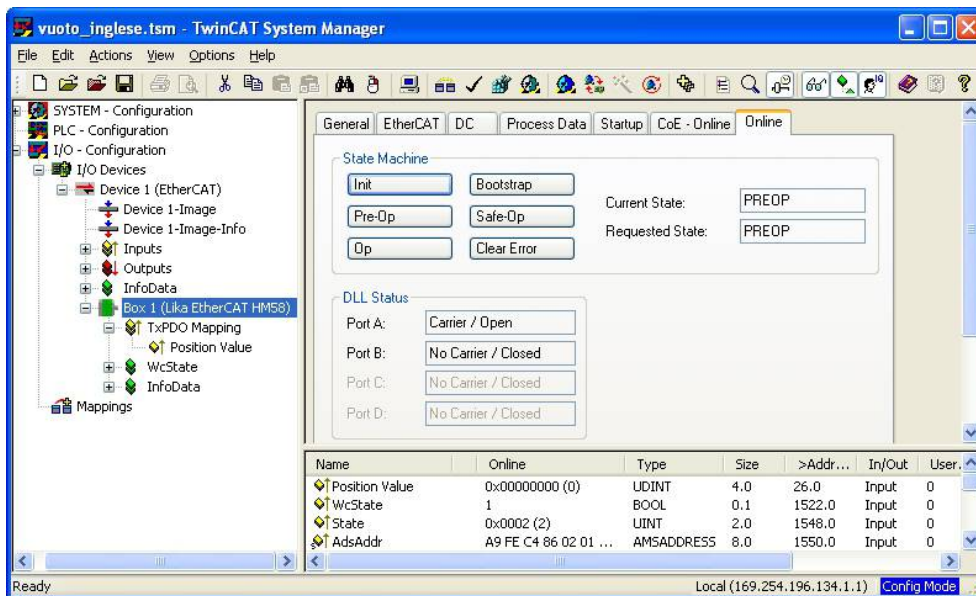
Objects can be read directly from the encoder; to do this click the **Advanced...** button: the **Advanced Settings** window will appear.



Select the **Dictionary** item in the left pane and then choose the **Online - via SDO Information** option in the **Dictionary** page; press **OK** button to confirm.

5.5 Online Data

In the left pane of the **TwinCAT System Manager** main window press the **Box (Lika EtherCAT EM58, ES58, HM58 or HS58)** item: some tabbed pages for configuring and managing the device will appear in the right pane. Enter the **Online** page to check the encoder status.



To display the encoder process data in real time, click the **Safe-OP** button if you want to display inputs only; click the **OP** button if you want to display both inputs and outputs.



WARNING

The structure of Data Objects (PDO and SDO) requires bytes to be sent from the Least Significant Byte (LSB) to the Most Significant Byte (MSB).

On the contrary in TwinCAT write and read data from MSB to LSB.

Furthermore in TwinCAT also strings must be entered in the reverse order:

- read default values: Data byte = 64 61 6F 6Chex = "daol" in ASCII code (means "load" in reverse);
- save parameters: Data byte = 65 76 61 73hex = "evas" in ASCII code (means "save" in reverse).

5.6 EEPROM upgrade



WARNING

The EEPROM upgrade process has to be accomplished by skilled and competent personnel. If the upgrade is not performed according to the instructions provided or a wrong or incompatible EEPROM program is installed then the unit may not be updated correctly, in some cases preventing the unit from working.



WARNING

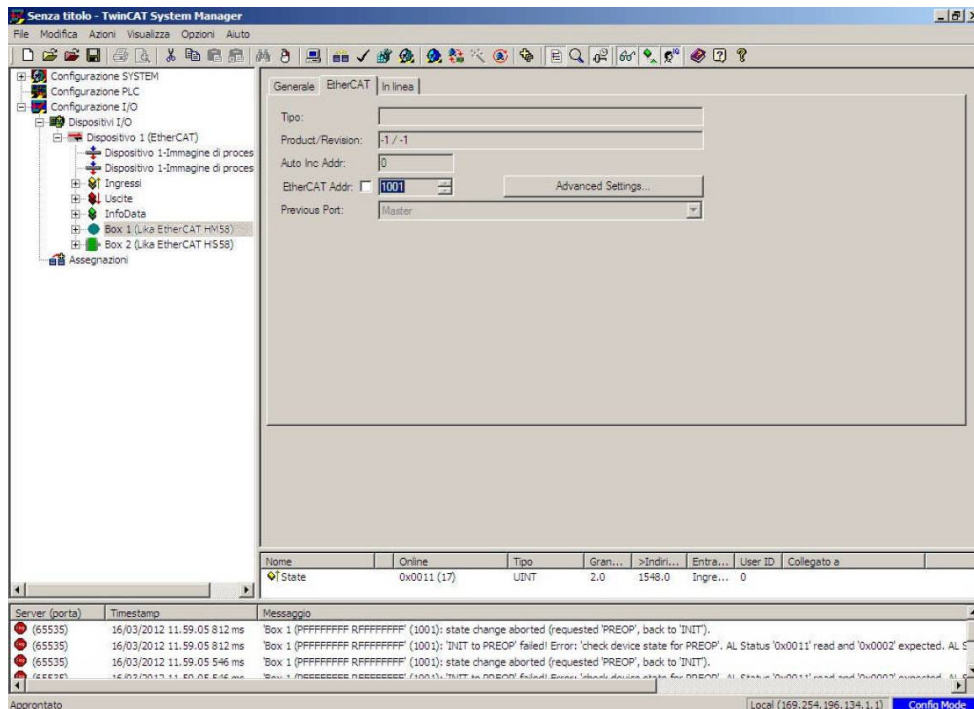
The XML file version, the firmware version and the EEPROM version must always comply. For example: if the firmware version is H1_S4 (Hardware version: 1; Software version: 4), it is mandatory that the EEPROM version is S4, therefore you must then install the XML file version V4.



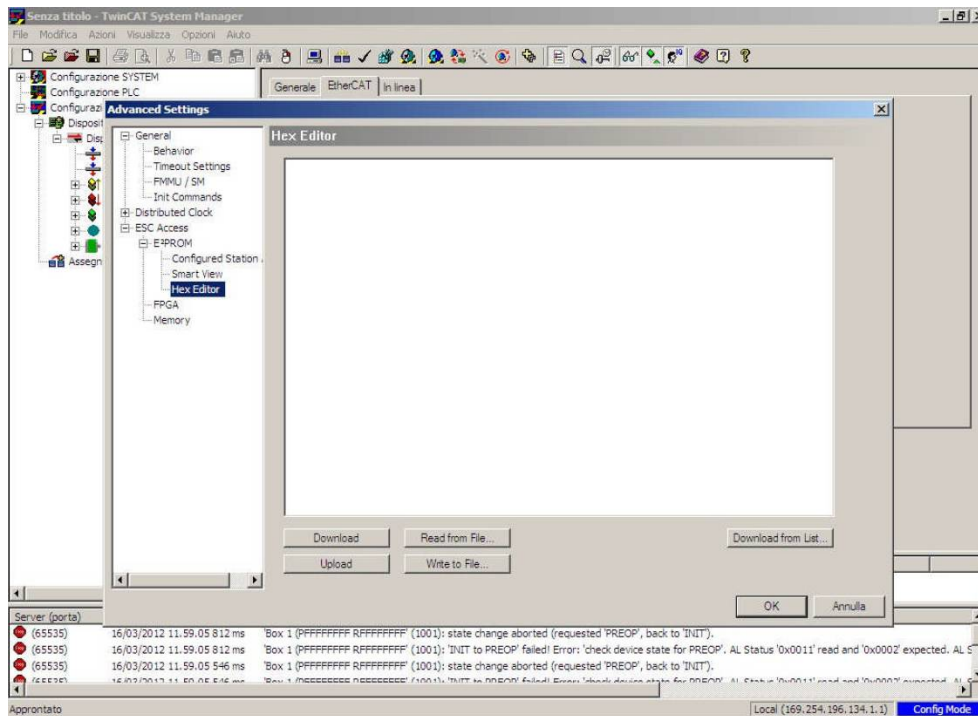
WARNING

It is mandatory that in an EtherCAT network all devices are provided with the same version of the firmware, EEPROM and XML file. So when you need to replace an old encoder installed in your network, then you must either upgrade all the encoders in the network to the last version compatible with the new encoder; or you must downgrade the new encoder to the older version compatible with the encoders already installed in the network.

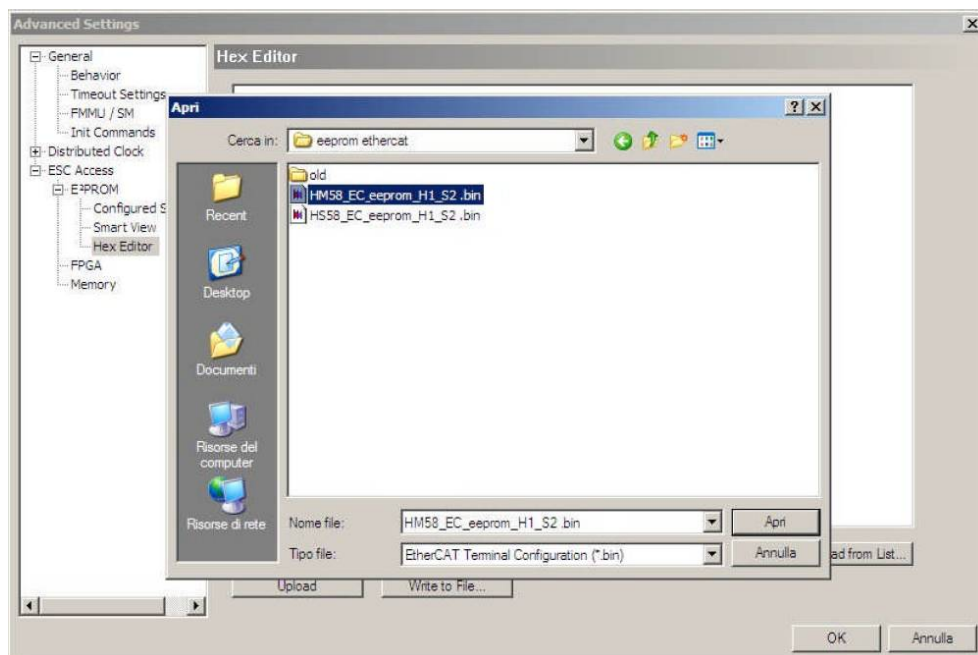
1. In the left pane of the **TwinCAT System Manager** main window press the **Box (Lika EtherCAT EM58, ES58, HM58 or HS58)** item of the encoder you need to update: some tabbed pages for configuring and managing the device will appear in the right pane. Enter the **EtherCAT** page.



- Press the **Advanced Settings...** button; the **Advanced Settings** page will appear; in the directory tree on the left expand the **ESC Access** directory, then expand the **E2PROM** directory, finally select the **HEX Editor** item.



- Press the **Read from File...** button and select the .BIN file provided to upgrade the EEPROM; please make sure you select the file of your own model you need to upgrade (for example: if you have to upgrade an H-series multiturn encoder then you must select the file **HM58_EC_eeeprom_Hx_Sy.bin**); finally press the **Open** button.

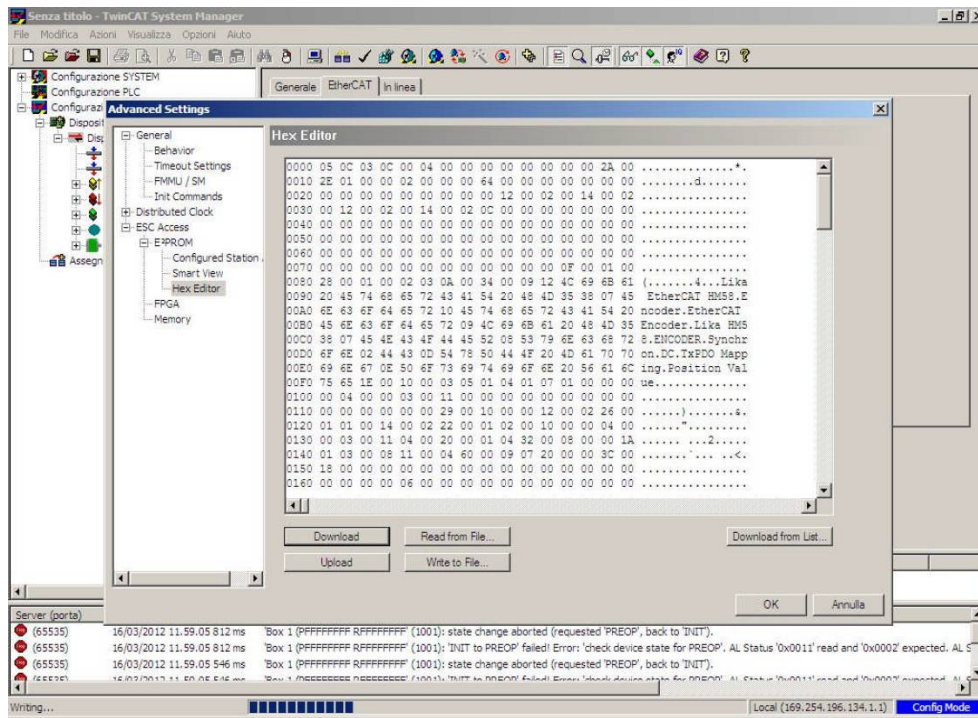




NOTE

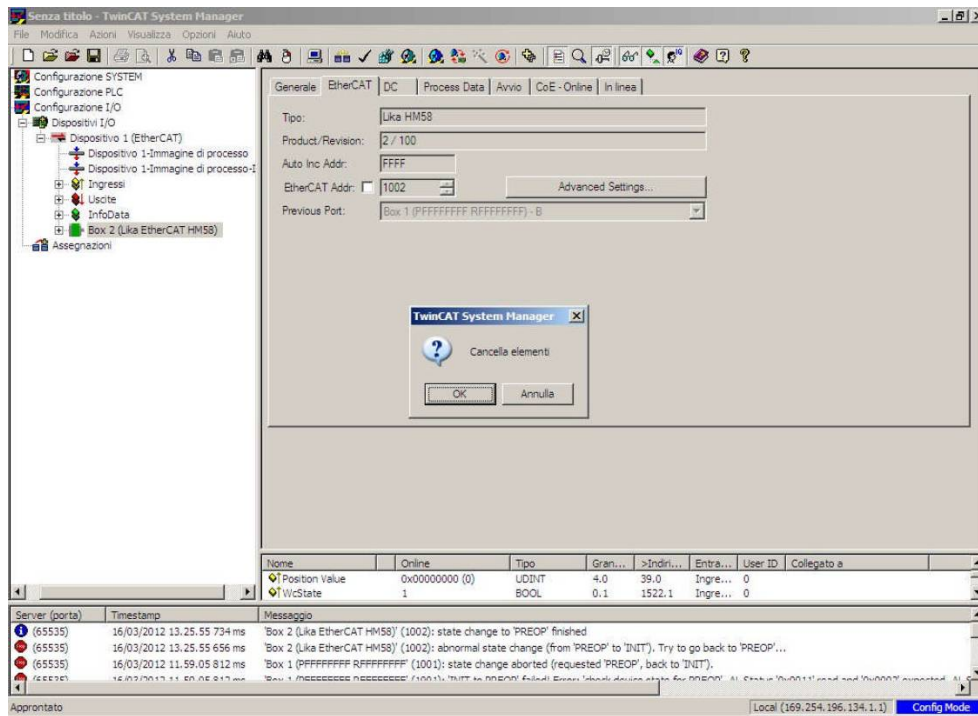
In the .BIN file Hx is the hardware version of the encoder, while Sy is the software version.

4. Move back to the previous **Advanced Settings** page and press the **Download** button. Now wait until the EEPROM writing process is carried out. The progress bar below in the page displays the progress of the operation. As soon as the process is completed press the **OK** button.

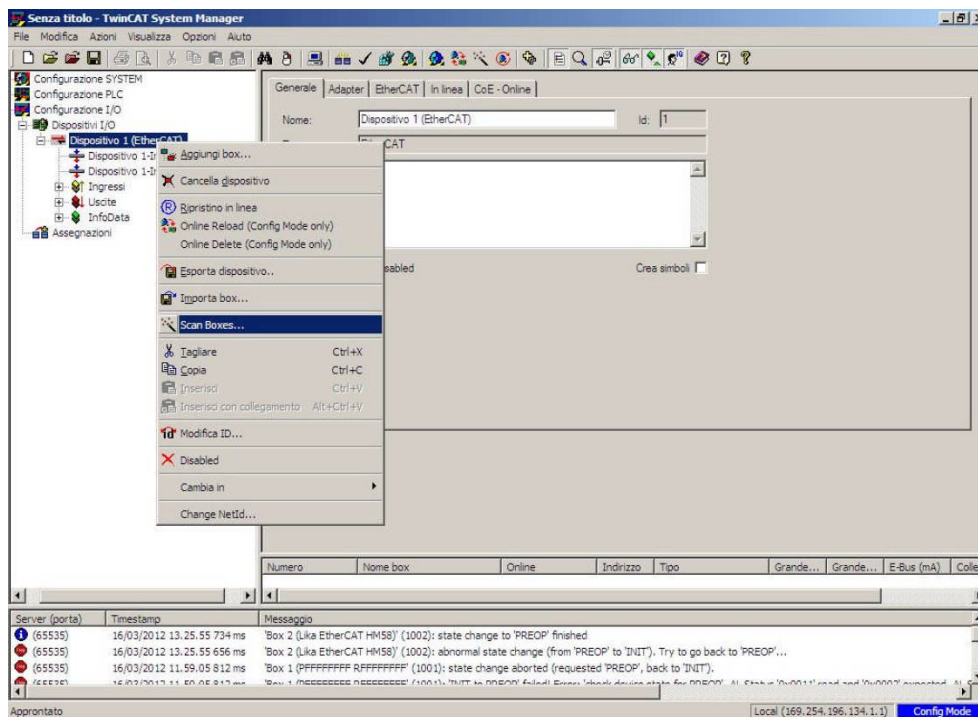


5. Now turn power off, then on again.

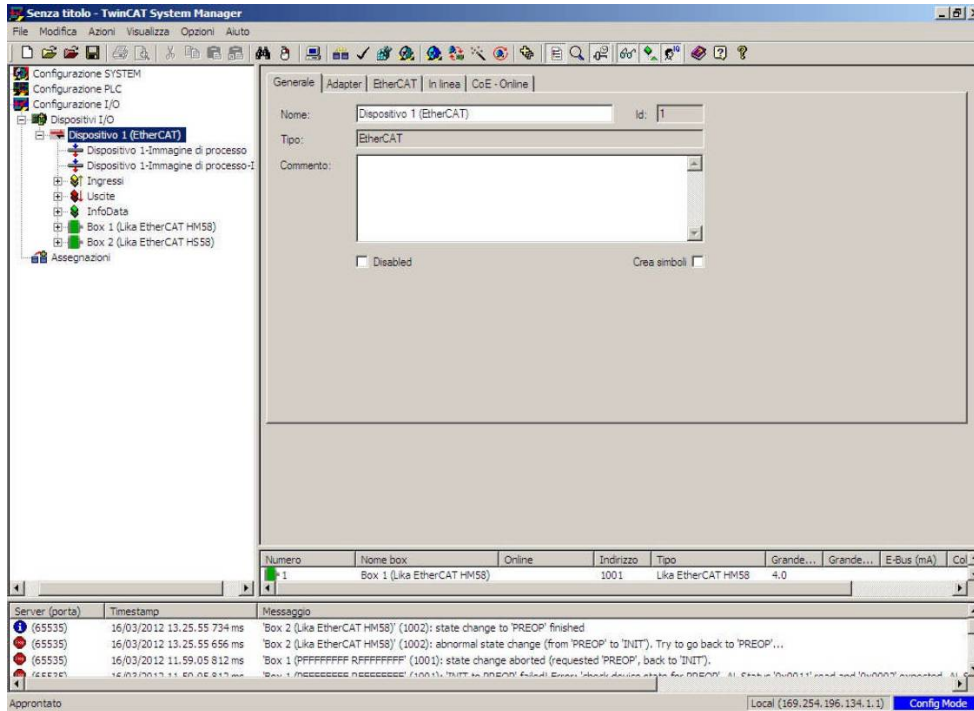
- In the left pane of the **TwinCAT System Manager** main window delete all the **Box (Lika EtherCAT EM58, ES58, HM58 or HS58)** items in the list. Select each box and then press the **DEL** key in the PC keyboard. Press **OK** to confirm.



- In the left pane of the **TwinCAT System Manager** main window select and right-click the **Device 1 (EtherCAT)** item; press the **Scan Boxes...** command in the menu.



- At the end of the scanning process all the devices available in the network are listed as shown in the Figure here below.



5.7 Firmware upgrade



WARNING

Firmware upgrade process has to be accomplished by skilled and competent personnel. If the upgrade is not performed according to the instructions provided or a wrong or incompatible firmware program is installed then the unit may not be updated correctly, in some cases preventing the unit from working.



WARNING

The XML file version, the firmware version and the EEPROM version must always comply. For example: if the firmware version is H1_S4 (Hardware version: 1; Software version: 4), it is mandatory that the EEPROM version is S4, therefore you must then install the XML file version V4.



WARNING

It is mandatory that in an EtherCAT network all devices are provided with the same version of the firmware, EEPROM and XML file. So when you need to replace an old encoder installed in your network, then you must either upgrade all the encoders in the network to the last version compatible with the new encoder; or you must downgrade the new encoder to the older version compatible with the encoders already installed in the network.

Firmware is a software program which controls the functions and operation of a device; the firmware program, sometimes referred to as "user program", is stored in the flash memory integrated inside the unit. Lika encoders are designed so that the firmware can be easily updated by the user himself. This allows Lika Electronic to make new improved firmware programs available during the lifetime of the product.

Typical reasons for the release of new firmware programs are the necessity to make corrections, improve and even add new functionalities to the device.

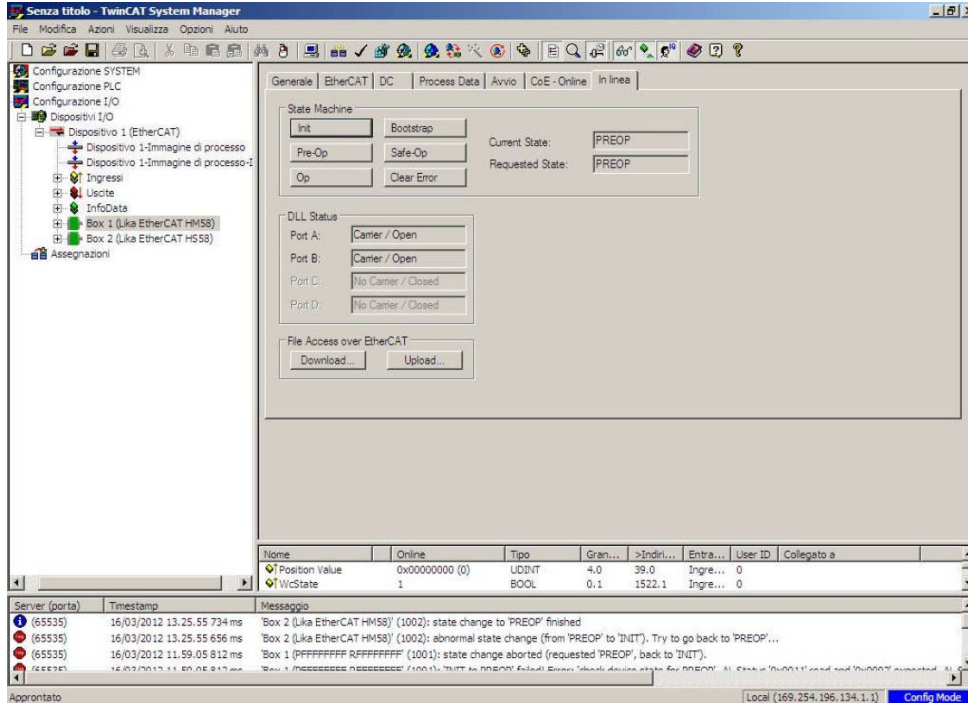
The firmware upgrading program consists of a single file having .EFW extension. It is downloadable from the web or released by Lika Electronic Technical Assistance & After Sale Service.



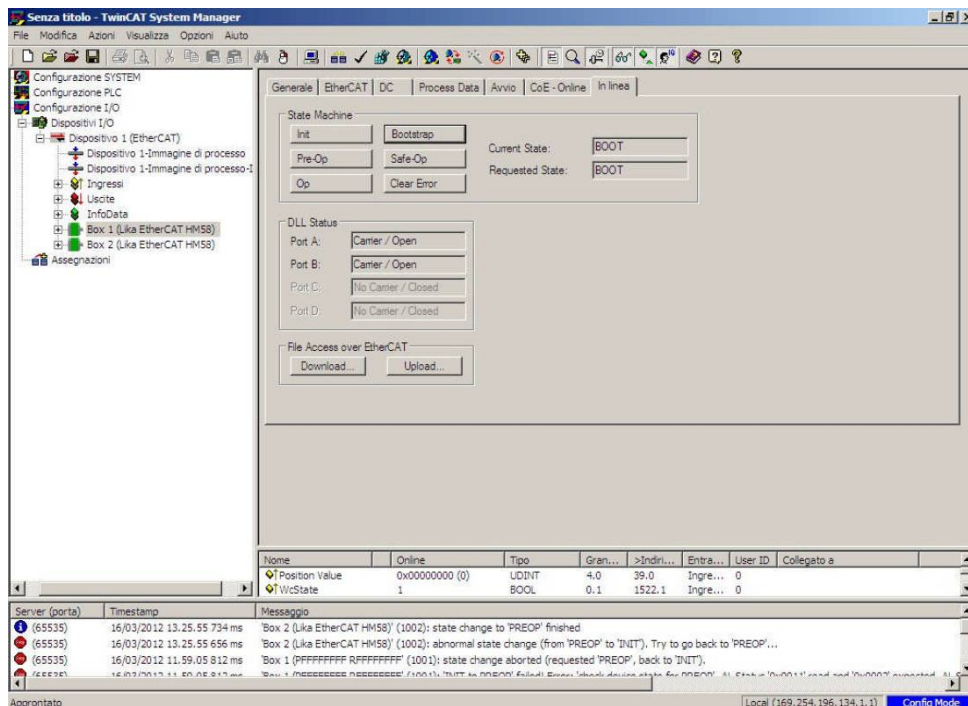
WARNING

You must upgrade the EEPROM before upgrading the firmware.

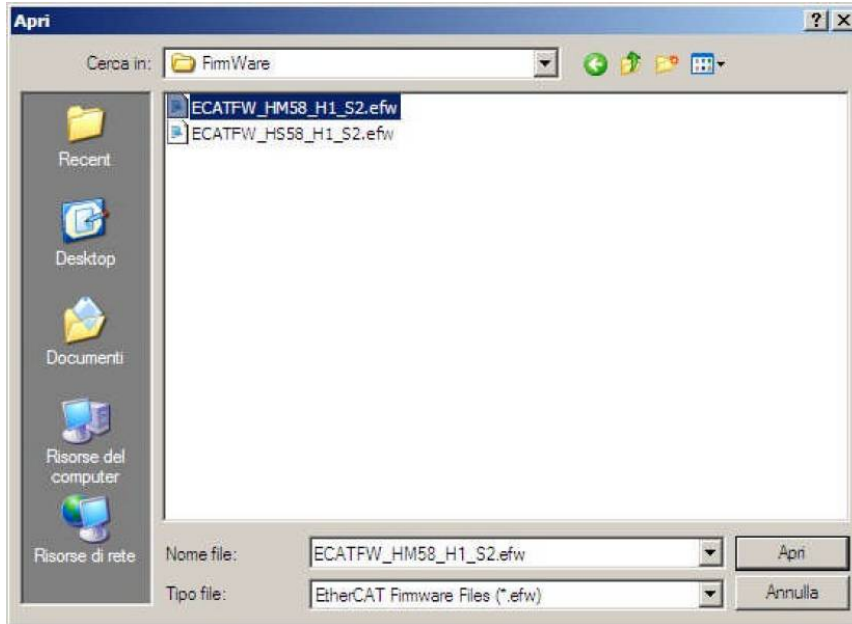
1. In the left pane of the **TwinCAT System Manager** main window press the **Box (Lika EtherCAT EM58, ES58, HM58 or HS58)** item of the encoder you need to update: some tabbed pages for configuring and managing the device will appear in the right pane. Enter the **Online** page.



2. Press the **Bootstrap** button in the **State Machine** box; in the **BOOT** state the encoder is ready to accept the firmware upgrade download process (the **BOOT** message appears next to the **Current State** item in the same box).



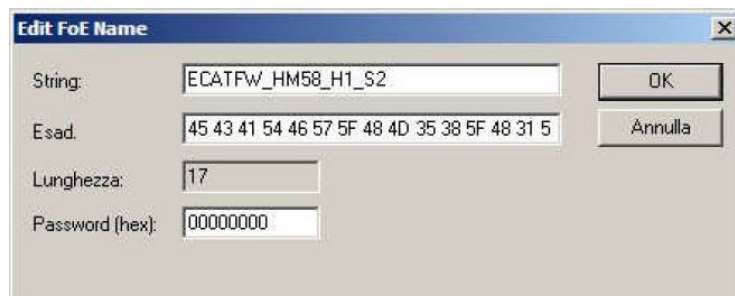
- Now press the **Download...** button in the **File Access Over EtherCAT** box; in the **Open** window that appears select the .EFW file provided to upgrade the firmware; please make sure you select the file of your own model you need to upgrade (for example: if you have to upgrade an H-series multiturn encoder then you must select the file ECATFW_HM58_Hx_Sy.efw); finally press the **Open** button.



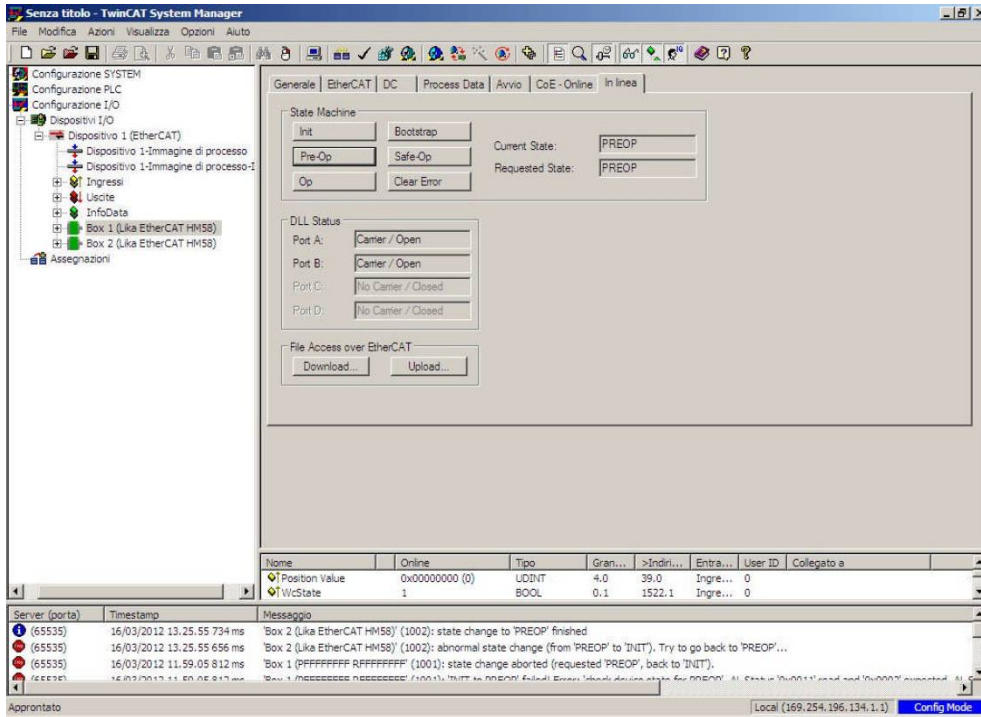
NOTE

In the .EFW file Hx is the hardware version of the encoder, while Sy is the software version.

- In the **Edit FoE Name** page that appears on the screen enter the password 0x00000000 next to the **Password (hex)** item below in the page and then press the **OK** button to confirm. Now wait until the firmware file saving process is carried out. The progress bar below in the page displays the progress of the operation.



- To check whether the firmware upgrade procedure has been completed successfully enter the **Online** page in the **TwinCAT System Manager** main window and press the **Pre-Op** button in the **State Machine** box; if everything is ok, the encoder enters the **PREOPERATIONAL** state (the **PREOP** message appears next to the **Current State** item in the same box).



6 - EtherCAT® interface

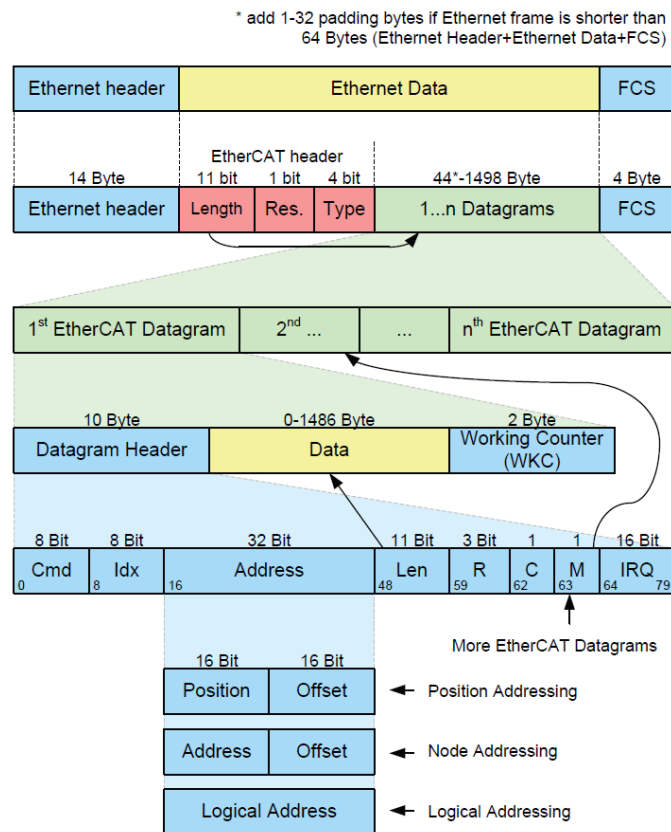
6.1 Basic Information on EtherCAT® Protocol

EtherCAT protocol is designed to use the standard Ethernet dataframes for issuing data; in addition, and as regards the hardware, it is not necessary to install dedicated Masters for establishing and managing the EtherCAT communication because standard Ethernet network cards can be used. This results in an advantage in terms of lower costs and simplicity of use because Ethernet network cards are used in standard personal computers and are easily commercially available.

An EtherCAT bus can be viewed as a single and large Ethernet device that receives and sends Ethernet telegrams; it can be considered an Ethernet subnet supported by an Ethernet dataframes structure.

However this "subnet" must be fitted with one only EtherCAT Master controller and several EtherCAT Slaves, but no Ethernet controller with downstream microprocessor must be present.

Here follows an Ethernet frame structure with EtherCAT:



Inside the Ethernet frames data are transmitted among Master and Slaves using PDO (process data objects) protocol. Each PDO message has inside one or more addresses for issuing data to Slaves; data + address/es (and additional elements such as a validation checksum) joined together forms an EtherCAT telegram (Datagram).

An EtherCAT frame can contain several telegrams and often a complete control cycle requires more than one frame.

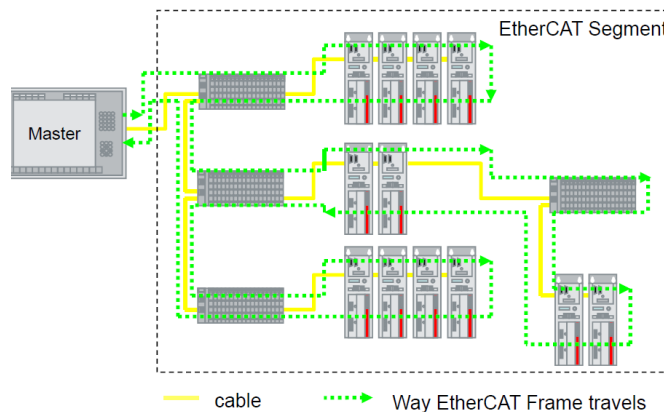
6.1.1 Data transfer

Usually, in a data bus system, Master controller sends online a data request and then waits for data to be processed and sent back from each Slave node; this does not comply with a real-time system because Master receives data from Slaves in different moments and the whole system cannot be synchronized.

In EtherCAT the real-time characteristic of the system is quite improved because data are processed "on-the-fly", using one single frame to acquire all data from all Slaves.

In fact the frame sent by the Master is read by each Slave node data is addressed to while the telegram passes through the device; similarly, input data is inserted while the telegram passes through. Then the telegram is forwarded to the next device. Telegrams are only delayed by a few nanoseconds.

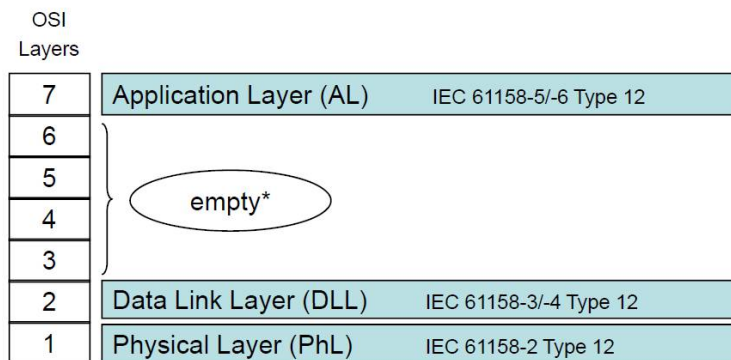
The last Slave issues back the complete frame to the Master with all the requested data (again passing through all the Slaves).



This efficient data flow is guaranteed by the 100BASE-TX full-duplex structure of EtherCAT bus which is fitted with two separated lines for transmitting and receiving data.

Moreover the protocols exchange takes place inside the hardware and it is thus independent from CPU and software processing.

6.1.2 ISO/OSI Layer model



* "empty" means that the layer behaviour exists, but is not shown explicitly.

6.1.3 Topology

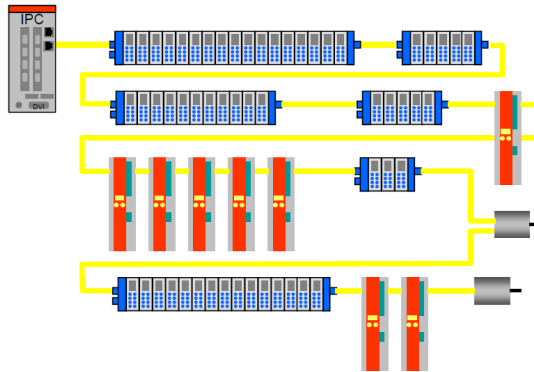
Several topologies of connection are supported by EtherCAT networks: line, tree, daisy-chain, star, ...). Furthermore EtherCAT networks can be configured in almost any topology in the same structure. The maximum length of the cable between two Slaves is 100 m; standard EtherCAT cables commercially available can be used.

The choice of the topology depends on the structural characteristics of the plant and it is made in order to reduce the complexity and time for cabling.

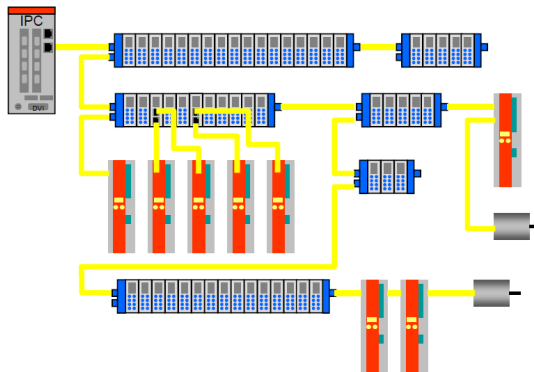
Inside an EtherCAT network up to 65535 devices can be connected.

Some topology examples are shown in the Figures below:

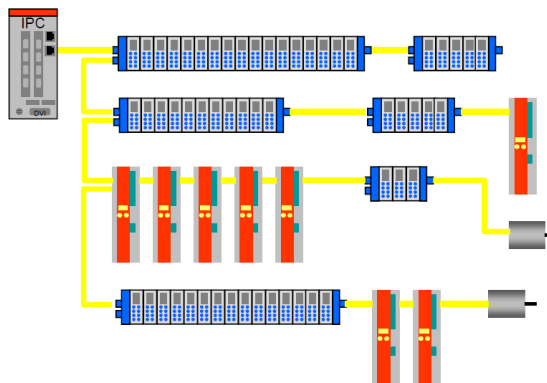
LINE topology:



TREE topology:



DAISY CHAIN with drop lines topology:



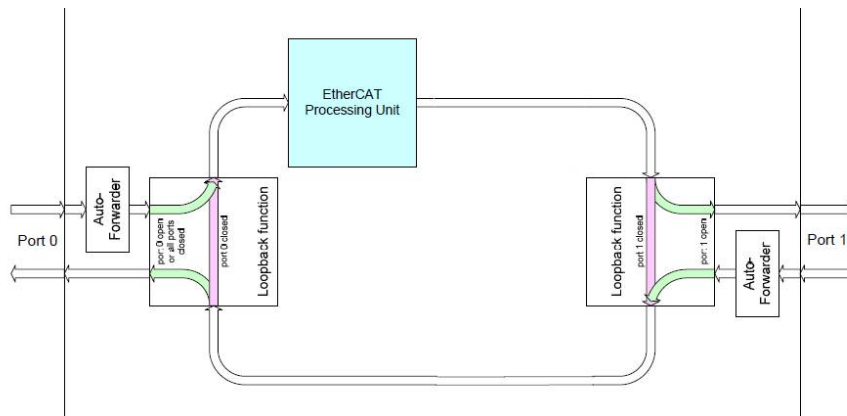
6.1.4 Line Termination

EtherCAT network needs no line termination because the line is terminated automatically; in fact every Slave is able to detect the presence of downstream Slaves.

An EtherCAT Slave is able to detect the presence of the signal in the outgoing line (Port 0) or in the return line (Port 1).

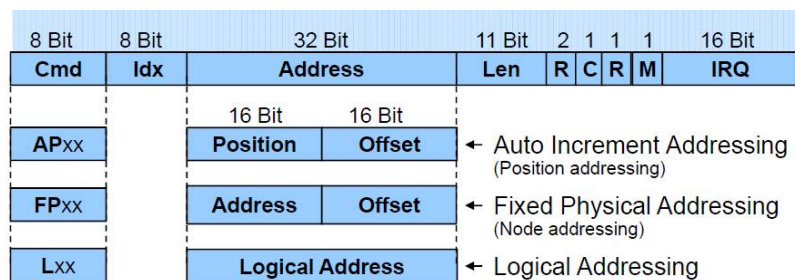
If the Slave is not able to detect the signal in its return line then it closes the communication ring by short-circuiting the TX signal of its outgoing line with the RX signal of its return line; in this way a telegram received through the outgoing line is processed and sent back through the TX of the return line.

The Slave sends a "carrier signal" or a telegram on TX of the outgoing line continuously and, once the next Slave is connected again, a signal on RX of the return line is detected again; so the short circuit is removed and the telegrams are sent on TX of the outgoing line.



6.1.5 Addressing

It is not necessary to assign a physical address to the device (for instance using a dip-switch) because the addressing of the Slave is automatic at power on during the initial scanning of hardware configuration.



The field for addressing is 32 bits long; there are three kinds of addressing:

- Auto Increment Addressing = Position Addressing: 16 bits indicates the physical position of the Slave inside the network while 16 bits are scheduled for local memory addressing; when the Slave receives the frame then it increments the position address and the Slave receiving address 0 is the addressed device;

- Fixed Addressing = 16 bits indicates the physical address of the Slave inside the network while 16 bits are scheduled for addressing the local memory;
- Logical Address = the Slave is not provided with its own individual address, but it can read and write data in a section of the total memory space available (4 Gigabytes).

6.1.6 Communication mode

Lika encoders with EtherCAT interface support the following operating modes:

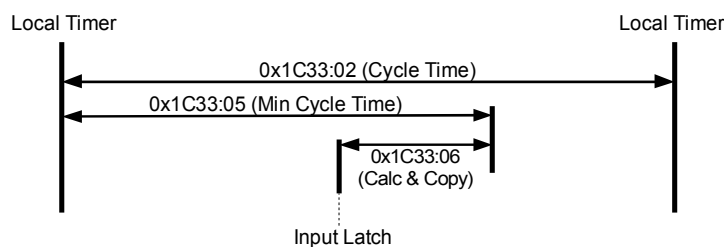
- FreeRun: asynchronous mode;
- SM3 event: synchronous mode;
- DC: distributed clock synchronization mode (synchronous mode).

For a system that requires high performances in real time (closed-loop applications) we suggest using DC mode; if real time requirements are not so mandatory SM3 or Freerun modes can be used instead.

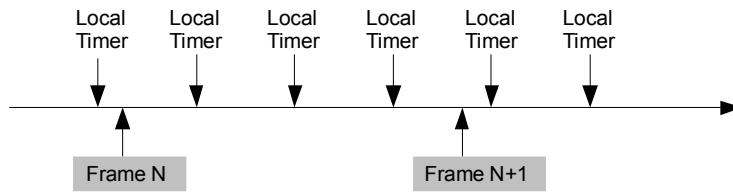
A reference parameter is the "Jitter": it represents the temporal fluctuation of the instant data sampling; in other words data sampled by the micro-controller is available in ECAT DPRAM memory after a certain time and the measure of the variability over time is the "jitter".

FreeRun

Asynchronous mode; encoder position is sampled directly from EtherCAT frame sent by the Master; position update is performed by an internal timer of the controller every 100 microseconds.



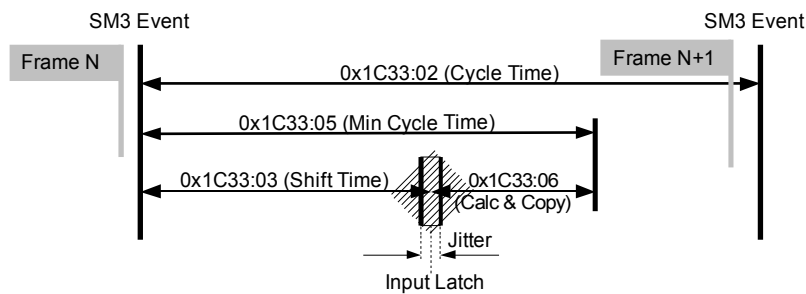
This operating mode has a high sampling jitter (up to 100 microseconds) and can be chosen only when cycle times are quite longer than the jitter if we want to ensure a sufficient real-time system performance.



Description	Min	Typ	Max	
Jitter	0		100	µsec
Cycle Time	1000		64000	µsec

Synchronous with SM3

In this mode data is sampled and then copied into Sync Manager buffer as soon as previous data was read from the Master (SM event); in this way new sampled data is synchronous with Master readings.



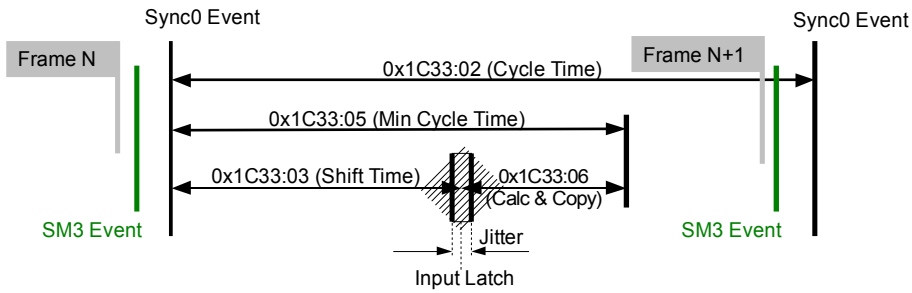
New data will be read by the Master at next cycle (following SM3 event), so if cycle time is too long, data could be relatively old for a real-time system. The main advantage is that data is updated exactly when Master is reading (synchronous mode).

Description	Min	Typ	Max	
Jitter	0		7,2	µsec
Cycle Time	62,5		64000	µsec

Synchronous with DC SYNC0

In this operating mode data is sampled and then copied into Sync Manager buffer simultaneously at SYNC0 event generated by the ESC capture/compare unit.

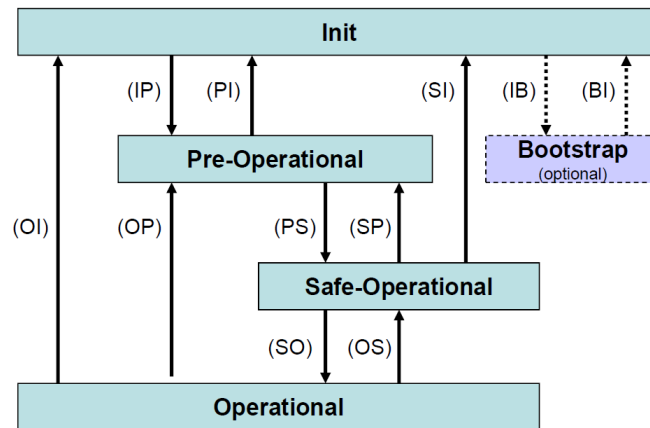
Time required for accomplishing these operations is set in object **1C33 Sync Manager input parameter**; in particular object **03 Shift Time** (1C33hex, sub3) and object **06 Calc and Copy time** (1C33hex, sub6).



In this operating mode "Jitter" is a fundamental parameter in the sampling of two consecutive data. The main advantage of this mode is that there is a direct relation between the sampling instant and the absolute time of the system; in this way, if we know the shift times of the Slaves, we can have an exact image of the system at a given moment (with a tolerance equal to jitter).

Description	Min	Typ	Max	
Jitter	0	100	200	µsec
Cycle Time	62,5		64000	µsec

6.1.7 EtherCAT State Machine (ESM)



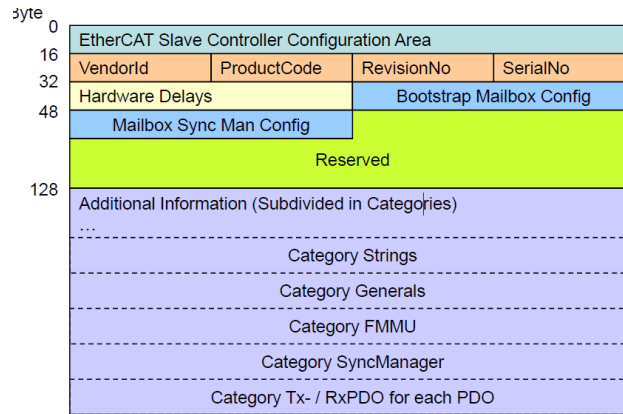
EtherCAT Slave is a state machine; communication and operating characteristics depend on the current state of the device:

- **INIT**: it is the default state after power on; in this state there is not direct communication between Master and Slave on the Application Layer; some configuration registers are initialized and the Sync Managers are configured;
- **PRE-OPERATIONAL**: in this state mailbox is active; both Master and Slave can use the mailbox and its protocols for exchanging specific initialization parameters of the application. Exchange of Process Data (PDO) is forbidden;
- **SAFE-OPERATIONAL**: in this state Master and Slave can issue only input process data, while output process data are still in **SAFE-OPERATIONAL** state;
- **OPERATIONAL**: in this state Master and Slave are enabled to send both input process data and output process data.

6.1.8 Slave configuration

The configuration of the Slave communication characteristics (Sync Manager configuration, addresses, synchronization modes, PDO mapping, ...) can be made both using the XML file (EtherCat Slave Information - ESI) or loading data directly from EEPROM (Slave Information Interface SII).

EEPROM content (SII):



6.1.9 Timing and Synchronization

The main feature of EtherCAT is its almost ideal representation of a real-time system.

Hence the Master has to synchronize all the Slaves at the same time in order to build a system where all nodes have the same reference time; this goal can be achieved by using "distributed clocks".

Master downloads its clock into one of the slaves (customarily the first Slave) which becomes the reference clock for all the Slaves in the network; so it has the task of synchronizing the other Slaves. Master controller periodically sends a special synchronization-telegram where the reference Slave writes its own "current time". This telegram is then sent to all the other Slaves that, in this way, provide for a new re-synchronization of their own clock in order to avoid possible drifts.

This synchronization of the reference time is very important in order to have a snapshot of the system and accordingly to take simultaneous actions in high sensitive applications such as the coordination in axis control operations.

Besides, EtherCAT Slave Controller (ESC) is fitted with a capture/compare unit that provides accurate synchronization signals (SYNCO or interrupts): they are sent to the local micro-controller so that it is able to synchronize its own clock to Slaves clock.

Sync Manager

Sync Manager has the task of synchronizing data transfer between Master and Slave and prevents the same memory area from being written by different events.

There are two synchronization modes:

- 3-Buffer Mode;
- 1-Buffer Mode.

Synchronisation mode is initialized through the XML file or by loading data directly from EEPROM (SII).

Buffered Mode (3-Buffer Mode)

In this mode new data can be accessed at any time by both Master EtherCAT and ESC controllers; no timing restrictions are imposed.

Three buffers are necessary (three consecutive memory areas); one buffer is always available to ESC controller for writing and one buffer always contains updated data to be read by the Master.

Customarily this is used for process data communication.

Mailbox Mode (1-Buffer Mode)

In this mode a "handshake" between Master and Slave must be used; in fact one only memory buffer is available to both Master and Slave for writing and reading; the Master (or the Slave) is enabled to write only when the buffer is empty, that is when the Slave (or the Master) has finished reading the data buffer. And vice versa: the Master (or the Slave) is enabled to read only when the buffer is empty, that is when the Slave (or the Master) has finished writing the data buffer.

6.2 CANopen Over EtherCAT (CoE)

Lika encoders are Slave devices and support "CanOpen Over EtherCAT" (COE) mode for data transfer. In particular, they support the "Device profile for encoders", Class 2.

For any omitted specification on CANopen® protocol, please refer to "CiA Draft Standard 301" and "CiA Draft Standard 406" documents available at the address www.can-cia.org.

For any omitted specification on EtherCAT® protocol, please refer to "ETG.1000 EtherCAT Specification" document available at the address www.ethercat.org.

6.2.1 XML file

EtherCAT® encoders are supplied with their own XML file **Lika_Ex58_Hx58_EC_Vx.xml** (see at www.lika.biz > **ROTARY ENCODERS** > **ABSOLUTE ENCODERS** > **ETHERCAT**). At the same address you can also download the most recent files to update both the firmware and the EEPROM of the encoder. For older versions of the files please contact Lika Electronic's After Sales Service Dept. For any information on the firmware upgrade procedure refer to the section "5.7 Firmware upgrade" on page 35. For any information on the EEPROM upgrade procedure refer to the section "5.6 EEPROM upgrade" on page 30).

XML file has to be installed on EtherCAT® Master device.



WARNING

Before installing the XML file please check that it is compatible with the firmware version and the EEPROM version of the device; the XML file version, the firmware version and the EEPROM version must always comply. For example: if the firmware version is H1_S4 (Hardware version: 1; Software version: 4), it is mandatory that the EEPROM version is S4, therefore you must then install the XML file version V4.



WARNING

It is mandatory that in an EtherCAT network all devices are provided with the same version of the firmware, EEPROM and XML file. So when you need to replace an old encoder installed in your network, then you must either upgrade all the encoders in the network to the last version compatible with the new encoder; or you must downgrade the new encoder to the older version compatible with the encoders already installed in the network.



WARNING

H- series has been implemented starting from version V1.
E- series has been implemented starting from version V4.

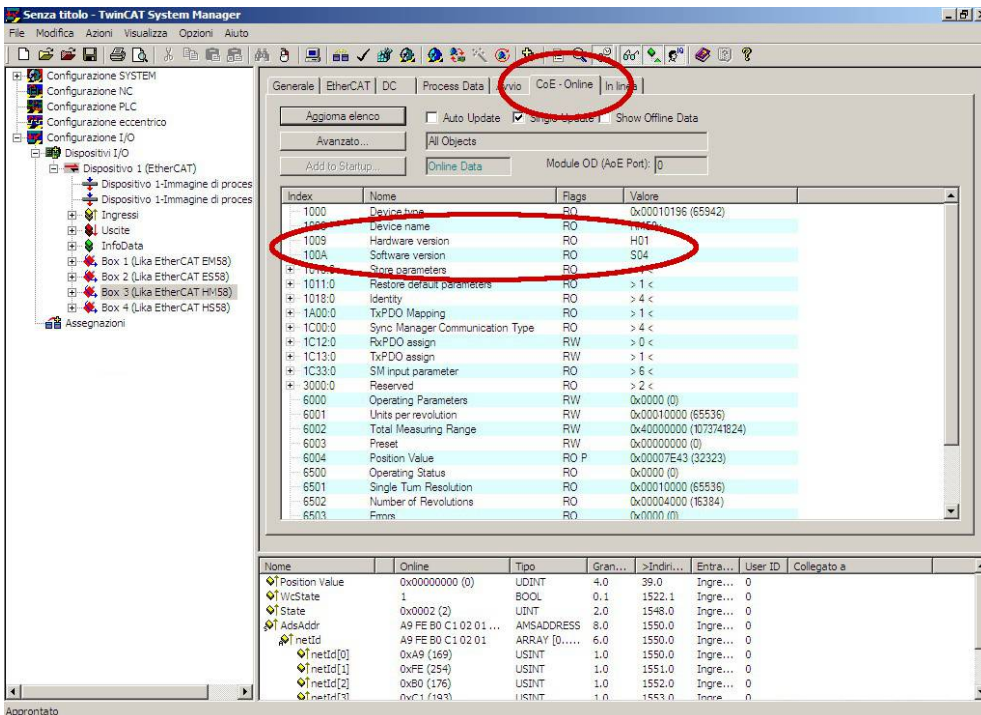


WARNING

In the H- series all versions are compatible for upgrade and downgrade except version V1.

In the E- series all versions are compatible for upgrade and downgrade starting from version V4.

If you want to know the firmware version of a device, press the **Box (Lika EtherCAT EM58, ES58, HM58 or HS58)** item in the left pane of the **TwinCAT System Manager** main window: some tabbed pages for configuring and managing the device will appear in the right pane. Enter the **CoE - Online** page and refer to the indexes Hardware version and Software version.



6.2.2 Communication messages

EtherCAT Datagram of CoE mode has the following structure:

Mbx Header	CoE Cmd			Cmd specific data
type = 3	Number	res	Type	
6 bytes	9 bits	3 bits	3 bits	0 ... 1478 bytes

Mbx Header = 3 CoE mode

Number = 0 in case of SDO messages

≠ 0 in case of PDO messages, it defines the type of service

- res reserved bits
- Type = 0 reserved
- = 1 Emergency message
- = 2 SDO request
- = 3 SDO response
- = 4 Transmitted PDO (TxPDO)
- = 5 Received PDO (RxPDO)
- = 6 Remote transmission request of TxPDO
- = 7 Remote transmission request of RxPDO
- = 8 SDO information
- = 9 ... 15 reserved

Cmd specific data PDO messages: are the process data, e.g. position value
 SDO messages: standard CANopen frame

Transmit (tx) or receive (rx) "Type" is viewed from Slave.

6.2.3 Process Data Objects (PDO)

PDO messages are used for transmitting or receiving process data in real time; data to be transmitted or received is defined in PDO Mapping and managed by Sync Manager PDO Mapping.

6.2.4 Service Data Objects (SDO)

SDO messages are issued via Mailbox (low priority data); Segmented SDO Service and SDO Complete Access are not supported (transfer of low size data and one sub-index at a time).

"CoE Cmd type" = 2 or 3

Structure of "Cmd specific data":

Cmd specific data				
SDO control	Index	Sub index	Data	Data optional
8 bits	16 bits	8 bits	32 bits	1 ... 1470 bytes

- SDO control standard CANopen SDO Service
- Index parameter index
- Sub index parameter sub-index
- Data parameter value
- Data optional optionally, more then 4 bytes of data can be sent in one frame. Full mailbox size usable.

Index and sub-index values are described in the "Object dictionary".

6.2.5 Object dictionary

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

Index-subindex Object name

[data types, attribute]

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

Unsigned16 data type:

Process data bytes	
byte 4	byte 5
LSByte	MSByte

Unsigned32 data type:

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

Standard objects

1000-00 Device type

[Unsigned32, ro]

Default = 0001 0196h = single turn encoder, DS 406
0002 0196h = multi turn encoder, DS 406

1008-00 Manufacturer Device Name

[String, ro]

It shows the manufacturer device name, expressed in ASCII code.

Default = 4553353878 = "ES58x" = single turn encoder E- series
454D353878 = "EM58x" = multi turn encoder E- series
4853353878 = "HS58x" = single turn encoder H- series
484D353878 = "HM58x" = multi turn encoder H- series

1009-00 Hardware version

[String, ro]

It shows the hardware version of the device.

100A-00 Software version

[String, ro]

It shows the software version of the device.

1010-01 Store parameters

[Unsigned32, rw]

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

Write "save" in ASCII code in the data bytes:

Master → Encoder

Cmd specific data							
Cmd	Index		Sub	Data			
23	10	10	01	73	61	76	65

Encoder → Master (confirmation)

Cmd specific data							
Cmd	Index		Sub	Data			
60	10	10	01	00	00	00	00

1011-01 Restore default parameters

[Unsigned32, rw]

This object allows you to restore all parameters to default values. The 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. A list of machine data and relevant default parameters preset by Lika Electronic engineers is available on page 64.

Write "load" in ASCII code in the data bytes:

Master → Encoder

Cmd specific data							
Cmd	Index		Sub	Data			
23	11	10	01	6C	6F	61	64

Encoder → Master (confirmation)

Cmd specific data							
Cmd	Index		Sub	Data			
60	11	10	01	00	00	00	00



NOTE

To save the default values execute the "Store parameters" function (see object **1010-01 Store parameters**). When the power is turned off, parameters not saved are lost.

1018 Identification object

01 Vendor number

[Unsigned32, ro]
Default = 0000 012Eh

02 Product number

[Unsigned32, ro]

03 Revision number

[Unsigned32, ro]

04 Serial number

[Unsigned32, ro]

1A00-01 PDO mapping parameter

[Unsigned32, rw]

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

Default = 6004 0020h

1C00 Sync Manager Communication Type

01 SM MailBox Receive (SM0)

[Unsigned8, ro]
Default = 1

02 SM MailBox Send (SM1)

[Unsigned8, ro]
Default = 2

03 SM PDO output (SM2)

[Unsigned8, ro]
Default = 3

04 SM PDO input (SM3)

[Unsigned8, ro]
Default = 4

1C12-00 Sync Manager RxPDO Assigned

[Unsigned8, ro]

This device does not support receive PDO messages.

Default = 0

1C13-01 Sync Manager TxPDO Assigned

[Unsigned32, ro]

This device uses TxPDO messages to send the position value.

Default = 0000 1A00hex

1C33 Sync Manager input parameter

Some sub-indexes of object 1C33hex are calculated dynamically and depend on both the encoder configuration (programmed resolution, counting direction, ...) and the synchronization mode (SM or DC).

01 Sync Type

[Unsigned16, rw]

Default = 1

0: FreeRun; see on page 43;

1: Synchronous with SM3 Event; see on page 44;

2: DC mode synchronous with SYNC0 event; see on page 45.

02 Cycle time

[Unsigned32, ro]

this parameter depends on the Sync Type selected:

if "FreeRun": interval between two position samplings (internal timer);

if "Synchronized with SM3": min. interval between two SM3 events;

if "Synchronized with DC Sync0": Sync0 cycle time.

03 Shift Time

[Unsigned32, ro]

Interval between the synchronization event and the moment of inputs latching from hardware. This parameter is calculated dynamically.

04 Sync modes supported

[Unsigned16, ro]

Default = 7

bit 0: FreeRun (supported)

bit 1: Synchronous with SM3 (supported)

bit 2: Synchronous with DC SYNC0 (supported)

05 Minimum cycle time

[Unsigned32, ro]

Max. duration of the encoder internal cycle time. This parameter is calculated dynamically and depends on the operating parameters and the position value.

06 Calc and Copy time

[Unsigned32, ro]

Time the internal micro-controller (DSP) needs to make calculations on latched optical reading of position and then copy updated data from local memory to ESC memory (Sync Manager) before they are available to EtherCAT. This parameter is calculated dynamically and depends on the operating parameters and the position value.

Device profile objects

6000-00 Operating parameters

[Unsigned16, rw]

Bit	Function	bit = 0	bit = 1
0	Code sequence	CW (clockwise)	CCW (counter clockwise)
1	not used		
2	Scaling function	disabled	enabled
3 ... 15	not used		

Default = 0000h

Code sequence

This is intended to set if the encoder counting increments either when the shaft is rotating clockwise (CW) or when the shaft is rotating counter-clockwise (CCW). CW and CCW rotations are viewed from shaft end. Setting 0 (bit 0 = 0) causes the encoder counting to increment when the shaft is rotating clockwise; setting 1 (bit 0 = 1) causes the encoder counting to increment when the shaft is rotating counter-clockwise.

Scaling function

This is meant to enable / disable the scaling parameters **6001-00 Counts per revolution** and **6002-00 Total resolution**. When the scaling function is disabled (bit 2 = 0), the encoder uses its own physical resolution (i.e. hardware counts per revolution and number of hardware revolutions, see the encoder identification label and the objects **6501-00 Hardware counts per revolution** and **6502-00 Hardware number of turns**); otherwise, when the scaling function is enabled (bit 2 = 1), the encoder uses the resolution set next to the parameters **6001-00 Counts per revolution** and **6002-00 Total resolution** complying with the following relation:

$$\text{Transmitted position} = \frac{\text{6001-00 Counts per revolution}}{\text{6501-00 Hardware counts per revolution}} \times \text{Real position} \leq \text{6002-00 Total resolution}$$

6001-00 Counts per revolution

[Unsigned32, rw]

This object sets the number of distinguishable steps per revolution.

This value is considered only if the bit 2 **Scaling function** in the **6000-00 Operating parameters** item is set to "1".

To avoid counting errors, check that

6501-00 Hardware counts per revolution

6001-00 Counts per revolution

is an integer value.

Furthermore, after having set a new value next to the **6001-00 Counts per revolution** parameter, make sure to meet also the following condition:

$$\frac{\text{6002-00 Total resolution}}{\text{6001-00 Counts per revolution}} \leq \text{Number of physical revolutions}$$

You are allowed to set whatever integer value lower than or equal to the **maximum number of physical steps per revolution** (see the hardware counts per revolution in the encoder identification label and the object **6501-00 Hardware counts per revolution**).

Default = 0000 2000h (8192) for EMxx
 0040 0000h (262144) for HSxx
 0001 0000h (65536) for HMxx



WARNING

Every time you change the value in this object then you are required to set a new preset value (see object **6003-00 Preset value**)

6002-00 Total resolution

[Unsigned32, rw]

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

This value is considered only if the bit 2 **Scaling function** in the **6000-00 Operating parameters** item is set to "=1".

This values results from:

6001-00 Counts per revolution * number of desired revolutions.

You are allowed to set whatever integer value lower than or equal to the **overall hardware resolution** (see the encoder identification label).

The overall hardware resolution results from:

6501-00 Hardware counts per revolution * 6502-00 Hardware number of turns.

Default = 0800 0000h (134217728) for EMxx
 0040 0000h (262144) for HSxx
 4000 0000h (1073741824) for HMxx



Example

Multi turn encoder HM5816/16384EC-6.

Resolution is:

- **Physical singleturn resolution:** 6501-00 Hardware counts per revolution = $65536 (2^{16})$
- **Physical multiturn resolution:** 6502-00 Hardware number of turns = 16384 giri (2^{14})
- **Total hardware resolution:** 6501-00 Hardware counts per revolution * 6502-00 Hardware number of turns = $1073741824 (2^{30})$

2048 counts/rev. * 1024 turns are required:

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



NOTE

To avoid counting errors we recommend values which are power of 2 (2^n : 2, 4, ..., 2048, 4096, 8192,...) to be set next to the objects 6001-00 Counts per revolution and 6002-00 Total resolution.



WARNING

If you have set the preset, when you change the value next to 6001-00 Counts per revolution and/or 6002-00 Total resolution parameters, then you must check the value in the 6003-00 Preset value parameter and perform the preset operation.

6003-00 Preset value

[Unsigned32, rw]

This object allows to set the encoder position to a Preset value. Preset function is meant to assign a certain value to a desired physical position of the encoder. 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. For instance, this can be useful for getting the zero point of the encoder and the zero point of the application to match.

Default = 0000 0000h



WARNING

Check the value in the 6003-00 Preset value parameter and perform the preset operation every time you change the value next to 6001-00 Counts per revolution and/or 6002-00 Total resolution parameters.



NOTE

- If the scaling function is disabled (bit 2 **Scaling function** in the parameter **6000-00 Operating parameters** = 0), **6003-00 Preset value** must be lower than or equal to the **total hardware resolution** (i.e. **6501-00 Hardware counts per revolution** * **6502-00 Hardware number of turns**).
- If the scaling function is enabled (bit 2 **Scaling function** in the parameter **6000-00 Operating parameters** = 1), **6003-00 Preset value** must be lower than or equal to **6002-00 Total resolution**.

Set Preset value **6003-00 Preset value** (preset = 1000 = 3E8h)

Master → Encoder

Cmd specific data							
Cmd	Index		Sub	Data			
23	03	60	00	E8	03	00	00

Encoder → Master (Set confirmation)

Cmd specific data							
Cmd	Index		Sub	Data			
60	03	60	00	00	00	00	00

6004-00 Position value

[Unsigned32, ro]

This object contains the position value of the encoder, transmitted value is calculated considering the scaling values, see bit 2 **Scaling function** of the object **6000-00 Operating parameters**.

6500-00 Operating status

[Unsigned16, ro]

Bit	Function	bit = 0	bit = 1
0	Code sequence	CW Clockwise	CCW Counter clockwise
1	not used		
2	Scaling function	Disabled	Enabled
3 ... 15	not used		

Code sequence

It shows the currently set value that is entered through the bit 0 **Code sequence** in the parameter **6000-00 Operating parameters**. If the bit is "0" the output encoder position value has been set to increment when the shaft rotates clockwise; if the bit is "1" instead the output encoder position value has been set to increment when the shaft rotates counter-clockwise. For any

further information on setting and using the counting direction function refer to the **6000-00 Operating parameters** parameter on page 56.

Scaling function

It shows the currently set value that is entered through the bit 2 **Scaling function** in the parameter **6000-00 Operating parameters**. In other words, it is intended to show whether the scaling function is enabled or disabled. If the value is "=0" the scaling function is disabled; if the value is "=1" instead the scaling function is enabled. For any further information on setting and using the scaling function refer to the **6000-00 Operating parameters** parameter on page 56.

6501-00 Hardware counts per revolution

[Unsigned32, ro]

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

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

6502-00 Hardware number of turns

[Unsigned32, ro]

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

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

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

6503-00 Errors

[Unsigned16, ro]

The corresponding bits of supported errors are set (see object **6504-00 Supported errors**).

6504-00 Supported errors

[Unsigned16, ro]

Default = 0000h (No errors supported).

6505-00 Warnings

[Unsigned16, ro]

Bits of supported warnings are set (see object 6506-00 Supported warnings).

6506-00 Supported warnings

[Unsigned16, ro]

bit 12: wrong parameters loaded from flash memory at power on.

Default = 1000h

6509-00 Offset value

[Unsigned32, ro]

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

**NOTE**

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

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

6.2.6 SDO Abort codes

SDO transfer could be unsuccessful; causes of error are listed and described in the SDO Abort Codes (see ETG1000.6 par. 5.6.2.7.2 table 40).

6.2.7 Emergency Error Codes

Emergency Service is used by server for transmitting diagnostic messages to the client using MailBox; Error Codes are listed and described in the ETG1000.6 par. 5.6.4.2 table 50.

Error Code		Error Register	Diagnostic Data				
Byte (0)	Byte (1)	Byte (2)	Byte (3)	Byte (4)	Byte (5)	Byte (6)	Byte (7)

Error Code	State Transition Errors of state machine: (for detailed description see ETG1000.6 par. 5.6.4.3) A000hex: transition error from PRE-OPERATIONAL to SAFE-OPERATIONAL A001hex: transition error from SAFE-OPERATIONAL to OPERATIONAL Encoder errors: 5000hex: Hardware error 5001hex: Diagnostic data (wrong parameters loaded from flash memory)
Error Register	EtherCAT state machine current status (ESM)
Diagnostic Data	information about possible error causes (see ETG1000.6 par. 5.6.4.3.2-5).

6.2.8 AL Status Error Codes

If the state transition requested by the Master through the "AL Control Register" is unsuccessful, Slave sets to 1 the "Error Indicator Bit" in "AL Status Register" and writes the cause of the error in "AL Status Code Register". Values and descriptions of "AL Status Code" are available in ETG1000.6 par.5.3.2 Table 11.

5.3 File Over EtherCAT (FoE)

Lika encoders are devices that allow the firmware update using the protocol "File over EtherCAT (FoE)".

For any specification on FoE protocol, please refer to "ETG.1000 EtherCAT Specification" document available at the address www.ethercat.org.

7 – Default parameters list

Default values are expressed in hexadecimal notation.

Parameters list	Default values		
6000-00 Operating parameters	0000		
6001-00 Counts per revolution	0000 2000h (8192) for EMxx 0040 0000h (262144) for HSxx 0001 0000h (65536) for HMxx		
6002-00 Total resolution	0800 0000h (134217728) for EMxx 0040 0000h (262144) for HSxx 4000 0000h (1073741824) for HMxx		
6003-00 Preset value	00000 0000		


This page intentionally left blank

This page intentionally left blank

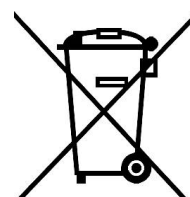
This page intentionally left blank



Document release	Description
1.0	1 st issue
1.1	General review
1.2	Section 5.3 added
1.3	Section 3 added, EEPROM and firmware upgrade instructions, complete review, Italian / English separate editions

	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 product datasheet for supply voltage rate.
---	---

Dispose separately



 **lika**

LIKA Electronic

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

Tel. +39 0445 806600

Fax +39 0445 806699

Italy: eMail info@lika.it - www.lika.it

World: eMail info@lika.biz - www.lika.biz