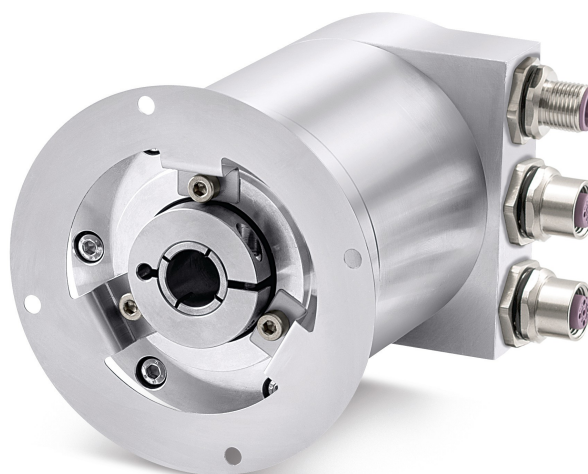


XAC81 EC

ATEX certification category 3
for use in zones 2, 22
and temperature class T5 (T100°C)



EtherCAT 



II 3GD, Ex nA IIB T5 Gc
II 3GD, Ex tc IIIC T100°C Dc, IP65

- 18-bit singleturn encoder version
- 27-bit & 30-bit multiturn encoder versions
- Complies with ETG.1000 specifications
- Implements CoE & FoE protocols and the EtherCAT State Machine

Suitable for the following models:

- XAC81-18-00-EC2-...
- XAC81-13-14-EC2-...
- XAC81-16-14-EC2-...

General Contents

1 - Safety summary	19
2 - Identification	22
3 - Certificates	23
4 - Mechanical installation	28
5 - Electrical connections	31
6 - Getting started	35
7 - Quick reference with TwinCAT	37
8 - EtherCAT® interface	58
9 - Default parameters list	91

This publication was produced by Lika Electronic s.r.l. 2024. 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.

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

General contents

User's guide.....	1
General contents.....	3
Subject Index.....	6
Typographic and iconographic conventions.....	7
Preliminary information.....	8
Glossary of EtherCAT terms.....	9
List of abbreviations.....	16
References.....	18
1 – Safety summary.....	19
1.1 Safety.....	19
1.2 Electrical safety.....	19
1.3 Mechanical safety.....	20
1.4 Operational safety.....	20
2 – Identification.....	22
3 – Certificates.....	23
3.1 Declaration of ATEX Conformity.....	23
3.2 EU Declaration of Conformity.....	25
3.3 Safety instructions.....	26
4 – Mechanical installation.....	28
4.1 Encumbrance sizes.....	28
4.2 Mounting the encoder.....	29
4.3 Solid shaft (LKM-1758) and Fixing plate (LKM-1520).....	30
5 – Electrical connections.....	31
5.1 PWR Power supply connector (Figure 1).....	32
5.2 P1 Port 1 and P2 Port 2 connectors (Figure 1).....	32
5.3 Network configuration: topologies, cables, hubs, switches - Recommendations.....	32
5.4 Addressing.....	33
5.5 Line Termination.....	33
5.6 Ground connection (Figure 1).....	34
5.7 EtherCAT states.....	34
6 – Getting started.....	35
6.1 General installation.....	35
6.2 Setting the scaling function and custom resolution.....	35
6.3 Reading the absolute position.....	36
6.4 Setting and executing the preset.....	36
6.5 Saving data.....	36
6.6 Restoring defaults.....	36
7 – Quick reference with TwinCAT.....	37
7.1 System configuration using TwinCAT software system from Beckhoff.....	37
7.1.1 Setting the Network Card.....	37
7.1.2 Add new I/O modules (Boxes).....	40
7.2 Setting the communication mode.....	44
7.2.1 Synchronous with SM3.....	44
7.2.2 Synchronous with DC (SYNCO).....	45
7.3 Process Data Objects.....	46
7.4 CoE Object Dictionary.....	47

7.5 Online Data.....	48
7.6 EEPROM upgrade.....	49
7.7 Firmware upgrade.....	54
8 – EtherCAT® interface	58
8.1 Basic Information on EtherCAT® protocol.....	58
8.1.1 Data transfer.....	59
8.1.2 ISO/OSI Layer model.....	60
8.1.3 Topology.....	60
8.1.4 Line Termination.....	61
8.1.5 Addressing.....	62
8.1.6 Communication mode.....	63
FreeRun.....	63
Synchronous with SM3.....	64
Synchronous with DC SYNC0.....	64
8.1.7 EtherCAT State Machine (ESM).....	65
8.1.8 Slave configuration.....	66
8.1.9 Timing and Synchronization.....	66
Sync Manager.....	66
Buffered Mode (3-Buffer Mode).....	67
Mailbox Mode (1-Buffer Mode).....	67
8.2 CANopen Over EtherCAT (CoE).....	68
8.2.1 XML file.....	68
8.2.2 Communication messages.....	69
8.2.3 Process Data Objects (PDO).....	70
8.2.4 Service Data Objects (SDO).....	70
8.2.5 Object dictionary.....	71
Communication Profile Area objects (DS301).....	73
1000-00 Device type	73
1008-00 Device Name	73
1009-00 Hardware version	73
100A-00 Software version	73
1010-01 Store parameters	74
1011-01 Restore default parameters	74
1018 Identity	75
01 Vendor ID.....	75
02 Product code.....	75
03 Revision.....	75
04 Serial number.....	75
1A00-01 PDO mapping parameter	76
01 Mapped Object 001.....	76
1C00 Sync Manager Communication Type	76
01 SM MailBox Receive (SM0).....	76
02 SM MailBox Send (SM1).....	76
03 SM PDO output (SM2).....	76
04 SM PDO input (SM3).....	76
1C12-00 Sync Manager RxPDO Assigned	77
1C13-01 Sync Manager TxPDO Assigned	77
01 Subindex 001.....	77
1C33 Sync Manager input parameter	77
01 Sync Type.....	77
02 Cycle time.....	77

03 Shift Time.....	77
04 Sync modes supported.....	78
05 Minimum cycle time.....	78
06 Calc and Copy time.....	78
Standardised Profile Area objects (DS406).....	79
6000-00 Operating parameters	79
Code sequence.....	79
Scaling function.....	79
6001-00 Units per revolution	80
6002-00 Total Measuring Range	81
6003-00 Preset	83
6004-00 Position value	85
6500-00 Operating status	85
Code sequence.....	85
Scaling function.....	86
6501-00 Hardware counts per revolution	86
6502-00 Hardware number of turns	86
6503-00 Errors	87
6504-00 Supported errors	87
6505-00 Warnings	87
6506-00 Supported warnings	87
6509-00 Offset	87
8.2.6 SDO Abort codes.....	88
8.2.7 Emergency Error Codes.....	89
8.2.8 AL Status Error Codes.....	89
8.3 File Over EtherCAT (FoE).....	90
9 – Default parameters list	91

Subject Index

1

1000-00 Device type.....	73
1008-00 Device Name.....	73
1009-00 Hardware version.....	73
100A-00 Software version.....	73
1010-01 Store parameters.....	74
1011-01 Restore default parameters.....	74
1018 Identity.....	75
1A00-01 PDO mapping parameter.....	76
1C00 Sync Manager Communication Type.....	76
1C12-00 Sync Manager RxPDO Assigned.....	77
1C13-01 Sync Manager TxPDO Assigned.....	77
1C33 Sync Manager input parameter.....	77

6

6000-00 Operating parameters.....	79
6001-00 Units per revolution.....	80
6002-00 Total Measuring Range.....	81
6003-00 Preset.....	83
6004-00 Position value.....	85
6500-00 Operating status.....	85
6501-00 Hardware counts per revolution.....	86
6502-00 Hardware number of turns.....	86
6503-00 Errors.....	87
6504-00 Supported errors.....	87
6505-00 Warnings.....	87
6506-00 Supported warnings.....	87
6509-00 Offset.....	87

B

BOOTSTRAP.....	34, 65
----------------	--------

C

Calc and Copy time.....	78
Code sequence.....	79, 85
Cycle time.....	77

D

Diagnostic data.....	89
----------------------	----

H

Hardware error.....	89
---------------------	----

I

INIT.....	34, 65
-----------	--------

M

Mapped Object 001.....	76
Minimum cycle time.....	78

O

OPERATIONAL.....	34, 65
------------------	--------

P

PRE-OPERATIONAL.....	34, 65
Product code.....	75

R

Revision.....	75
---------------	----

S

SAFE-OPERATIONAL.....	34, 65
Scaling function.....	79, 86
Serial number.....	75
Shift Time.....	77
SM MailBox Receive (SM0).....	76
SM MailBox Send (SM1).....	76
SM PDO input (SM3).....	76
SM PDO output (SM2).....	76
Subindex 001.....	77
Sync modes supported.....	78
Sync Type.....	77

V

Vendor ID.....	75
----------------	----

W




Wrong parameters loaded from flash memory at power-on.....	87
--	----

Typographic and iconographic conventions

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

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

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

	This icon, followed by the word 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 XAC81 series with EtherCAT interface**:

- **XAC81-18-00-EC2-...** **18 bit singleturn encoder**
- **XAC81-13-14-EC2-...** **13+14 bit multiturn encoder**
- **XAC81-16-14-EC2-...** **16+14 bit multiturn encoder**

For complete technical information please [refer to the product datasheet](#).

EtherCAT is the open standard for Industrial Ethernet. Its technology enables improved performances and allows to meet efficiency and productivity requirements in any complex industrial system. Real time communication, deterministic synchronization, high speed up to 100 Mbit/s full duplex over long distances, flexible network topologies, complete diagnostics, IT integration are among the key benefits. EtherCAT encoders comply with ETG.1000 specifications and implement the CoE (CANopen over EtherCAT) and FoE (for firmware update) protocols as well as the EtherCAT State Machine. Thus they offer full scaling, preset, code sequence, position and velocity readout, diagnostic information etc.

XAC81 ATEX encoders are category 3 "normal level of protection" devices for use in Zones 2 and 22 and in the temperature class T5 (T100°C). For complete information on ATEX certification please refer to the enclosed documentation.

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.

Glossary of EtherCAT terms

EtherCAT, like many other networking systems, has a set of unique terminology. Table below contains a few of the technical terms used in this guide to describe the EtherCAT interface. They are listed in alphabetical order.

Acknowledge telegram (AT)	Telegram, in which each Slave inserts its data.
Actual value	Value of a variable at a given instant.
Algorithm	Completely determined finite sequence of operations by which the values of the output data can be calculated from the values of the input data.
Application	Function or data structure for which data is consumed or produced. Software functional element specific to the solution of a problem in industrial-process measurement and control.
Application class	Configuration of a Drive Object with a set of functional objects and supported by standard telegrams.
Application mode	Type of application that can be requested from a PDS.
Application objects	Multiple object classes that manage and provide a run time exchange of messages across the network and within the network device.
Application process	Part of a distributed application on a network, which is located on one device and unambiguously addressed.
Application relationship	Cooperative association between two or more application-entity-invocations for the purpose of exchange of information and coordination of their joint operation. This relationship is activated either by the exchange of application-protocol-data-units or as a result of preconfiguration activities.
Attribute	Description of an externally visible characteristic or feature of an object, property or characteristic of an entity. The attributes of an object contain information about variable portions of an object. Typically, they provide status information or govern the operation of an object. Attributes may also affect the behaviour of an object. Attributes are divided into class attributes and instance attributes.
Axis	Logical element inside an automation system (e.g. a motion control system) that represents some form of movement.
Basic Slave	Slave device that supports only physical addressing of data.
Behaviour	Indication of how an object responds to particular events.
Bit	Unit of information consisting of a 1 or a 0. This is the smallest data unit that can be transmitted.

CANopen	Application layer protocol as defined in EN 50325-4.
Channel	Representation of a single physical or logical management object of a Slave to control conveyance of data.
CIP™	Common Industrial Protocol (see IEC 61158 Type 2, IEC 61784-1 and IEC 61784-2 CPF2).
Class	Description of a set of objects that share the same attributes, operations, methods, relationships, and semantics.
Client	Object which uses the services of another (Server) object to perform a task. Initiator of a message to which a Server reacts.
Clock synchronization	Representation of a sequence of interactions to synchronize the clocks of all time receivers by a time Master.
Commands	Set of commands from the application control program to the PDS to control the behaviour of the PDS or functional elements of the PDS.
Communication cycle	Accumulation of all telegrams between two Master synchronization telegrams.
Communication object	Component that manages and provides a run time exchange of messages across the network.
Connection	Logical binding between two application objects within the same or different devices.
Consume	Act of receiving data from a provider.
Consumer	Node or sink receiving data from a provider.
Control	Purposeful action on or in a process to meet specified objectives.
Control device	Physical unit that contains – in a module/subassembly or device – an application program to control the PDS.
Control unit	Control device.
Control word	Two adjacent bytes inside the Master data telegram containing commands for the addressed drive.
Controller	Controlling device which is associated with one or more drives (axes) a host for the overall automation.
Conveyance path	Unidirectional flow of APDUs across an application relationship.
Cycle time	Time span between two consecutive cyclically recurring events.
Cyclic	Events which repeat in a regular and repetitive manner.
Cyclic data	Part of the telegram which does not change its meaning during cyclic operation of the interface. High priority real-time data that is transferred by a CIP Motion connection on a periodic basis.
Data	Generic term used to refer to any information carried over a

	fieldbus.
Data consistency	Means for coherent transmission and access of the input-or output-data object between and within Client and Server.
Data exchange	Demand dependent; non cyclic transmission (service channel).
Data type	Relation between values and encoding for data of that type according to the definitions of IEC 61131-3. Set of values together with a set of permitted operations.
Data type object	Entry in the object dictionary indicating a data type.
Default gateway	Device with at least two interfaces in two different IP subnets acting as router for a subnet.
Device	Field device. Networked independent physical entity of an industrial automation system capable of performing specified functions in a particular context and delimited by its interfaces. Entity that performs control, actuating and/or sensing functions and interfaces to other such entities within an automation system. Physical entity connected to the fieldbus composed of at least one communication element (the network element) and which may have a control element and/or a final element (transducer, actuator, etc.).
Device profile	Collection of device dependent information and functionality providing consistency between similar devices of the same device type. Representation of a device in terms of its parameters and behaviour according to a device model that describes the device's data and behaviour as viewed through a network, independent from any network technology.
Diagnosis information	All data available at the Server for maintenance purposes.
Distributed clocks	Method to synchronize Slaves and maintain a global time base.
DL	Data-link-layer.
DLPDU	Data-link-protocol-data-unit.
Drive Object	Functional element of a Drive Unit.
Drive Unit	Logical device which comprises all functional elements related to one central processing unit.
Error	Discrepancy between a computed, observed or measured value or condition and the specified or theoretically correct value or condition.
Error class	General grouping for related error definitions and corresponding error codes.
Error code	Identification of a specific type of error within an error class.
EtherCAT State Machine	EtherCAT Slave is a state machine; communication and operating characteristics depend on the current state of the

	device.
Event	Instance of a change of conditions.
Event data	Medium priority real-time data that is transferred by a CIP Motion connection only after a specified event occurs.
Feed forward	Command value used to compensate the lag in the control loop.
Feedback variable	Variable which represents a controlled variable and which is returned to a comparing element.
Fieldbus memory management unit	Function that establishes one or several correspondences between logical addresses and physical memory.
Fieldbus memory management unit entity	Single element of the fieldbus memory management unit: one correspondence between a coherent logical address space and a coherent physical memory location.
Frame	Denigrated synonym for DLPDU.
FreeRun	Asynchronous communication mode.
Full Slave	Slave device that supports both physical and logical addressing of data.
Functional element	Entity of software or software combined with hardware, capable of accomplishing a specified function of a device.
HMI	Human Machine Interface.
Host	Device that covers the automation functionality of an automation device.
I/O data	Input data and output data that would typically need to be updated on a regular basis (e.g. periodic change of state), such as commands, set-points, status and actual values.
Identification number (IDN)	Designation of operating data under which a data block is preserved with its attribute, name, unit, minimum and maximum input values, and the data.
Index	Address of an object within an application process.
Input data	Data transferred from an external source into a device, resource or functional element.
Interface	Shared boundary between two entities defined by functional characteristics, signal characteristics, or other characteristics as appropriate.
Little endian	Data representation of multi-octet fields where the least significant octet is transmitted first.
Logical power drive system	Model which includes PDS and communication network accessible through the generic PDS interface.
Mapping	Correspondence between two objects in that way that one object is part of the other object.
Mapping parameters	Set of values defining the correspondence between application objects and process data objects.

Master	Device that controls the data transfer on the network and initiates the media access of the Slaves by sending messages and that constitutes the interface to the control system. Node, which assigns the other nodes the right to transmit.
Master data telegram (MDT)	Telegram, in which the Master inserts its data.
Medium	Cable, optical fibre or other means by which communication signals are transmitted between two or more points.
Message	Ordered series of octets intended to convey information. Normally used to convey information between peers at the application layer.
Model	Mathematical or physical representation of a system or a process, based with sufficient precision upon known laws, identification or specified suppositions.
Motion	Any aspect of the dynamics of an axis.
Motion Axis Object	Object that defines the attributes, services, and behaviour of a motion device based axis (or PDS) according to the CIP Motion specification, including Communications, Device Control, and Basic Drive FE elements as defined in IEC 61800-7.
Network	Set of nodes connected by some type of communication medium, including any intervening repeaters, bridges, routers and lower-layer gateways.
Node	Single DL-entity as it appears on one local link. End-point of a link in a network or a point at which two or more links meet [derived from IEC 61158-2].
Object	Abstract representation of a particular component within a device. An object can be: <ol style="list-style-type: none"> 1. an abstract representation of the capabilities of a device. Objects can be composed of any or all of the following components: <ul style="list-style-type: none"> ◦ data (information which changes with time); ◦ configuration (parameters for behaviour); ◦ methods (things that can be done using data and configuration); 2. a collection of related data (in the form of variables) and methods (procedures) for operating on that data that have clearly defined interface and behaviour.
Object dictionary	Data structure addressed by Index and Sub-index that contains description of data type objects, communication objects and application objects. List of objects with unique 16-bit index and 8-bit sub-index as defined in EN 50325-4.
Operating cycle	Period of the control loop within the drive or the control unit.
Operating mode	Characterization of the way and the extent to which the human operator intervenes in the control equipment.

Output data	Data originating in a device, resource or functional element and transferred from them to external systems.
P-Device	Field device and the host for the Drive Objects.
Parameter	Data element that represents device information that can be read from or written to a device, for example through the network or a local HMI.
PDO	Process Data Object.
PDS	Power Drive System.
Process data	Collection of application objects designated to be transferred cyclically or acyclically for the purpose of measurement and control.
Process Data Object (PDO)	Communication object with real-time capability. Structure described by mapping parameters containing one or several process data entities.
Producer	Node or source sending data to one or many consumers.
Profile	Representation of a PDS interface in terms of its parameters, parameter assemblies and behaviour according to a communication profile and a device profile.
Protocol	Convention about the data formats, time sequences, and error correction in the data exchange of communication systems.
Reference variable	Input variable to a comparing element in a controlling system which sets the desired value of the controlled variable and is deducted from the command variable.
Resource	Processing or information capability.
Segment	Collection of one real Master with one or more Slaves.
Server	Object which provides services to another (Client) object.
Service	Operation or function than an object and/or object class performs upon request from another object and/or object class.
Service data	Lower priority real-time data associated with a service message from the controller that is transferred by a CIP Motion connection on a periodic basis.
Set-point	Value or variable used as output data of the application control program to control the PDS.
Slave	DL-entity accessing the medium only after being initiated by the preceding Slave or the Master. Node, which is assigned the right to transmit by the Master.
Standard telegram	Set of input data and output data for an application mode.
Status	Set of information from the PDS to the application control program reflecting the state or mode of the PDS or a functional element of the PDS.
Status word	Two adjacent bytes inside the drive telegram containing status information.

Subindex	Sub-address of an object within the object dictionary.
Supervisor	Engineering device which manages provisions of configuration data (parameter sets) and collections of diagnosis data from P-Devices and/or controllers.
Switch	MAC bridge as defined in IEEE 802.1D.
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. Collection of control elements to coordinate access to concurrently used objects.
Sync manager channel	Single control elements to coordinate access to concurrently used objects.
Synchronised	Condition where the local clock value on the drive is locked onto the Master clock of the distributed System Time.
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.
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.
System Time	Absolute time value as defined in the CIP Sync specification in the context of a distributed time system where all devices have a local clock that is synchronised with a common Master clock.
Telegram	Message.
Time stamp	System Time stamp value associated with the CIP Motion connection data that conveys the absolute time when the associated data was captured, or that can also be used to determine when the associated data shall be applied.
Topology	Physical network architecture with respect to the connection between the stations of the communication system.
Type	Hardware or software element which specifies the common attributes shared by all instances of the type.
Use case	Class specification of a sequence of actions, including variants, that a system (or other entity) can perform, interacting with actors of the system.
Variable	Software entity that may take different values, one at a time.

List of abbreviations

Table below contains a list of abbreviations (in alphabetical order) which may be used in this guide to describe the EtherCAT interface.

Term	Description
ADS	Automation Device Specification
AL	Application layer
AoE	ADS over EtherCAT
AP (-task)	Application (-task) on top of the stack
API	Application Programming Interface
ASCII	American Standard Code for Information Interchange
CoE	CANopen over EtherCAT
COS	Change of State
DC	Distributed Clocks
DL	Data Link Layer
DPM	Dual port memory
E2PROM (EEPROM)	Electrically erasable Programmable Read-only Memory
EoE	Ethernet over EtherCAT
ESC	EtherCAT Slave Controller
ESM	EtherCAT State Machine
ETG	EtherCAT Technology Group
EtherCAT	Ethernet for Control and Automation Technology
FMMU	Fieldbus Memory Management Unit
FoE	File Access over EtherCAT
IEEE	Institute of Electrical and Electronics Engineers
LFW	Loadable firmware
LOM	Linkable object modules
LSB	Least significant byte
MSB	Most significant byte
OD	Object dictionary
ODV3	Object dictionary Version 3
PHY	Physical Interface (Ethernet)
PDO	Process Data Object (process data channel)

Term	Description
RTR	Remote Transmission Request
RxPDO	Receive PDO
SDO	Service Data Object (representing an acyclic data channel)
SHM	Shared memory
SM	Sync Manager
SoE	Servo Profile over EtherCAT
SSC	SoE Service Channel
TxPDO	Transmit PDO
VoE	Vendor Profile over EtherCAT
XML	eXtensible Markup Language

References

- [1] IEC 61158 Part 2-6 Type 12 documents (also available for members of EtherCAT Technology Group as specification documents ETG-1000)
- [2] Proceedings of EtherCAT Technical Committee Meeting from February 9th, 2005.
- [3] IEC 61800-7
- [4] EtherCAT Specification Part 5 – Application Layer services specification. ETG.1000.5.
- [5] EtherCAT Specification Part 6 – Application Layer protocol specification. ETG.1000.6.
- [6] EtherCAT Indicator and Labeling Specification. ETG.1300.
- [7] EtherCAT Protocol Enhancements. ETG.1020.
- [8] EtherCAT Slave Information Annotation ETG 2001.
- [9] EtherCAT Slave Information Specification ETG.2000.

1 – Safety summary



1.1 Safety

- Always comply with the information in this manual concerning the ATEX products;
- always adhere to the professional safety and accident prevention regulations applicable to your country during device installation and operation;
- installation and maintenance operations have to be carried out by qualified personnel only, with power supply disconnected and stationary mechanical parts;
- device must be used only for the purpose appropriate to its design: use for purposes other than those for which it has been designed could result in serious personal and/or the environment damage;
- high current, voltage and moving mechanical parts can cause serious or fatal injury;
- failure to comply with these precautions or with specific warnings elsewhere in this manual violates safety standards of design, manufacture, and intended use of the equipment;
- Lika Electronic 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 "5 - Electrical connections" section on page 31;
- connect +Vdc and 0Vdc and check the power supply is correct first before connecting the communication ports;
- in classified areas the electrical connection of the device must be carried out in compliance with the methods indicated in the directive EN IEC 60079-0:2018-07 and according to EN 60079-14;
- in compliance with the 2014/30/EU norm on electromagnetic compatibility, the following precautions must be taken:
 - before handling and installing, discharge electrical charge from your body and tools which may come in touch with the device;
 - power supply must be stabilized without noise, install EMC filters on device power supply if needed;
 - always use shielded cables (twisted pair cables whenever possible);
 - avoid cables runs longer than necessary;
 - avoid running the signal cable near high voltage power cables;
 - mount the device as far as possible from any capacitive or inductive noise source, shield the device from noise source if needed;
 - to guarantee a correct working of the device, avoid using strong magnets on or near by the unit;



- minimize noise by connecting the cable shield and/or the connector and/or the frame to ground. Provide the ground connection as close as possible to the encoder. We suggest using the connection point provided in the encoder's enclosure (see the Figure 1) to connect the unit to ground. Fix the ground cable to the connection point by means of a ring crimp terminal and a M4 bolt. The connection point to ground might be provided also on the installation side by connecting the cable shield to the earth point. It is user's duty to find the best solution to minimize the interference. Please make sure that ground is not affected by noise;
- the rated supply voltage must not exceed 30Vdc;
- in classified areas the electrical connection of the device must be carried out in compliance with the methods stated in the EN IEC 60079-0:2018-07 regulation;
- the user has to take suitable measures in order to prevent longer or continuous disturbances that cause the supply voltage to exceed by 10% or more;
- the product must be protected against overheating due to mechanical or electrical overloading.



1.3 Mechanical safety

- Install the device following strictly the information in the "4 - Mechanical installation" section on page 28;
- 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 hollow shaft: the encoder can be mounted directly on a shaft whose diameter has to respect the technical characteristics specified in the purchase order.



1.4 Operational safety

- The described products are category 3 certified and are designed for use in potentially explosive zones 2 (according to EN60079-15) and 22 (according to EN60079-31). They are suitable for a safe use and a normal level of protection in areas in which explosive atmospheres caused by mixture of gases and vapours (zone 2) or dust (zone 22) are unlikely to occur and if they do occur do so for a short period of time. They comply with the constructional safety requirements of the temperature class T5 (T100°C = +212°F). **They are not to be used in Zones 0, 1, 20 and 21;**
- protection requirements: Zone 2, gas explosion protection (G): protection by non sparking equipment: the construction minimizes the

occurrence of sparks, arcs or hot surfaces, which in normal operation might give the risk of explosion; Zone 22, dust explosion protection (D): the construction of the product protects against the penetration of dust in hazardous quantities (min. IP5x) and guarantees that the surface temperature is under the ignition temperature of dust/air mixture as well as under the smouldering temperature of dust deposits;

- the manufacturer's specifications (temperature, speed, ...) must under no circumstances be exceeded;
- max. permissible environmental temperature: -20°C +40°C (-4°F +104°F) at continuous rotational speed of max. 6000 rpm;
- the maximum surface temperature (in °C) of the product must not exceed 2/3 of the ignition temperature of the dust/air mixture;
- in continuous operation the plastic parts must be able to withstand a temperature 10°C higher than the maximum temperature that can be reached in the hottest point of the product while the environmental temperature is the highest allowed during operation.

2 - Identification

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



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

3 – Certificates

3.1 Declaration of ATEX Conformity



Carrè, 17.05.2021

Declaration of Conformity

The manufacturer:
LIKA ELECTRONIC SRL
Via S. Lorenzo, 25
36010 Carrè (VI) – Italy

hereby declares that the following products (only when correctly assembled):

- | | | |
|------------------|----------------------|----------------------|
| • XAC8018/1PB-14 | • XAC8013/16384PB-14 | • XAC8016/16384PB-14 |
| • XAC8018/1CB-14 | • XAC8013/16384CB-14 | • XAC8016/16384CB-14 |
| • XAC8018/1FD-14 | • XAC8013/16384FD-14 | • XAC8016/16384FD-14 |
| • XAC8118/1PT-14 | • XAC8113/16384PT-14 | • XAC8116/16384PT-14 |
| • XAC8118/1EC-14 | • XAC8113/16384EC-14 | • XAC8116/16384EC-14 |
| • XAC8118/1MT-14 | • XAC8113/16384MT-14 | • XAC8116/16384MT-14 |
| • XAC8118/1PL-14 | • XAC8113/16384PL-14 | • XAC8116/16384PL-14 |
| • XAC8118/1EP-14 | • XAC8113/16384EP-14 | • XAC8116/16384EP-14 |

with the marking:

Ex II 3G Ex nA IIB T5 Gc
EX II 3D Ex tc IIIC T100° Dc IP65

may be used in Zone 2 and Zone 22 hazardous areas.

Gas explosion protected equipment for Zone 2, according to EN60079-15

Zone 2 includes areas in which an explosive atmosphere consisting of a mixture of air and flammable substances in the form of gas or vapour is not likely to occur in normal operation but, if it does occur, will persist for a short period only.

Dust explosion protected equipment for Zone 22, according to EN60079-31

Zone 22 includes areas in which an explosive atmosphere in the form of a cloud of dust in air is not likely to occur in normal operation but, if it does occur, will persist for a short period only.

Lika Electronic Srl
Smart encoders & actuators



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

Tel. +39 0445 806600
Fax +39 0445 806699
info@lika.biz • www.lika.biz

P.I./C.F. IT 00817760242
VAT # IT 00817760242
R.E.A. 165423/Vicenza

Protection requirements:
Zone 22, dust explosion protection (D)

Protection by means of the housing: the construction of the product protects against the penetration of dust in hazardous quantities (min. IP5x) and guarantees that the surface temperature is under the ignition temperature of dust/air mixtures as well as under the smouldering temperature of dust deposits.

Zone 2, gas explosion protection (G)

Protection by non sparking equipment: the construction minimizes the occurrence of sparks, arcs or hot surfaces, which in normal operation might give the risk of explosion.

The manufacturer's specification of the purchased product such as operating temperature, shaft rotational speed, shaft load, max. supply voltage, etc. must under no circumstances be exceeded.


Protection requirements for rotary encoders of category 3GD:

- The product must be correctly assembled (encoder body and connection cap)
- The degree of protection in the installation condition must be at least IP6x.
- The maximum surface temperature of the product (in °C) must not exceed two-thirds of the ignition temperature of the dust/air mixture.
- During normal operation no sparks, arcs and non-permissible temperatures should occur.
- The upper continuous operating temperature of plastic parts must be 10°C higher than the maximum temperature that can arise at the hottest point of the product, based on the highest environmental temperature possible in service.

Obligations of the user:

- The rated supply voltage must not exceed 30Vdc.
- The manufacturer's specification concerning the range of the operating temperature (see product data sheet) has to be fully observed.
- The product must be protected against overheating due to mechanical or electrical overloading.
- The user has to take suitable measures in order to prevent longer or continuous disturbances that cause the supply voltage to be exceeded by 10% or more.
- The user has the obligation to protect in an appropriate way the area where electrical connections are carried out (make sure that the product is not connected to power supply when connecting) and ensure that this area also complies with Zone 2/22 requirements.
- The listed products do not meet the "impact test" requirements of EN IEC 60079-0:2018, section 26.4.2. but can however be used in Zone 2/22 areas where the user ensures that impacts are avoided by appropriate protection.

LIKA Electronic SRL


 Marco CALABRESE
 Managing Director

 Lika Electronic Srl
 Smart encoders & actuators

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

 Tel. +39 0445 806600
 Fax +39 0445 806699
 info@lika.biz • www.lika.biz

 P.I./C.F. IT 00817760242
 VAT # IT 00817760242
 R.E.A. 165423/Vicenza

3.2 EU Declaration of Conformity

EU Declaration of Conformity encoders series XAC80-XAC81

- 1) Certificate: Nr. 5
- 2) Manufacturer: LIKA ELECTRONIC SRL
Via S. Lorenzo, 25
36010 Carrè (VI) - Italy
VAT# IT00817760242
- 3) Scope of the certificate: encoder series
- XAC8018/1PB-14 • XAC8013/16384PB-14 • XAC8016/16384PB-14
 - XAC8018/1CB-14 • XAC8013/16384CB-14 • XAC8016/16384CB-14
 - XAC8018/1FD-14 • XAC8013/16384FD-14 • XAC8016/16384FD-14
 - XAC8118/1PT-14 • XAC8113/16384PT-14 • XAC8116/16384PT-14
 - XAC8118/1EC-14 • XAC8113/16384EC-14 • XAC8116/16384EC-14
 - XAC8118/1MT-14 • XAC8113/16384MT-14 • XAC8116/16384MT-14
 - XAC8118/1PL-14 • XAC8113/16384PL-14 • XAC8116/16384PL-14
 - XAC8118/1EP-14 • XAC8113/16384EP-14 • XAC8116/16384EP-14
- 4) This certificate has been issued under the responsibility of the manufacturer indicated in point 2).
- 5) The scope of the certificate indicated in point 3) is in conformity with the essential Health and Safety regulations and legislative regulations of the directives:
2014/34/EU "ATEX"
2014/30/EU "Electromagnetic compatibility"
- 6) Compliance with harmonized regulations, technical specifications and other documents is assured by compliance with the following norms:
EN IEC 60079-0:2018-07
EN 60079-31:2014-07
EN 61000-6-4, EN 61000-6-2
EN 60079-15
- 7) ATEX marking on the equipment:



II 3 GD, Ex nA IIB T5 Gc, Ex tc IIIC T100° Dc, IP65

Carrè, 17.05.2021


Marco CALABRESE
Legal representative

Lika Electronic Srl
Smart encoders & actuators



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

Tel. +39 0445 806600
Fax +39 0445 806699
info@lika.biz • www.lika.biz

P.I./C.F. IT 00817760242
VAT # IT 00817760242
R.E.A. 165423/Vicenza

3.3 Safety instructions



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

Smart encoders & actuators

SAFETY INSTRUCTIONS encoders series XAC80-XAC81

1) Marking:



II3 GD, Ex nA IIB T5 Gc, Ex tc IIIC T100° Dc, IP65

Explosion-proof encoder manufactured in compliance with the following regulations:

EN IEC 60079-0:2018-07
EN 60079-31:2014-07
EN 61000-6-4, EN 61000-6-2
EN 60079-15

- **Ex:** Equipment for use in potentially explosive atmospheres.
- **d:** Protection by explosion-proof housing.
- **IIB:** Electrical apparatus for use in potentially explosive atmospheres other than mines susceptible to firedamp.
- **T5:** Maximum housing surface temperature 100°C.
- **Gc:** Level of protection (EPL). The equipment can be used in Zone 2.
- **tc:** Electrical apparatus with protective housing for use in the presence of combustible dust.
- **IIIC:** Equipment or protective housing intended for use in potentially explosive atmospheres with presence of combustible dust.
- **T100°C:** Maximum surface temperature.
- **Dc:** Level of protection (EPL). The equipment can be used in Zone 22.
- **IP65:** Degree of IP protection for dust-proof housing.

Equipment intended for use in the following Zones:

Zone 2: Mixture of gas/air, vapour/air, mist/air

Zone 22: Mixture of dust/air

ATTENTION: Equipment not to be used in Zones 0, 1 and 21

Lika Electronic Srl
Smart encoders & actuators



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

Tel. +39 0445 806600
Fax +39 0445 806699
info@lika.biz • www.lika.biz

P.I./C.F. IT 00817760242
VAT # IT 00817760242
R.E.A. 165423/Vicenza



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

Smart encoders & actuators

- 2) The equipment has to be installed only by qualified personnel and according to the applicable regulations.
- 3) Do not tool or drill the equipment.
- 4) If the connection cap needs to be opened, please carefully replace it after and ensure that the seal is as tight as it was before
- 5) After connection, please carefully tighten the cable glands and the connectors.
- 6) Use the encoder's fixing plate for installation and against rotation.
- 7) Protect the device against shock and mechanical damages.
- 8) Use the product according to the indicated degree of IP protection.
- 9) Maximum permissible environmental temperature -20°C to +40°C (at continuous rotational speed of max. 6000 rpm).
- 10) In classified areas the electrical connection of the device has to be carried out according to the methods of EN IEC 60079-0:2018-07.
- 11) Connect the device according to the electrical connections scheme on the user manual.
- 12) Provide a ground connection (GND) using the ground screw on the housing.

LIKA ELECTRONIC SRL
Carrè, 17.05.2021

Lika Electronic Srl
Smart encoders & actuators



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

Tel. +39 0445 806600
Fax +39 0445 806699
info@lika.biz • www.lika.biz

P.I./C.F. IT 00817760242
VAT # IT 00817760242
R.E.A. 165423/Vicenza

4 – Mechanical installation



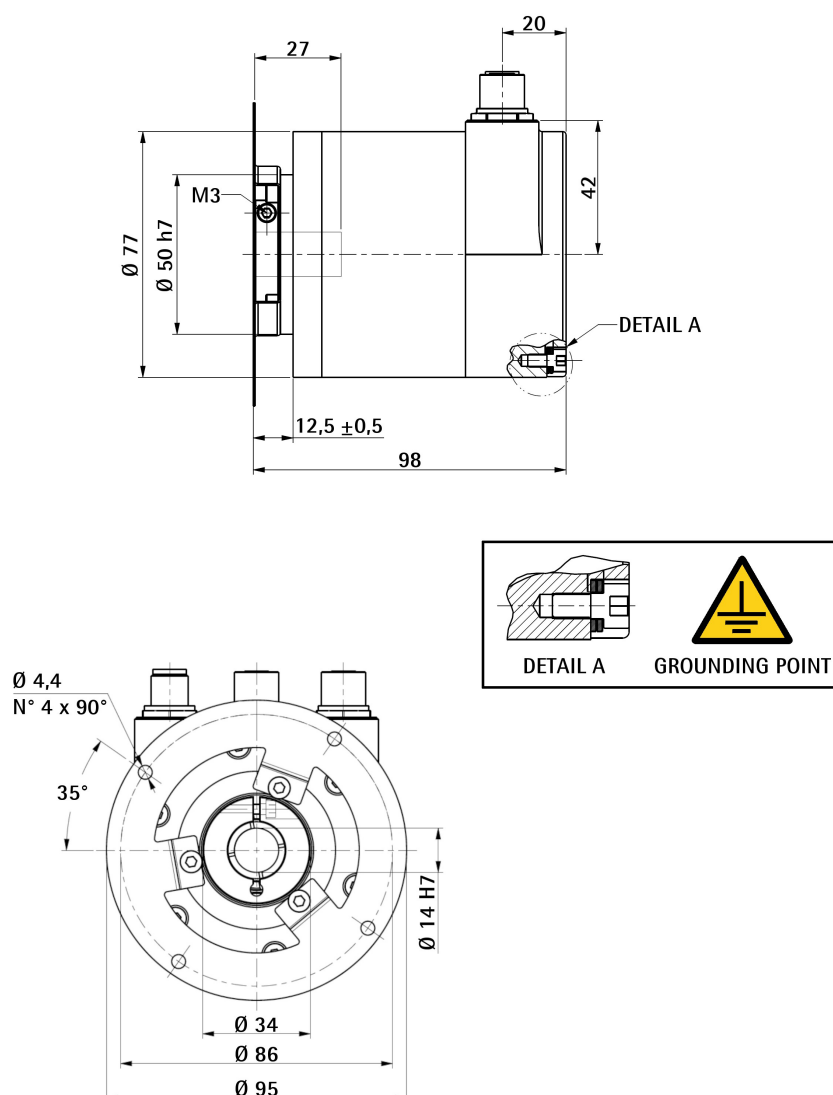
WARNING

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

For any information on the mechanical data and the electrical characteristics of the encoder please refer to the technical catalog.

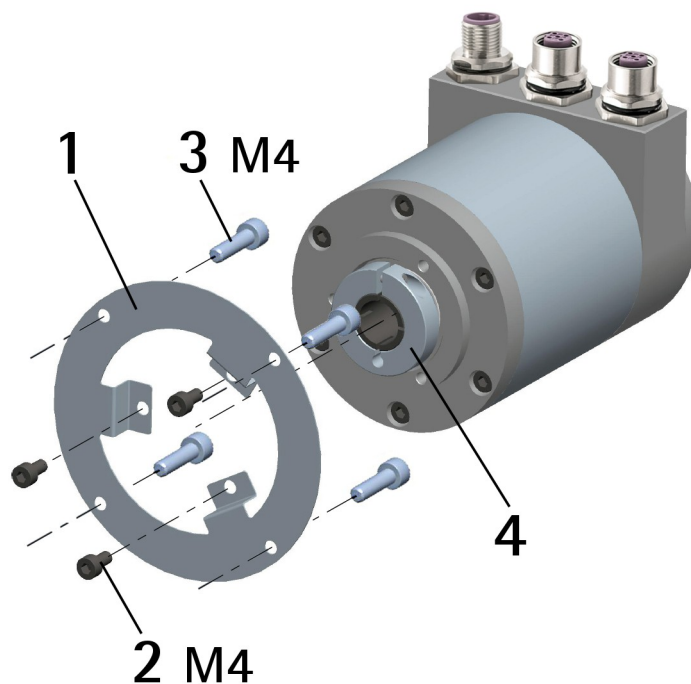
4.1 Encumbrance sizes

(values are expressed in mm)



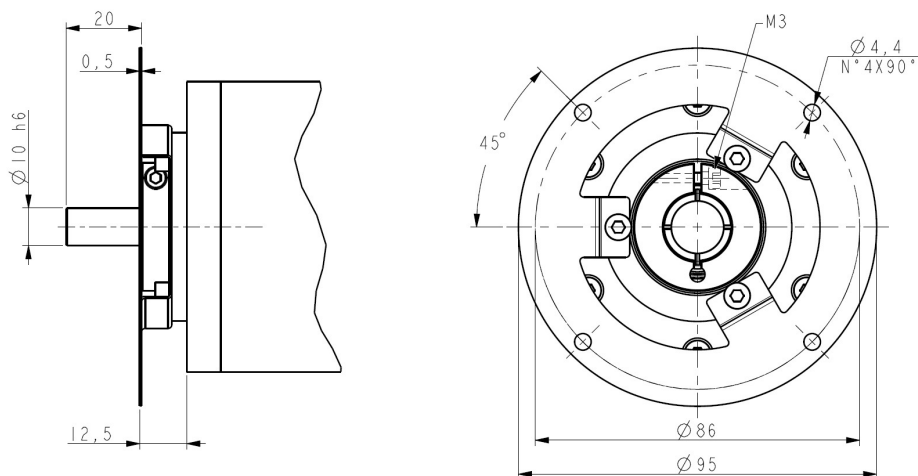
4.2 Mounting the encoder

- Fasten the fixing plate **1** to the encoder using the three M4 screws **2** provided with the device;
- mount the encoder on the motor shaft using the reducing sleeve (if supplied); avoid forcing the encoder shaft;
- fasten the fixing plate **1** to the rear of the motor using four M4 cylindrical head screws **3**;
- fix the collar **4** to the encoder shaft (apply threadlocker to screw).



4.3 Solid shaft (LKM-1758) and Fixing plate (LKM-1520)

LKM-1758 is an optional feature and has to be ordered separately.



WARNING

Unit with solid shaft: in order to guarantee 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 respected.



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 out technical personnel for the complete availability of custom made shafts.



No user serviceable parts inside the encoder!

5 - Electrical connections



WARNING

Power supply must be turned off before performing any electrical connection! Installation, electrical connection and maintenance operations must be carried out by qualified personnel only, with power supply disconnected. Mechanical components must be in stop. Do not open the encoder.



No user serviceable parts inside the encoder!

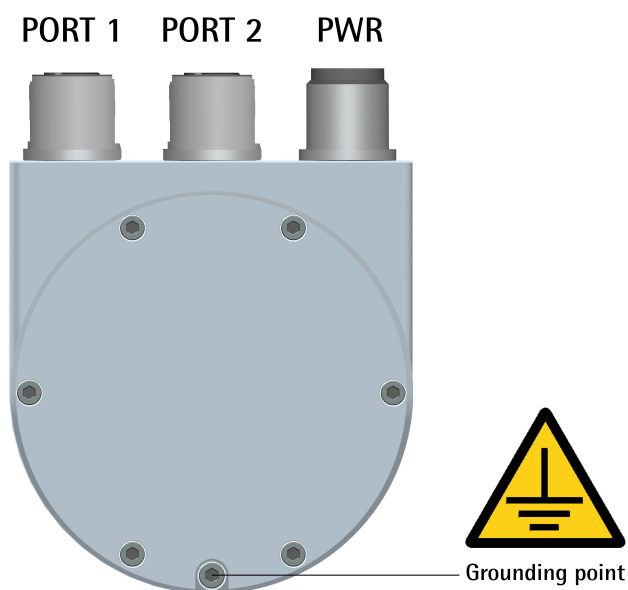
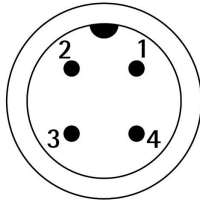


Figure 1 - Electrical connection and grounding point

5.1 PWR Power supply connector (Figure 1)

M12 4-pin male connector with A coding is used for power supply.



Description	Pin
+10Vdc +30Vdc	1
n.c.	2
0Vdc	3
n.c.	4

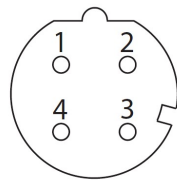


WARNING

Connect +Vdc and 0Vdc and check the power supply is correct first before connecting the communication ports.

5.2 P1 Port 1 and P2 Port 2 connectors (Figure 1)

Two M12 4-pin female connectors with D coding are used for Ethernet connection through port 1 and port 2.



Description	Pin
Tx Data +	1
Rx Data +	2
Tx Data -	3
Rx Data -	4

The M12 connectors have pin-out in compliance with the EtherCAT® standard. Therefore you can use standard EtherCAT cables commercially available. The Ethernet interface supports 100 Mbit/s, fast Ethernet, full duplex operation. Please note that input ECATIN (BUS IN) and output ECATOUT (BUS OUT) connectors are not interchangeable! BUS IN connector must be networked towards the EtherCAT Master.



WARNING

Connect +Vdc and 0Vdc and check the power supply is correct first before connecting the communication ports.

5.3 Network configuration: topologies, cables, hubs, switches - Recommendations

Cables and possible connectors comply with the EtherCAT specifications. Cables are CAT-5 shielded cables.

Line, tree or star: EtherCAT supports almost any topology. The bus or line structure known from the fieldbuses thus also becomes available for Ethernet, without the quantity limitations implied by cascaded switches or hubs.

The Fast Ethernet physics (100BASE-TX) enables a cable length of 100 m (328 ft) between two devices. Since up to 65,535 devices can be connected, the size of the network is almost unlimited.

The Ethernet protocol according to IEEE 802.3 remains intact right up to the individual device; no sub-bus is required. In order to meet the requirements of a modular device like an electronic terminal block, the physical layer in the coupling device can be converted from twisted pair or optical fiber to LVDS (alternative Ethernet physical layer, standardized in [4.5]). A modular device can thus be extended very cost-efficiently. Subsequent conversion from the backplane physical layer LVDS to the 100BASE-TX physical layer is possible at any time – as usual with Ethernet.

For a complete list of the available cordsets and connection kits please refer to the product datasheet ("Accessories" list).

5.4 Addressing

It is not necessary to assign a physical address to the device because the addressing of the Slave is automatic at power-on during the initial scanning of the hardware configuration.

The field for addressing is 32-bit long, there are three kinds of addressing:

- Auto Increment Addressing = Position Addressing: 16 bits indicate 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 indicate 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).

For complete information refer to the "8.1.5 Addressing" section on page 62.

5.5 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 the downstream Slaves. For complete information refer to the "8.1.4 Line Termination" section on page 61.

5.6 Ground connection (Figure 1)

To minimize noise connect properly the shield and/or the frame to ground. Connect properly the cable shield to ground on user's side. Make sure that ground is not affected by noise. It is recommended to provide the ground connection as close as possible to the device. We suggest using the ground point provided in the enclosure (see Figure 1).

5.7 EtherCAT states

EtherCAT Slave is a state machine; the communication and the 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 the Master and the Slave on the Application Layer; some configuration registers are initialized and the Sync Managers are configured.
- **PRE-OPERATIONAL** (PREOP): in this state the mailbox is active; both the Master and the Slave can use the mailbox and its protocols for exchanging specific initialization parameters of the application. Exchange of Process Data (PDO) is forbidden. In this state the FoE protocol is used for firmware download.
- **SAFE-OPERATIONAL** (SAFEOP): in this state the Master and the Slave can issue only input process data, while the output process data is still in the **SAFE-OPERATIONAL** state;
- **OPERATIONAL** (OP): in this state the Master and the Slave are enabled to send both input process data and output process data.
- **BOOTSTRAP** (BOOT): no process data communication. Communication only via mailbox on Application Layer available. Special mailbox configuration is possible, e.g. larger mailbox size.

6 – Getting started



The following instructions are provided to allow the operator to set up the device for standard operation in a quick and safe mode.

6.1 General installation

- Mechanically install the device (see on page 28);
- execute the electrical connection and switch on the +10Vdc +30Vdc power supply, see on page 31 ff; check the soundness of the connection;
- switch off the power supply and execute the network connection, then switch on the power supply again, see on page 31 ff; check the soundness of the connection;
- you do not need to set the node address and the transmission rate (see on page 62);
- you do not need to set any line termination (see on page 61);
- install the XML file, see on page 40 ff;
- if you want to use the physical resolution (see the **6501-00 Hardware counts per revolution** object and the **6502-00 Hardware number of turns** object), please check that the **Scaling function** parameter is disabled (the bit 2 in the **6000-00 Operating parameters** object = 0; see on page 79);
- otherwise, if you need a custom resolution, enable the **Scaling function** parameter (the bit 2 in the **6000-00 Operating parameters** object = 1; see on page 79) and then set the resolution you need for your application next to the **6001-00 Units per revolution** and **6002-00 Total Measuring Range** objects (see on page 80);
- if you need you can enter the Preset value next to the **6003-00 Preset** object and then set it in the desired position; see on page 83;
- save the new setting values (use the **1010-01 Store parameters** object; see on page 74).

6.2 Setting the scaling function and custom resolution

- If you want to use the physical resolution of the encoder, please check that the **Scaling function** parameter is disabled (the bit 2 in the **6000-00 Operating parameters** object is = "0", see on page 79); in this case, the device uses the physical resolution (see the **6501-00 Hardware counts per revolution** and **6502-00 Hardware number of turns** objects) to arrange the absolute position value.
It is also possible to use the software tool, see the "7.4 CoE Object Dictionary" section on page 47.

- On the contrary, if you need a custom resolution, you must enable the scaling function by setting the **Scaling function** parameter (the bit 2 in the **6000-00 Operating parameters** object) to ="1" first and then set the required resolution parameters:
 - set the singleturn resolution next to the **6001-00 Units per revolution** object, see on page 80;
 - set the total resolution next to the **6002-00 Total Measuring Range** object, see on page 81.

It is also possible to use the software tool, see the "7.4 CoE Object Dictionary" section on page 47.

6.3 Reading the absolute position

To read the position value you can choose among the following methods.

- To read the absolute position of the encoder see the **6004-00 Position value** object on page 85; the **6004-00 Position value** object is mapped in the **1A00-01 PDO mapping parameter** object, sub-index 001 **01 Mapped Object 001**, see on page 76.
- Use the software tool, see the "7.3 Process Data Objects" section on page 46.

6.4 Setting and executing the preset

To set and execute the preset you can choose among the following methods.

- Enter a suitable value next to the **6003-00 Preset** object, see on page 83; the preset value is activated as soon as the value is confirmed.
- Use the software tool, see the "7.4 CoE Object Dictionary" section on page 47.

6.5 Saving data

To save the parameters permanently use the **1010-01 Store parameters** object, see on page 74.

6.6 Restoring defaults

To restore the default parameters use the **1011-01 Restore default parameters** object, see on page 74.

7 – Quick reference with TwinCAT

Lika encoders are Slave devices and support "CANopen over EtherCAT" (CoE) mode for data transfer. In particular, they support the "CANopen DS 301 Communication profile".

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

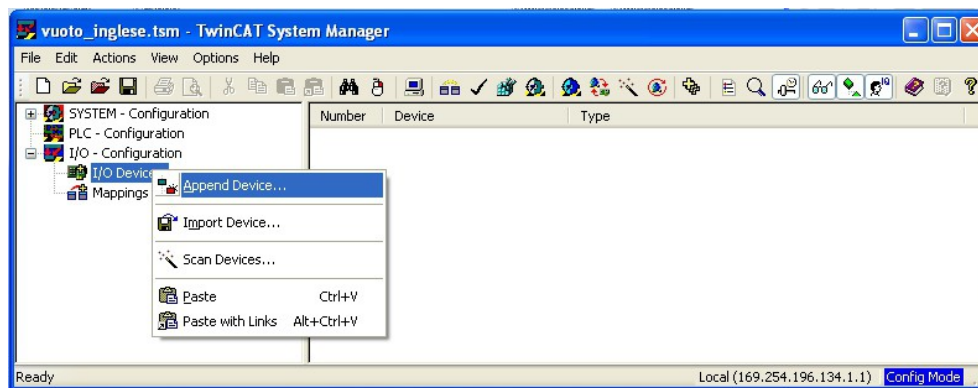
For any omitted specification on CANopen® protocol, please refer to "CiA Draft Standard Proposal 301. Application Layer and Communication Profile" and "CiA Draft Standard 406. Device profile for encoders" documents available at the address www.can-cia.org.

7.1 System configuration using TwinCAT software system from Beckhoff

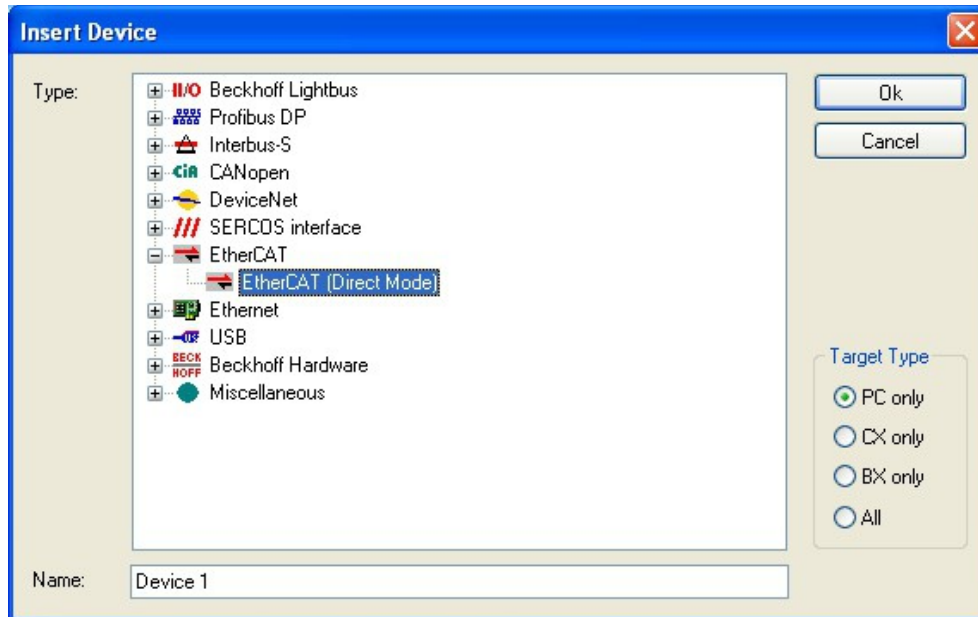
7.1.1 Setting the Network Card

Launch **TwinCAT System Manager**.

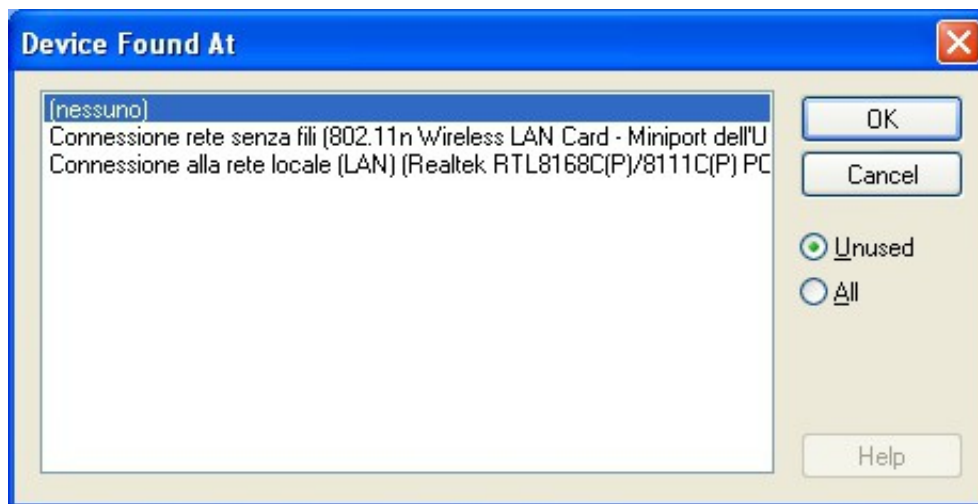
In the left pane of the main window extend the devices tree and select the **I/O Devices** item; right-click the **I/O Devices** item and then press the **Append Device...** command.



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



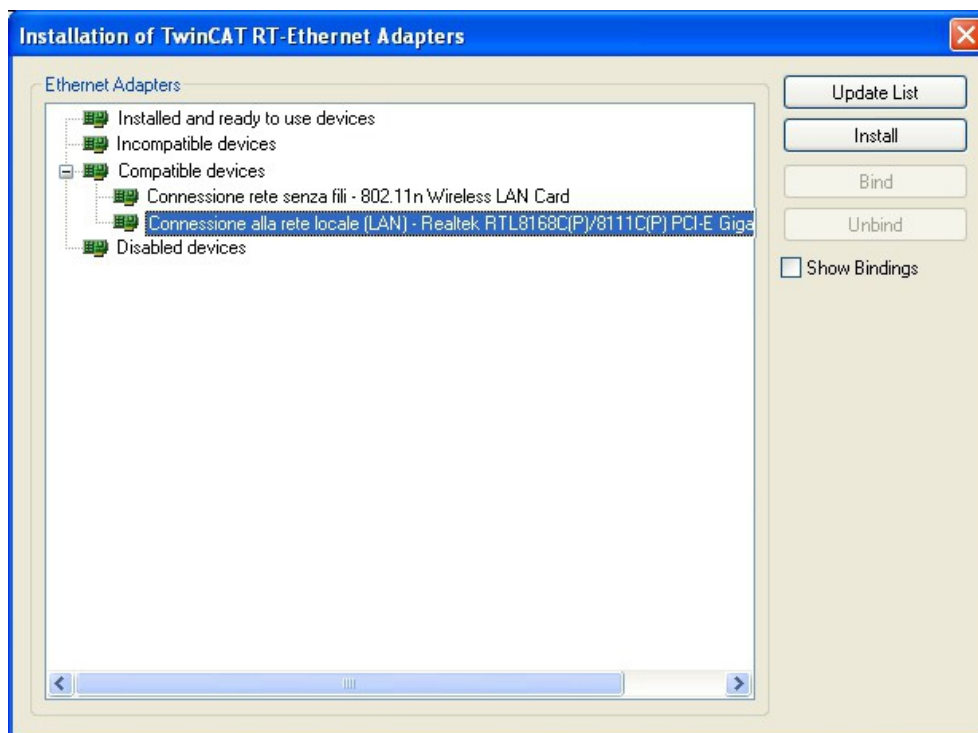
If a network card has been already installed properly, the following window will appear and show the list of the installed devices.



Select the network card you want to use and then confirm the choice by 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.

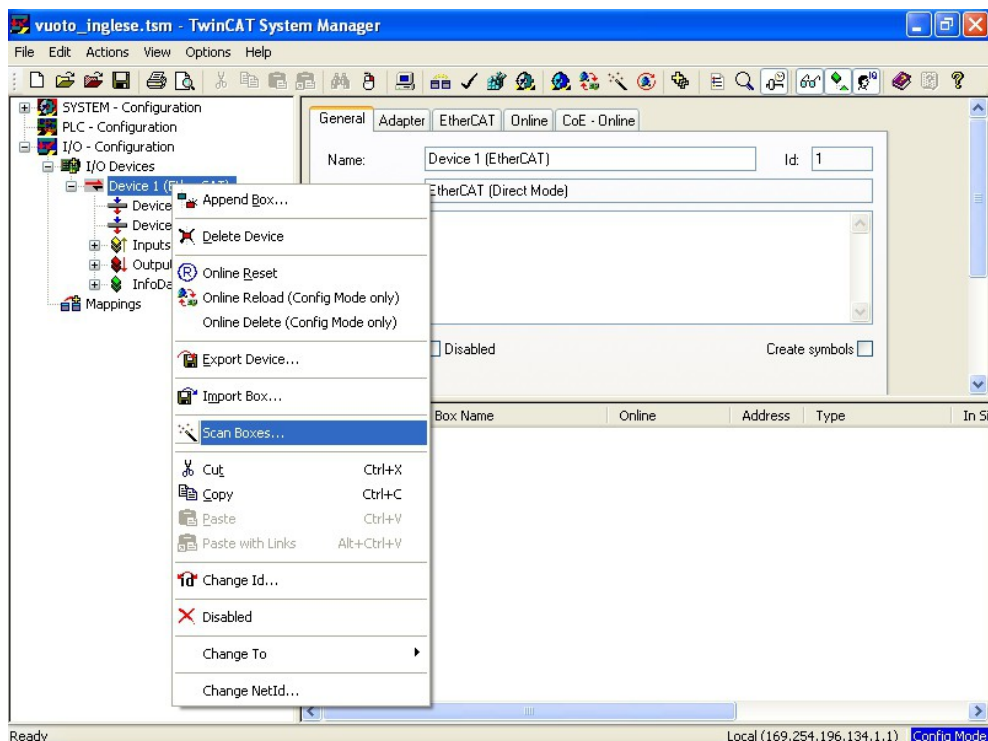
The **Installation of TwinCAT RT-Ethernet Adapters** window will appear.



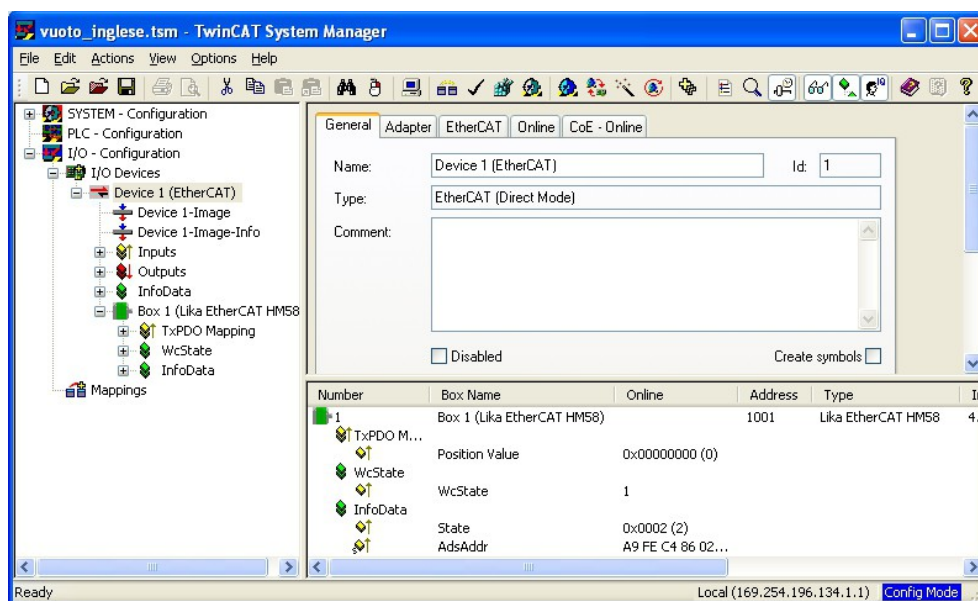
Now select the **Compatible Devices** item and choose the network card you want to install; finally press the **Install** button to confirm your choice.

7.1.2 Add new I/O modules (Boxes)

If one or more 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 the 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**). For older versions of the file please contact Lika Electronic's After Sales Service Dept.



WARNING

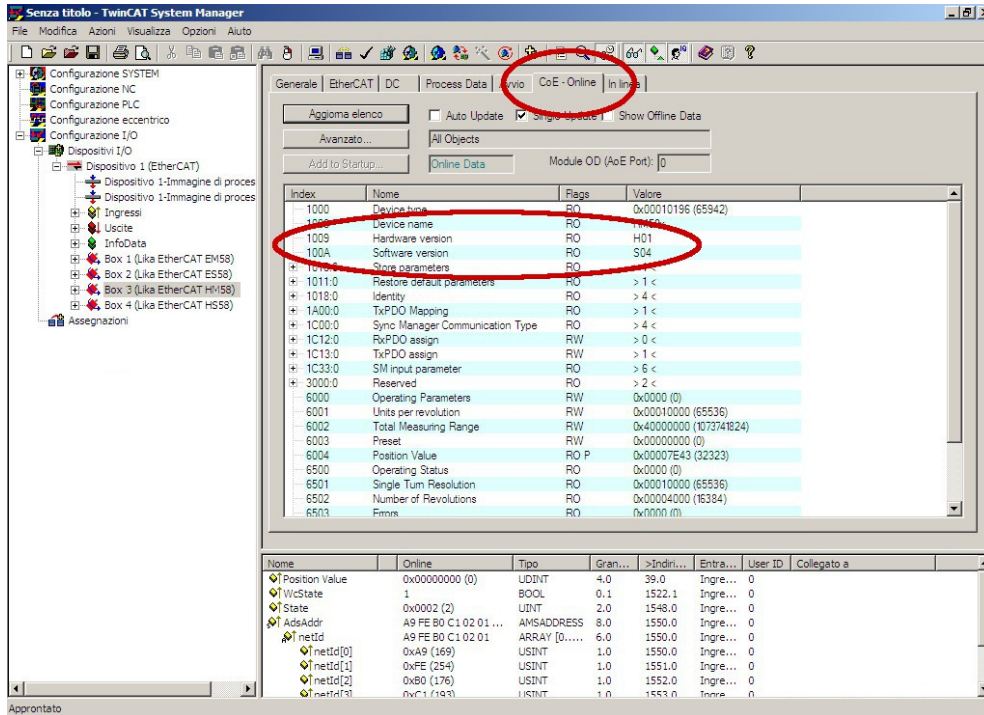
Before installing the XML file please check that it is compatible with the firmware version and the EEPROM version of the device; the XML file version, the firmware version and the EEPROM version must always comply. For example: if the firmware version is H1_S4 (Hardware version: 1; Software version: 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 "7.6 EEPROM upgrade" section on page 49. For a description of the firmware upgrade procedure please refer to the "7.7 Firmware upgrade" section on page 54.



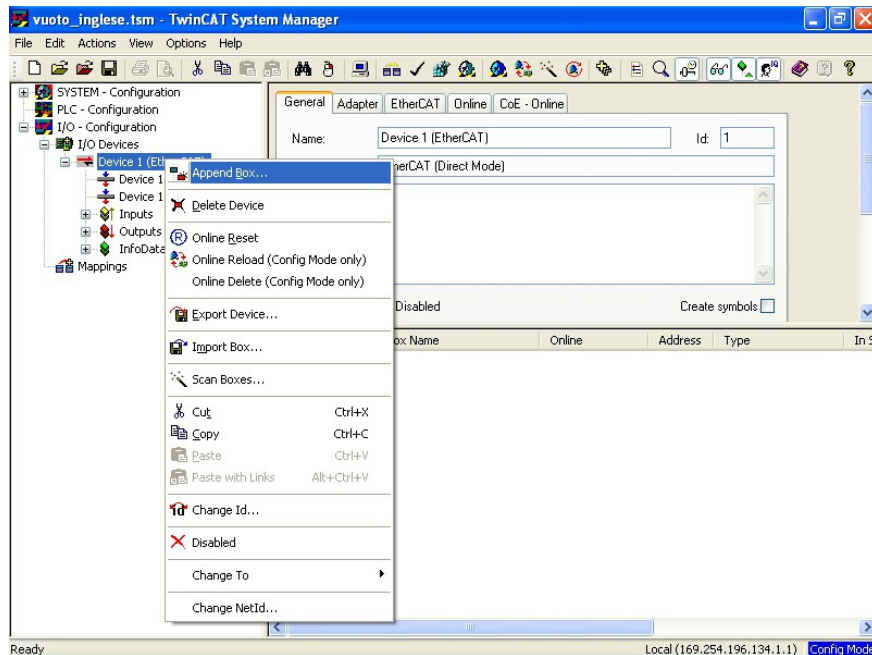
WARNING

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.

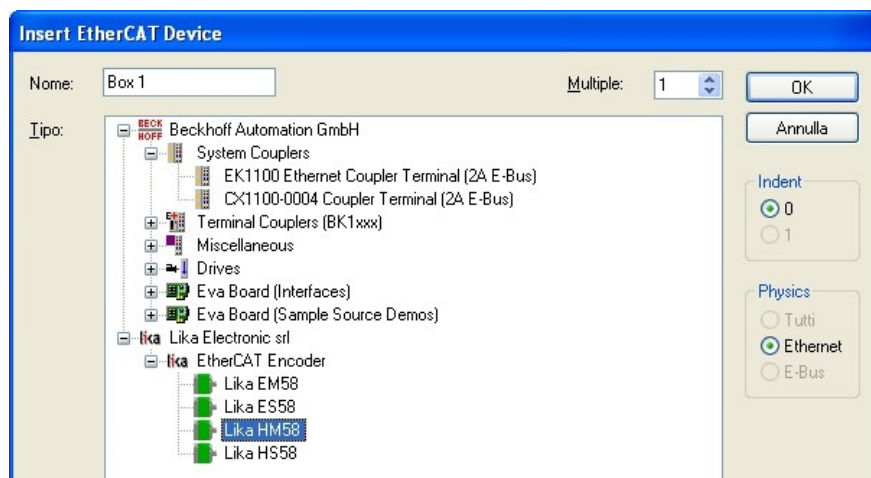
If you want to know the firmware version of a device, press the **Box (Lika EtherCAT HS58 -for XAC81-18-00-... or Lika EtherCAT EM58 -for XAC81-13-14-... or Lika EtherCAT HM58 -for XAC81-16-14-...)** item in the left pane of the **TwinCAT System Manager** main window: some tabbed pages for configuring and managing the device will appear in the right pane. Enter the **CoE - Online** page and refer to the **1009-00 Hardware version** and **100A-00 Software version** indexes.



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



The **Insert EtherCAT Device** window will appear.



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:

- Install **Lika HS58** item for **XAC81-18-00-EC2-...** model.
- Install **Lika EM58** item for **XAC81-13-14-EC2-...** model.
- Install **Lika HM58** item for **XAC81-16-14-EC2-...** model.



WARNING

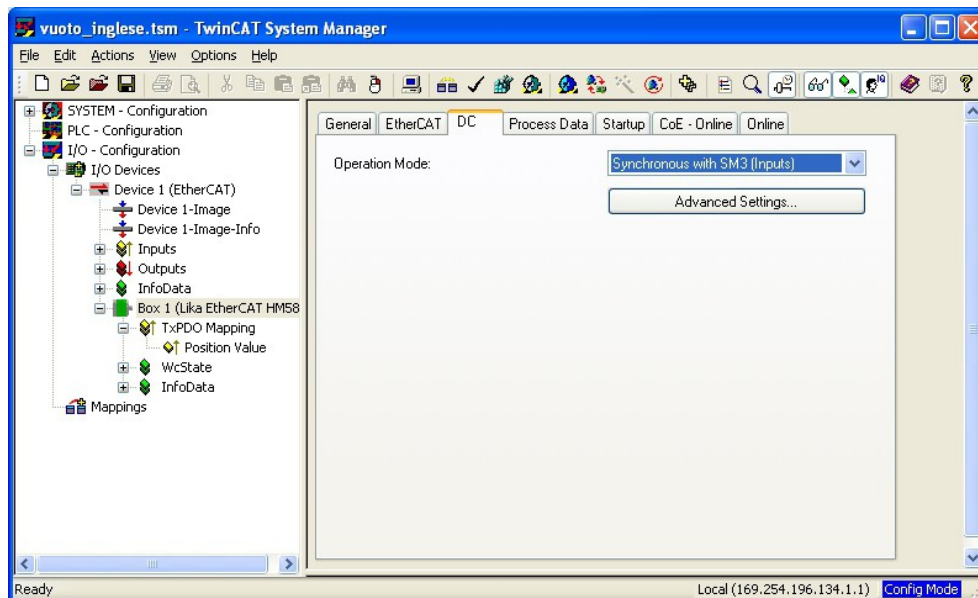
Please do not install **Lika ES58** item.

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

7.2 Setting the communication mode

7.2.1 Synchronous with SM3

In the left pane of the **TwinCAT System Manager** main window press the **Box** (**Lika EtherCAT HS58** -for XAC81-18-00-... or **Lika EtherCAT EM58** -for XAC81-13-14-... or **Lika EtherCAT HM58** -for XAC81-16-14-...) 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.

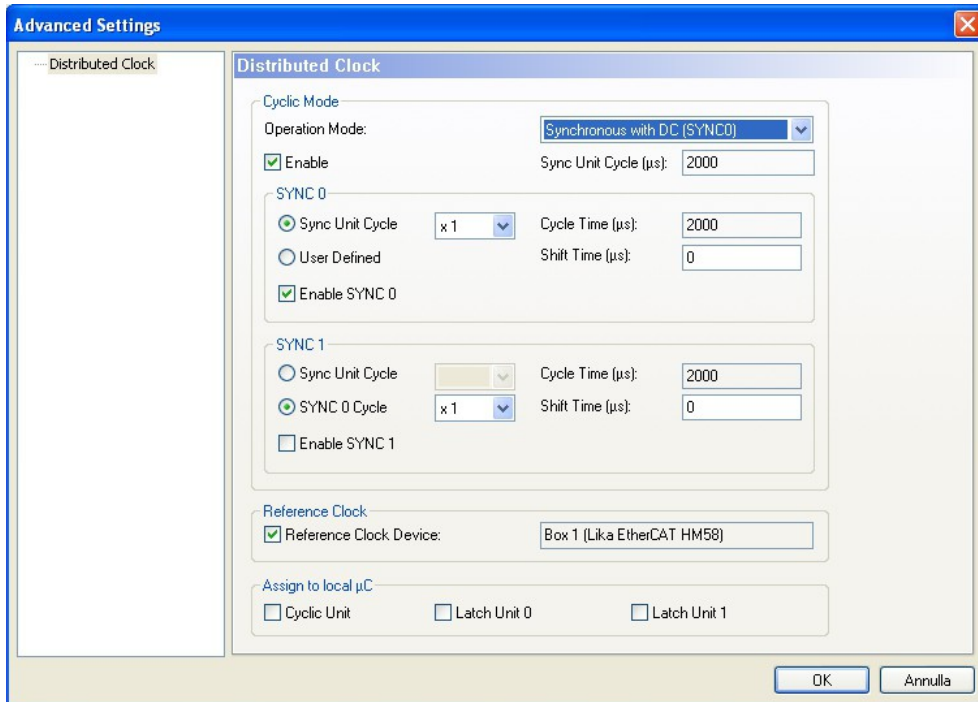


For any further information on the Synchronous with SM3 operation mode please refer to the "Synchronous with SM3" section on page 64 and to the **1C33 Sync Manager input parameter** object on page 77.

7.2.2 Synchronous with DC (SYNC0)

In the left pane of the **TwinCAT System Manager** main window press the **Box (Lika EtherCAT HS58 -for XAC81-18-00-... or Lika EtherCAT EM58 -for XAC81-13-14-... or Lika EtherCAT HM58 -for XAC81-16-14-...)** 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.

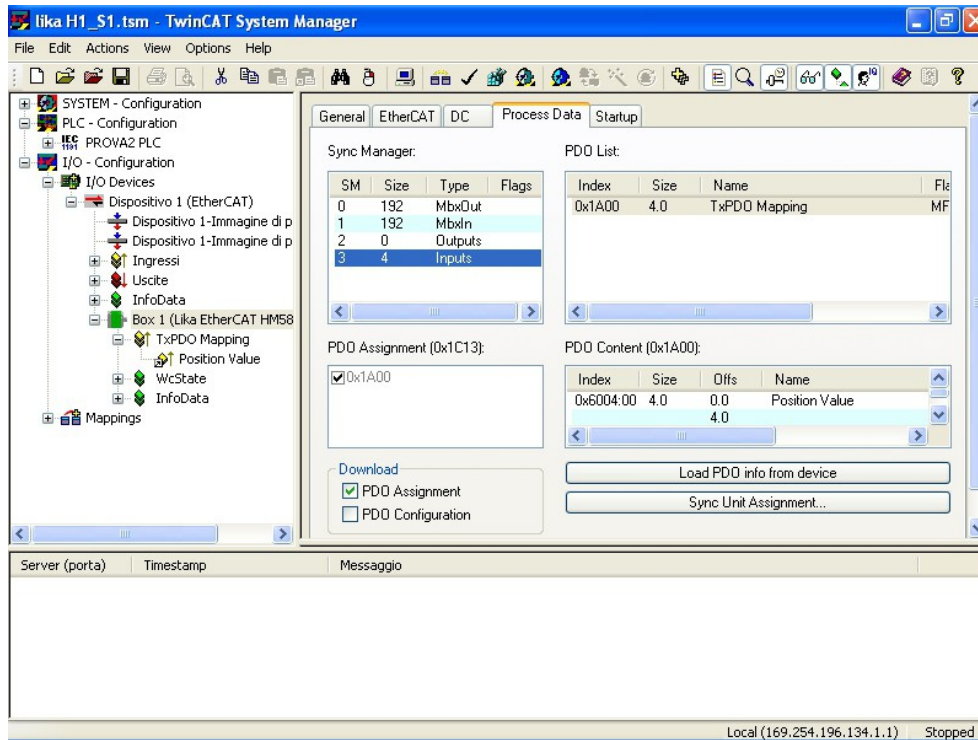


In the section tabbed **SYNC 0** set the cycle time next to the **Sync Unit Cycle** box; sync time is calculated as multiple (or sub-multiple) of the value set in the **Sync Unit Cycle (µs)** item right above.

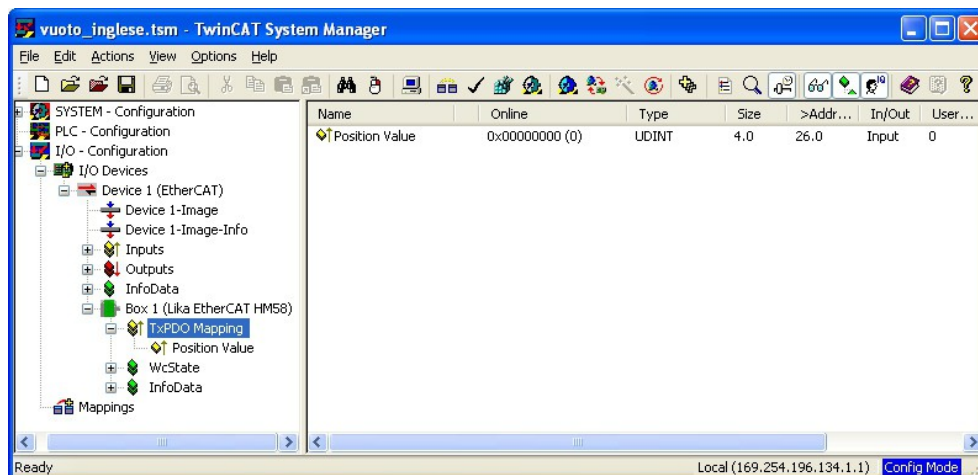
For any further information on the Synchronous with DC operation mode please refer to the "Synchronous with DC SYNC0" section on page 64 and to the **1C33 Sync Manager input parameter** object on page 77.

7.3 Process Data Objects

In the left pane of the **TwinCAT System Manager** main window press the **Box (Lika EtherCAT HS58 -for XAC81-18-00-... or Lika EtherCAT EM58 -for XAC81-13-14-... or Lika EtherCAT HM58 -for XAC81-16-14-...)** item. Expand the box to see Process Data Outputs (PDO). 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 (TxPDO Mapping) are shown.

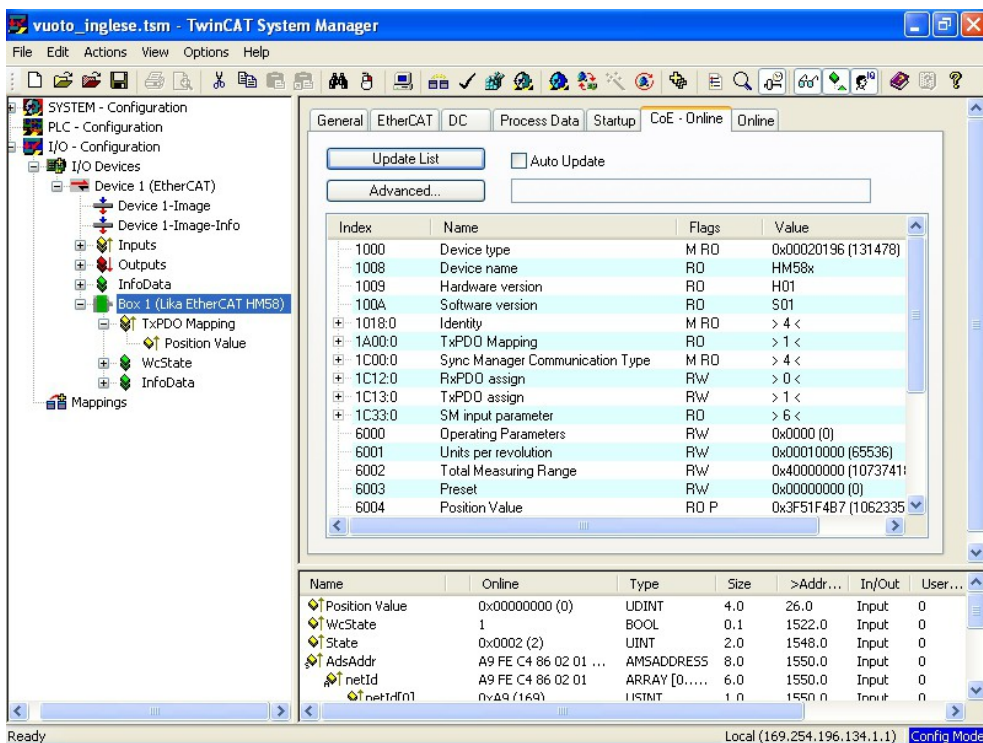


Process data objects can be displayed also by pressing the **TxPDO Mapping** item in the left pane of the **TwinCAT System Manager** main window; data is listed in the right pane.

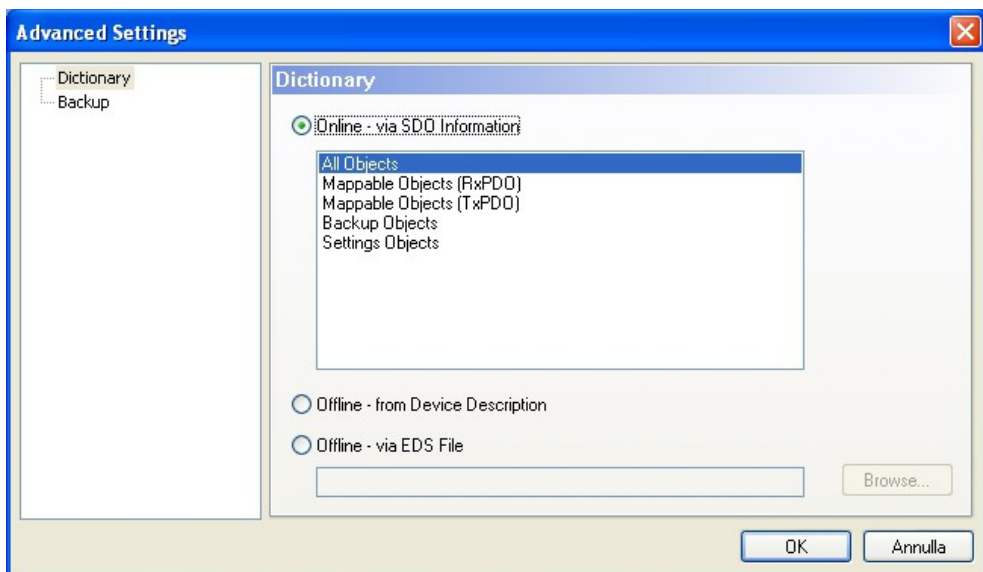


7.4 CoE Object Dictionary

In the left pane of the **TwinCAT System Manager** main window press the **Box (Lika EtherCAT HS58 -for XAC81-18-00-... or Lika EtherCAT EM58 -for XAC81-13-14-... or Lika EtherCAT HM58 -for XAC81-16-14-...)** 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 object dictionary is shown. This is the offline version of the object dictionary as read from the XML file.



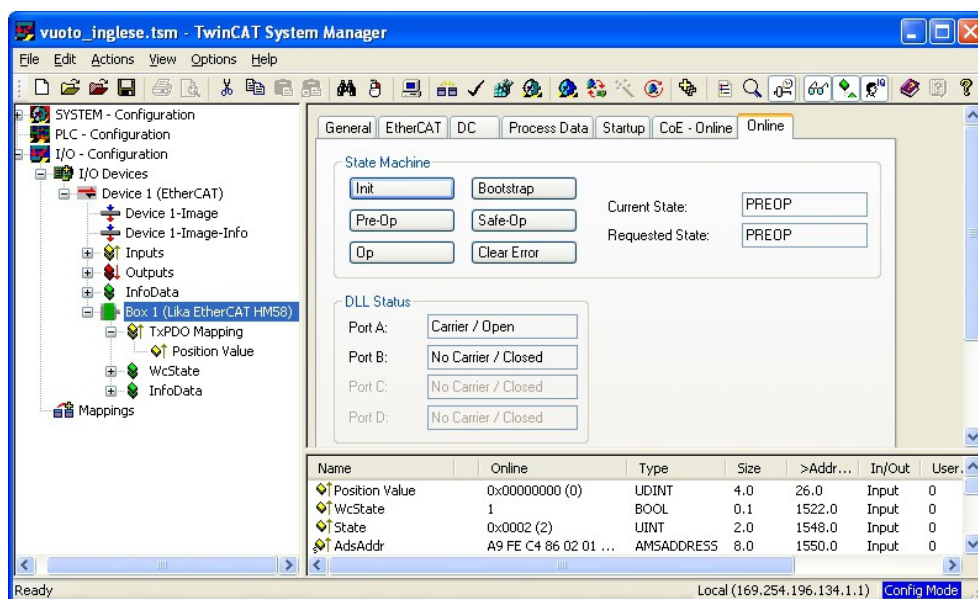
Objects can also be read directly from the encoder; to do this click the **Advanced...** button: the **Advanced Settings** window will appear.



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

7.5 Online Data

In the left pane of the **TwinCAT System Manager** main window press the **Box (Lika EtherCAT HS58 -for XAC81-18-00-... or Lika EtherCAT EM58 -for XAC81-13-14-... or Lika EtherCAT HM58 -for XAC81-16-14-...)** item: some tabbed pages for configuring and managing the device will appear in the right pane. Enter the **Online** page to check the state of the encoder.



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 (PDOs and SDOs) 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 (i.e. "load" if read backwards);
- save parameters: Data byte = 65 76 61 73hex = "**evas**" in ASCII code (i.e. "save" if read backwards).

7.6 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

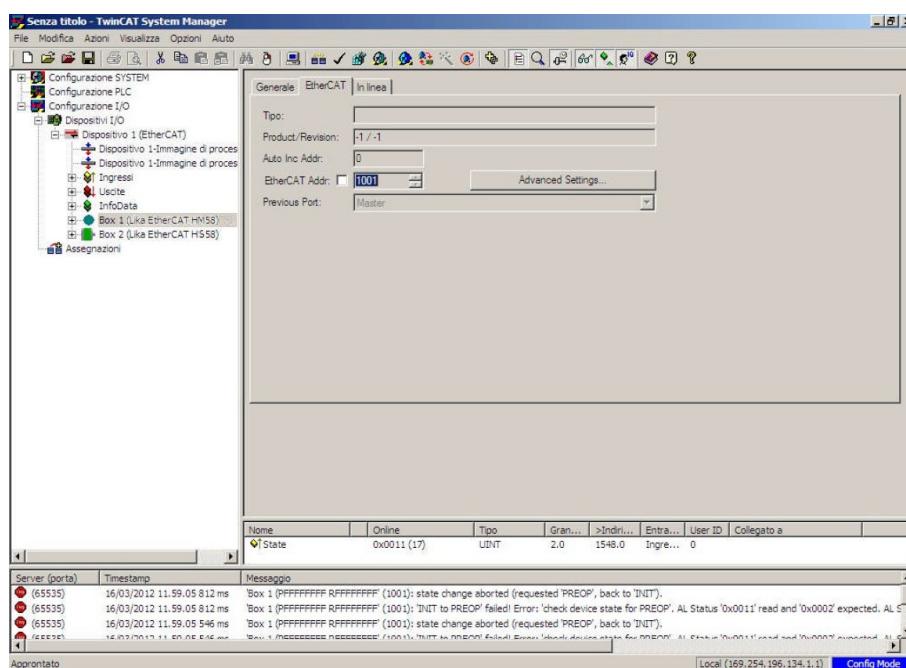
The XML file version, the firmware version and the EEPROM version must always comply. For example: if the firmware version is H1_S4 (Hardware version: 1; Software version: 4), it is mandatory that the EEPROM version is S4, therefore you must then install the XML file version V4.



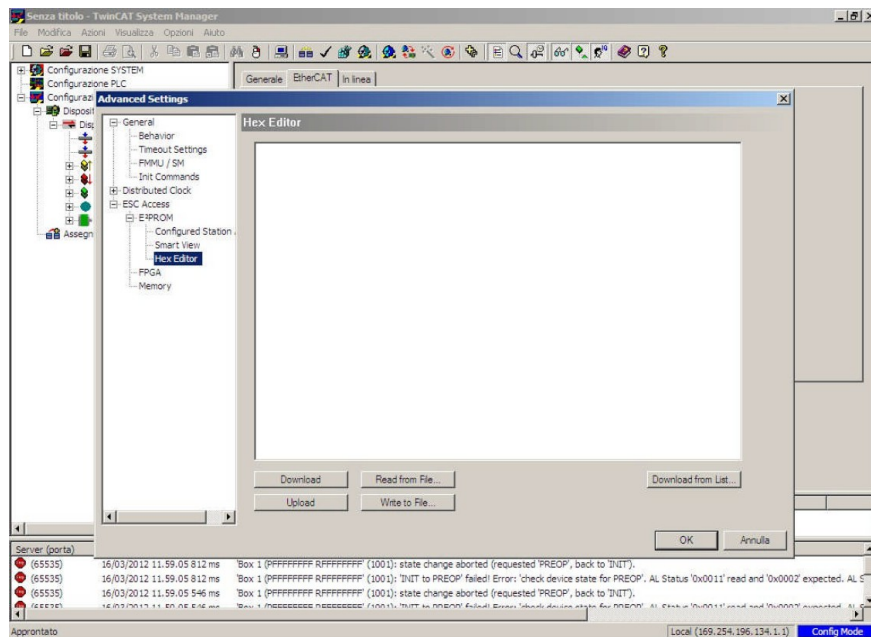
WARNING

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.

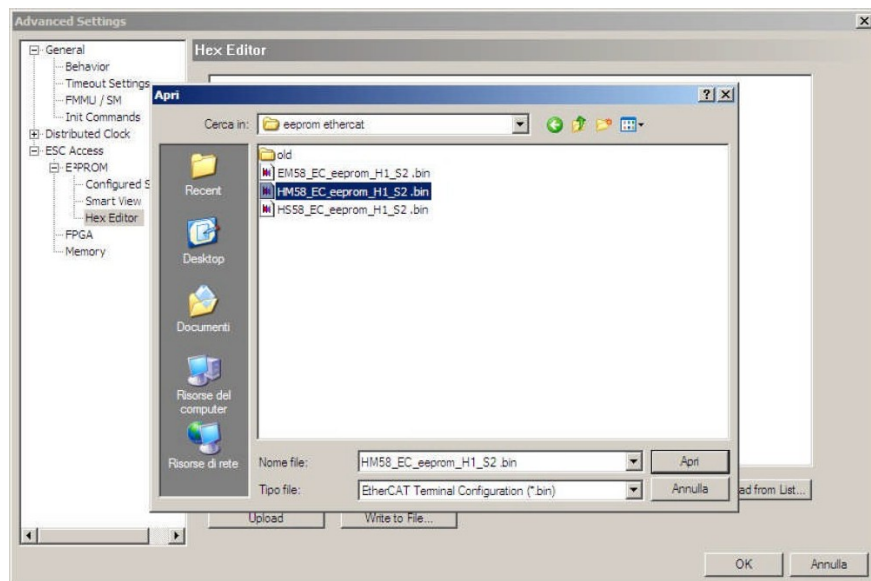
1. In the left pane of the **TwinCAT System Manager** main window press the **Box (Lika EtherCAT HS58 -for XAC81-18-00-... or Lika EtherCAT EM58 -for XAC81-13-14-... or Lika EtherCAT HM58 -for XAC81-16-14-...)** item of the encoder you need to update: some tabbed pages for configuring and managing the device will appear in the right pane. Enter the **EtherCAT** page.



2. 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.



3. Press the **Read from File...** button and select the .BIN file provided by Lika Electronic to upgrade the EEPROM; please make sure you select the file suitable for the model you need to upgrade (if you have to upgrade an XAC81-18-00-EC2-... singleturn encoder then you must select the file HS58_EC_eeprom_Hx_Sy.bin); if you have to upgrade an XAC81-13-14-EC2-... multiturn encoder then you must select the file EM58_EC_eeprom_Hx_Sy.bin; if you have to upgrade an XAC81-16-14-EC2-... multiturn encoder then you must select the file HM58_EC_eeprom_Hx_Sy.bin); finally press the **Open** button.

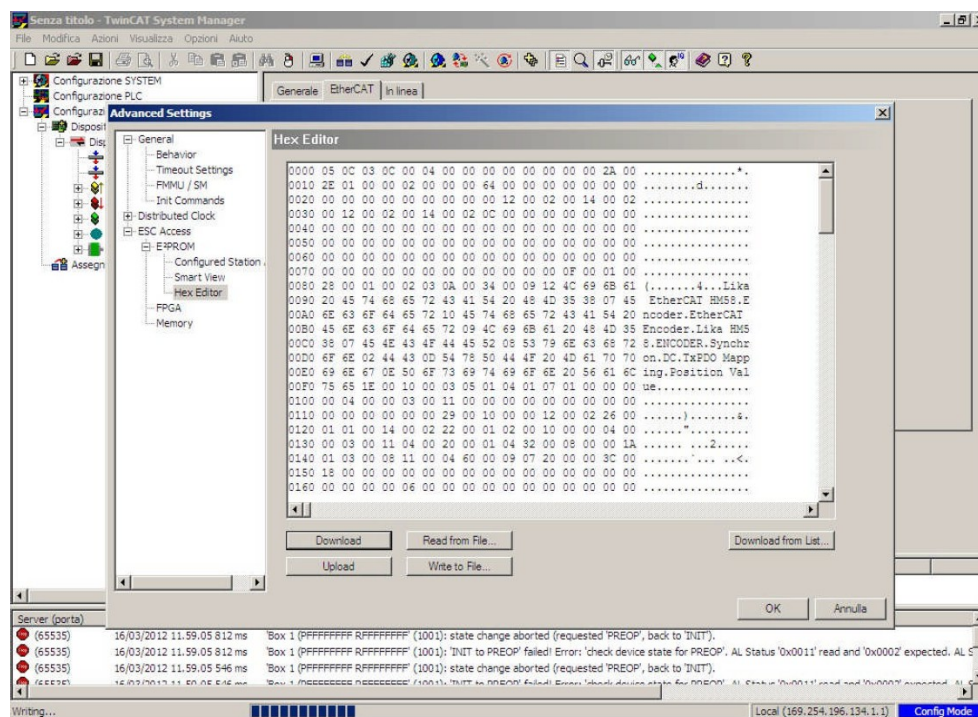




NOTE

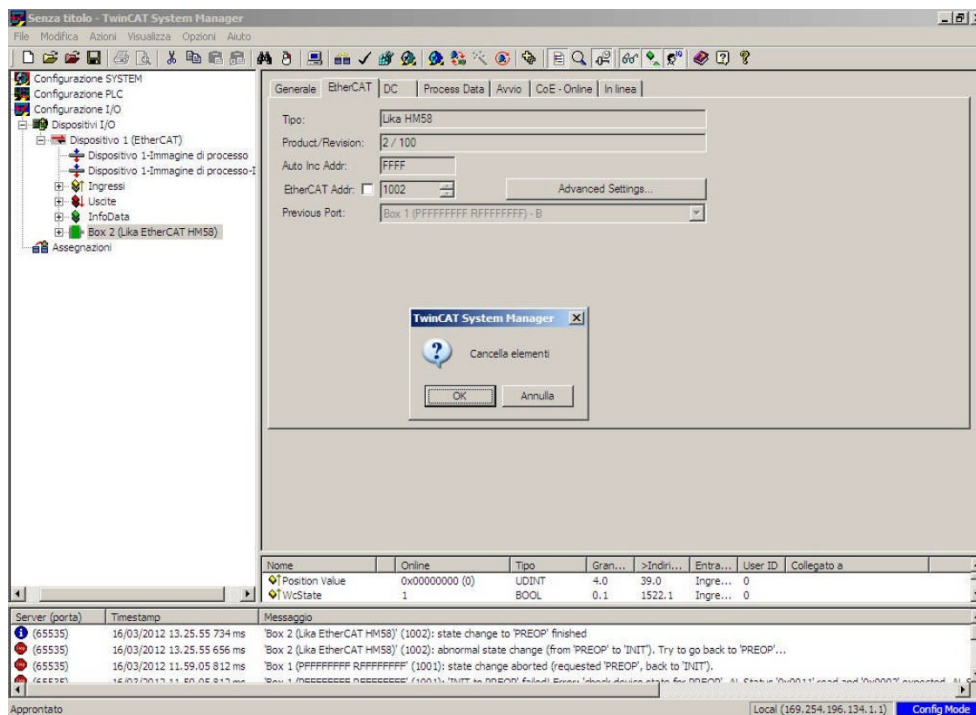
In the .BIN file Hx is the hardware version of the encoder, while Sy is the software version.

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 carried out press the **OK** button.

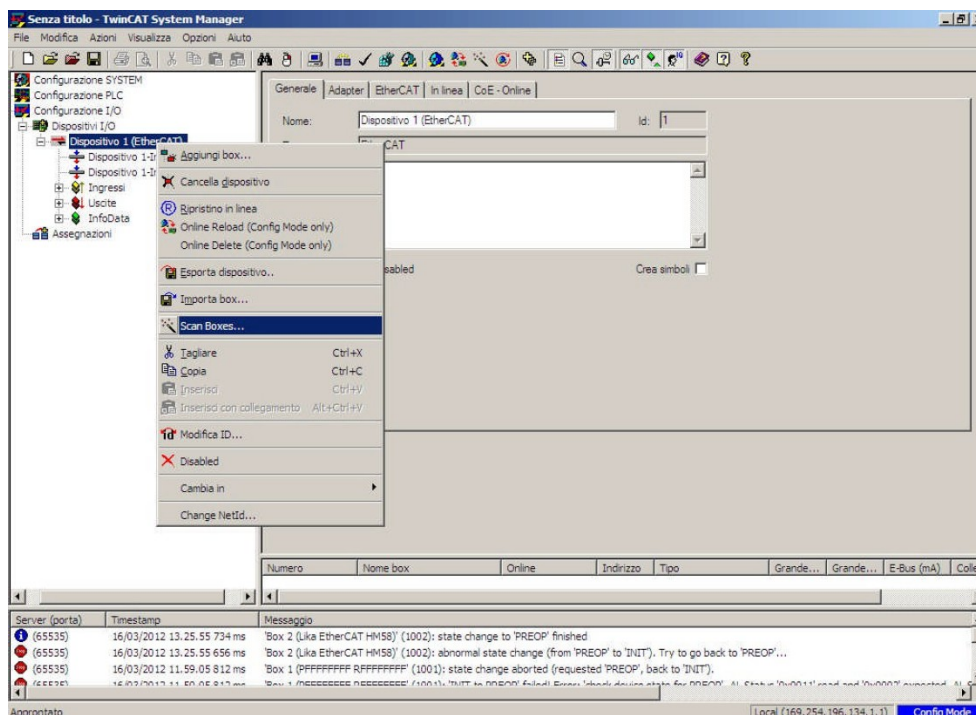


5. Now turn the power supply off, then on again.

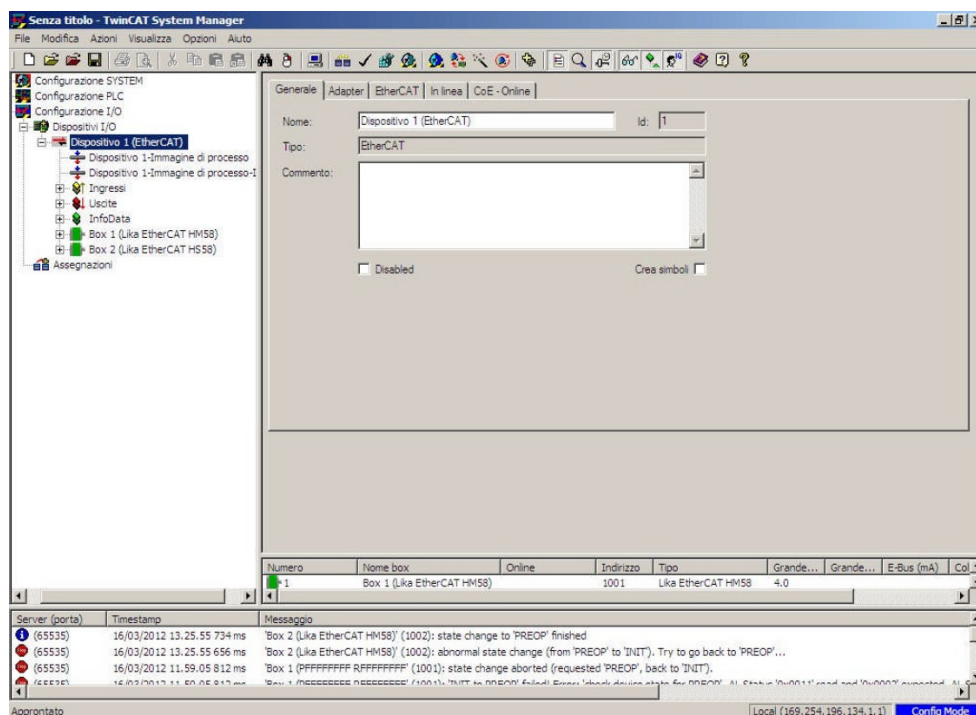
- In the left pane of the **TwinCAT System Manager** main window delete all the **Box (Lika EtherCAT HS58 -for XAC81-18-00-... or Lika EtherCAT EM58 -for XAC81-13-14-... or Lika EtherCAT HM58 -for XAC81-16-14-...)** items in the list. Select each box and then press the **DEL** key in the PC keyboard. 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.



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



7.7 Firmware upgrade



WARNING

The 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 operating.



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.

The 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.

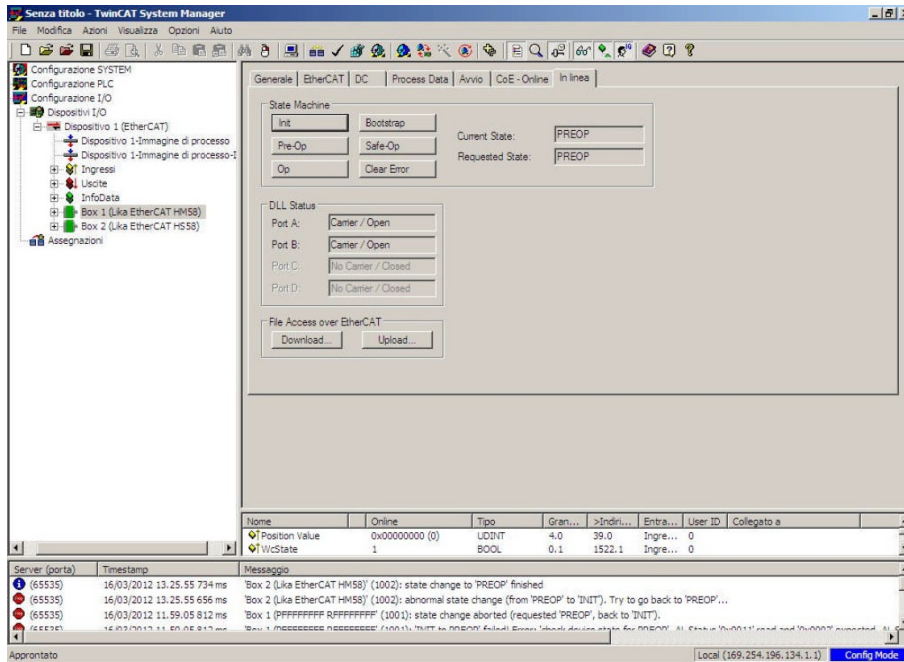
The firmware upgrading program consists of a single file having .EFW extension. It is released by Lika Electronic Technical Assistance & After Sale Service.



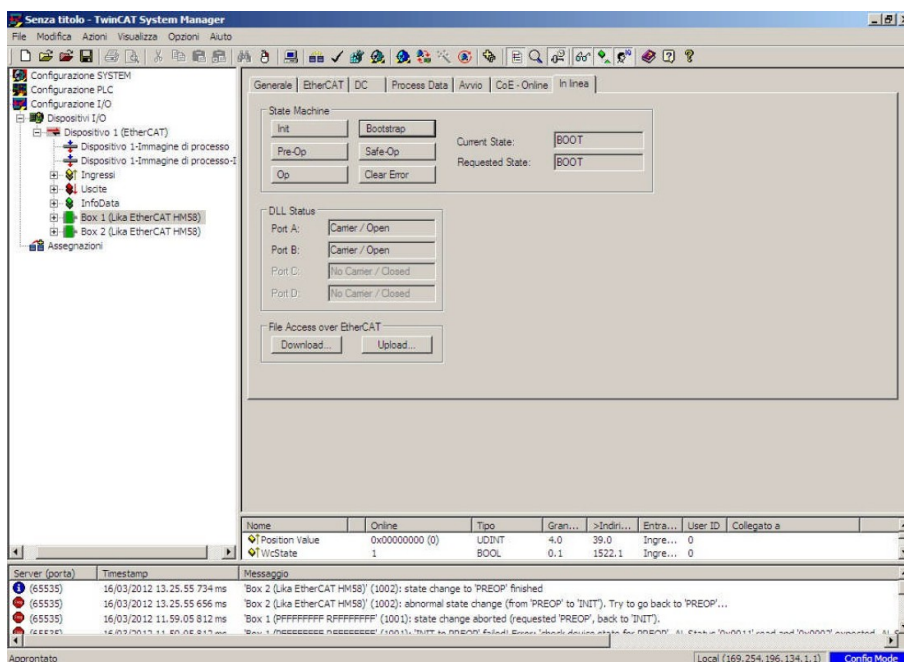
WARNING

You must upgrade the EEPROM before upgrading the firmware.

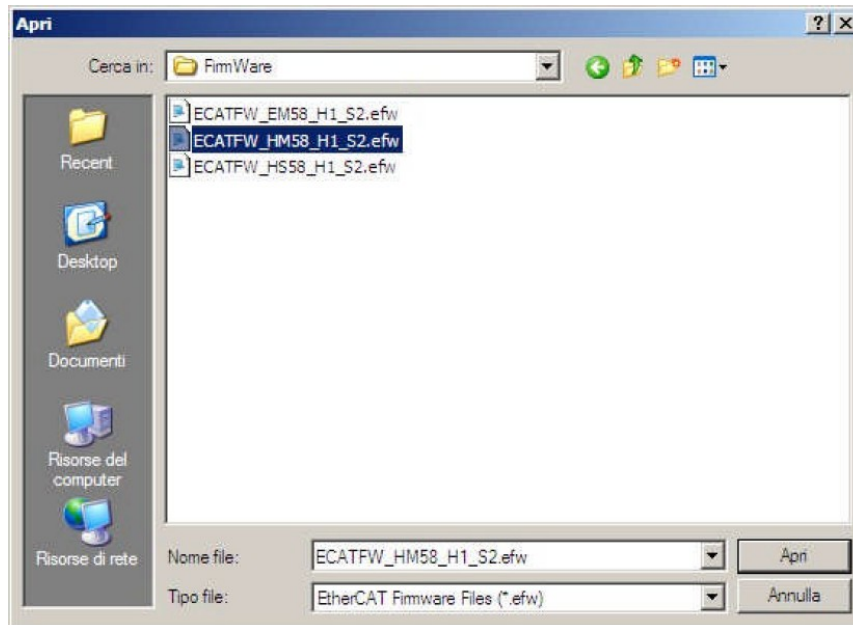
1. In the left pane of the **TwinCAT System Manager** main window press the **Box (Lika EtherCAT HS58 -for XAC81-18-00-... or Lika EtherCAT EM58 -for XAC81-13-14-... or Lika EtherCAT HM58 -for XAC81-16-14-...)** item of the encoder you need to update: some tabbed pages for configuring and managing the 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).



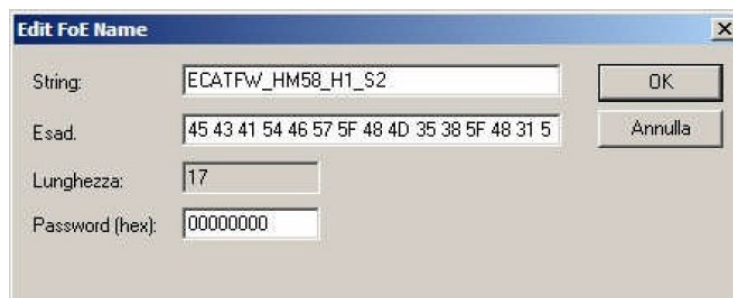
- Now press the **Download...** button in the **File Access Over EtherCAT** box; in the **Open** window that appears select the .EFW file provided to upgrade the firmware; please make sure you select the exact file of model you need to upgrade (if you have to upgrade an XAC81-18-00-EC2 singleturn encoder then you must select the file **ECATFW_HS58_Hx_Sy.efw**; if you have to upgrade an XAC81-13-14-EC2 multiturn encoder then you must select the file **ECATFW_EM58_Hx_Sy.efw**; if you have to upgrade an XAC81-16-14-EC2 multiturn encoder then you must select the file **ECATFW_HM58_Hx_Sy.efw**); finally press the **Open** button.



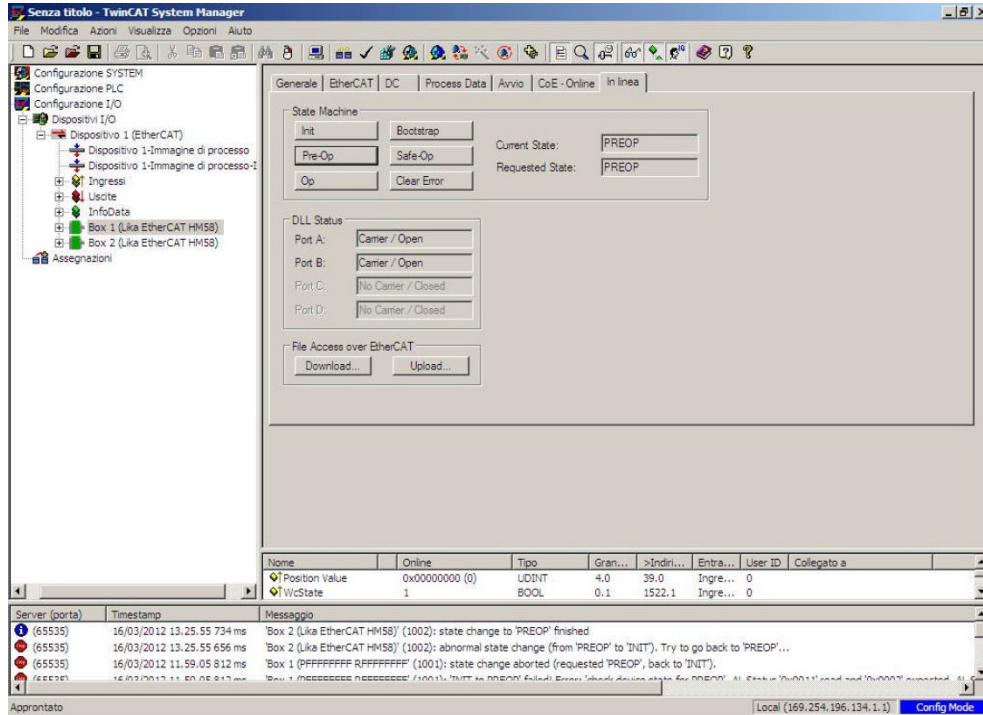
NOTE

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 00000000hex 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).



8 – EtherCAT® interface

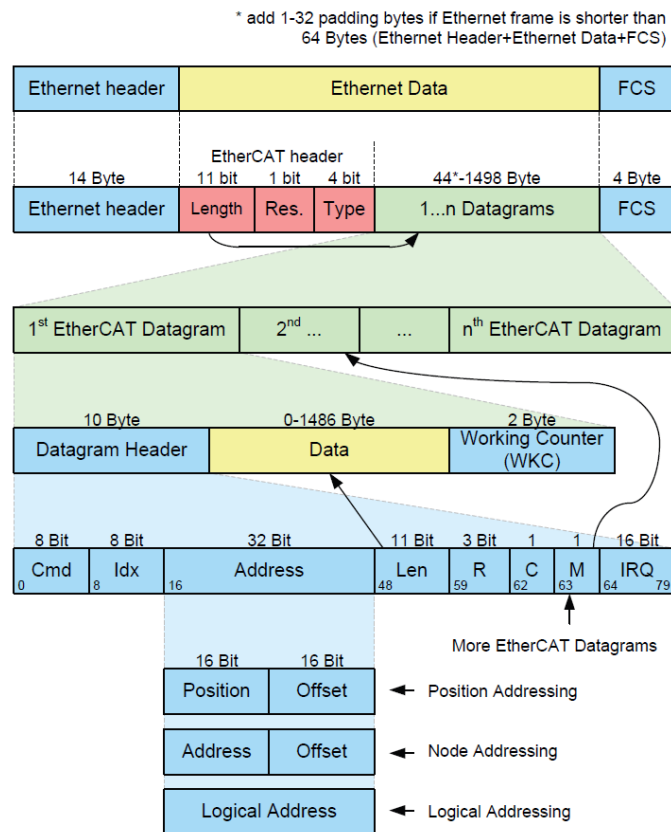
8.1 Basic Information on EtherCAT® protocol

The 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 a great 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 is transmitted among Master and Slaves using PDO (Process Data Objects) protocol. Each PDO message has inside one or more addresses for issuing data to the Slaves; data + address/es (and additional elements such as a validation checksum) joined together form an EtherCAT telegram (Datagram).

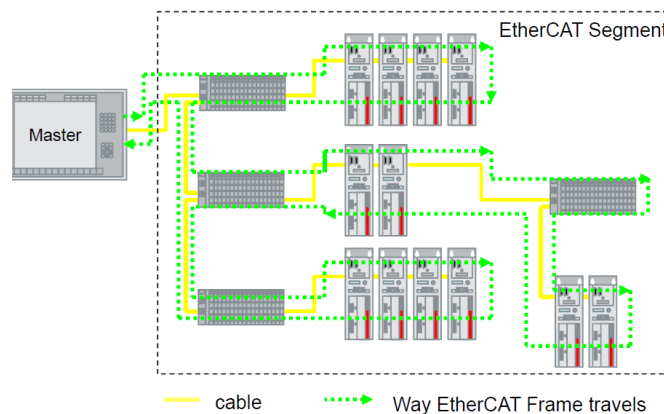
An EtherCAT frame can contain several telegrams and a complete control cycle often requires more than one frame.

8.1.1 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 the Master receives data from the 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 the 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.

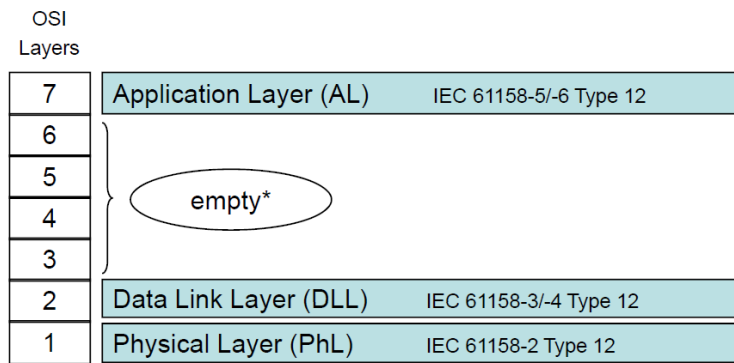
The last Slave issues back the complete frame to the Master with all the requested data (again passing through all the Slaves).



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 the CPU and the software processing.

8.1.2 ISO/OSI Layer model



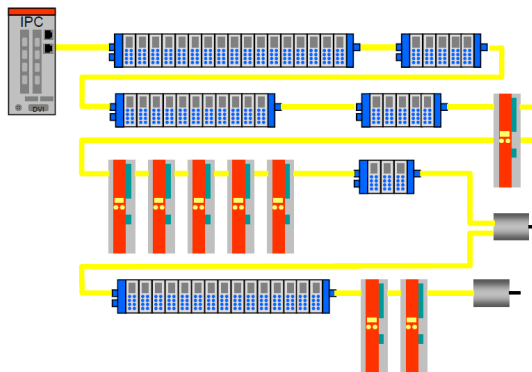
* "Empty" means that the layer behaviour exists, but is not shown explicitly.

8.1.3 Topology

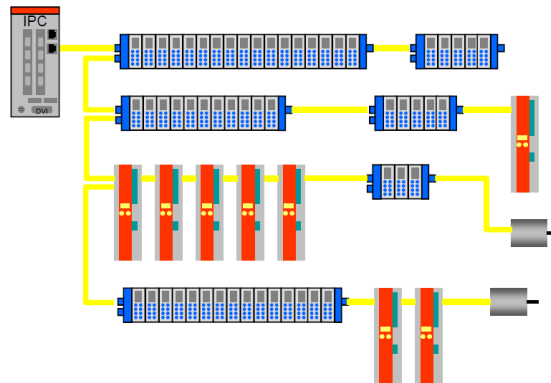
Several topologies of connection are supported by the EtherCAT networks: line, tree, daisy-chain, star, ...). EtherCAT networks can be configured in almost any topology in the same structure. The maximum length of the cable between two Slaves is 100 m / 328 ft; standard EtherCAT cables commercially available 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 65,535 devices can be connected. Some topology examples are shown in the Figures below:

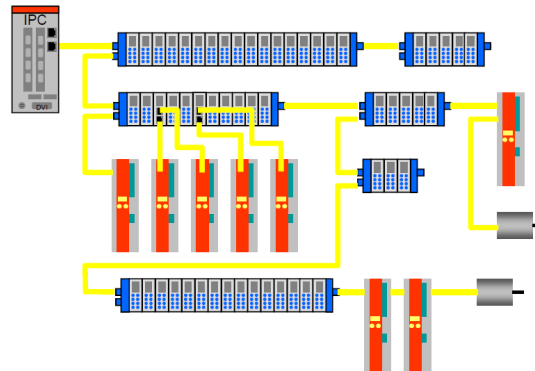
LINE topology:



TREE topology:



DAISY CHAIN with drop lines topology:



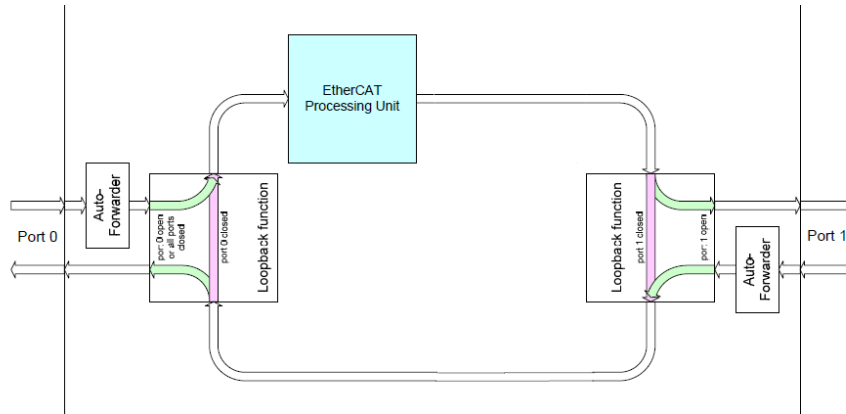
8.1.4 Line Termination

The 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.



8.1.5 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 the hardware configuration.

8 Bit	8 Bit	32 Bit		11 Bit	2	1	1	1	16 Bit
Cmd	Idx	Address		Len	R	C	R	M	IRQ
APxx		16 Bit	16 Bit						
		Position	Offset						
FPxx		Address	Offset						
Lxx		Logical Address							

← Auto Increment Addressing
(Position addressing)

← Fixed Physical Addressing
(Node addressing)

← Logical Addressing

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

- Auto Increment Addressing = Position Addressing: 16 bits indicate 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 the address 0 is the addressed device;
- Fixed Addressing = 16 bits indicate the physical address of the Slave inside the network while 16 bits are scheduled for addressing the local memory;
- Logical Addressing = 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).

8.1.6 Communication mode

Lika encoders with EtherCAT interface support the following operating modes:

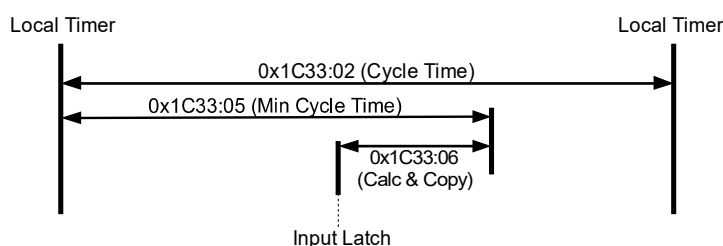
- FreeRun: asynchronous mode;
- SM3 event: synchronous mode;
- DC: distributed clock synchronization mode (synchronous mode).

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.

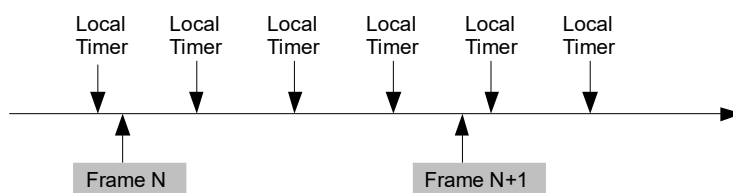
A reference parameter is the "Jitter": it represents the temporal fluctuation of the instant data sampling; in other words data sampled by the micro-controller is available in ECAT DPRAM memory after a certain time and the measure of the variability over time is the "jitter".

FreeRun

Asynchronous mode; the encoder position is sampled directly from the EtherCAT frame sent by the Master; the 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.

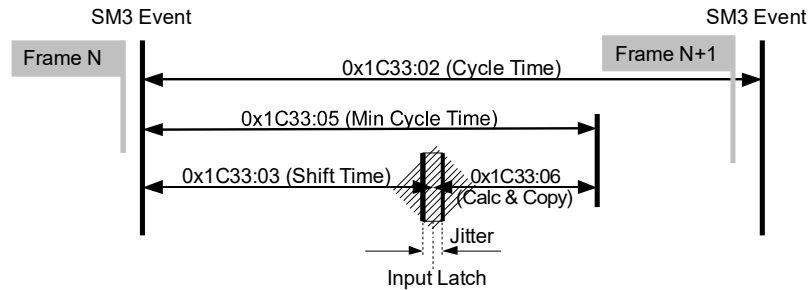


Description	Min	Typ	Max	
Jitter	0		100	µsec
Cycle Time	1000		64000	µsec

See the [1C33 Sync Manager input parameter](#) entry on page 77.

Synchronous with SM3

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



New data will be read by the Master at the next cycle (following SM3 event), so if the 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 the Master is reading (synchronous mode).

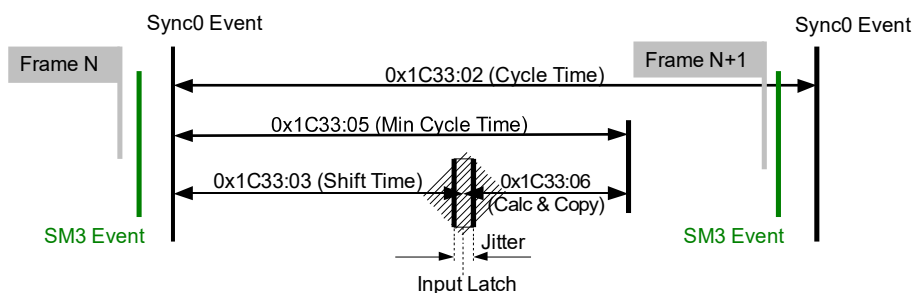
Description	Min	Typ	Max	
Jitter	0		7.2	ns
Cycle Time	62.5		64000	μs

See the [1C33 Sync Manager input parameter](#) entry on page 77.

Synchronous with DC SYNC0

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

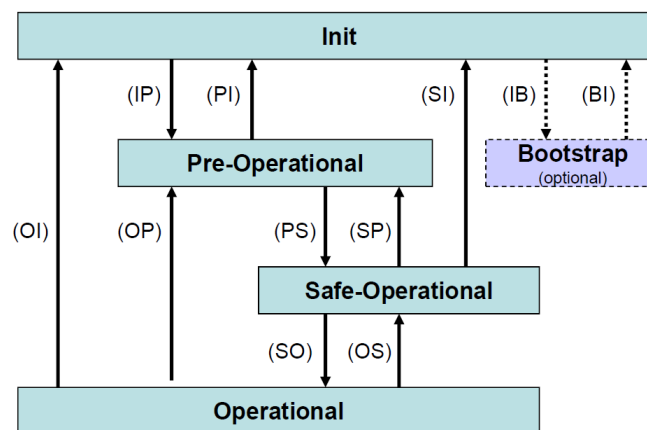
Time required for accomplishing these operations is set in the [1C33 Sync Manager input parameter](#) object; in particular in the **03 Shift Time** entry (1C33hex, sub3) and in the **06 Calc and Copy time** entry (1C33hex, sub6).



In this operating mode the "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 the jitter).

Description	Min	Typ	Max	
Jitter	0	100	200	µsec
Cycle Time	62.5		64000	µsec

8.1.7 EtherCAT State Machine (ESM)



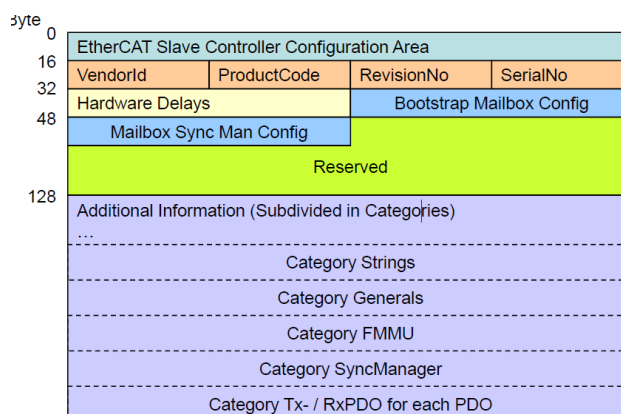
EtherCAT Slave is a state machine; the communication and the operating characteristics depend on the current state of the device:

- **INIT**: it is the default state after power-on; in this state there is not direct communication between the Master and the Slave on the Application Layer; some configuration registers are initialized and the Sync Managers are configured.
- **PRE-OPERATIONAL** (PREOP): in this state the mailbox is active; both the Master and the 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** (SAFEOP): in this state the Master and the Slave can issue only input process data, while the output process data is still in the **SAFE-OPERATIONAL** state;
- **OPERATIONAL** (OP): in this state the Master and the Slave are enabled to send both input process data and output process data.
- **BOOTSTRAP** (BOOT): no process data communication. Communication only via mailbox on Application Layer available. Special mailbox configuration is possible, e.g. larger mailbox size. In this state usually the FoE protocol is used for firmware download.

8.1.8 Slave configuration

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

EEPROM content (SII):



8.1.9 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".

The 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. The 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 an up-to-date "snapshot" of the system and accordingly take simultaneous actions in high sensitive applications such as the coordination in axis control operations. Besides, the 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 the Slaves clock.

Sync Manager

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

There are two synchronization modes:

- Buffered Mode (3-Buffer Mode);

- Mailbox 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 the EtherCAT Master and the ESC controllers; no timing restrictions are imposed.

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

Customarily this mode is used for cyclic data exchange, i.e. process data communication.

Mailbox Mode (1-Buffer Mode)

In this mode a "handshake" between the Master and the Slave must be used; in fact one only memory buffer is available to both the Master and the 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. The mailbox mode is typically used for application layer protocols and exchange of acyclic data (e.g. parameter settings).

The encoder features four Sync Managers, see the [1C00 Sync Manager Communication Type](#) object on page 76:

- **Sync Manager 0 (SM MailBox Receive, SM0)**
Used for mailbox write transfers (Master to Slave).
The module has a configurable write mailbox size with default size of 276 bytes, corresponding to 255 bytes plus relevant protocol headers and padding.
- **Sync Manager 1 (SM MailBox Send, SM1)**
Used for mailbox read transfers (Slave to Master).
The module has a configurable read mailbox size with default size of 276 bytes, corresponding to 255 bytes plus relevant protocol headers and padding.
- **Sync Manager 2 (SM PDO output, SM2)**
It contains the RxPDOs (i.e., Sync Manager 2 holds the Read Process Data).
- **Sync Manager 3 (SM PDO input, SM3)**
It contains the TxPDOs (i.e., Sync Manager 3 holds the Write Process Data).

8.2 CANopen Over EtherCAT (CoE)

Lika encoders are Slave devices and support the "CanOpen Over EtherCAT" (COE) mode for data transfer. In particular, they support the "CANopen DS 301 Communication profile", Class 2 and the "CANopen DS 406 Device profile for encoders".

For any omitted specification on CANopen® protocol, please refer to the "CiA Draft Standard Proposal 301. Application Layer and Communication Profile" and to the "CiA Draft Standard 406. Device profile for encoders" documents available at the address www.can-cia.org.

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

8.2.1 XML file

XAC81 EtherCAT® encoders are supplied with an XML file **Lika_Ex58_Hx58_EC_Vx.xml** (download it from Lika's corporate web site www.lika.biz). The XML file has to be installed on EtherCAT® Master device.

Please install one of the following encoders:

- Install **Lika HS58** encoder for **XAC81-18-00-EC2-...** singleturn model.
- Install **Lika EM58** encoder for **XAC81-13-14-EC2-...** multiturn model.
- Install **Lika HM58** encoder for **XAC81-16-14-EC2-...** multiturn model.



WARNING

Please do not install **Lika ES58** encoder.

For any information on the firmware upgrade procedure refer to the "7.7 Firmware upgrade" section on page 54. For any information on the EEPROM upgrade procedure refer to the "7.6 EEPROM upgrade" section on page 49.



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.

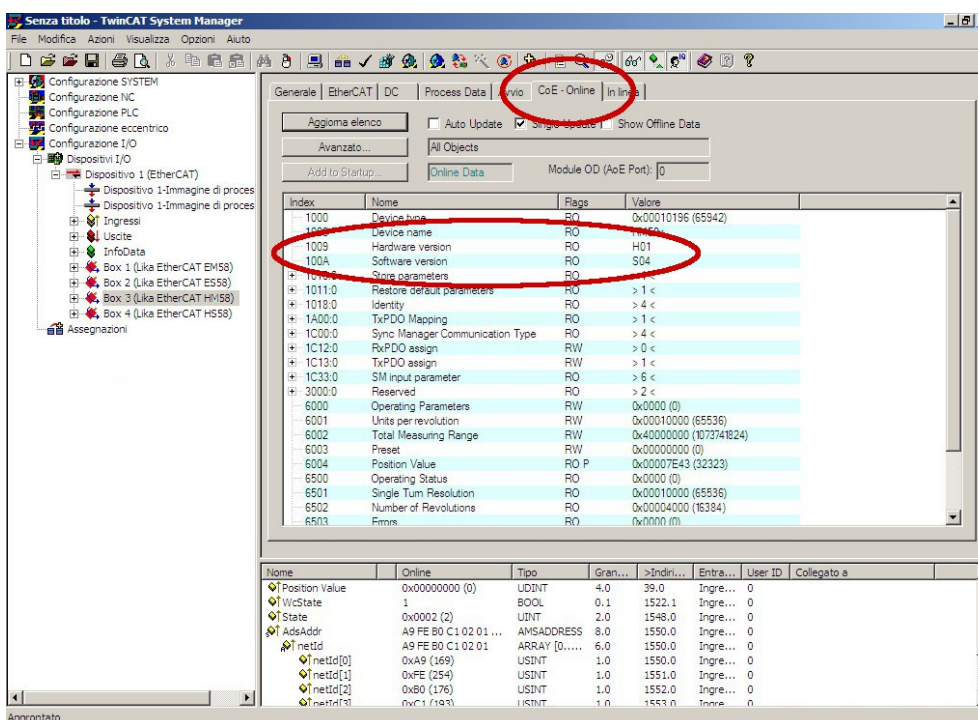


WARNING

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.

If you want to know the firmware version of a device, press the **Box (Lika EtherCAT HS58 -for XAC81-18-00-... or Lika EtherCAT EM58 -for XAC81-13-14-... or Lika EtherCAT HM58 -for XAC81-16-14-...)** 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 **1009-00 Hardware version** and **100A-00 Software version** indexes.



8.2.2 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 the Slave side.

8.2.3 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.

8.2.4 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".

8.2.5 Object dictionary

The most important part of a device profile is the Object Dictionary. The Object Dictionary is essentially a grouping of objects accessible via the network in an ordered, pre-defined mode. Each object within the dictionary is addressed using a 16-bit index.

The Object Dictionary can contain a maximum of 65,536 entries.

The user-related objects are grouped in three main areas: the Communication Profile Area, the Manufacturer Specific Profile Area and the Standardised Device Profile Area. The objects are all described in the XML file.

The **Communication Profile Area** at indexes from 1000h to 1FFFh contains the communication specific parameters for the EtherCAT network. These entries are common to all devices. PDO objects and SDO objects are described in this section. The Communication Profile Area objects comply with the "CiA Draft Standard Proposal 301. Application Layer and Communication Profile". Refer to the "Communication Profile Area objects (DS301)" section on page 73.

The **Manufacturer Specific Profile Area** at indexes from 2000h to 5FFFh is free to add manufacturer-specific functionality. No items are available in this area.

The **Standardised Device Profile Area** at indexes from 6000h to 9FFFh contains all data objects common to a class of devices that can be read or written via the network. The device profiles may use entries from 6000h to 9FFFh to describe the device parameters and the device functionality. The Standardised Device Profile Area objects comply with the "CiA Draft Standard 406 CANopen Device profile for encoders". Refer to the "Standardised Profile Area objects (DS406)" section on page 79.

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

Signed8 / Unsigned8 data type:

Process data bytes							
byte 4							
7	6	5	4	3	2	1	0
MSbit		...				LSbit	

Signed16 / Unsigned16 data type:

Process data bytes	
byte 4	byte 5
LSByte	MSByte

Signed32 / Unsigned32 data type:

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

**NOTE**

Always save the new values after setting in order to store them in the non-volatile memory permanently. Use the **1010-01 Store parameters** object, see on page 74.

Should the power supply be turned off all data that has not been saved previously will be lost!

Communication Profile Area objects (DS301)

1000-00 Device type

[Unsigned32, ro]

It contains information about the device type. The object describes the type of device and its functionality.

Default = 0001 0196h = singleturn encoder, in compliance with DS 406
 0002 0196h = multiturn encoder, in compliance with DS 406

1008-00 Device Name

[String, ro]

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

Default = 4853353878h = "HS58x" = singleturn encoder version,
 XAC81-18-00-EC2-... singleturn encoder
 454D353878h = "EM58x" = multiturn encoder version,
 XAC81-13-14-EC2-... multiturn encoder
 484D353878h = "HM58x" = multiturn encoder version,
 XAC81-16-14-EC2-... multiturn encoder

1009-00 Hardware version

[String, ro]

It shows the hardware version of the device, expressed in hexadecimal ASCII code.



EXAMPLE

483031h = H01 = Hardware version 01

Default = device dependent

100A-00 Software version

[String, ro]

It shows the software version of the device, expressed in hexadecimal ASCII code.



EXAMPLE

533032h = S02 = Software version 02

Default = device dependent

1010-01 Store parameters

[Unsigned32, rw]

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

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

Master → Encoder

Cmd specific data							
Cmd	Index		Sub	Data			
23	10	10	01	73	61	76	65
				s	a	v	e

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 the operator 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 91.

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

Master → Encoder

Cmd specific data							
Cmd	Index		Sub	Data			
23	11	10	01	6C	6F	61	64
				l	o	a	d

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 the **1010-01 Store parameters** object). When the power is turned off, parameters not saved are lost.

1018 Identity

[Unsigned8, ro]

This object contains general information about the device. Sub-index 00 contains the number of the entries.

Default = 4

01 Vendor ID

[Unsigned32, ro]

It provides the manufacturer-specific vendor ID. The EtherCAT vendor ID is the same as the CANopen vendor ID.

Default = 0000 012Eh

02 Product code

[Unsigned32, ro]

The manufacturer-specific product code identifies a specific device version.

Default = 0000 0001h Lika EtherCAT HS58, singleturn encoder (i.e. XAC81-18-00-...)
 0000 0002h Lika EtherCAT EM58 and HM58, multiturn encoder (i.e. XAC81-13-14-... and XAC81-16-14-...)

03 Revision

[Unsigned32, ro]

The manufacturer-specific revision number consists of a major revision number and a minor revision number. The major revision number identifies a specific device behaviour. The minor revision number identifies different version with the same device behaviour.

Default = 0001 0001h Lika EtherCAT EM58 series (i.e. XAC81-13-14-...)
 0001 0000h Lika EtherCAT HS58 and HM58 series (i.e. XAC81-18-00-... and XAC81-16-14-...)

7	...	0	15	...	8	23	...	16	31	...	24
Minor revision number						Major revision number					
LSB					MSB	

04 Serial number

[Unsigned32, ro]

It provides the Serial Number of the device. It is 0 if no serial number is provided.

Default = 0000 0000h (= not used)

1A00-01 PDO mapping parameter

[Unsigned8, ro]

This object contains the mapping parameters for the PDOs the EtherCAT device is able to transmit. Sub-index 00 contains the number of entries.

01 Mapped Object 001

[Unsigned32, rw]

Sub-index 01 contains the information of the mapped application object 001. The object describes the content of the PDO by its index, sub-index and length. The length section contains the length of the application object in bits. This may be used to verify the mapping.

7	0	15	8	23	16	31	24								
Length		Sub-Index		Index											
LSB								MSB							

Default = 6004 0020h = **6004-00 Position value** object, length 32 bits

1C00 Sync Manager Communication Type

[Unsigned8, ro]

This object contains the number and type of Sync Manager Communication Types supported by the encoder. Sub-index 00 specifies the number of Sync Manager channels. Refer also to the "Sync Manager" section on page 66.

01 SM MailBox Receive (SM0)

[Unsigned8, ro]

Used for mailbox write transfers (Master to Slave).

Default = 01h

02 SM MailBox Send (SM1)

[Unsigned8, ro]

Used for mailbox read transfers (Slave to Master).

Default = 02h

03 SM PDO output (SM2)

[Unsigned8, ro]

It contains the RxPDOs (i.e. Sync Manager 2 holds the Read Process Data).

Default = 03h

04 SM PDO input (SM3)

[Unsigned8, ro]

It contains the TxPDOs (i.e. Sync Manager 3 holds the Write Process Data).

Default = 04h

1C12-00 Sync Manager RxPDO Assigned

[Unsigned8, ro]

This object specifies whether the device uses Receive PDO messages. This device does not support Receive PDO messages.

Default = 00h

1C13-01 Sync Manager TxPDO Assigned

[Unsigned32, ro]

This object specifies whether the device uses Transmit PDO messages. Sub-index 00 specifies the number of entries, i.e. the number of assigned TxPDOs.

01 Subindex 001

This device uses TxPDO messages to send the position value.

Default = 0000 1A00h = **1A00-01 PDO mapping parameter** object

1C33 Sync Manager input parameter

1C33 Sync Manager input parameter object contains the input synchronization parameters. Some of them are calculated dynamically and depend on both the encoder configuration (programmed resolution, counting direction, ...) and the synchronization mode (SM or DC). Sub-index 00 contains the number of entries.

01 Sync Type

[Unsigned16, rw]

It allows to select the synchronization mode. For more information refer to page 63.

0: FreeRun: see on page 63;

1: Synchronous with SM3 Event: see on page 64;

2: DC mode synchronous with SYNC0 event: see on page 64.

Default = 1

02 Cycle time

[Unsigned32, ro]

This parameter depends on the **01 Sync Type** selected. Application cycle time, i.e. interval between two position samplings (internal timer). The value is expressed in nanoseconds (ns).

If 0 = "FreeRun": interval between two position samplings (internal timer).

If 1 = "Synchronous with SM3": minimum interval between two SM3 events.

If 2 = "DC mode synchronous with SYNC0 event": 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 and expressed in nanoseconds (ns).

04 Sync modes supported

[Unsigned16, ro]

It shows the list of the supported synchronization modes.

bit 0: FreeRun (supported)

bit 1: Synchronous with SM3 (supported)

bit 2: Synchronous with DC SYNC0 (supported)

Default = 7

05 Minimum cycle time

[Unsigned32, ro]

Min. duration of the encoder internal cycle time. This parameter is calculated dynamically and depends on the operating parameters and the position value. It is expressed in nanoseconds (ns).

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. It is expressed in nanoseconds (ns).

**NOTE**

Always save the new values after setting in order to store them in the non-volatile memory permanently. Use the **1010-01 Store parameters** object, see on page 74.

Should the power supply be turned off all data that has not been saved previously will be lost!

Standardised Profile Area objects (DS406)

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 values are highlighted in bold

Default = 0000h

Code sequence

This is intended to set whether the count is increasing (count up information) when the shaft of the rotary encoder rotates clockwise (CW) or counter-clockwise (CCW). Setting 0 (bit 0 = 0) causes the encoder counting to increase when the encoder shaft rotates clockwise; setting 1 (bit 0 = 1) causes the encoder counting to increase when the encoder shaft rotates counter-clockwise. CW and CCW rotations are viewed from the shaft end.

To know whether the **Code sequence** is currently set to CW or CCW, you can read the bit 0 **Code sequence** in the **6500-00 Operating status** object, see on page 85.



WARNING

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

Scaling function

This is meant to disable (0) / enable (1) the scaled parameters **6001-00 Units per revolution** and **6002-00 Total Measuring Range**.

When the scaling function is disabled (bit 2 = 0), the encoder uses the physical singleturn resolution and the physical multiturn resolution (i.e. the hardware counts per revolution and the 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**) to arrange the absolute position information; the **6001-00 Units per revolution** and **6002-00 Total Measuring Range** objects are ignored.

On the contrary, when the scaling function is enabled (bit 2 = 1), the user is allowed to enter the custom singleturn resolution in the **6001-00 Units per revolution** object and the custom total resolution in the **6002-00 Total Measuring Range** object and these values are used to calculate the position information.

To know whether the **Scaling function** is currently enabled, you can read the bit 2 **Scaling function** of the **6500-00 Operating status** object, see on page 86.



WARNING

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

6001-00 Units per revolution

[Unsigned32, rw]



WARNING

This object is active only if the bit 2 **Scaling function** in the **6000-00 Operating parameters** object is set to "1"; otherwise it is ignored and the system uses the physical values (**6501-00 Hardware counts per revolution** and **6502-00 Hardware number of turns**) to calculate the position information.

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

To avoid counting errors, check that

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

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

Default = 0040 0000h (262,144)	for XAC81-18-00-EC singleturn encoder
0000 2000h (8,192)	for XAC81-13-14-EC multiturn encoder
0001 0000h (65,536)	for XAC81-16-14-EC multiturn encoder



WARNING

When you set a new value next to the **6001-00 Units per revolution** object, please always check also the **6002-00 Total Measuring Range** object value and be sure that the resulting number of revolutions complies with the **Hardware number of revolutions** of the device (see the **6502-00 Hardware number of turns** object).

$$\frac{\text{6002-00 Total Measuring Range}}{\text{6001-00 Units per revolution}} \leq \text{Number of physical revolutions}$$

Let's suppose that the XAC81-16-14-EC2-... multiturn encoder is programmed as follows:

6001-00 Units per revolution: 65,536 cpr

6002-00 Total Measuring Range = 268 435 456₁₀ = 65,536 (cpr) * 4,096 (rev.)

Let's set a new singleturn resolution, for instance: **6001-00 Units per revolution** = 360 cpr.

If we do not change the **6002-00 Total Measuring Range** value at the same time, we will get the following result:

$$\text{Number of revolutions} = \frac{268\,435\,456 \text{ (6002-00 Total Measuring Range)}}{360 \text{ (6001-00 Units per revolution)}} = 745,654.04...$$

As you can see, the encoder is required to carry out more than 745,000 revolutions, this cannot be because the hardware number of revolutions can be max. 16,384. When this happens, the encoder falls into an error.



WARNING

When you enable the scaling function (bit 2 **Scaling function** = 1), please enter scaled values next to the **6001-00 Units per revolution** and **6002-00 Total Measuring Range** objects that are consistent with the physical values. In the case of inconsistent values, the system will warn about the wrong parametrization and fault condition by means of the dedicated objects.



WARNING

Every time you change the scaled values (see the **6001-00 Units per revolution** and **6002-00 Total Measuring Range** objects), then you are required to set a new preset value (see the **6003-00 Preset** object).

6002-00 Total Measuring Range

[Unsigned32, rw]



WARNING

This object is active only if the bit 2 **Scaling function** in the **6000-00 Operating parameters** object is set to "1"; otherwise it is ignored and the system uses the physical values (**6501-00 Hardware counts per revolution**

and **6502-00 Hardware number of turns**) to calculate the position information.

This object sets a custom number of distinguishable steps over the total measuring range. The total resolution of the encoder results from the product of **6001-00 Units per revolution** by the required **Number of revolutions**.

You are allowed to set whatever integer value less than or equal to the **overall hardware resolution** (see the encoder identification label as well as **6501-00 Hardware counts per revolution** and **6502-00 Hardware number of turns** objects). The overall hardware resolution results from:

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

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

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

$$\text{Number of revolutions} = \frac{\text{6002-00 Total Measuring Range}}{\text{6001-00 Units per revolution}}$$

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

Default = 0040 0000h (262,144)	for XAC81-18-00-EC singleturn encoder
0800 0000h (134 217 728)	for XAC81-13-14-EC multiturn encoder
4000 0000h (1 073 741 824)	for XAC81-16-14-EC multiturn encoder



WARNING

When you set a new value next to the **6002-00 Total Measuring Range** object, please always check also the **6001-00 Units per revolution** object value and be sure that the resulting number of revolutions complies with the Hardware number of revolutions of the device (max. 16,384 revolutions).

Let's suppose that the XAC81-16-14-EC2-... multiturn encoder is programmed as follows:

6001-00 Units per revolution: 65,536 cpr

6002-00 Total Measuring Range = 1 073 741 824₁₀ = 65,536 (cpr) * 16,384 (rev.)

Let's set a new total resolution, for instance: **6002-00 Total Measuring Range** = 360.

As the **6002-00 Total Measuring Range** must be greater than or equal to the **6001-00 Units per revolution**, the above setting is not allowed.



WARNING

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



EXAMPLE

We install the XAC81-**16-14**-EC2-... multiturn rotary encoder.

The physical resolution is as follows:

- **Physical singleturn resolution:** **6501-00 Hardware counts per revolution** = 65,536 (2^{16})
- **Physical multiturn resolution:** **6502-00 Hardware number of turns** = 16,384 turns (2^{14})
- **Total hardware resolution:** **6501-00 Hardware counts per revolution** * **6502-00 Hardware number of turns** = 1 073 741 824 ($2^{16} * 2^{14} = 2^{30}$)

In the specific installation **2,048 counts/rev. * 1,024 turns** are required:

- Enable the scaling function: **6000-00 Operating parameters**, bit 2 **Scaling function** = "1"
- Singleturn resolution: **6001-00 Units per revolution** = 2,048 (0000 0800h)
- Multiturn resolution: **6002-00 Total Measuring Range** = 2,048 * 1,024 = 2 097 152 (0020 0000h)



NOTE

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



WARNING

If **6001-00 Units per revolution** and/or **6002-00 Total Measuring Range** values change, the **6003-00 Preset** object must be updated according to the new resolution. A new preset operation is required.

6003-00 Preset

[Unsigned32, rw]

This object allows to set the encoder position to a Preset value. The Preset function is meant to assign a desired value to a physical position of the encoder. The chosen physical position will get the value set next to this object and all the previous and following positions will get a value according to it. This function can be useful, for instance, when the zero position of the encoder and the zero position of the axis need to match. The preset value will be set for the position

of the encoder in the moment when the preset value is sent. We suggest setting the preset value when the encoder is in stop.

Default = 0000 0000h



EXAMPLE

Let's take a look at the following example to better understand the preset function and the meaning and use of the related objects and commands: **6003-00 Preset** and **6509-00 Offset**.

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

Transmitted value = read position (it does not matter whether the position is physical or scaled) + **6003-00 Preset** - **6509-00 Offset**.

If you never set the **6003-00 Preset** and you never performed the preset setting, then the transmitted value and the read position are necessarily the same as **6003-00 Preset** = 0 and **6509-00 Offset** = 0.

When you set the **6003-00 Preset** and then execute the preset setting, the system saves the current encoder position in the **6509-00 Offset** object. It follows that the transmitted value and the **6003-00 Preset** are the same as read position - **6509-00 Offset** = 0; in other words, the value set next to the **6003-00 Preset** object is paired with the current position of the encoder as you wish.

For example, let's assume that the value "50" is set next to the **6003-00 Preset** object and you execute the preset setting when the encoder position is "1000". In other words, you want to receive the value "50" when the encoder reaches the physical position "1000".

We will obtain the following information sequence:

Transmitted value = read position ("1000") + **6003-00 Preset** ("50") - **6509-00 Offset** ("1000") = 50.

The following transmitted value will be:

Transmitted value = read position ("1001") + **6003-00 Preset** ("50") - **6509-00 Offset** ("1000") = 51.

And so on.

To set the preset value you must send the following command:

Set the Preset value 6003-00 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



NOTE

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



WARNING

Check the value in the **6003-00 Preset** object and perform the preset operation every time you change the value next to the **Code sequence** parameter or the **6001-00 Units per revolution** and/or **6002-00 Total Measuring Range** objects.

6004-00 Position value

[Unsigned32, ro]

This object contains the information about the current position of the encoder. The output value is scaled according to the scaling parameters, if the scaling function is enabled, see the bit 2 **Scaling function** of the **6000-00 Operating parameters** object. The **6004-00 Position value** object is mapped in the **1A00-01 PDO mapping parameter** object, see on page 76.

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 value that is currently set through the bit 0 **Code sequence** in the **6000-00 Operating parameters** object. If the bit is "0" the output encoder position value has been set to increase when the encoder shaft rotates clockwise; if the bit is "1" instead the output encoder position value has been set to increase when the encoder shaft rotates counter-clockwise. To set the code sequence to either CW or CCW you must set the bit 0 **Code sequence** in the **6000-00 Operating parameters** object to 0 / 1. For any further information on setting and using the counting direction function refer to the **6000-00 Operating parameters** object on page 79.

Scaling function

It shows the value that is currently set through the bit 2 **Scaling function** in the **6000-00 Operating parameters** object. In other words, it is intended to show whether the scaling function is enabled or disabled. If the bit is "0", the scaling function is disabled; if the value is "1" instead the scaling function is enabled. To disable / enable the scaling function you must set the bit 2 **Scaling function** in the **6000-00 Operating parameters** object to 0 / 1. For any further information on setting and using the scaling function refer to the **6000-00 Operating parameters** object on page 79.

6501-00 Hardware counts per revolution

[Unsigned32, ro]



WARNING

This object is active only if the bit 2 **Scaling function** in the **6000-00 Operating parameters** object is set to "0"; otherwise it is ignored and the system uses the custom values (**6001-00 Units per revolution** and **6002-00 Total Measuring Range**) to calculate the position information.

This object is intended to show the number of physical distinguishable steps provided per each turn by the hardware (physical singleturn resolution, see the hardware counts per revolution in the encoder identification label).

If you want to set a custom singleturn resolution see the **6001-00 Units per revolution** object.

6502-00 Hardware number of turns

[Unsigned32, ro]



WARNING

This object is active only if the bit 2 **Scaling function** in the **6000-00 Operating parameters** object is set to "0"; otherwise it is ignored and the system uses the custom values (**6001-00 Units per revolution** and **6002-00 Total Measuring Range**) to calculate the position information.

This object is intended to show the number of physical turns provided by the hardware of the encoder (physical multiturn resolution, see the hardware revolutions in the encoder identification label).

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

If you want to set a custom number of turns see the **6001-00 Units per revolution** and **6002-00 Total Measuring Range** objects.

6503-00 Errors

[Unsigned16, ro]

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

6504-00 Supported errors

[Unsigned16, ro]

This object contains the information on the error alarms supported by the encoder. No error alarms are supported in this encoder.

Default = 0000h (No errors supported).

6505-00 Warnings

[Unsigned16, ro]

The corresponding bits of supported warnings are set (see the **6506-00 Supported warnings** object).

6506-00 Supported warnings

[Unsigned16, ro]

This object contains the information on the warnings supported by the encoder.

bits 0 ... 11 = not supported

bit 12 = **Wrong parameters loaded from flash memory at power-on**

bits 13 ... 15 = not supported

Default = 1000h

6509-00 Offset

[Unsigned32, ro]

This object contains the Offset value. As soon as you activate the preset, the current position of the encoder is saved in this object. The offset value is then used in the preset function in order to calculate the encoder position value to be transmitted. To zero set the value in this object you must upload the factory default values (see the **1011-01 Restore default parameters** object on page 74).

For any further information on the preset function and the meaning and use of the related objects **6003-00 Preset** and **6509-00 Offset** refer to page 83.

**NOTE**

To save the new parameters execute the store parameters function (see the **1010-01 Store parameters** object on page 74).

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

8.2.6 SDO Abort codes

SDO transfer could be unsuccessful; causes of error are listed and described in the SDO Abort Codes. Here follows the list of the available SDO Abort Codes. For complete information see ETG1000.6 "EtherCAT Specification – Part 6. Application Layer protocol specification", par. 5.6.2.7.2 table 40.

Abort code	Description
0503 0000h	Toggle bit not changed.
0504 0000h	SDO protocol timeout.
0504 0001h	Client/Server command specifier not valid or unknown.
0504 0005h	Out of memory.
0601 0000h	Unsupported access to an object.
0601 0001h	Attempt to read a write only object.
0601 0002h	Attempt to write a read only object.
0602 0000h	The object does not exist in the object dictionary.
0604 0041h	The object cannot be mapped into the PDO.
0604 0042h	The number and length of the objects to be mapped would exceed PDO length.
0604 0043h	General parameter incompatibility reason.
0604 0047h	General internal incompatibility in the device.
0606 0000h	Access failed due to a hardware error.
0607 0010h	Data type does not match, length of service parameter does not match
0607 0012h	Data type does not match, length of service parameter too high
0607 0013h	Data type does not match, length of service parameter too low
0609 0011h	Subindex does not exist.
0609 0030h	Value range of parameter exceeded (only for write access).
0609 0031h	Value of parameter written too high.
0609 0032h	Value of parameter written too low.
0609 0036h	Maximum value is less than minimum value.
0800 0000h	General error
0800 0020h	Data cannot be transferred or stored to the application.
0800 0021h	Data cannot be transferred or stored to the application because of local control.
0800 0022h	Data cannot be transferred or stored to the application because of the present device state.
0800 0023h	Object dictionary dynamic generation fails or no object dictionary is present.

8.2.7 Emergency Error Codes

Emergency Service is used by the Server for transmitting diagnostic messages to the client using MailBox; Error Codes are listed and described in the ETG1000.6 "EtherCAT Specification – Part 6. Application Layer protocol specification", 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	<p>State Transition Errors of state machine: (for detailed description see ETG1000.6 par. 5.6.4.3)</p> <p>A000hex: transition error from PRE-OPERATIONAL to SAFE-OPERATIONAL</p> <p>A001hex: transition error from SAFE-OPERATIONAL to OPERATIONAL</p> <p>Encoder errors:</p> <p>5000hex: Hardware error</p> <p>5001hex: Diagnostic data (wrong parameters loaded from flash memory)</p>						
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).						

8.2.8 AL Status Error Codes

If the state transition requested by the Master through the "AL Control Register" is unsuccessful, the 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 "EtherCAT Specification – Part 6. Application Layer protocol specification", par.5.3.2 Table 11.

8.3 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**.

Please refer also to the "7.7 Firmware upgrade" section on page 54.

9 – Default parameters list

Default values are expressed in hexadecimal notation.

Parameters list	Default values		
1000-00 Device type	0001 0196 = singleturn encoder 0002 0196 = multiturn encoder		
1008-00 Device Name	4853353878 = "HS58x" for XAC81-18-00-EC singleturn encoder 454D353878 = "EM58x" for XAC81-13-14-EC multiturn encoder 484D353878 = "HM58x" for XAC81-16-14-EC multiturn encoder		
1009-00 Hardware version	device dependent		
100A-00 Software version	device dependent		
1018 Identity 01 Vendor ID 02 Product code 03 Revision 04 Serial number	0000 012E 0000 0001 = Lika EtherCAT HS58, singleturn encoder (i.e. XAC81-18-00) 0000 0002 = Lika EtherCAT EM58 and HM58, multiturn encoder (i.e. XAC81-13-14 and XAC81-16-14) 0001 0001 = Lika EtherCAT EM58 series (i.e. XAC81-13- 14) 0001 0000 = Lika EtherCAT HS58 and HM58 series (i.e. XAC81-18-00 and XAC81- 16-14) 0000 0000		
1A00-01 PDO mapping parameter 01 Mapped Object 001	6004 0020		
6000-00 Operating parameters	0000		
6001-00 Units per revolution	0040 0000 (262,144) for XAC81-18-00-EC singleturn encoder		

	0000 2000 (8,192) for XAC81-13-14-EC multiturn encoder 0001 0000 (65,536) for XAC81-16-14-EC multiturn encoder		
6002-00 Total Measuring Range	0040 0000 (262,144) for XAC81-18-00-EC singleturn encoder 0800 0000 (134 217 728) for XAC81-13-14-EC multiturn encoder 4000 0000 (1 073 741 824) for XAC81-16-14-EC multiturn encoder		
6003-00 Preset	00000 0000		

This page intentionally left blank

This page intentionally left blank

This page intentionally left blank

Document release	Release date	Description	HW	SW	XML file version
1.0	18.04.2024	First issue	H02	S05	V6



Dispose separately

lika

Lika Electronic

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

Tel. +39 0445 806600

Fax +39 0445 806699



info@lika.biz • www.lika.biz