

## EM58 MT HS58 MT HM58 MT



Questo prodotto è ancora in produzione, ma in fase di progressiva dismissione, pertanto non è consigliato per nuove applicazioni.

Raccomandiamo l'utilizzo del nuovo encoder EXM58-EX058.

#### Descrive i seguenti modelli:

- EM58, EM58S MT
- EMC58, EMC59, EMC60 MT
- HS58, HS58S MT
- HSC58, HSC59, HSC60 MT
- HM58, HM58S MT
- HMC58, HMC59, HMC60 MT

#### Indice generale

Informazioni preliminari	8
1 - Norme di sicurezza	18
2 - Identificazione	20
3 - Istruzioni di montaggio	21
4 - Connessioni elettriche	26
5 - Quick reference	35
6 - Interfaccia MODBUS®/TCP	36

This publication was produced by Lika Electronic s.r.l. 2023. All rights reserved. Tutti i diritti riservati. Alle Rechte vorbehalten. Todos los derechos reservados. Tous droits réservés.

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

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

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

The logo for Lika Electronic s.r.l. consists of the word "lika" in a bold, lowercase, sans-serif font. The letter "i" has a dot above it. The logo is positioned in the bottom right corner of the page.

# Indice generale

Manuale d'uso.....	1
Indice generale.....	3
Indice analitico.....	6
Convenzioni tipografiche e iconografiche.....	7
Informazioni preliminari.....	8
Glossario dei termini MODBUS/TCP.....	9
Lista delle abbreviazioni.....	16
Referenze.....	17
<b>1 Norme di sicurezza.....</b>	<b>18</b>
1.1 Sicurezza.....	18
1.2 Avvertenze elettriche.....	18
1.3 Avvertenze meccaniche.....	19
<b>2 Identificazione.....</b>	<b>20</b>
<b>3 Istruzioni di montaggio.....</b>	<b>21</b>
3.1 Encoder con asse sporgente.....	21
3.1.1 Fissaggio standard.....	21
3.1.2 Fissaggio con graffe (codice LKM-386).....	22
3.1.3 Fissaggio con campana (codice PF4256).....	22
3.2 Encoder con asse cavo.....	23
3.2.1 EMC58, HSC58, HMC58.....	23
3.2.2 EMC59, HSC59, HMC59.....	24
3.2.3 EMC60, HSC60, HMC60.....	25
<b>4 Connessioni elettriche.....</b>	<b>26</b>
4.1 Connettori M12.....	26
4.1.1 Connettore alimentazione PWR (Figura 1).....	26
4.1.2 Connettori P1 Porta 1 e P2 Porta 2 (Figura 1).....	27
4.1.3 Configurazione di rete: topologie, cavi, hub, switch - Raccomandazioni.....	27
4.2 Collegamento a terra.....	28
4.3 Collegamento della calza.....	28
4.4 Indirizzo MAC e indirizzo IP.....	28
4.5 Impostazione dell'indirizzo IP e dei parametri di configurazione della rete.....	29
4.6 Resistenza di terminazione.....	30
4.7 LED di diagnostica (Figura 1).....	30
4.8 DIP A: reset dei parametri di configurazione della rete ai valori di fabbrica.....	32
4.9 Coperchio encoder (Figura 2).....	33
<b>5 Quick reference.....</b>	<b>35</b>
5.1 Getting started.....	35
<b>6 Interfaccia MODBUS®/TCP.....</b>	<b>36</b>
6.1 Principi guida del protocollo MODBUS.....	36
6.2 Descrizione generale del frame MODBUS.....	37
6.3 MODBUS su Application Data Unit TCP/IP.....	38
6.4 PDU MODBUS.....	40
6.5 Codici funzione.....	40
6.5.1 Codici funzione implementati.....	41
<b>03 Read Holding Register.....</b>	<b>41</b>
<b>04 Read Input Register.....</b>	<b>43</b>

06 Write Single Register.....	45
16 Write Multiple Register.....	47
6.6 Stati encoder.....	52
WAIT_PROCESS.....	52
ERROR.....	52
PROCESS_ACTIVE.....	52
EXCEPTION.....	52
7 Parametri di programmazione.....	53
7.1 Parametri disponibili.....	53
7.1.1 Parametri Holding Register.....	53
Watchdog timeout [82].....	54
Current position [95-96].....	54
Speed value [97-98].....	55
Status word [99-100].....	55
Counts per revolution [101-102].....	55
Total Resolution [103-104].....	57
Preset value [105-106].....	59
Speed format [107-108].....	60
Operating parameters [109-110].....	60
Scaling function.....	60
Code sequence.....	61
Control Word [111-112].....	62
Save parameters.....	62
Restore default parameters.....	62
Perform counting preset.....	63
Singleturn resolution [113-114].....	64
Number of revolutions [115-116].....	64
Supported alarms [117-118].....	65
Supported warnings [119-120].....	66
Alarm registers [121-122].....	66
Machine data not valid.....	66
Setting data not valid.....	67
Flash memory error.....	67
Warnings register [123-124].....	67
Wrong parameters list [125-126].....	67
Counts per revolution error.....	68
Total resolution error.....	68
Preset value error.....	68
Offset value error.....	68
Offset value [127-128].....	68
DSC Firmware Version [129-130].....	69
PCB Hardware Version [131-132].....	69
7.1.2 Parametri Input Register.....	70
Current position [1-2].....	70
Speed value [3-4].....	70
Status word [5-6].....	70
Scaling function.....	71
Code sequence.....	71
Alarm.....	71

7.2 Exception response e codici eccezione.....	73
<b>8 Web Server Integrato.....</b>	<b>76</b>
8.1 Web server integrato – Informazioni preliminari.....	76
8.2 Pagina Home del Web server.....	77
8.3 Posizione e velocità dell'encoder.....	78
8.3.1 Note specifiche sull'utilizzo di Internet Explorer.....	79
8.4 Encoder information (registri MODBUS).....	80
8.5 Impostazione del valore di Preset.....	81
8.6 Impostazione dei registri.....	83
8.7 Upgrade del firmware.....	87
8.8 Configurazione della rete.....	93
<b>9 Esempi di programmazione.....</b>	<b>97</b>
9.1 Utilizzo del codice funzione 03 Read Holding Register.....	97
9.2 Utilizzo del codice funzione 04 Read Input Register.....	98
9.3 Utilizzo del codice funzione 06 Write Single Register.....	99
9.4 Utilizzo del codice funzione 16 Write Multiple Register.....	100
<b>10 Lista parametri di default.....</b>	<b>102</b>

# Indice analitico

## A

Alarm.....	71
Alarm registers [121-122].....	66

## C

Code sequence.....	61, 71
Control Word [111-112].....	62
Counts per revolution [101-102].....	55
Counts per revolution error.....	68
Current position [1-2].....	70
Current position [95-96].....	54

## D

DSC Firmware Version [129-130].....	69
-------------------------------------	----

## E

ERROR.....	52
EXCEPTION.....	52

## F

Flash memory error.....	67
-------------------------	----

## M

Machine data not valid.....	66
-----------------------------	----

## N

Number of revolutions [115-116].....	64
--------------------------------------	----

## O

Offset value [127-128].....	68
Offset value error.....	68
Operating parameters [109-110].....	60

## P

PCB Hardware Version [131-132].....	69
Perform counting preset.....	63
Preset value [105-106].....	59
Preset value error.....	68
PROCESS_ACTIVE.....	52

## R

Restore default parameters.....	62
---------------------------------	----

## S

Save parameters.....	62
Scaling function.....	60, 71
Setting data not valid.....	67
Singleturn resolution [113-114].....	64
Speed format [107-108].....	60
Speed value [3-4].....	70
Speed value [97-98].....	55
Status word [5-6].....	70
Status word [99-100].....	55
Supported alarms [117-118].....	65
Supported warnings [119-120].....	66

## T

Total Resolution [103-104].....	57
Total resolution error.....	68

## W




WAIT_PROCESS.....	52
Warnings register [123-124].....	67
Watchdog timeout [82].....	54
Wrong parameters list [125-126].....	67

# Convenzioni tipografiche e iconografiche

Per rendere più agevole la lettura di questo testo sono state adottate alcune convenzioni grafiche e iconografiche. In particolare:

- i parametri e gli oggetti sia propri dell'interfaccia che del dispositivo Lika sono evidenziati in **VERDE**;
- gli allarmi sono evidenziati in **ROSSO**;
- gli stati sono evidenziati in **FUCSIA**.

Scorrendo il testo sarà inoltre possibile imbattersi in alcune icone che evidenziano porzioni di testo di particolare interesse o rilevanza. Talora esse possono contenere prescrizioni di sicurezza atte a richiamare l'attenzione sui rischi potenziali legati all'utilizzo del dispositivo. Si raccomanda di seguire attentamente le prescrizioni elencate nel presente manuale al fine di salvaguardare la sicurezza dell'utilizzatore oltre che le performance del dispositivo. I simboli utilizzati nel presente manuale sono i seguenti:

	Questa icona, accompagnata dal termine <b>ATTENZIONE</b> , evidenzia le porzioni di testo che contengono informazioni della massima importanza per l'operatore concernenti l'uso corretto e sicuro del dispositivo. Le istruzioni accompagnate da questo simbolo devono essere seguite scrupolosamente dall'operatore. La loro mancata osservanza può generare malfunzionamenti e danni sia al dispositivo che alla macchina sulla quale il dispositivo è installato e procurare lesioni anche gravi agli operatori al lavoro in prossimità.
	Questa icona, accompagnata dal termine <b>NOTA</b> , evidenzia le porzioni di testo che contengono notazioni importanti ai fini di un uso corretto e performante del dispositivo. Le istruzioni accompagnate da questo simbolo devono essere tenute bene in considerazione da parte dell'operatore. La loro mancata osservanza può procurare l'esecuzione di procedure errate di settaggio da parte dell'utilizzatore e conseguentemente un funzionamento errato o inadeguato del dispositivo.
	Questa icona evidenzia le porzioni di testo che contengono suggerimenti utili per agevolare l'operatore nel settaggio e l'ottimizzazione del dispositivo. Talora il simbolo è accompagnato dal termine <b>ESEMPIO</b> quando le istruzioni di impostazione dei parametri siano seguite da esemplificazioni che ne chiarifichino l'utilizzo.

# Informazioni preliminari

Questo manuale ha lo scopo di descrivere le caratteristiche tecniche, l'installazione e l'utilizzo corretto e sicuro dei seguenti encoder **con interfaccia MODBUS/TCP**:

**EMxxx-13-14-MT2-...** (encoder multigiro, risoluzione 13 +14 bit)

**HSxxx-18-00-MT2-...** (encoder monogiro, risoluzione 18 bit)

**HMxxx-16-14-MT2-...** (encoder multigiro, risoluzione 16 +14 bit)

Per le specifiche tecniche riferirsi al datasheet del prodotto.

Per una più agevole consultazione, questo manuale può essere diviso in due sezioni principali.

Nella prima sezione sono fornite le informazioni generali riguardanti il trasduttore comprendenti le norme di sicurezza, le istruzioni di montaggio meccanico e le prescrizioni relative alle connessioni elettriche.

Nella seconda sezione, intitolata **Interfaccia MODBUS/TCP**, sono invece fornite tutte le informazioni sia generali che specifiche relative all'interfaccia MODBUS. In questa sezione sono descritte le caratteristiche dell'interfaccia e i parametri MODBUS che l'unità implementa.



# Glossario dei termini MODBUS/TCP

MODBUS/TCP, come molte altre interfacce di collegamento in rete, si avvale di una terminologia specifica. La tabella qui sotto contiene alcuni dei termini tecnici che sono utilizzati in questa guida per descrivere l'interfaccia MODBUS/TCP. Sono elencati in ordine alfabetico.

<b>ADU</b>	Application Data Unit (Unità Dati Applicazione), è il frame dati del protocollo MODBUS. Prende la forma di un header (intestazione) a 7 byte (MBAP Header: identificatore della transazione + identificatore del protocollo + campo della lunghezza + identificatore dell'unità) e della Protocol Data Unit (PDU: codice funzione + dati). La ADU MODBUS/TCP è inserita nel campo dati di un frame TCP standard e inviata via TCP sulla porta registrata 502, che è specificamente riservata per le applicazioni MODBUS. Pertanto questo pacchetto è incapsulato dai frame di dati imposti dagli stack TCP/IP dei protocolli (TCP/IP/MAC) prima di essere trasmesso sulla rete. Riferirsi a pagina 38.
<b>Application Process</b>	L'Application Process è il task sull'Application Layer.
<b>Application protocol</b>	MODBUS è un protocollo a livello applicativo o struttura a messaggi che definisce le regole per l'organizzazione e l'interpretazione dei dati indipendentemente dal supporto di trasmissione dei dati. TCP/IP garantisce solamente che i messaggi applicazione siano trasferiti tra i dispositivi sulla Ethernet Local-Area Network (LAN), non garantisce che i dispositivi comprendano effettivamente ed interagiscano gli uni con gli altri. Per MODBUS/TCP, questa capacità è fornita dal protocollo MODBUS a livello applicazione.
<b>Bus</b>	Un bus è un supporto di comunicazione che collega diversi nodi. I dati possono essere trasferiti tramite circuiti seriali o paralleli, ossia tramite conduttori elettrici o fibra ottica.
<b>Client</b>	Un Client è un qualsiasi dispositivo di rete che invia richieste di dati ai Server. MODBUS/TCP segue il modello Client/Server. I Master MODBUS sono intesi come Client, mentre gli Slave MODBUS sono intesi come Server.
<b>Codice eccezione</b>	Codice che gli Slave devono restituire in caso di problemi. Tutte le eccezioni sono segnalate mediante l'aggiunta di 0x80 al codice funzione della richiesta. Riferirsi alla pagina 73.
<b>Codice funzione</b>	MODBUS è un protocollo di tipo richiesta/risposta e offre servizi specificati da codici funzione. Il codice funzione è inviato da un Client a un Server e indica il tipo di azione che il Server deve eseguire. I codici funzione MODBUS sono elementi delle PDU di richiesta/risposta MODBUS. Il campo codice funzione di una data unit MODBUS è codificato in un byte. I codici validi sono compresi nel range

	1 ... 255 decimali (il range 128 – 255 è riservato e utilizzato per le exception response). Il codice funzione "0" non è valido. Gli encoder Lika implementano solamente codici funzione pubblici. Riferirsi alla pagina 40.
<b>Codifica dati</b>	MODBUS utilizza una rappresentazione 'big-Endian' per dati e indirizzi. Questo significa che, quando s invia una quantità numerica maggiore del singolo byte, il byte più significativo è inviato per primo. Riferirsi a pagina 38.
<b>Comunicazione deterministica</b>	Descrive un processo di comunicazione il cui comportamento nel tempo può essere esattamente predetto. In altri termini il momento in cui il messaggio raggiunge il destinatario è predittibile.
<b>Conferma MODBUS</b>	Una Conferma MODBUS (MODBUS Confirmation) è il messaggio di risposta ricevuto dal lato Client.
<b>Determinismo</b>	E' la capacità del protocollo di comunicazione di garantire che un messaggio sia inviato o ricevuto in un lasso di tempo finito e predittibile.
<b>DHCP</b>	DHCP (Dynamic Host Configuration Protocol) è un protocollo di rete standardizzato network usato nelle reti Internet Protocol (IP) per la distribuzione dinamica dei parametri di configurazione della rete, per esempio gli indirizzi IP per interfacce e servizi. All'avvio un server DHCP assegna indirizzi IP dinamici e gli indirizzi possono cambiare nel tempo. I server DHCP nella rete acquisiscono la richiesta e mettono a disposizione del Client un indirizzo IP. Il Client fa propria la prima offerta che riceve e il server DHCP a sua volta informa il Client del suo successo nell'attribuzione dell'indirizzo IP per un arco di tempo specificato.
<b>DNS</b>	DNS (Domain Name System) è un sistema di denominazione distribuito di livello gerarchico per computer, servizi o qualsiasi risorsa connessa a Internet o a una rete privata. DNS è un servizio di risoluzione del nome host che può essere usato per determinare l'indirizzo IP di un computer a partire dal suo nome host. Questo permette agli utenti di lavorare con i nomi host, come per esempio www.example.com, piuttosto che con indirizzi IP, come per esempio 192.168.5.102 o 192.168.12.68.
<b>Exception code</b>	Codice che gli Slave devono restituire in caso di problemi. Tutte le eccezioni sono segnalate mediante l'aggiunta di 0x80 al codice funzione della richiesta. Riferirsi alla pagina 73.
<b>Exception response</b>	MODBUS lavora seguendo l'usuale modello Client/Server (Master/Slave): il Client (Master) invia un telegramma di richiesta (request telegram, service request) al Server (Slave) e il Server risponde con un telegramma di risposta (response telegram). Se il Server non è in grado di processare una richiesta, invierà di ritorno un codice funzione di errore (exception response) che è il codice funzione originale più 80H (ossia, con il bit più significativo impostato a 1). Riferirsi alle pagine 40 e 73.

<b>Function code</b>	<p>MODBUS è un protocollo di tipo richiesta/risposta e offre servizi specificati da codici funzione. Il codice funzione è inviato da un Client a un Server e indica il tipo di azione che il Server deve eseguire. I codici funzione MODBUS sono elementi delle PDU di richiesta/risposta MODBUS.</p> <p>Il campo codice funzione di una data unit MODBUS è codificato in un byte. I codici validi sono compresi nel range 1 ... 255 decimali (il range 128 – 255 è riservato e utilizzato per le exception response). Il codice funzione "0" non è valido. Gli encoder Lika implementano solamente codici funzione pubblici. Riferirsi alla pagina 40.</p>
<b>Holding register</b>	<p>Nel modello dati MODBUS, un Holding register è il dato in uscita. Un Holding register ha una dimensione di 16 bit, è modificabile da un programma applicativo e permette un accesso sia in lettura-scrittura che in sola lettura. Riferirsi alla pagina 53.</p>
<b>Host</b>	<p>Un computer o altro dispositivo in una rete TCP/IP.</p>
<b>IEEE 1588</b>	<p>Questo standard definisce un protocollo che abilita la sincronizzazione dei clock nei dispositivi in una rete distribuita (per esempio connessi via Ethernet).</p>
<b>Incapsulamento</b>	<p>Il termine "incapsulamento" si riferisce all'azione di impacchettamento (integrazione) del messaggio MODBUS all'interno del contenitore TCP, del contenitore IP e del contenitore MAC.</p>
<b>Indicazione MODBUS</b>	<p>Una Indicazione MODBUS (MODBUS Indication) è il messaggio di richiesta ricevuto dal lato Server.</p>
<b>Indirizzo broadcast</b>	<p>Un indirizzo IP con una porzione host che corrisponde a quella di tutti.</p>
<b>Indirizzo di rete</b>	<p>Un indirizzo IP con una porzione host costituita da soli zeri.</p>
<b>Indirizzo IP</b>	<p>L'indirizzo IP è un numero a 32 bit che identifica in maniera univoca un host (computer o altro dispositivo, per esempio una stampante o un router) in una rete TCP/IP. Gli indirizzi IP sono normalmente espressi nel formato a decimali puntati, con quattro numeri separati da punti, per esempio 192.168.123.132. Un indirizzo IP consta di due parti. La prima parte di un indirizzo IP è usata come indirizzo di rete, la seconda invece come indirizzo host. Considerando l'esempio 192.168.123.132, se dividiamo il numero nelle due parti otterremo quanto segue: 192.168.123. = rete; .132 = Host. 0: 192.168.123.0 = indirizzo di rete; 0.0.0.132 = indirizzo host. Riferirsi alla pagina 28.</p>
<b>Indirizzo MAC</b>	<p>L'indirizzo MAC è un identificatore univoco a livello globale formato da due parti: i primi 3 byte rappresentano l'ID del costruttore (manufacturer ID) e sono rilasciati dall'autorità di standardizzazione IEE; gli altri 3 byte rappresentano un numero consecutivo aggiunto dal costruttore. Riferirsi alla pagina 28.</p>

<b>Input register</b>	Nel modello dati MODBUS, un Input register è il dato in ingresso. Un Input register ha una dimensione di 16 bit, è messo a disposizione da un sistema I/O e permette l'accesso in sola lettura. Riferirsi alla pagina 70.
<b>Internet</b>	L'insieme globale delle reti che sono collegate insieme e condividono un range comune di indirizzi IP.
<b>InterNIC</b>	L'organizzazione responsabile per l'amministrazione degli indirizzi IP in Internet.
<b>IP</b>	Il protocollo di rete usato per la trasmissione di pacchetti di rete in una rete TCP/IP o in Internet.
<b>Isocrono</b>	Riguarda i processi che richiedono il coordinamento dei tempi per avere successo. Il trasferimento isocrono dei dati garantisce che il flusso dei dati sia continuo e a una velocità costante, in stretto sincronismo con la capacità dei dispositivi collegati.
<b>Legacy Ethernet</b>	Ethernet nella sua standardizzazione in IEEE 802.3 (funzionamento non-deterministico in ambienti senza criticità temporali).
<b>Master</b>	Un Master è qualsiasi dispositivo di rete che invia richieste dati agli Slave.
<b>MBAP Header</b>	L'MBAP header (MODBUS Application Header) è un'intestazione (header) a 7 byte aggiunta all'inizio del messaggio e usata in TCP/IP per identificare la MODBUS Application Data Unit. Contiene i seguenti dati: <ul style="list-style-type: none"> <li>• Transaction Identifier: 2 byte impostati dal Client per identificare in maniera univoca ciascuna richiesta. Questi byte vengono ripetuti dal Server dal momento che le sue risposte potrebbero non essere ricevute nello stesso ordine delle richieste.</li> <li>• Protocol Identifier: 2 byte inviati dal Client, sempre = 00 00</li> <li>• Dimensione: 2 byte a identificare il numero di byte nel messaggio a seguire.</li> <li>• Unit Identifier: 1 byte inviato dal Client e ripetuto dal Server per l'identificazione di una Slave remoto collegato in una linea seriale o in altri bus.</li> </ul> Riferirsi alla pagina 38.
<b>Media Access Control (MAC)</b>	Uno dei sottolivelli del livello Data Link preposto al controllo di chi accede al medium per inviare un messaggio.
<b>Message</b>	Il servizio di messaggistica MODBUS permette una comunicazione di tipo Client/Server tra dispositivi collegati nella rete Ethernet TCP/IP. Il modello Client / Server si basa su quattro tipi di messaggi: <ul style="list-style-type: none"> <li>• Richiesta MODBUS (MODBUS Request)</li> <li>• Conferma MODBUS (MODBUS Confirmation)</li> <li>• Indicazione MODBUS (MODBUS Indication)</li> <li>• Risposta MODBUS (MODBUS Response)</li> </ul>

	I servizi di messaggistica MODBUS sono usati per lo scambio di informazioni.
<b>MODBUS Confirmation</b>	Una Conferma MODBUS (MODBUS Confirmation) è il messaggio di risposta ricevuto dal lato Client.
<b>MODBUS Indication</b>	Una Indicazione MODBUS (MODBUS Indication) è il messaggio di richiesta ricevuto dal lato Server.
<b>MODBUS Request</b>	Una Richiesta MODBUS (MODBUS Request) è il messaggio inviato nella rete dal Client per dare inizio a una transazione. Riferirsi a pagina 40.
<b>MODBUS Response</b>	Una Risposta MODBUS (MODBUS Response) è il messaggio di risposta inviato dal Server. Riferirsi a pagina 40.
<b>Otteto</b>	Un numero a 8 bit, 4 dei quali comprendono un indirizzo IP a 32 bit. Hanno un range 00000000-11111111 corrispondente ai valori decimali 0-255.
<b>Pacchetto</b>	Una unità di dati trasmessa su una rete TCP/IP o una rete WAN (wide area network).
<b>PDU</b>	<p>La Protocol Data Unit (PDU) rappresenta il codice funzione MODBUS e il campo dati nel loro formato originale. E' impacchettato insieme con l'MBAP Header a formare l'Application Data Unit (ADU).</p> <p>Il protocollo MODBUS definisce tre PDU. Esse sono:</p> <ul style="list-style-type: none"> <li>• MODBUS Request PDU, mb_req_pdu</li> <li>• MODBUS Response PDU, mb_rsp_pdu</li> <li>• MODBUS Exception Response PDU, mb_excep_rsp_pdu</li> </ul> <p>Riferirsi a pagina 40.</p>
<b>Porta</b>	E' un indirizzo usato localmente al livello transport (in un nodo) e identifica mittente e destinazione del pacchetto all'interno dello stesso nodo. I numeri di porta si dividono in numeri di porta noti (well-known, 0-1023), numeri di porta dell'utente registrato (1024-49151) e numeri di porta dinamici privati (49152-65535). In TCP, il numero di porta 0 è riservato e non può essere usato. Le porte permettono a TCP/IP di effettuare il multiplexing e il demultiplexing di una sequenza di datagrammi IP che devono arrivare a molti differenti (simultanei) processi di applicazione. MODBUS/TCP usa la porta nota 502 per attendere (listen, rimanere in ascolto) e ricevere messaggi MODBUS sulla rete Ethernet.
<b>Read Holding Register (03, 0003hex)</b>	Questo codice funzione è utilizzato per LEGGERE il contenuto di un blocco contiguo di holding register in un dispositivo remoto; in altre parole, permette di leggere i valori impostati in un gruppo di parametri di lavoro disposti in sequenza. Riferirsi alla pagina 41.
<b>Read Input Register (04, 0004hex)</b>	Questo codice funzione è usato per LEGGERE da 1 a 125 registri di ingresso contigui in un dispositivo remoto; in altre parole, permette la lettura di alcuni valori di risultato e dei messaggi di stato / allarme in un dispositivo remoto. Riferirsi

	alla pagina 43.
<b>Real-time</b>	Real-time significa che un sistema processa gli eventi esterni in un lasso di tempo definito. Se la reazione di un sistema è predittibile, si parla di sistema deterministico. Le condizioni generali per il real-time sono pertanto: risposta deterministica e tempo di risposta definito.
<b>Register</b>	Le funzioni MODBUS operano sui registri di memoria per configurare, monitorare e controllare gli I/O del dispositivo. Riferirsi alla pagina 53.
<b>Registro</b>	Le funzioni MODBUS operano sui registri di memoria per configurare, monitorare e controllare gli I/O del dispositivo. Riferirsi alla pagina 53.
<b>Rete</b>	Con rete si intende l'insieme dei computer in un singolo segmento di rete fisica; altrimenti, è un range di indirizzi di rete IP che è allocato da un amministratore di sistema.
<b>Richiesta MODBUS</b>	Una Richiesta MODBUS (MODBUS Request) è il messaggio inviato nella rete dal Client per dare inizio a una transazione. Riferirsi a pagina 40.
<b>Risposta MODBUS</b>	Una Risposta MODBUS (MODBUS Response) è il messaggio di risposta inviato dal Server. Riferirsi a pagina 40.
<b>Router</b>	Un dispositivo che trasferisce il traffico di rete tra reti con diverso IP.
<b>Server</b>	Un Server è un qualsiasi programma che attende che gli siano inviate richieste di dati. I Server non avviano la comunicazione con i Client, ma inviano solo le risposte. MODBUS/TCP segue il modello Client/Server. I Master MODBUS sono da intendersi come Client, mentre gli Slave MODBUS sono da intendersi come Server.
<b>Service request</b>	E' la richiesta MODBUS (MODBUS Request), ossia il messaggio inviato nella rete da un Client per dare inizio a una transazione.
<b>Slave</b>	Uno Slave è un qualsiasi programma che attende che gli sia inviata una richiesta dati. Gli Slave non avviano la comunicazione con i Master, ma inviano solo le risposte.
<b>Subnet Mask</b>	Un numero a 32 bit usato per distinguere le parti network e host di un indirizzo IP. In altri termini, è usato dal protocollo TCP/IP per determinare se un host si trova in una sottorete (subnet) locale oppure in una rete remota.
<b>Subnet o Subnetwork</b>	Una rete più piccola creata dividendo una rete più grande in parti uguali.
<b>TCP/IP</b>	Usato a livello globale, è l'insieme dei protocolli, degli standard e delle utility comunemente usate in Internet e nella grandi reti. Il sistema Ethernet è progettato esclusivamente per il trasporto dei dati. Può essere paragonato a un'autostrada intesa come un sistema per il trasporto di merci e passeggeri. I

	<p>dati sono appunto trasportati dai protocolli.</p> <p>Si può paragonare alle auto e ai veicoli commerciali che trasportano merci e passeggeri sull'autostrada.</p> <p>I compiti gestiti dal Transmission Control Protocol (TCP) di base e dall'Internet Protocol (IP) (abbreviati in TCP/IP):</p> <ol style="list-style-type: none"> <li>1. Il mittente spezzetta i dati in una sequenza di pacchetti.</li> <li>2. I pacchetti sono trasportati tramite Ethernet al corretto destinatario.</li> <li>3. Il destinatario riassume i pacchetti di dati nell'ordine corretto.</li> <li>4. I pacchetti corrotti sono inviati nuovamente fino a che il destinatario non riconosce l'avvenuto trasferimento con successo.</li> </ol>
<b>Topologia</b>	<p>Struttura della rete. Strutture usate comunemente:</p> <ul style="list-style-type: none"> <li>• topologia Line (lineare);</li> <li>• topologia Ring (ad anello);</li> <li>• topologia Star (a stella);</li> <li>• topologia Tree (ad albero).</li> </ul> <p>Riferirsi alla pagina 27.</p>
<b>Velocità di trasmissione</b>	<p>Velocità di trasferimento dei dati (in bps, bit al secondo).</p> <p>Riferirsi alla pagina 27.</p>
<b>Wide area network (WAN)</b>	<p>Una rete di grandi dimensioni che è un insieme di reti più piccole divise da router. Internet è un esempio di una WAN di dimensioni estremamente grandi.</p>
<b>Write Multiple Register (16, 0010hex)</b>	<p>Questo codice è usato per SCRIVERE un blocco di registri contigui (da 1 a 123 registri) in un dispositivo remoto. Riferirsi alla pagina 47.</p>
<b>Write Single Register (06, 0006hex)</b>	<p>Questo codice funzione è usato per SCRIVERE un singolo holding register in un dispositivo remoto. Riferirsi alla pagina 45.</p>

# Lista delle abbreviazioni

La tabella qui sotto contiene una lista delle abbreviazioni (in ordine alfabetico) che possono essere utilizzate in questo manuale per descrivere l'interfaccia MODBUS/TCP.

<b>ADU</b>	Application Data Unit
<b>HDLC</b>	High level Data Link Control
<b>HMI</b>	Human Machine Interface
<b>I/O</b>	Input/Output (ingressi/uscite)
<b>IETF</b>	Internet Engineering Task Force
<b>IP</b>	Internet Protocol (protocollo Internet)
<b>MAC</b>	Media Access Control
<b>MB</b>	MODBUS Protocol (protocollo MODBUS)
<b>MBAP</b>	MODBUS Application Protocol
<b>MBAP header</b>	MODBUS Application Header
<b>PDU</b>	Protocol Data Unit
<b>PLC</b>	Programmable Logic Controller
<b>TCP</b>	Transmission Control Protocol



# Referenze

- [1] MODBUS Application Protocol Specification, Versione V1.1b3
- [2] MODBUS messaging on TCP/IP implementation guide, Versione V1.0b
- [3] RFC 791, Internet Protocol, Sep81 DARPA
- [4] RFC 1122 Requirements for Internet Hosts -- Communication Layers
- [5] IEC 61918 Industrial communication networks – Installation of communication networks in industrial premises
- [6] IEC 61784-5-13 Industrial communication networks – Profiles – Part 5-13: Installation of fieldbuses – Installation profiles for CPF 13

# 1 Norme di sicurezza



## 1.1 Sicurezza

- Durante l'installazione e l'utilizzo del dispositivo osservare sempre le norme di prevenzione e sicurezza sul lavoro previste nel proprio paese;
- l'installazione e le operazioni di manutenzione devono essere eseguite da personale qualificato, in assenza di tensione e parti meccaniche in movimento;
- utilizzare il dispositivo esclusivamente per la funzione per cui è stato costruito: ogni altro utilizzo potrebbe risultare pericoloso per l'utilizzatore e per l'ambiente;
- alte correnti, tensioni e parti in movimento possono causare lesioni serie o fatali;
- non utilizzare in ambienti esplosivi o infiammabili;
- il mancato rispetto delle norme di sicurezza o delle avvertenze specificate in questo manuale è considerato una violazione delle norme di sicurezza standard previste dal costruttore o richieste dall'uso per cui lo strumento è destinato;
- Lika Electronic non si assume alcuna responsabilità per eventuali danni o lesioni derivanti dall'inosservanza delle norme di sicurezza da parte dell'utilizzatore.



## 1.2 Avvertenze elettriche

- Effettuare le connessioni elettriche esclusivamente in assenza di tensione;
- rispettare le connessioni riportate nella sezione "4 - Connessioni elettriche" a pagina 26;
- in conformità alla normativa 2014/30/UE sulla compatibilità elettromagnetica rispettare le seguenti precauzioni:
  - prima di maneggiare e installare il dispositivo eliminare la presenza di carica elettrostatica dal proprio corpo e dagli utensili che verranno in contatto con il dispositivo;
  - alimentare il dispositivo con tensione stabilizzata e priva di disturbi; se necessario, installare appositi filtri EMC all'ingresso dell'alimentazione;
  - utilizzare sempre cavi schermati e possibilmente "twistati";
  - non usare cavi più lunghi del necessario;
  - evitare di far passare il cavo dei segnali del dispositivo vicino a cavi di potenza;
  - installare il dispositivo il più lontano possibile da possibili fonti di interferenza o schermarlo in maniera efficace;
  - per garantire un funzionamento corretto del dispositivo, evitare l'utilizzo di apparecchiature con forte carica magnetica in prossimità dell'unità;
  - collegare la calza del cavo e/o la custodia del connettore e/o il corpo del dispositivo a un buon punto di terra; assicurarsi che il punto di terra sia privo di disturbi. Il collegamento a terra può essere effettuato sul lato dispositivo e/o sul lato utilizzatore; è compito dell'utilizzatore valutare la soluzione



migliore da adottare per minimizzare i disturbi. Si consiglia di effettuare il collegamento a terra il più vicino possibile all'encoder. Per la messa a terra si consiglia di utilizzare il punto di collegamento previsto sul coperchio del dispositivo (utilizzare 1 vite TCEI M3 x 6 a testa cilindrica con 2 rondelle zigrinate).



### 1.3 Avvertenze meccaniche

- Montare il dispositivo rispettando rigorosamente le istruzioni riportate nella sezione "3 - Istruzioni di montaggio" a pagina 21;
- effettuare il montaggio meccanico esclusivamente in assenza di parti meccaniche in movimento;
- non disassemblare il dispositivo;
- non eseguire lavorazioni meccaniche sul dispositivo o sull'albero;
- dispositivo elettronico delicato: maneggiare con cura; evitare urti o forti sollecitazioni sia all'asse che al corpo del dispositivo;
- utilizzare il dispositivo in accordo con le caratteristiche ambientali previste dal costruttore;
- encoder con asse sporgente: utilizzare giunti elastici per collegare encoder e motore; rispettare le tolleranze di allineamento ammesse dal giunto elastico;
- encoder con asse cavo: l'encoder può essere montato direttamente su un albero che rispetti le caratteristiche definite nel foglio d'ordine e fissato mediante il collare e, ove previsto, un pin antirotazione.

## 2 Identificazione

Il dispositivo è identificato mediante il **codice di ordinazione**, il **numero di serie** e l'**indirizzo MAC** stampati sull'etichetta applicata al dispositivo stesso; i dati sono ripetuti anche nei documenti di trasporto che lo accompagnano. Citare sempre il codice di ordinazione, il numero di serie e l'indirizzo MAC quando si contatta Lika Electronic per l'acquisto di un ricambio o nella necessità di assistenza tecnica. Per ogni informazione sulle caratteristiche tecniche del dispositivo fare riferimento al catalogo del prodotto.



**Attenzione:** gli encoder con codice di ordinazione finale "/Sxxx" possono avere caratteristiche meccaniche ed elettriche diverse dallo standard ed essere provvisti di documentazione aggiuntiva per cablaggi speciali (Technical info).

### 3 Istruzioni di montaggio



**ATTENZIONE**

L'installazione e le operazioni di manutenzione devono essere eseguite da personale qualificato, in assenza di tensione e componenti meccaniche in movimento.

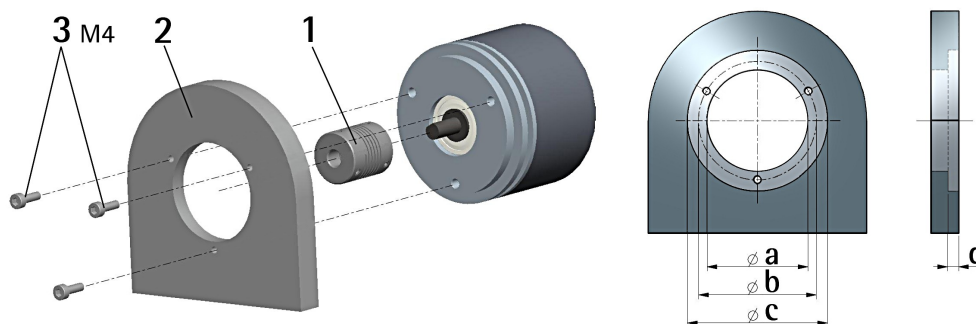
Per ogni informazione sulle caratteristiche meccaniche e i dati elettrici dell'encoder referirsi al datasheet del prodotto.

I valori sono espressi in millimetri (mm).

#### 3.1 Encoder con asse sporgente

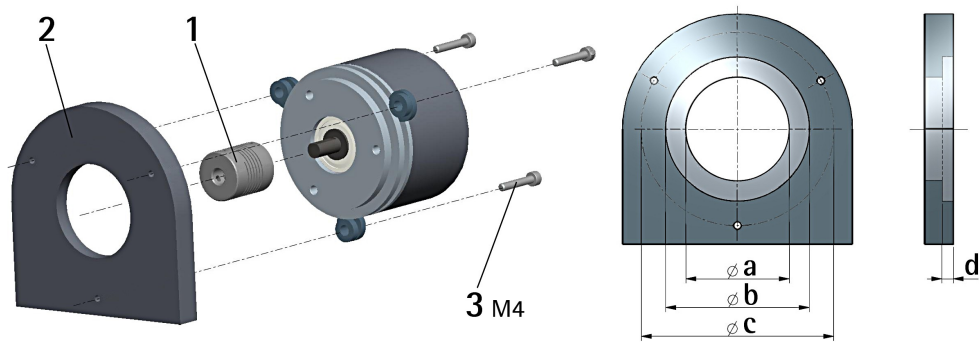
- Fissare il giunto elastico **1** all'encoder;
- fissare l'encoder alla flangia **2** o alla campana utilizzando le viti **3**;
- fissare la flangia **2** al supporto o la campana al motore;
- fissare il giunto elastico **1** al motore;
- assicurarsi che le tolleranze di disallineamento ammesse dal giunto elastico **1** siano rispettate.

##### 3.1.1. Fissaggio standard



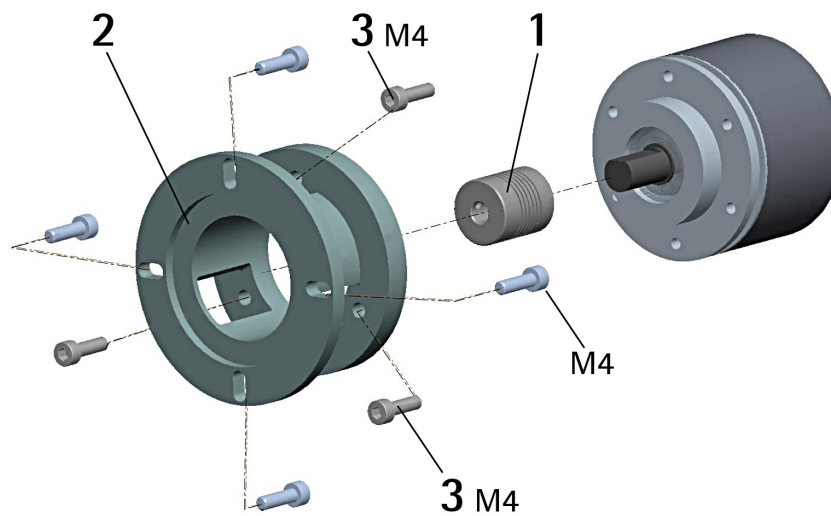
	a [mm]	b [mm]	c [mm]	d [mm]
EM58, HS58, HM58	-	42	50 F7	4
EM58S, HS58S, HM58S	36 H7	48	-	-

3.1.2 Fissaggio con graffe (codice LKM-386)



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

3.1.3 Fissaggio con campana (codice PF4256)



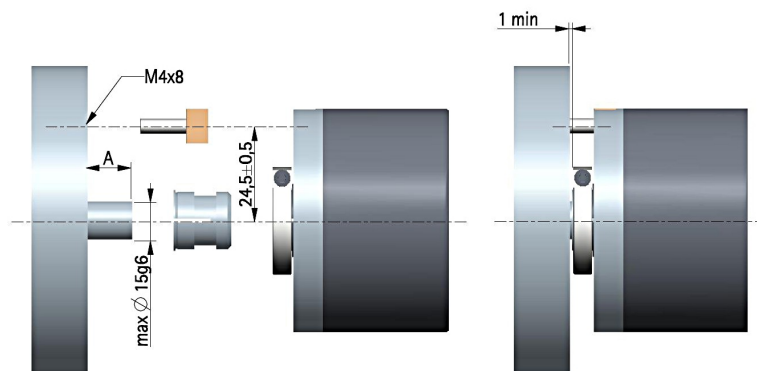
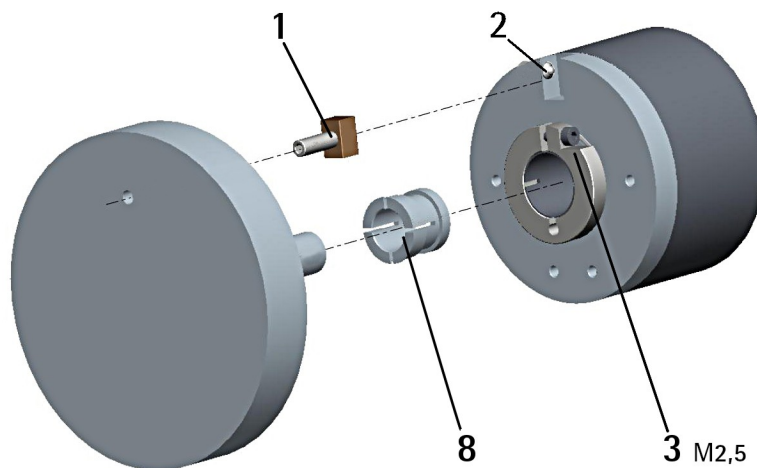
**NOTA**

Si raccomanda di utilizzare giunti elastici per collegare encoder ad asse sporgente e motore; rispettare le tolleranze di disallineamento ammesse dal giunto elastico.

### 3.2 Encoder con asse cavo

#### 3.2.1 EMC58, HSC58, HMC58

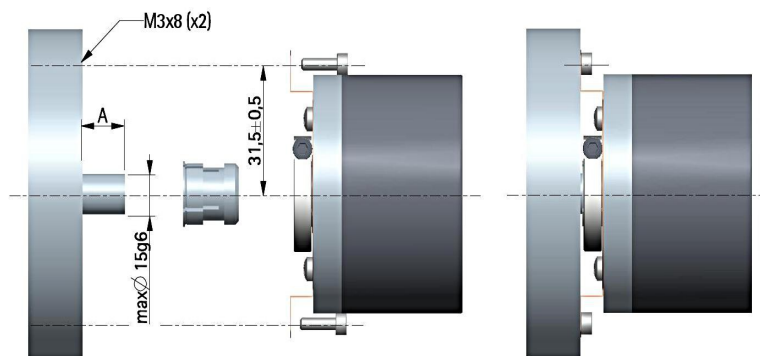
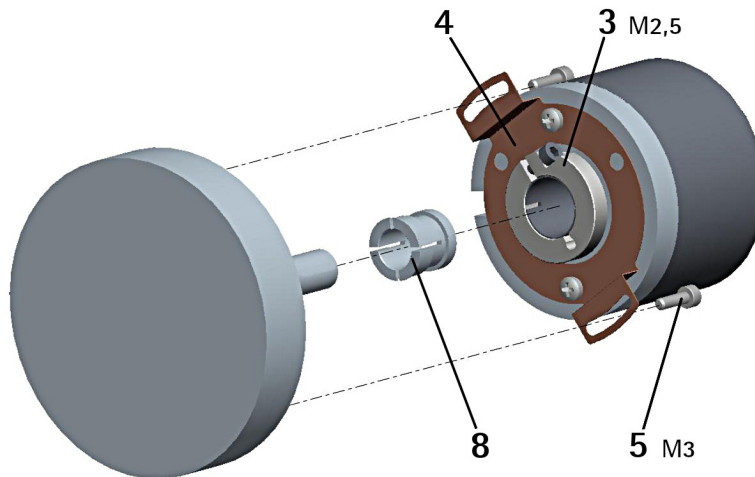
- Fissare il pin antirotazione **1** sul retro del motore (fissaggio con controdado);
- inserire l'encoder sull'albero del motore utilizzando la boccola di riduzione **8** (se fornita). Evitare sforzi sull'albero encoder;
- inserire il pin antirotazione **1** nella fresatura della flangia encoder; esso rimane così in posizione grazie al grano **2** prefissato da Lika;
- fissare il collare **3** dell'albero encoder (fissare la vite **3** con frenafiletto).



A = min. 8 mm, max. 18 mm

### 3.2.2 EMC59, HSC59, HMC59

- Inserire l'encoder sull'albero del motore utilizzando la boccia di riduzione **8** (se fornita); evitare sforzi sull'albero encoder;
- fissare la molla di fissaggio **4** sul retro del motore utilizzando due viti M3 x 8 a testa cilindrica **5**;
- fissare il collare **3** dell'albero encoder mediante la vite predisposta (fissare la vite **3** con frenafiletto).

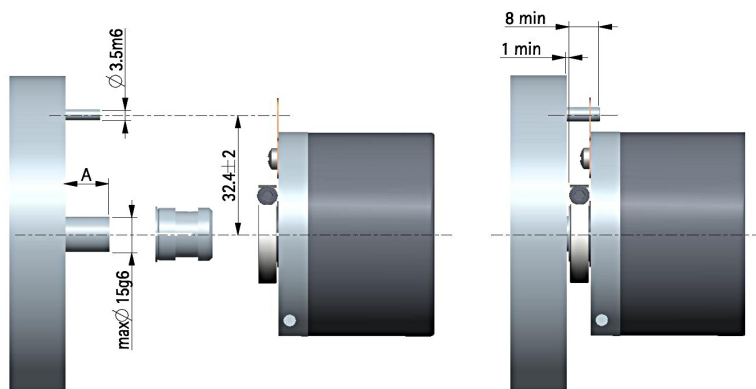
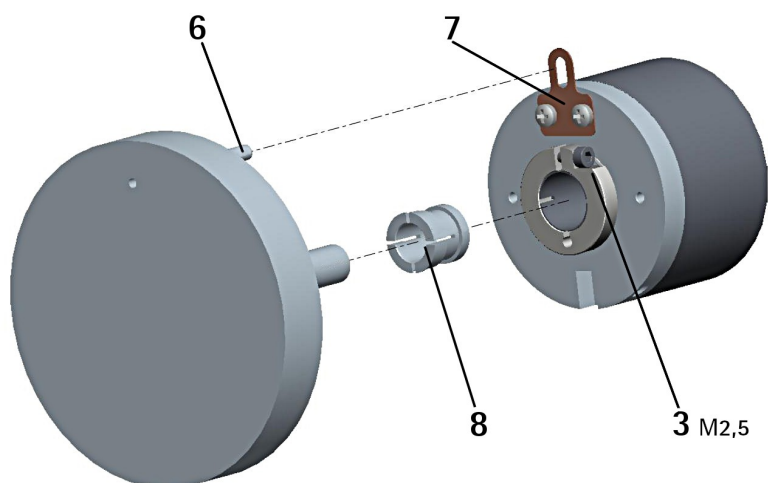


A = min. 8 mm, max. 18 mm



### 3.2.3 EMC60, HSC60, HMC60

- Rimuovere il pin antirotazione **1** (si veda la Figura a pagina 23);
- fissare la spina temprata **6** sul retro del motore;
- inserire l'encoder sull'albero del motore utilizzando la boccola di riduzione **8** (se fornita); evitare sforzi sull'albero encoder;
- assicurarsi che il pin antirotazione **6** sia inserito nella molla di fissaggio **7**;
- fissare il collare **3** dell'albero encoder mediante la vite predisposta (fissare la vite **3** con frenafiletto).



A = min. 8 mm, max. 18 mm



#### NOTA

Si raccomanda di non eseguire lavorazioni meccaniche con trapani o fresatrici sull'albero dell'encoder. Si potrebbero procurare danni irrimediabili ai componenti interni con immediata perdita della garanzia. Si prega di contattare il nostro servizio tecnico per informazioni sulla gamma di alberi "personalizzati".

## 4 Connessioni elettriche



### ATTENZIONE

L'alimentazione deve essere tolta prima di eseguire qualsiasi connessione elettrica!

L'installazione, le connessioni elettriche e le operazioni di manutenzione devono essere eseguite esclusivamente da personale qualificato, in assenza di tensione e componenti meccaniche in movimento.

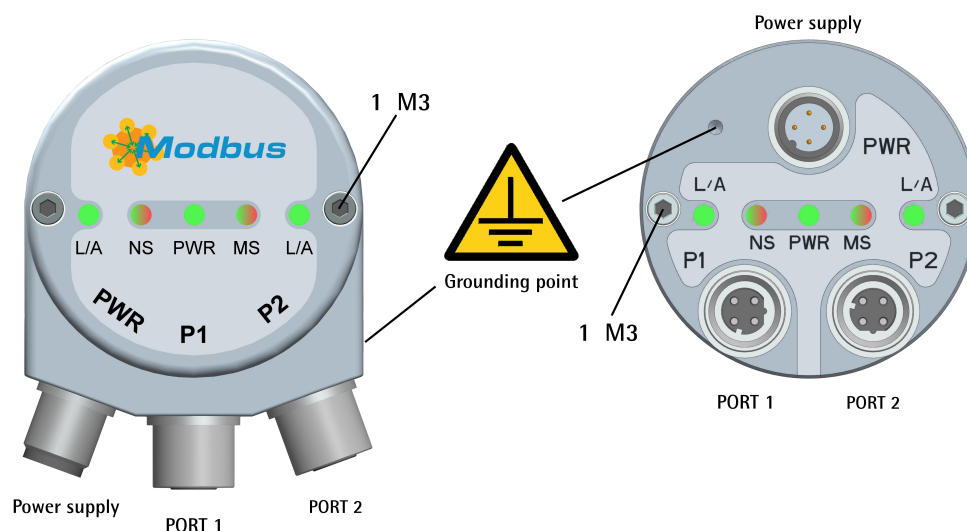


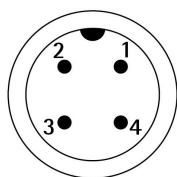
Figura 1 - Connettori e LED di diagnostica

### 4.1 Connettori M12

Sul coperchio sono montati tre connettori M12 con pin-out conforme allo standard Ethernet. Pertanto è possibile utilizzare cavi Ethernet standard disponibili in commercio. La PORTA 1 e la PORTA 2 sono intercambiabili.

#### 4.1.1 Connettore alimentazione PWR (Figura 1)

Il connettore M12 4 pin maschio codifica A è utilizzato per l'alimentazione.

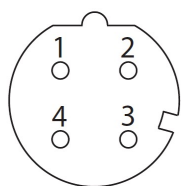


Descrizione	Pin
+10Vdc +30Vdc	1
n.c.	2
0Vdc	3
n.c.	4

n.c. = non collegato

### 4.1.2 Connettori P1 Porta 1 e P2 Porta 2 (Figura 1)

Due connettori M12 4 pin femmina codifica D sono utilizzati per il collegamento Ethernet attraverso la porta 1 e la porta 2.



Descrizione	Pin
<b>Tx Data +</b>	1
<b>Rx Data +</b>	2
<b>Tx Data -</b>	3
<b>Rx Data -</b>	4

L'interfaccia Ethernet supporta un funzionamento a 10/100 Mbit/s, full/half duplex.

I connettori M12 P1 PORTA 1 e P2 PORTA 2 hanno un pin-out in conformità allo standard Ethernet. Pertanto è possibile utilizzare cavi Ethernet standard disponibili in commercio, per maggiori informazioni si veda più oltre.

Le porte sono uguali e intercambiabili – se è richiesta una sola connessione, si può usare indifferentemente l'una o l'altra porta.

### 4.1.3 Configurazione di rete: topologie, cavi, hub, switch - Raccomandazioni

Dato che utilizzano Ethernet, le reti MODBUS/TCP supportano molteplici tipologie di connessione: lineare, ad albero, daisy-chain, a stella, ... Inoltre le reti MODBUS/TCP possono essere configurate in pressoché ogni topologia nella stessa struttura.

La connessione di un encoder può essere eseguita direttamente con una scheda di rete oppure indirettamente con uno switch, un hub o una rete aziendale.

I cavi e i connettori sono conformi alle specifiche Ethernet IEEE 802.3.

Se si utilizza una connessione diretta a un computer/controllore senza interporre componenti di reti, occorre usare un cavo di rete standard, "dritto" ("straight"), non un cavo "incrociato" ("crossover").

Per conseguire una velocità di trasferimento dei dati fino a 100 Mbit/s bisogna usare un cavo almeno di tipo CAT-5 (categoria 5). Nel caso fosse presente nella rete un componente di rete che non supporta il fast Ethernet, l'encoder abbasserà automaticamente la velocità di trasmissione a 10 Mbit/s.

E' possibile utilizzare cavi standard Ethernet disponibili sul mercato.

Per informazioni complete riferirsi a IEC 61918, IEC 61784-5-13 e IEC 61076-2-101.

Per aumentare l'immunità ai disturbi si possono usare solo cavi S/FTP o SF/FTP (CAT-5).

Occorre rispettare obbligatoriamente la lunghezza massima del cavo (100 metri) predefinita da Ethernet 100Base-TX.

Per quanto riguarda il cablaggio e le misure EMC, bisogna considerare gli standard IEC 61918 e IEC 61784-5-13.

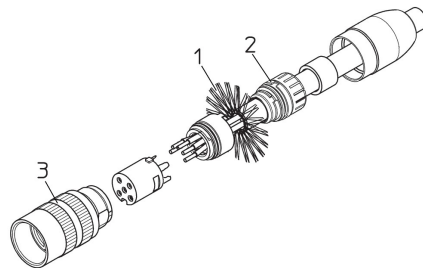
Per una lista completa delle prolunghe e dei kit di collegamento disponibili riferirsi al datasheet di prodotto (elenco in "Accessories").

#### 4.2 Collegamento a terra

Per minimizzare i disturbi collegare con cura la calza del cavo e/o la custodia del connettore e/o il corpo del dispositivo a un buon punto di terra. Collegare opportunamente la calza del cavo a terra sul lato utilizzatore. I cavi intestati serie EC- di Lika Electronic prevedono il collegamento della calza alla ghiera del connettore per la messa a terra attraverso il corpo del dispositivo. I connettori volanti serie E- di Lika Electronic utilizzano invece un connettore plastico; pertanto non è possibile la raccolta calza. Nel caso in cui si utilizzi un connettore metallico collegare opportunamente la calza del cavo attenendosi alle istruzioni del costruttore. In tutti i casi assicurarsi che il punto di terra sia privo di disturbi e il più vicino possibile al dispositivo. Per la messa a terra si consiglia di utilizzare il punto di collegamento previsto sul coperchio del dispositivo (si veda la Figura 1, utilizzare una vite TCEI M3 x 4 a testa cilindrica con due rondelle zigrinate).

#### 4.3 Collegamento della calza

Districare la calza **1** e tagliarla alla giusta misura; quindi piegarla sul particolare **2**; infine posizionare la ghiera **3** assicurandosi che la calza **1** e la ghiera **3** siano adeguatamente in contatto.



#### 4.4 Indirizzo MAC e indirizzo IP

E' possibile identificare l'unità nella rete attraverso l'**indirizzo MAC** e l'**indirizzo IP**. L'indirizzo MAC è un identificatore permanente e univoco a livello globale assegnato all'unità per la comunicazione sul livello fisico; mentre l'indirizzo IP è il nome dell'unità all'interno di una rete che utilizza il protocollo Internet. L'indirizzo MAC ha una dimensione di 6 byte e non può essere modificato. Consiste di due parti, i numeri sono espressi in notazione esadecimale: i primi tre byte sono utilizzati per identificare il costruttore (OUI, ossia Organizationally Unique Identifier, Identificativo Unico a livello di Organizzazione) e sono forniti dall'autorità di standardizzazione IEE; mentre gli altri tre byte rappresentano un numero consecutivo gestito dal costruttore e costituiscono l'identificativo specifico dell'unità. L'indirizzo MAC è leggibile sull'etichetta applicata al corpo dell'encoder ed è visualizzato anche nella pagina **Encoder Information** del web server.

L'indirizzo MAC ha la seguente struttura:

Valore bit 47 ... 24			Valore bit 23 ... 0		
10	B9	FE	X	X	X
<b>Codice azienda (OUI)</b>			<b>Numero consecutivo</b>		

L'indirizzo IP deve essere assegnato dall'utilizzatore a ciascuna interfaccia dell'unità che deve essere collegata in rete, l'indirizzo IP di default assegnato da Lika Electronic è 192.168.1.10, mentre la subnet mask di default è 255.255.255.0.

Per impostare i parametri di configurazione della rete riferirsi alla sezione seguente.

#### 4.5 Impostazione dell'indirizzo IP e dei parametri di configurazione della rete



##### ATTENZIONE

Solo tecnici competenti, ben istruiti e dotati di adeguata esperienza, esperti di architettura dei calcolatori, di progettazione di reti e di sistemi operativi devono avere accesso alla configurazione dei parametri di configurazione della rete.

L'impostazione inappropriata dei parametri di rete si traduce in un funzionamento errato del sistema.



##### ATTENZIONE

L'indirizzo MODBUS TCP/IP e i parametri di comunicazione possono essere impostati solo via software.

La tabella seguente riassume l'indirizzo IP di default e i parametri di configurazione della rete.

<b>Parametro IP</b>	<b>Valore</b>
Indirizzo IP	192.168.1.10
Subnet mask	255.255.255.0
Indirizzo Gateway	0.0.0.0

Per configurare la rete e impostare i parametri di comunicazione specifici, l'operatore deve accedere alla pagina **Network IP Configuration** del Web server.

E' valida qualsiasi impostazione nel range: 0.0.0.0 ... 255.255.255.255 in conformità con le regole del protocollo Internet.

Per qualsiasi informazione sulla pagina **Network IP Configuration** riferirsi alla sezione "8.8 Configurazione della rete" a pagina 93.


**NOTA**

Se per qualsiasi ragione fosse necessario ripristinare i valori di fabbrica (valori di default) dei parametri di configurazione della rete, occorre accedere al DIP switch DIP A ospitato all'interno del coperchio. Per informazioni complete riferirsi alla sezione "4.8 DIP A: reset dei parametri di configurazione della rete ai valori di fabbrica" a pagina 32.

**4.6 Resistenza di terminazione**

Non sono necessarie terminazioni di linea in quanto la chiusura della rete MODBUS/TCP avviene in modo automatico; ogni Slave infatti è in grado di rilevare la presenza o meno di altri Slave a valle.

**4.7 LED di diagnostica (Figura 1)**

Cinque LED montati sul coperchio dell'encoder (si veda la Figura 1) hanno lo scopo di mostrare visivamente lo stato di funzionamento o di guasto dell'encoder e dell'interfaccia MODBUS. Il significato di ciascun LED è spiegato nelle seguenti tabelle.

LED	Descrizione
<b>LED L/A Collegamento/ Attività (Link/Activity) sulla porta 1 P1 (verde)</b>	Mostra lo stato e l'attività del link fisico (porta 1 P1).
<b>OFF</b>	Link non attivo, nessuna attività sulla porta 1 P1.
<b>ON</b>	Link sulla porta 1 P1 attivo, nessuna attività.
<b>LAMPEGGIANTE</b>	Attività sulla porta 1 P1.

LED	Descrizione
<b>LED di errore NS Stato Rete (Network State) (verde / rosso)</b>	Mostra lo stato corrente della rete.
<b>OFF</b>	Nessun indirizzo IP rilevato o stato di guasto.
<b>ON verde</b>	E' stato ricevuto almeno un messaggio MODBUS.

<b>LAMPEGGIANTE verde</b>	L'encoder sta attendendo di ricevere il primo messaggio MODBUS.
<b>ON rosso</b>	Rilevato conflitto di indirizzo IP o ERRORE FATALE.
<b>LAMPEGGIANTE rosso</b>	Timeout di connessione, non è stato ricevuto nessun messaggio MODBUS entro il tempo di timeout impostato (si veda il registro <b>Watchdog timeout [82]</b> a pagina 54).

LED	Descrizione
<b>LED PWR Alimentazione (Power) (verde)</b>	Mostra lo stato di alimentazione del dispositivo.
<b>OFF</b>	L'encoder non è alimentato (OFF).
<b>ON</b>	L'encoder è alimentato (ON).

LED	Descrizione
<b>LED MS Stato Modulo (Module Status) (verde / rosso)</b>	Mostra lo stato del dispositivo MODBUS/TCP.
<b>OFF</b>	Non c'è alimentazione al dispositivo (OFF).
<b>ON verde</b>	Il dispositivo è in stato operativo normale.
<b>ON rosso</b>	Guasto grave, ERRORE FATALE.
<b>LAMPEGGIANTE rosso</b>	Guasto minore.

LED	Descrizione
<b>LED L/A Collegamento/ Attività (Link/Activity) sulla porta 2 P2 (verde)</b>	Mostra lo stato e l'attività del link fisico (porta 2 P2).
<b>OFF</b>	Link non attivo, nessuna attività sulla porta 2 P2.
<b>ON</b>	Link sulla porta 2 P2 attivo, nessuna attività.

<b>LAMPEGGIANTE</b>	Attività sulla porta 2 P2.
---------------------	----------------------------

**NOTA**

Se entrambi il LED NS Stato Rete e il LED MS Stato Modulo sono rossi, si è verificato un errore fatale.

**NOTA**

All'avvio, il dispositivo esegue un test hardware per verificare il funzionamento dei LED.

Sia il LED NS Stato Rete che il LED MS Stato Modulo si accendono.

**4.8 DIP A: reset dei parametri di configurazione della rete ai valori di fabbrica****ATTENZIONE**

Per accedere al DIP switch DIP A predisposto per il reset dei parametri di configurazione della rete ai valori di fabbrica (valori di default) occorre rimuovere il coperchio. Per ogni informazione sulla rimozione del coperchio riferirsi alla sezione seguente "4.9 Coperchio encoder (Figura 2)".

**ATTENZIONE**

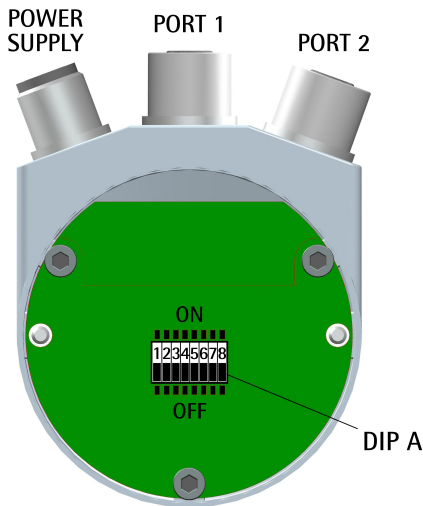
Prima di ripristinare il coperchio è fondamentale assicurarsi che il corpo dell'encoder e il coperchio siano allo stesso potenziale!

Se per qualsiasi ragione fosse necessario ripristinare i valori di fabbrica (valori di default) dei parametri di configurazione della rete (indirizzo IP, Subnet mask, DHCP), accedere al DIP switch DIP A procedendo come descritto di seguito:

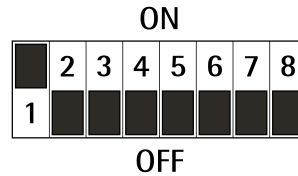
**ATTENZIONE**

Prestare la massima attenzione ai circuiti e collegamenti interni mentre il coperchio è aperto.





- Togliere l'alimentazione;
- rimuovere il coperchio come spiegato nella successiva sezione "4.9 Coperchio encoder (Figura 2)";
- impostare lo switch hardware 1 a ON;



- ricollegare il coperchio, quindi ridare tensione e attendere il completamento del processo di inizializzazione;

- togliere l'alimentazione;
- rimuovere nuovamente il coperchio e riportare ora lo switch hardware 1 a OFF;
- ripristinare il coperchio come spiegato nella successiva sezione "4.9 Coperchio encoder (Figura 2)";
- ridare tensione all'encoder per ripristinarne il normale funzionamento.

La tabella seguente riassume l'indirizzo IP e i parametri di configurazione della rete dopo l'operazione di reset.

Parametro IP	Valore
Indirizzo IP	192.168.1.10
Subnet mask	255.255.255.0
DHCP	OFF

#### 4.9 Coperchio encoder (Figura 2)



##### ATTENZIONE

Non rimuovere o connettere il coperchio dell'encoder se il dispositivo è alimentato. Alcuni componenti interni potrebbero danneggiarsi.

Il DIP switch DIP A predisposto per il reset dei parametri di configurazione della rete ai valori di fabbrica (valori di default) è ospitato all'interno del coperchio dell'encoder (si veda a pagina 32). Per accedere a questo elemento è pertanto necessario rimuovere il coperchio.

**NOTA**

Eeguire questa operazione con estrema prudenza per non danneggiare i componenti interni.

Per togliere il coperchio svitare le due viti **1**. Prestare la massima attenzione alla disconnessione del connettore interno.

Avere cura di ripristinare il coperchio al termine delle operazioni. Ricollegare con cura il connettore interno. Fissare le viti **1** applicando una coppia di serraggio di circa 2.5 Nm.

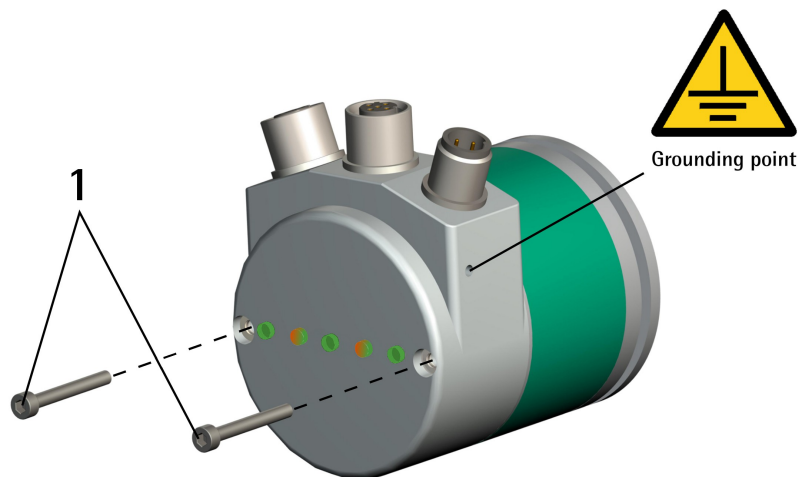


Figura 2 - Rimozione del coperchio dell'encoder

**ATTENZIONE**

Prima di ripristinare il coperchio è fondamentale assicurarsi che il corpo dell'encoder e il coperchio siano allo stesso potenziale!

## 5 Quick reference

### 5.1 Getting started

Le istruzioni che seguono permettono all'operatore un set up rapido e sicuro dell'encoder in una modalità di funzionamento standard.

- Installare il dispositivo meccanicamente;
- eseguire le connessioni elettriche e di rete;
- se richiesto, impostare i parametri di comunicazione per permettere all'unità l'accesso alla rete MODBUS/TCP, si veda la sezione "4.5 Impostazione dell'indirizzo IP e dei parametri di configurazione della rete" a pagina 29; i parametri di configurazione della rete di default sono i seguenti:

Parametro IP	Valore
Indirizzo IP	192.168.1.10
Subnet mask	255.255.255.0
Gateway di default	0.0.0.0

- Accendere il dispositivo fornendo tensione +10Vdc +30Vdc;
- se si deve usare la risoluzione fisica dell'unità, verificare che il bit **Scaling function** sia disabilitato (il bit 0 nei registri **Operating parameters [109-110]** = 0; si veda a pagina 60); in questo caso l'encoder utilizza i valori dei registri **Singleturn resolution [113-114]** e **Number of revolutions [115-116]** per calcolare il valore della posizione assoluta;
- se invece è richiesta una risoluzione specifica, abilitare il bit **Scaling function** (il bit 0 nei registri **Operating parameters [109-110]** = 1; si veda a pagina 60);
- impostare quindi il valore desiderato per la risoluzione monogiro nei registri **Counts per revolution [101-102]**, si veda a pagina 55;
- impostare il valore desiderato per la risoluzione totale nei registri **Total Resolution [103-104]**, si veda a pagina 57;
- ora, se richiesto, è possibile impostare un valore di Preset nei registri **Preset value [105-106]** e attivarlo mediante l'esecuzione del comando **Perform counting preset** disponibile nei registri **Control Word [111-112]**, si veda a pagina 59;
- salvare i nuovi valori impostati (usare il comando **Save parameters** disponibile nei registri **Control Word [111-112]**, si veda a pagina 62).



#### NOTA

Il protocollo MODBUS/TCP non richiede alcun file di configurazione.

## 6 Interfaccia MODBUS®/TCP

Gli encoder MODBUS/TCP di Lika sono dispositivi Slave (Server) e implementano il protocollo di applicazione MODBUS (livello 7 del modello OSI) e il protocollo "MODBUS messaging on TCP/IP" (Ethernet: livelli 1 & 2 del modello OSI; TCP/IP: livelli 3 & 4 del modello OSI).

Per ogni ulteriore informazione o specifica omessa riferirsi ai documenti "MODBUS Application Protocol Specification, Version V1.1b3" e "MODBUS messaging on TCP/IP implementation guide, Version V1.0b" disponibili all'indirizzo [www.modbus.org](http://www.modbus.org).

### 6.1 Principi guida del protocollo MODBUS

MODBUS è un protocollo di comunicazione del livello di applicazione, posizionato al livello 7 del modello OSI, che permette una comunicazione Client/Server tra dispositivi collegati su tipi diversi di bus o reti. In particolare, il servizio di messaggistica MODBUS/TCP permette una comunicazione Client/Server tra dispositivi collegati sulla rete Ethernet TCP/IP.

Il protocollo Modbus è stato sviluppato nel 1979 da Modicon, per sistemi di automazione industriale e controllori programmabili Modicon. Da allora è diventato un metodo standard industriale per il trasferimento di informazioni I/O discrete/analogiche e dati di registro tra dispositivi industriali di controllo e monitoraggio.

I dispositivi MODBUS comunicano utilizzando un metodo Master-Slave (Client-Server) nel quale solo un dispositivo (il Master/Client) può dare inizio alle transazioni (chiamate query, richieste). Gli altri dispositivi (gli Slave/Server) rispondono fornendo al Master i dati richiesti, oppure eseguendo l'azione richiesta tramite la query. Uno Slave è un qualsiasi dispositivo periferico (un trasduttore I/O, una valvola, un drive di rete, o un altro dispositivo di misura) che processa le informazioni e trasmette il proprio output al Master tramite MODBUS.

I Master possono rivolgersi a singoli Slave oppure avviare messaggi di richiesta broadcast rivolti a tutti gli Slave. Gli Slave rispondono a tutte le query loro rivolte individualmente, ma non rispondono alle query di tipo broadcast. Gli Slave non prendono l'iniziativa di avviare uno scambio di messaggi, semplicemente rispondono alle richieste inviate dal Master.

MODBUS/TCP (noto anche come MODBUS-TCP o MODBUS TCP/IP) è semplicemente il protocollo MODBUS RTU con una interfaccia TCP attiva su Ethernet.

La struttura dello scambio dati MODBUS è definito dal protocollo di applicazione che stabilisce le regole per l'organizzazione e l'interpretazione dei dati indipendentemente dal mezzo di trasmissione dai dati.

TCP/IP si riferisce al Transmission Control Protocol e all'Internet Protocol, che descrivono il mezzo di trasmissione per lo scambio dati MODBUS/TCP.

Tra i vantaggi rilevanti di MODBUS/TCP troviamo:

- MODBUS/TCP prende semplicemente il set di istruzioni MODBUS e impacchetta il tutto in TCP/IP;
- supporta lo standard Ethernet e non necessita di Master o chipset dedicati; si possono utilizzare schede Ethernet per PC e PC standard per comunicare in qualsiasi rete Ethernet;
- non richiede alcun file di configurazione;
- non necessita di alcun software specifico grazie alla possibilità di integrare un web server: è progettato per offrire funzioni utili e informazioni complete sul dispositivo al quale è possibile accedere tramite Internet.

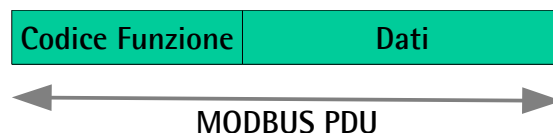
In particolare permette di:

- visualizzare e verificare i parametri correntemente impostati;
- impostare i parametri di comunicazione della rete;
- impostare i parametri di lavoro del dispositivo;
- aggiornare il firmware;
- monitorare il dispositivo e accedere ad alcune funzioni di manutenzione avanzate.

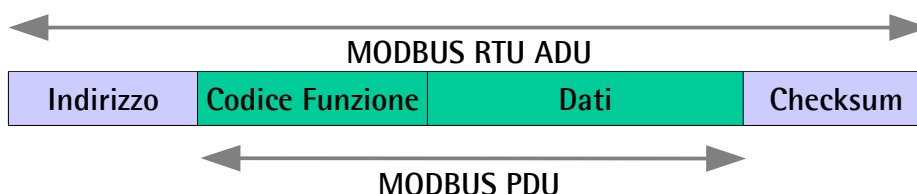
Al web server è possibile accedere da qualsiasi PC su cui sia installato un web browser.

### 6.2 Descrizione generale del frame MODBUS

Il protocollo di applicazione MODBUS definisce una semplice **Protocol Data Unit (PDU)** indipendentemente dai livelli di comunicazione sottostanti:



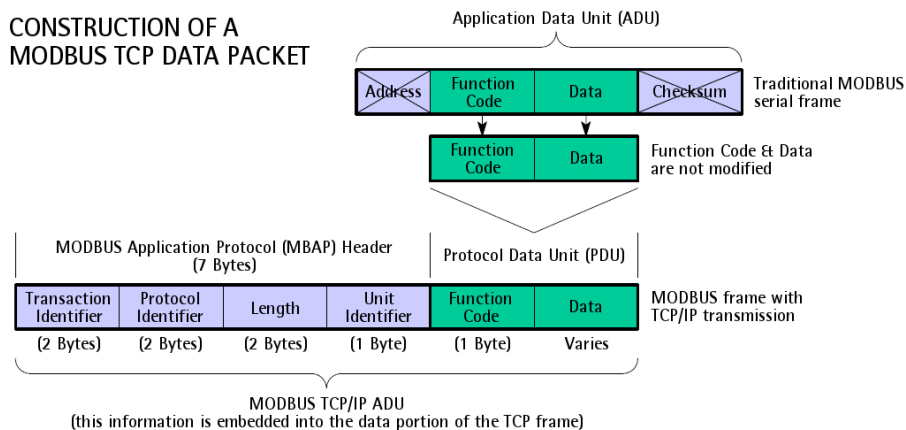
La mappatura del protocollo MODBUS su un bus specifico o su una rete introduce alcuni campi ulteriori nella **Application Data Unit (ADU)**. Il Client che avvia una transazione MODBUS prepara la Application Data Unit MODBUS e aggiunge poi i campi al fine di ottenere la ADU di comunicazione appropriata. Il codice funzione indica al Server che tipo di azione eseguire.



### 6.3 MODBUS su Application Data Unit TCP/IP

MODBUS/TCP utilizza TCP/IP ed Ethernet per trasportare i dati della struttura di messaggio MODBUS tra dispositivi compatibili. Ossia, MODBUS/TCP combina una rete fisica (Ethernet) con uno standard di collegamento (TCP/IP), e un metodo standard di rappresentazione dei dati (MODBUS come protocollo di applicazione). Essenzialmente, il messaggio MODBUS/TCP è semplicemente una comunicazione MODBUS incapsulata in un involucro Ethernet TCP/IP.

In pratica, MODBUS TCP integra un data frame MODBUS standard in un frame TCP, senza l'indirizzo e il checksum di MODBUS RTU, come mostrato nel diagramma seguente.



Come si può notare, la Protocol Data Unit viene integrata nella sua forma originaria.

La Application Data Unit (ADU) di MODBUS/TCP prende la forma di una header a 7 byte (MBAP Header -MODBUS Application Protocol Header: Transaction Identifier -Identificatore di transazione- + Protocol Identifier -Identificatore di protocollo- + campo dimensione + Unit Identifier -Identificatore di unità) e la protocol data unit (MODBUS PDU: Codice funzione + Dati).

- **MBAP HEADER** (MODBUS Application Protocol Header). Una header dedicata è utilizzata in TCP/IP per identificare la MODBUS Application Data Unit. La MBAP Header contiene i seguenti campi:
  - **Transaction Identifier (Identificatore di transazione):** ha una dimensione di 2 byte ed è utilizzato per l'abbinamento delle transazioni, vale a dire per l'identificazione di una transazione Richiesta / Risposta MODBUS. E' inizializzato dal Client, il Server poi copia nella risposta il Transaction Identifier ricevuto con la richiesta.
  - **Protocol Identifier (Identificatore di protocollo):** ha una dimensione di 2 byte ed è utilizzato per il multiplexing intra-sistema. Il protocollo MODBUS è identificato dal valore 0. E' inizializzato dal Client, il Server copia poi nella risposta il Protocol Identifier ricevuto con la richiesta.
  - **Dimensione (Length):** ha una dimensione di 2 byte e rappresenta il numero di byte dei campi che seguono, incluso lo Unit Identifier, il codice funzione e il campo dati. E' inizializzato dal Client nella richiesta ed è inizializzato dal Server nella risposta.

- **Unit Identifier (Identificatore dell'unità):** ha una dimensione di 1 byte ed è utilizzato per scopi di routing intra-sistema. Tipicamente si usa per comunicare a uno Slave MODBUS+ o a uno Slave MODBUS su linea seriale attraverso un gateway tra una rete Ethernet TCP/IP e una linea seriale MODBUS. Questo campo è impostato dal Client MODBUS nella richiesta e deve essere restituito dal Server nella risposta con lo stesso valore. In una applicazione Server MODBUS/TCP tipica, lo Unit Identifier è impostato a 00 hex o FF hex.
- **CODICE FUNZIONE (FUNCTION CODE):** il codice funzione indica al Server che tipo di azione eseguire. Il codice funzione è seguito da un campo **DATI** che contiene i parametri di richiesta e risposta. Tutte le richieste e le risposte MODBUS sono create in modo tale che il destinatario possa verificare che un messaggio è concluso. Per i codici funzione in cui la PDU MODBUS ha una lunghezza fissa, il codice funzione da solo è sufficiente. Per i codici funzione che implicano una quantità variabile di dati nella richiesta o nella risposta, il campo dati include un conteggio dei byte. Per ogni ulteriore informazione sui codici funzione implementati riferirsi alla sezione "6.5 Codici funzione" a pagina 40.
- **DATI (DATA):** il campo **DATI** dei messaggi contiene i byte per le informazioni aggiuntive e le specifiche di trasmissione che il Server utilizza per dare seguito all'azione definita dal **CODICE FUNZIONE**. Questo può includere elementi come per esempio indirizzi discreti e indirizzi di registro, la quantità degli elementi da gestire e il conteggio dei byte di dati effettivi presenti nel campo. La struttura del campo **DATI** dipende da ciascun **CODICE FUNZIONE** (riferirsi alla sezione "6.5 Codici funzione" a pagina 40).

La Application Data Unit MODBUS/TCP completa è integrata all'interno del campo dati di un frame TCP standard e inviata via TCP alla **porta registrata 502**, che è specificamente riservata per le applicazioni MODBUS. I Client e i Server MODBUS/TCP attendono e ricevono i dati MODBUS attraverso la porta 502.



**NOTA**

MODBUS usa una rappresentazione 'big Endian' per indirizzi e dati. Questo significa che quando è trasmessa una quantità numerica maggiore di un singolo byte, il byte più significativo (MSB, most significant byte) viene trasmesso per primo. Quindi per esempio:

Dimensione registro	valore	
16 bit	1234hex	il primo byte trasmesso è 12hex, quindi 34hex

## 6.4 PDU MODBUS

Il protocollo MODBUS definisce tre PDU, esse sono:

- **MODBUS Request PDU (PDU di richiesta MODBUS);**
- **MODBUS Response PDU (PDU di risposta MODBUS);**
- **MODBUS Exception Response PDU (PDU di risposta di eccezione).**

La **MODBUS Request PDU** consiste di {function\_code, request\_data}, dove:  
function\_code = codice funzione MODBUS [1 byte];  
request\_data = questo campo dipende dal codice funzione utilizzato e solitamente contiene informazioni quali coordinate di variabili, valori di variabili, offset dati, codici di sottofunzioni, ecc. [n byte].

La **MODBUS Response PDU** consiste di {function\_code, response\_data}, dove:  
function\_code = codice funzione MODBUS [1 byte];  
response\_data = questo campo dipende dal codice funzione utilizzato e solitamente contiene informazioni quali coordinate di variabili, valori di variabili, offset dati, codici di sottofunzioni, ecc. [n byte].

La **MODBUS Exception Response PDU** consiste di {exception-function\_code, exception\_code}, dove:  
exception-function\_code = codice funzione MODBUS + 0080 hex [1 byte];  
exception\_code = codice di eccezione MODBUS, riferirsi alla tabella "MODBUS Exception Codes" nella sezione 7 del documento "MODBUS Application Protocol Specification V1.1b", [1 byte].

La dimensione della PDU MODBUS è soggetta ai limiti dimensionali ereditati dalla prima implementazione di MODBUS sulla rete di Linea Seriale (max. ADU RS-485 = 256 byte).

Pertanto:

**PDU MODBUS per la comunicazione di linea seriale** = 256 – indirizzo del Server (1 byte) – CRC (2 byte) = 253 byte.

Di conseguenza:

**ADU RS-232 / RS-485** = 253 byte + indirizzo del Server (1 byte) + CRC (2 byte) = **256 byte**.

**ADU TCP MODBUS** = 253 byte + MBAP (7 byte) = **260 byte**.

## 6.5 Codici funzione

Come detto in precedenza, i codici funzione (function code) esplicitano al Server il tipo di azione da eseguire.

Il campo codice funzione di una Protocol Data Unit MODBUS è codificato in un byte. I codici validi sono compresi nel range da 1 a 255 decimale (il range da 128 a 255 è riservato e utilizzato per le Exception Response). Quando un messaggio è inviato da un Client a un dispositivo Server il campo codice funzione informa il Server sul tipo di azione da eseguire. Il codice funzione "0" non è valido.



Esistono tre categorie di codici funzione MODBUS, esse sono:

- **codici funzione pubblici;**
- **codici funzione definiti dall'utilizzatore;**
- **codici funzione riservati.**

I **codici funzione pubblici** sono compresi nel range tra 1 e 64, tra 73 e 99 e tra 111 e 127; questi codici funzione sono definiti e approvati dalla comunità di MODBUS-IDA.org che ne gestisce la conformità e ne garantisce l'unicità; sono documentati pubblicamente.

La gamma di codici funzione da 65 a 72 e da 100 a 110 raggruppa i **codici funzione definiti dall'utilizzatore**: l'utilizzatore può scegliere e implementare un codice funzione che non sia supportato dalle specifiche; è chiaro che questo non può garantire l'unicità del codice funzione scelto.

I **codici funzione riservati** non sono disponibili in alcun modo per un utilizzo pubblico.

### 6.5.1 Codici funzione implementati

Gli encoder MODBUS/TCP di Lika implementano esclusivamente i codici funzione pubblici descritti qui di seguito.

#### 03 Read Holding Register

FC = 03 (0003 hex) ro

Questo codice funzione è usato per LEGGERE i valori in un blocco contiguo di Holding Register (indirizzi di riferimento 4X) in un dispositivo remoto; in altri termini, permette la lettura dei valori impostati in un insieme di parametri di lavoro in successione nel dispositivo. La Request PDU specifica l'indirizzo del primo registro del gruppo e il numero di registri del gruppo. Nella PDU i registri sono indirizzati a partire da zero. Ne consegue che i registri numerati 1-16 hanno indirizzo 0-15.

Il valore del registro nel messaggio di risposta è confezionato in due byte per registro, con i valori binari allineati a destra all'interno di ogni byte. Per ciascun registro, il primo byte contiene i bit di livello alto (msb, bit più significativi) mentre il secondo contiene i bit di livello basso (lsb, bit meno significativi).

Per una lista completa degli holding register accessibili utilizzando il codice funzione **03 Read Holding Register** riferirsi alla sezione "7.1.1 Parametri Holding Register" a pagina 53.

#### Request PDU

Codice funzione	1 byte	<b>0003 hex</b>
Indirizzo iniziale	2 byte	da 0000 hex a FFFF hex
Numero di registri	2 byte	da 1 a 125 (007D hex)

**Response PDU**

Codice funzione	1 byte	0003 hex
Numero byte	1 byte	2 x N*
Valore dei registri	N* x 2 byte	

\*N = Numero di registri

**Exception Response PDU**

Codice errore	1 byte	0083 hex (=0003 hex + 0080 hex)
Codice eccezione	1 byte	01 o 02 o 03 o 04



Esempio di richiesta di lettura dei registri **Preset value [105-106]** (indirizzo 104-105).

Request		Response	
Nome campo	(Hex)	Nome campo	(Hex)
Funzione	03	Funzione	03
Indirizzo iniziale Hi	00	Numero byte	04
Indirizzo iniziale Lo	68	Registro 105 valore Hi	00
Numero di registri Hi	00	Registro 105 valore Lo	00
Numero di registri Lo	02	Registro 106 valore Hi	05
		Registro 106 valore Lo	DC

Come si evince dalla tabella, i registri **Preset value [105-106]** (indirizzo 104-105) contengono i valori 00 00 hex e 05 DC hex rispettivamente, ossia 1500 in notazione decimale.

La ADU MODBUS/TCP necessaria per la richiesta di lettura dei registri **Preset value [105-106]** (indirizzo 104-105) è la seguente:

**MBAP Header + Request PDU** (in notazione esadecimale)

[00][01][00][00][00][06][00][03][00][68][00][02]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][06] = Dimensione

[00] = Identificatore di unità

[03] = codice funzione **03 Read Holding Register**

[00][68] = indirizzo iniziale (registri **Preset value [105-106]**, indirizzo 104-105)

[00][02] = numero di registri richiesti

La ADU MODBUS/TCP necessaria per restituire il valore dei registri **Preset value [105-106]** (indirizzo 104-105) è la seguente:

**MBAP Header + Response PDU** (in notazione esadecimale)

[00][01][00][00][00][07][00][03][04][00][00][05][DC]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][07] = Dimensione

[00] = Identificatore di unità

[03] = codice funzione **03 Read Holding Register**

[04] = numero di byte (2 byte per ciascun registro)

[00][00] = valore del registro 105, 00 00 hex = 0 dec

[05][DC] = valore del registro 106, 05 DC hex = 1500 dec

#### **04 Read Input Register**

FC = 04 (0004 hex)

Questo codice funzione è utilizzato per LEGGERE registri di input contigui compresi tra 1 e 125 (indirizzi di riferimento 3X) in un dispositivo remoto; in altri termini, permette di leggere alcuni valori di risultato e messaggi di allarme / stato in un dispositivo remoto. La Request PDU specifica l'indirizzo del primo registro da leggere e il numero di registri. Nella PDU i registri sono indirizzati a partire da zero. Ne consegue che i registri di ingresso numerati 1-16 hanno indirizzo 0-15.

I valori del registro nel messaggio di risposta sono confezionati in due byte per registro, con i valori binari allineati a destra all'interno di ogni byte. Per ciascun registro, il primo byte contiene i bit di livello alto (msb, bit più significativi) mentre il secondo contiene i bit di livello basso (lsb, bit meno significativi).

Per la lista completa dei registri di input accessibili utilizzando il codice funzione **04 Read Input Register** riferirsi alla sezione "7.1.2 Parametri Input Register" a pagina 70.

#### **Request PDU**

Codice funzione	1 byte	<b>0004 hex</b>
Indirizzo iniziale	2 byte	da 0000 hex a FFFF hex
Numero di registri di input	2 byte	da 0000 hex a 007D hex

**Response PDU**

Codice funzione	1 byte	<b>0004 hex</b>
Numero byte	1 byte	2 x N*
Valore dei registri di input	N* x 2 byte	

\*N = Numero di registri

**Exception Response PDU**

Codice errore	1 byte	<b>0084 hex (=0004 hex + 0080 hex)</b>
Codice eccezione	1 byte	01 o 02 o 03 o 04



Esempio di richiesta di lettura dei registri **Current position [1-2]** (indirizzo 0-1).

Request		Response	
Nome campo	(Hex)	Nome campo	(Hex)
Funzione	<b>04</b>	Funzione	<b>04</b>
Indirizzo iniziale Hi	<b>00</b>	Numero byte	<b>04</b>
Indirizzo iniziale Lo	<b>00</b>	Registro 1 valore Hi	<b>00</b>
Numero di Reg. Input Hi	<b>00</b>	Registro 1 valore Lo	<b>00</b>
Numero di Reg. Input Lo	<b>02</b>	Registro 2 valore Hi	<b>2F</b>
		Registro 2 valore Lo	<b>F0</b>

Come si evince dalla tabella, i registri **Current position [1-2]** (indirizzo 0-1) contengono i valori 00 00 hex e 2F F0 hex rispettivamente, ossia 12272 in notazione decimale.

La ADU MODBUS/TCP necessaria per la richiesta di lettura dei registri **Current position [1-2]** (indirizzo 0-1) è la seguente:

**MBAP Header + Request PDU (in notazione esadecimale)**

[00][01][00][00][00][06][00][04][00][00][00][02]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][06] = Dimensione

[00] = Identificatore di unità

[04] = codice funzione **04 Read Input Register**

[00][00] = indirizzo iniziale (registri **Current position [1-2]**, indirizzo 0-1)

[00][02] = numero di registri richiesti

La ADU MODBUS/TCP necessaria per restituire il valore dei registri **Current position [1-2]** (indirizzo 0-1) è la seguente:

**MBAP Header + Response PDU** (in notazione esadecimale)

[00][01][00][00][00][07][00][04][04][00][00][2F][F0]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][07] = Dimensione

[00] = Identificatore di unità

[04] = codice funzione **04 Read Input Register**

[04] = numero di byte (2 byte per ciascun registro)

[00][00] = valore del registro 1, 00 00 hex = 0 dec

[2F][F0] = valore del registro 2, 2F F0 hex = 12272 dec

## 06 Write Single Register

FC = 06 (0006 hex)

Questo codice funzione è utilizzato per la SCRITTURA di un singolo Holding Register (indirizzi di riferimento 4X) in un dispositivo remoto. La Request PDU specifica l'indirizzo del registro da scrivere. I registri sono indirizzati a partire da zero. Ne consegue che il registro numerato con 1 ha indirizzo 0.

La risposta positiva rispecchia nella sua struttura la domanda ed è inviata dopo che il valore richiesto è stato scritto.

Per la lista completa degli holding register accessibili utilizzando il codice funzione **06 Write Single Register** riferirsi alla sezione "7.1.1 Parametri Holding Register" a pagina 53.

### Request PDU

Codice funzione	1 byte	<b>0006 hex</b>
Indirizzo registro	2 byte	da 0000 hex a FFFF hex
Valore registro	2 byte	da 0000 hex a FFFF hex

**Response PDU**

Codice funzione	1 byte	<b>0006 hex</b>
Indirizzo registro	2 byte	da 0000 hex a FFFF hex
Valore registro	2 byte	da 0000 hex a FFFF hex

**Exception Response PDU**

Codice errore	1 byte	<b>0086 hex (=0006 hex + 0080 hex)</b>
Codice eccezione	1 byte	01 o 02 o 03 o 04



Esempio di richiesta di scrittura del registro **Watchdog timeout [82]** (indirizzo 81): si vuole abilitare la funzione di Watchdog e impostare un timeout di 10 ms.

Request		Response	
Nome campo	(Hex)	Nome campo	(Hex)
Funzione	<b>06</b>	Funzione	<b>06</b>
Indirizzo registro Hi	<b>00</b>	Indirizzo registro Hi	<b>00</b>
Indirizzo registro Lo	<b>51</b>	Indirizzo registro Lo	<b>51</b>
Valore registro Hi	<b>00</b>	Valore registro Hi	<b>00</b>
Valore registro Lo	<b>0A</b>	Valore registro Lo	<b>0A</b>

Come si evince dalla tabella, nel registro **Watchdog timeout [82]** (indirizzo 81) è impostato il valore 00 0A hex (10 dec): la funzione di Watchdog è abilitata e il timeout è impostato a 10 ms.

La ADU MODBUS/TCP necessaria per la richiesta di scrittura del valore 00 0A hex nel registro **Watchdog timeout [82]** (indirizzo 81) è la seguente:

**MBAP Header + Request PDU** (in notazione esadecimale)

[00][01][00][00][00][06][00][06][00][51][00][0A]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][06] = Dimensione

[00] = Identificatore di unità

[06] = codice funzione **06 Write Single Register**

[00][51] = indirizzo del registro **Watchdog timeout [82]**, 51 hex = 81 dec

[00][0A] = valore da impostare nel registro

La ADU MODBUS/TCP necessaria per restituire una risposta a seguito della richiesta di scrittura del valore 00 0A hex nel registro **Watchdog timeout [82]** (indirizzo 81) è la seguente:

**MBAP Header + Response PDU** (in notazione esadecimale)

[00][01][00][00][00][06][00][06][00][51][00][0A]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][06] = Dimensione

[00] = Identificatore di unità

[06] = codice funzione **06 Write Single Register**

[00][51] = indirizzo del registro **Watchdog timeout [82]**, 51 hex = 81 dec

[00][0A] = valore impostato nel registro

### 16 Write Multiple Register

FC = 16 (0010 hex)

Questo codice funzione è utilizzato per la SCRITTURA di un blocco di Holding Register contigui (indirizzi di riferimento 4X, registri da 1 a 123) in un dispositivo remoto.

I valori da scrivere sono specificati nel campo dati della richiesta. I dati sono confezionati in due byte per registro.

La risposta positiva restituisce il codice funzione, l'indirizzo iniziale e il numero di registri scritti.

Per la lista completa degli holding register accessibili utilizzando il codice funzione **16 Write Multiple Register** riferirsi alla sezione "7.1.1 Parametri Holding Register" a pagina 53.

### Request PDU

Codice funzione	1 byte	<b>0010 hex</b>
Indirizzo iniziale	2 byte	da 0000 hex a FFFF hex
Numero di registri	2 byte	da 0001 hex a 007B hex
Numero byte	1 byte	2 x <b>N*</b>
Valore registri	<b>N*</b> x 2 byte	valore

\*N = Numero di registri

**Response PDU**

Codice funzione	1 byte	<b>0010 hex</b>
Indirizzo iniziale	2 byte	da 0000 hex a FFFF hex
Numero di registri	2 byte	da 1 a 123 (007B hex)

**Exception Response PDU**

Codice errore	1 byte	<b>0090 hex (= 0010 hex + 0080 hex)</b>
Codice eccezione	1 byte	01 o 02 o 03 o 04



Esempio di richiesta di scrittura del valore 00 00 08 00 hex (=2048 dec) nei registri **Counts per revolution [101-102]** (indirizzo 100-101) e del valore 00 80 00 00 hex (=8388608 dec) nei registri **Total Resolution [103-104]** (indirizzo 102-103).

<b>Request</b>		<b>Response</b>	
Nome campo	(Hex)	Nome campo	(Hex)
Funzione	<b>10</b>	Funzione	<b>10</b>
Indirizzo iniziale Hi	<b>00</b>	Indirizzo iniziale Hi	<b>00</b>
Indirizzo iniziale Lo	<b>64</b>	Indirizzo iniziale Lo	<b>64</b>
Numero di registri Hi	<b>00</b>	Numero di registri Hi	<b>00</b>
Numero di registri Lo	<b>04</b>	Numero di registri Lo	<b>04</b>
Numero byte	<b>08</b>		
Registro 101 valore Hi	<b>00</b>		
Registro 101 valore Lo	<b>00</b>		
Registro 102 valore Hi	<b>08</b>		
Registro 102 valore Lo	<b>00</b>		
Registro 103 valore Hi	<b>00</b>		
Registro 103 valore Lo	<b>80</b>		
Registro 104 valore Hi	<b>00</b>		
Registro 104 valore Lo	<b>00</b>		

Come si evince dalla tabella, nei registri **Counts per revolution [101-102]** all'indirizzo 100-101 sono impostati i valori 00 00 hex e 08 00 hex, ossia 2048 in



notazione decimale; mentre nei registri **Total Resolution [103-104]** all'indirizzo 102-103 sono impostati i valori 00 80 hex e 00 00 hex, ossia 8388608 in notazione decimale. Pertanto l'encoder sarà programmato per avere una risoluzione monogiro di 2048 cpr e 4096 giri (= 8388608 / 2048).

La ADU MODBUS/TCP necessaria per la richiesta di scrittura del valore 2048 dec nei registri **Counts per revolution [101-102]** (indirizzo 100-101) e del valore 8388608 dec nei registri **Total Resolution [103-104]** (indirizzo 102-103) è la seguente:

**MBAP Header + Request PDU** (in notazione esadecimale)

[00][01][00][00][00][0F][00][10][00][64][00][04][08][00][00][08][00][00][80][00][00]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][0F] = Dimensione

[00] = Identificatore di unità

[10] = codice funzione **16 Write Multiple Register**

[00][64] = indirizzo iniziale (registri **Counts per revolution [101-102]**, indirizzo 100-101)

[00][04] = numero di registri richiesti

[08] = numero di byte (2 byte per ciascun registro)

[00][00] = valore da impostare nel registro 101, 00 00 hex

[08][00] = valore da impostare nel registro 102, 08 00 hex (00 00 08 00 hex = 2048 dec)

[00][80] = valore da impostare nel registro 103, 00 80 hex

[00][00] = valore da impostare nel registro 104, 00 00 hex (00 80 00 00 hex = 8388608 dec)

La ADU MODBUS/TCP necessaria per restituire una risposta a seguito di una richiesta di scrittura del valore 2048 nei registri **Counts per revolution [101-102]** (indirizzo 100-101) e del valore 8388608 nei registri **Total Resolution [103-104]** (indirizzo 102-103) è la seguente:

**MBAP Header + Response PDU** (in notazione esadecimale)

[00][01][00][00][00][06][00][10][00][64][00][04]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][06] = Dimensione

[00] = Identificatore di unità

[10] = codice funzione **16 Write Multiple Register**

[00][64] = indirizzo iniziale (registri **Counts per revolution [101-102]**, indirizzo 100-101)

[00][04] = numero di registri scritti



Esempio di richiesta di scrittura dei registri **Operating parameters [109-110]** (indirizzo 108-109): si vogliono impostare la funzione di scaling (bit 0 **Scaling function** = 1) e il conteggio crescente con rotazione oraria dell'albero dell'encoder (bit 1 **Code sequence** = 0).

Request		Response	
Nome campo	(Hex)	Nome campo	(Hex)
Funzione	10	Funzione	10
Indirizzo iniziale Hi	00	Indirizzo iniziale Hi	00
Indirizzo iniziale Lo	6C	Indirizzo iniziale Lo	6C
Numero di registri Hi	00	Numero di registri Hi	00
Numero di registri Lo	02	Numero di registri Lo	02
Numero byte	04		
Registro 109 valore Hi	00		
Registro 109 valore Lo	00		
Registro 110 valore Hi	00		
Registro 110 valore Lo	01		

Come si evince dalla tabella, nei registri **Operating parameters [109-110]** (indirizzo 108-109) è impostato il valore 00 00 00 01 hex, ossia 0000 0000 0000 0000 0000 0000 0000 0001 in notazione binaria: il bit 0 **Scaling function** = 1; il bit 1 **Code sequence** = 0; gli altri bit non sono utilizzati, pertanto il loro valore è 0.

La ADU MODBUS/TCP necessaria per la richiesta di impostazione della funzione di scaling (bit 0 **Scaling function** = 1) e del conteggio crescente con rotazione oraria dell'albero dell'encoder (bit 1 **Code sequence** = 0) nei registri **Operating parameters [109-110]** (indirizzo 108-109) è la seguente:

**MBAP Header + Request PDU** (in notazione esadecimale)

[00][01][00][00][00][0B][00][10][00][6C][00][02][04][00][00][00][01]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][0B] = Dimensione

[00] = Identificatore di unità

[10] = codice funzione **16 Write Multiple Register**

[00][6C] = indirizzo iniziale (registri **Operating parameters [109-110]**, indirizzo 108-109)

[00][02] = numero di registri richiesti

[04] = numero di byte (2 byte per ciascun registro)  
 [00][00] = valore da impostare nel registro 109, 00 00 hex  
 [00][01] = valore da impostare nel registro 110, 00 01 hex

La ADU MODBUS/TCP necessaria per restituire una risposta a seguito della richiesta di impostazione della funzione di scaling (bit 0 **Scaling function** = 1) e del conteggio crescente con rotazione oraria dell'albero dell'encoder (bit 1 **Code sequence** = 0) nei registri **Operating parameters [109-110]** (indirizzo 108-109) è la seguente:

**MBAP Header + Response PDU** (in notazione esadecimale)

[00][01][00][00][00][06][00][10][00][6C][00][02]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][06] = Dimensione

[00] = Identificatore di unità

[10] = codice funzione **16 Write Multiple Register**

[00][6C] = indirizzo iniziale (registri **Operating parameters [109-110]**, indirizzo 108-109)

[00][02] = numero di registri scritti



### ATTENZIONE

Per motivi di sicurezza, durante il funzionamento dell'encoder si deve sempre prevedere un continuo scambio di dati tra Master e Slave per monitorare l'effettiva attività di comunicazione; questo si rende necessario per evitare situazioni di pericolo nel caso in cui insorgessero guasti nella rete di comunicazione.

A questo scopo il sistema prevede l'implementazione e la possibilità di abilitazione della funzione di Watchdog. La funzione di Watchdog consiste in un timer di sicurezza che, grazie a un timeout, permette di rilevare condizioni di loop o di deadlock. Per esempio, nel caso in cui si interrompesse la comunicazione mentre è attivo un comando, il sistema di sicurezza Watchdog interverrebbe immediatamente comandando l'attivazione di un allarme.

Il registro **Watchdog timeout [82]** è usato per disabilitare/abilitare la funzione di Watchdog oltreché per impostare il valore del timeout espresso in millisecondi. L'impostazione del registro a 0 procura la disabilitazione della funzione di Watchdog. Qualsiasi valore maggiore di 0 abilita la funzione di Watchdog e imposta il timeout del Watchdog. Quando la funzione di Watchdog è abilitata, se il dispositivo non riceve un messaggio dal Server entro il tempo impostato, il sistema forza l'uscita dell'encoder dalla rete e contemporaneamente l'attivazione dello stato **EXCEPTION**. Il LED di errore NS Stato Rete inizia a lampeggiare rosso (si veda a pagina 30).

### 6.6 Stati encoder

La tabella in basso descrive gli stati che l'encoder può assumere durante il funzionamento nella rete MODBUS/TCP.

Stato encoder	Descrizione
<b>WAIT_PROCESS</b>	In attesa di richiesta MODBUS. Non appena riceve una richiesta MODBUS l'encoder passa allo stato <b>PROCESS_ACTIVE</b> .
<b>ERROR</b>	E' stato rilevato un conflitto di indirizzi IP nella rete MODBUS. Il LED di errore NS Stato Rete si accende rosso (si veda a pagina 30).
<b>PROCESS_ACTIVE</b>	Se non riceve richieste entro il tempo preimpostato, l'encoder passa allo stato <b>WAIT_PROCESS</b> .
<b>EXCEPTION</b>	Si è verificato un timeout del Watchdog, qualsiasi richiesta MODBUS sarà ignorata. Il LED di errore NS Stato Rete inizia a lampeggiare rosso (si veda a pagina 30).

## 7 Parametri di programmazione

### 7.1 Parametri disponibili

Di seguito sono riportati i parametri disponibili per gli encoder MODBUS, per ognuno è indicato:

#### Nome parametro [Numero registro]

[indirizzo registro, tipo variabile, attributo]

- Il numero e l'indirizzo del registro sono espressi in notazione decimale.
- Attributo:
  - ro = variabile accessibile in sola lettura
  - rw = variabile accessibile in lettura e scrittura

I registri MODBUS hanno una dimensione di 16 bit; tutti i parametri encoder invece -fatta eccezione per il parametro **Watchdog timeout [82]**- si compongono di 2 registri, ossia hanno una lunghezza di 32 bit (indipendentemente dalla loro grandezza, che sia pari a 32 o a 16 bit).

word	MSW			LSW		
bit	31	...	16	15	...	0
	msb		lsb	msb		lsb

#### 7.1.1 Parametri Holding Register

Gli **Holding register** (parametri **Dati Macchina**) sono registri con riferimento 4X, accessibili sia in scrittura che in lettura; per la lettura del valore impostato nel parametro utilizzare il codice funzione **03 Read Holding Register** (lettura di registri multipli); per la scrittura di un valore nel parametro utilizzare il codice funzione **06 Write Single Register** (scrittura di un registro singolo) oppure il codice funzione **16 Write Multiple Register** (scrittura di registri multipli); per ogni ulteriore informazione sui codici funzione implementati riferirsi alla sezione "6.5.1 Codici funzione implementati" a pagina 41.



#### NOTA

Salvare sempre i nuovi valori dopo l'impostazione in modo da conservarli nella memoria non volatile in maniera permanente. Usare la funzione **Save parameters** disponibile nei registri **Control Word [111-112]**, si veda a pagina 62.

Nel caso in cui si togliesse l'alimentazione, tutti i dati non salvati in precedenza sarebbero persi!

**Watchdog timeout [82]**

[81, Unsigned16, rw]

Per motivi di sicurezza, durante il funzionamento dell'encoder si deve sempre prevedere un continuo scambio di dati tra Master e Slave per monitorare l'effettiva attività di comunicazione; questo si rende necessario per evitare situazioni di pericolo nel caso in cui insorgessero guasti nella rete di comunicazione.

A questo scopo il sistema prevede l'implementazione e la possibilità di abilitazione della funzione di Watchdog. La funzione di Watchdog consiste in un timer di sicurezza che, grazie a un timeout, permette di rilevare condizioni di loop o di deadlock. Per esempio, nel caso in cui si interrompesse la comunicazione mentre è attivo un comando, il sistema di sicurezza Watchdog interverrebbe immediatamente comandando l'attivazione di un allarme.

Il registro **Watchdog timeout [82]** è usato per disabilitare/abilitare la funzione di Watchdog oltreché per impostare il valore del timeout espresso in millisecondi. L'impostazione del registro a 0 procura la disabilitazione della funzione di Watchdog. Qualsiasi valore maggiore di 0 abilita la funzione di Watchdog e imposta il timeout del Watchdog. Quando la funzione di Watchdog è abilitata, se il dispositivo non riceve un messaggio dal Server entro il tempo impostato, il sistema forza l'uscita dell'encoder dalla rete e contemporaneamente l'attivazione dello stato **EXCEPTION**. Il LED di errore NS Stato Rete inizia a lampeggiare rosso (si veda a pagina 30).

Default = 0 (min. = 0, max. = 65535)

**NOTA**

Non appena la funzione di Watchdog è abilitata (**Watchdog timeout [82]** > 0), per attivare il timeout occorre inviare almeno un comando MODBUS.

**NOTA**

Il valore di **Watchdog timeout [82]** è immediatamente salvato in memoria non appena impostato; pertanto non è necessario memorizzarlo mediante la funzione **Save parameters**.

**Current position [95-96]**

[94-95, Unsigned16, ro]

I registri di input **Current position [1-2]** sono disponibili anche come holding register all'indirizzo 94-95 e accessibili utilizzando il codice funzione **03 Read Holding Register**. Per ogni informazione riferirsi a pagina 70.

**Speed value [97-98]**

[96-97, Signed16, ro]

I registri di input **Speed value [3-4]** sono disponibili anche come holding register all'indirizzo 96-97 e accessibili utilizzando il codice funzione **03 Read Holding Register**. Per ogni informazione riferirsi a pagina 70.

**Status word [99-100]**

[98-99, Unsigned16, ro]

I registri di input **Status word [5-6]** sono disponibili anche come holding register all'indirizzo 98-99 e accessibili utilizzando il codice funzione **03 Read Holding Register**. Per ogni informazione riferirsi alla pagina 70.

**Counts per revolution [101-102]**

[100-101, Unsigned32, rw]



**ATTENZIONE**

Questi registri sono attivi solamente se il bit 0 **Scaling function** dei registri **Operating parameters [109-110]** è impostato a "1"; diversamente sono ignorati e il sistema utilizza i valori fisici (si vedano i registri **Singleturn resolution [113-114]** e **Number of revolutions [115-116]**) per calcolare il valore di informazione.

Questi registri impostano il numero di informazioni per giro desiderate, ossia il valore richiesto per la risoluzione monogiro.

E' possibile impostare qualsiasi valore intero minore o uguale al **numero massimo di informazioni per giro fisiche** (si vedano i registri **Singleturn resolution [113-114]**).

Se si imposta un valore fuori range (per esempio maggiore del numero massimo di informazioni per giro fisiche), il valore non è accettato.

Per evitare errori di conteggio, assicurarsi che:

$$\frac{\text{Singleturn resolution [113-114]}}{\text{Counts per revolution [101-102]}} = \text{valore intero.}$$

Default = 8.192 (min. = 1, max. = 8.192)	per la serie EM58
262.144 (min. = 1, max. = 262.144)	per la serie HS58
65.536 (min. = 1, max. = 65.536)	per la serie HM58

**ATTENZIONE**

Per evitare errori di conteggio accertarsi sempre di assicurare la seguente condizione:

$$\frac{\text{Total Resolution [103-104]}}{\text{Counts per revolution [101-102]}} = \text{potenza di 2.}$$

Inoltre, dopo l'impostazione di un nuovo valore nei registri **Counts per revolution [101-102]**, accertarsi di assicurare la seguente condizione:

$$\frac{\text{Total Resolution [103-104]}}{\text{Counts per revolution [101-102]}} \leq \text{numero di giri fisici (si veda Number of revolutions [115-116])}$$

Immaginiamo per esempio che l'encoder HM58-16-14-MT2-... sia programmato come segue:

**Counts per revolution [101-102]**: 8.192

**Total Resolution [103-104]** = 33.554.432 = 8.192 (cpr) \* 4.096 (giri)

Impostiamo ora una nuova risoluzione monogiro, per esempio: **Counts per revolution [101-102]** = 360.

Se non modifichiamo contestualmente anche il valore della risoluzione totale **Total Resolution [103-104]**, risulterà che:

$$\text{Numero di giri} = \frac{33.554.432 \text{ (Total Resolution [103-104])}}{360 \text{ (Counts per revolution [101-102])}} = 93.206,755\dots$$

Come si evince, sarebbero richiesti all'encoder più di 93.000 giri, il che non può essere dato che il numero massimo di giri fisici è, come detto, 16.384 (si vedano i registri **Singleturn resolution [113-114]**). Quando questo accade, il valore non è accettato.

**ATTENZIONE**

Se si è impostato il preset, ogniqualvolta si modifica il valore dei registri **Counts per revolution [101-102]**, occorre poi verificare il valore dei registri **Preset value [105-106]**; se necessario bisognerà impostare un nuovo preset ed eseguire la sua attivazione (bit 11 **Perform counting preset** nei registri **Control Word [111-112]** = 1).



**Total Resolution [103-104]**

[102-103, Unsigned32, rw]



**ATTENZIONE**

Questi registri sono attivi solamente se il bit 0 **Scaling function** dei registri **Operating parameters [109-110]** è impostato a "=1"; diversamente sono ignorati e il sistema utilizza i valori fisici (si vedano i registri **Singleturn resolution [113-114]** e **Number of revolutions [115-116]**) per calcolare il valore di posizione.

Questi registri impostano il numero di informazioni desiderate per l'intero range di misura (risoluzione totale dell'encoder). La risoluzione totale dell'encoder risulta dal prodotto del valore di **Counts per revolution [101-102]** per il **numero di giri richiesti**.

E' possibile impostare qualsiasi valore intero minore o uguale alla **risoluzione fisica totale** (**Singleturn resolution [113-114]** \* **Number of revolutions [115-116]**).

Se si imposta un valore fuori range (per esempio maggiore della risoluzione fisica totale), il valore non è accettato.

Default = 134.217.728 (min. = 1, max. = 134.217.728) per la serie EM58  
 262.144 (min. = 1, max. = 262.144) per la serie HS58  
 1.073.741.824 (min. = 1, max. = 1.073.741.824) per la serie HM58



**ATTENZIONE**

Per evitare errori di conteggio accertarsi sempre di assicurare la seguente condizione:

$$\frac{\text{Total Resolution [103-104]}}{\text{Counts per revolution [101-102]}} = \text{potenza di 2.}$$

Inoltre, dopo l'impostazione di un nuovo valore nei registri **Total Resolution [103-104]**, controllare sempre anche i registri **Counts per revolution [101-102]** e accertarsi di assicurare la seguente condizione:

$$\frac{\text{Total Resolution [103-104]}}{\text{Counts per revolution [101-102]}} \leq \text{numero di giri fisici (si veda Number of revolutions [115-116])}$$

Immaginiamo per esempio che l'encoder HM58-16-14-MT2-... sia programmato come segue:

**Counts per revolution [101-102]:** 8.192

**Total Resolution [103-104]** = 33.554.432 = 8.192 (cpr) \* 4.096 (giri)

Impostiamo ora una nuova risoluzione totale, per esempio: **Total Resolution [103-104]** = 360.

Dato che il valore di **Total Resolution [103-104]** deve essere maggiore o uguale a quello di **Counts per revolution [101-102]**, l'impostazione sopra non è ammessa. Quando questo accade, il valore non è accettato.



#### ATTENZIONE

se si è impostato il preset, ogniqualvolta si modifica il valore dei registri **Total Resolution [103-104]**, occorre poi verificare il valore dei registri **Preset value [105-106]**; se necessario bisognerà impostare un nuovo preset ed eseguire la sua attivazione (bit 11 **Perform counting preset** dei registri **Control Word [111-112]** = 1).



#### ESEMPIO

Vogliamo installare l'encoder multigiro HM58-16-14-MT2-... .

La sua risoluzione fisica è la seguente (si veda il codice di ordinazione):

- Conteggi fisici per giro: **Singleturn resolution [113-114]** = 65.536 ( $2^{16}$ )
- Numero di giri fisici: **Number of revolutions [115-116]** = 16.384 ( $2^{14}$ )
- Risoluzione totale fisica: **Singleturn resolution [113-114]** \* **Number of revolutions [115-116]** = 1.073.741.824 ( $2^{30}$ )

Nella specifica installazione sono richiesti 2.048 conteggi/giro \* 1.024 giri:

- Abilitare la funzione di scaling: **Operating parameters [109-110]**, bit 0 **Scaling function** = "1"
- Conteggi per giro: **Counts per revolution [101-102]** = 2.048 (0000 0800 hex)
- Risoluzione totale: **Total Resolution [103-104]** = 2.048 \* 1.024 = 2.097.152 (0020 0000 hex)



#### NOTA

Consigliamo di impostare nei registri **Counts per revolution [101-102]** e **Total Resolution [103-104]** valori che siano una potenza di 2 ( $2^n$ : 2, 4, ..., 2048, 4096, 8192,...) per evitare errori di conteggio.



#### ATTENZIONE

Se si modificano i valori di **Counts per revolution [101-102]** e/o **Total Resolution [103-104]**, occorre aggiornare il valore di **Preset value [105-106]** secondo la nuova risoluzione. E' altresì richiesta una nuova operazione di preset.

**Preset value [105-106]**

[104-105, Unsigned32, rw]

Questi registri permettono di impostare la posizione dell'encoder a un valore di Preset. Con la funzione di Preset è possibile assegnare un valore desiderato a una posizione fisica dell'albero dell'encoder. La posizione fisica scelta assumerà perciò il valore impostato in questi registri e tutte le posizioni precedenti e successive assumeranno un valore conseguente. Questa funzione si rivela utile, per esempio, per far sì che lo zero dell'encoder e lo zero dell'applicazione coincidano. Il valore di preset sarà assegnato alla posizione dell'encoder al momento dell'invio del comando **Perform counting preset** disponibile nei registri **Control Word [111-112]**. Consigliamo di attivare il preset quando l'encoder è fermo.

Default = 0 (min. = 0, max. = 134.217.727 *)	per la serie EM58
0 (min. = 0, max. = 262.143 *)	per la serie HS58
0 (min. = 0, max. = 1.073.741.823 *)	per la serie HM58

\* Si veda la nota in basso.

**ESEMPIO**

Forniamo qui di seguito un esempio per comprendere meglio la funzione di preset e il significato dei registri e dei comandi correlati: **Preset value [105-106]**, **Offset value [127-128]** e **Perform counting preset**.

La posizione encoder trasmessa è il risultato del seguente calcolo:

**Quota trasmessa = posizione letta** (sia essa fisica o "scalata") + **Preset value [105-106]** - **Offset value [127-128]**.

Se non si è mai impostato il **Preset value [105-106]** e comunque eseguito un comando di attivazione del preset (comando **Perform counting preset** in **Control Word [111-112]**), quota trasmessa e posizione letta coincidono in quanto **Preset value [105-106]** = 0 e **Offset value [127-128]** = 0.

Quando si imposta un **Preset value [105-106]** e si esegue poi il comando **Perform counting preset**, il sistema memorizza l'attuale posizione dell'encoder nei registri **Offset value [127-128]**. Ne risulta che quota trasmessa e **Preset value [105-106]** coincidono in quanto **posizione letta - Offset value [127-128]** = 0; in altri termini, a una desiderata posizione dell'encoder corrisponde il valore impostato nei registri **Preset value [105-106]**.

Per esempio, supponiamo di impostare il valore "50" nei registri **Preset value [105-106]** e di eseguire il comando **Perform counting preset** in corrispondenza della posizione encoder "1000". Vogliamo cioè che alla posizione "1000" sia restituita la quota "50".

Risulta perciò che:

**Quota trasmessa = posizione letta** (= "1000") + **Preset value [105-106]** (= "50") - **Offset value [127-128]** (= "1000") = 50.

La successiva quota trasmessa sarà poi:

**Quota trasmessa = posizione letta** (= "1001") + **Preset value [105-106]** (= "50") - **Offset value [127-128]** (= "1000") = 51.

E così via.



**NOTA**

- Se **Scaling function** è disabilitato (il bit 0 dei registri **Operating parameters [109-110]** = 0), il **Preset value [105-106]** deve essere minore o uguale alla "Risoluzione totale fisica" - 1, ossia (**Singleturn resolution [113-114]** \* **Number of revolutions [115-116]**) - 1.
- Se **Scaling function** è abilitato (il bit 0 dei registri **Operating parameters [109-110]** = 1), il **Preset value [105-106]** deve essere minore o uguale a **Total Resolution [103-104]** - 1.



**ATTENZIONE**

Verificare il valore nei registri **Preset value [105-106]** e se necessario eseguire l'attivazione del preset (bit 11 **Perform counting preset** dei registri **Control Word [111-112]** = 1) ogniqualvolta si imposta un nuovo **Code sequence** o si abilita la **Scaling function** o si modificano i valori scalati (registri **Counts per revolution [101-102]** e / o **Total Resolution [103-104]**).

**Speed format [107-108]**

[106-107, Unsigned16, rw]

Questi registri impostano l'unità di misura per il valore di velocità (si vedano i registri **Speed value [3-4]** a pagina 70).

0 = steps/s: numero di conteggi al secondo;

1 = rpm: giri al minuto.

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

**Operating parameters [109-110]**

[108-109, Unsigned16, rw]

Bit	Funzione	bit = 0	bit = 1
0	<b>Scaling function</b>	disabilitato	abilitato
1	<b>Code sequence</b>	Rotazione oraria	Rotazione antioraria
2 ... 31	non usati		

I valori di default sono evidenziati in grassetto

Default = 0000 0000 hex (min. = 0000 0000 hex, max. = 0000 0003 hex)

**Byte 0**

**Scaling function**

bit 0

Se questa funzione è disabilitata (il bit 0 **Scaling function** = 0), il dispositivo utilizza la risoluzione fisica per fornire il

valore della posizione assoluta (si vedano i registri **Singleturn resolution [113-114]** e **Number of revolutions [115-116]**); se questa funzione è abilitata (il bit 0 **Scaling function** = 1), il dispositivo utilizza la risoluzione specifica impostata nei registri **Counts per revolution [101-102]** e **Total Resolution [103-104]** nel rispetto della seguente relazione:

Posizione trasmessa =

$$\frac{\text{Counts per revolution [101-102]}}{\text{Singleturn resolution [113-114]}} * \text{Posizione reale} \leq \text{Total Resolution [103-104]}$$



**NOTA**

Per sapere se la **Scaling function** è correntemente abilitata, è possibile leggere il bit 0 **Scaling function** nei registri di input **Status word [5-6]**, si veda a pagina 71.



**ATTENZIONE**

Ogniqualevolta si abilita/disabilita la funzione di scaling e/o si modificano i valori scalati (si vedano i registri **Counts per revolution [101-102]** e **Total Resolution [103-104]**), sarà poi necessario impostare un nuovo valore di preset (si vedano i registri **Preset value [105-106]**) e quindi salvare i nuovi parametri (si veda la funzione **Save parameters**).

**Code sequence**

bit 1

Imposta se il valore di posizione trasmesso dall'encoder incrementa (informazione crescente) quando l'albero dell'encoder ruota in senso orario (0 = orario) oppure antiorario (1 = antiorario). Se il bit 1 **Code sequence** = 0, il valore della posizione assoluta **incrementa** quando l'albero dell'encoder ruota **in senso orario**; al contrario, se il bit 1 **Code sequence** = 1, il valore della posizione assoluta **incrementa** quando l'albero dell'encoder ruota **in senso antiorario**. Le rotazioni oraria e antioraria sono da intendersi guardando l'encoder dal lato albero.



**ATTENZIONE**

La modifica di questo valore influisce necessariamente anche sulla posizione calcolata dal controllore. Dopo la sua impostazione è pertanto obbligatorio eseguire un nuovo preset (si vedano i registri **Preset value [105-106]**) e salvare i parametri.



**NOTA**

Per sapere se **Code sequence** è correntemente abilitato, è possibile leggere il bit 1 **Code sequence** nei registri di input **Status word [5-6]**, si veda a pagina 71.

bit 2 ... 7 Non utilizzati.

Byte 1 ... 3 Non utilizzati.

### Control Word [111-112]

[110-111, Unsigned16, rw]

Questa variabile contiene i comandi da inviare in tempo reale allo Slave per controllarlo.

Bit	Funzione	bit = 0	bit = 1
0 ... 8	non usati		
9	Save parameters		
10	Restore default parameters		
11	Perform counting preset		
12 ... 31	non usati		

Byte 0 Non usato.

### Byte 1

bit 8 Non usato.

### Save parameters

bit 9

Questa funzione permette il salvataggio di tutti i parametri nella memoria non volatile. I dati vengono salvati nella memoria non volatile in corrispondenza del fronte di salita del bit; in altri termini, il salvataggio dei dati è eseguito ogniqualvolta il bit commuta dallo stato logico basso ("0") allo stato logico alto ("1"). Il bit deve poi essere riportato allo stato logico basso ("0") per rendere nuovamente disponibile la funzione.



#### NOTA

Salvare sempre i nuovi valori dopo l'impostazione in modo da conservarli nella memoria non volatile in maniera permanente.

Nel caso in cui si togliesse l'alimentazione, tutti i dati non salvati in precedenza sarebbero persi!

### Restore default parameters

bit 10

Questa funzione permette all'operatore il ripristino di tutti i parametri ai valori di default (i valori di default sono

impostati durante la messa a punto in azienda del dispositivo dai tecnici di Lika Electronic e ne permettono il funzionamento in modalità generica e sicura). Questa funzione può essere utile, per esempio, per ripristinare i valori di fabbrica nel caso in cui l'encoder sia stato programmato in maniera non corretta e non si riesca a riportarlo a un funzionamento regolare.

I parametri di default sono ripristinati in corrispondenza del fronte di salita del bit; in altri termini, l'operazione di caricamento dei parametri di default è eseguita ogniqualvolta il bit commuta dallo stato logico basso ("0") allo stato logico alto ("1"). Il bit deve poi essere riportato allo stato logico basso ("0") per rendere nuovamente disponibile la funzione. La lista completa dei dati macchina e dei relativi parametri di default preimpostati dai tecnici di Lika Electronic sono disponibili a pagina 102.



#### ATTENZIONE

L'esecuzione di questo comando procura la sovrascrittura di tutti i parametri impostati precedentemente!

#### Perform counting preset

bit 11

Questo comando è utilizzato per attivare il valore di preset nell'encoder. A seguito dell'invio di questo comando, il valore di posizione trasmesso per la corrente posizione dell'encoder assume il valore impostato nei registri **Preset value [105-106]** e tutte le posizioni precedenti e successive assumono un valore conseguente. L'operazione è eseguita in corrispondenza del fronte di salita del bit, ossia ogniqualvolta il bit commuta dallo stato logico basso ("0") allo stato logico alto ("1"). Il bit deve poi essere riportato allo stato logico basso ("0") per rendere nuovamente disponibile la funzione. Con l'invio di questo comando, la posizione corrente dell'encoder è memorizzata temporaneamente all'interno dei registri **Offset value [127-128]**. Per ogni ulteriore informazione sulla funzione di preset e sul significato e l'uso dei relativi registri e comandi **Preset value [105-106]**, **Offset value [127-128]** e **Perform counting preset** riferirsi alla pagina 59.



#### ATTENZIONE

Per memorizzare permanentemente la posizione corrente dell'encoder nei registri **Offset value [127-128]**, eseguire il comando **Save parameters**. Nel caso in cui si togliesse l'alimentazione senza preventivo salvataggio dei dati, il valore **Offset value [127-128]** non salvato sarebbe perso!

bit 12 ... 15 Non usati.

Byte 2 e 3 Non usati.



**NOTA**

Salvare sempre i nuovi valori dopo l'impostazione in modo da conservarli nella memoria non volatile in maniera permanente. Usare la funzione **Save parameters** disponibile nei registri **Control Word [111-112]**, si veda a pagina 62.

Nel caso in cui si togliesse l'alimentazione, tutti i dati non salvati in precedenza sarebbero persi!

**Singleturn resolution [113-114]**

[112-113, Unsigned32, ro]



**ATTENZIONE**

Questi registri sono attivi solamente se il bit 0 **Scaling function** dei registri **Operating parameters [109-110]** è impostato a "0"; diversamente sono ignorati e il sistema utilizza i valori specifici impostati nei registri **Counts per revolution [101-102]** e **Total Resolution [103-104]** per calcolare l'informazione della posizione.

Questi registri visualizzano il numero di informazioni fisiche che l'hardware è in grado di restituire per giro (risoluzione monogiro fisica).

Se si desidera impostare una risoluzione specifica per l'applicazione, si vedano i registri **Counts per revolution [101-102]**.

Default = 8.192	per la serie EM58
262.144	per la serie HS58
65.536	per la serie HM58

**Number of revolutions [115-116]**

[114-115, Unsigned32, ro]



**ATTENZIONE**

Questi registri sono attivi solamente se il bit 0 **Scaling function** dei registri **Operating parameters [109-110]** è impostato a "0"; diversamente sono ignorati e il sistema utilizza i valori specifici impostati nei registri **Counts per revolution [101-102]** e **Total Resolution [103-104]** per calcolare l'informazione della posizione.



Questi registri visualizzano il numero di giri fisici che l'hardware è in grado di realizzare (numero di rotazioni fisiche).

La **Risoluzione totale fisica** risulta dal prodotto del valore di **Singleturn resolution [113-114]** moltiplicato per il valore di **Number of revolutions [115-116]**.

Se si desidera impostare un numero di giri specifico per l'applicazione, si vedano i registri **Counts per revolution [101-102]** e **Total Resolution [103-104]**.

Default = 16.384	per la serie EM58
1	per la serie HS58
16.384	per la serie HM58

### Supported alarms [117-118]

[116-117, Unsigned16, ro]

Bit	Funzione	bit = 0	bit = 1
0 ... 11	non usati		
12	<b>Machine data not valid</b>	Allarme non supportato	Allarme supportato
13	<b>Setting data not valid</b>	Allarme non supportato	Allarme supportato
14	<b>Flash memory error</b>	Allarme non supportato	Allarme supportato
15 ... 31	non usati		

Questi registri contengono l'informazione sugli allarmi supportati dall'encoder. I messaggi di allarme disponibili sono descritti nei registri **Alarm registers [121-122]**.

Gli allarmi supportati sono elencati qui di seguito:

**Byte 0** Non usato.

**Byte 1**

bit 8 ... 11 Non usati.

**Machine data not valid**

bit 12

**Setting data not valid**

bit 13

**Flash memory error**

bit 14

bit 15 Non usato.

**Byte 2 e 3** Non usati.

Default = 0000 7000h (= 00000 0000 0000 0000 0111 0000 0000 0000 = gli allarmi ai bit 12, 13 e 14 dei registri **Alarm registers [121-122]** sono supportati).

### Supported warnings [119-120]

[118-119, Unsigned16, ro]

Questi registri contengono l'informazione sulle avvertenze supportate dall'encoder. Non ci sono avvertenze supportate in questo encoder.

Default = 0

### Alarm registers [121-122]

[120-121, Unsigned16, ro]

Bit	Funzione	bit = 0	bit = 1
0 ... 11	non usati		
12	<b>Machine data not valid</b>	Allarme non attivo	Allarme attivo
13	<b>Setting data not valid</b>	Allarme non attivo	Allarme attivo
14	<b>Flash memory error</b>	Allarme non attivo	Allarme attivo
15 ... 31	non usati		

Questi registri visualizzano gli allarmi correntemente attivi nel dispositivo. Viene attivato un allarme quando un malfunzionamento nell'encoder può portare a un valore di posizione non corretto. Se si verifica un allarme, il relativo bit è portato al livello logico alto (1) finché l'allarme non è cancellato e l'encoder non è in grado di restituire un valore di posizione accurato.

I messaggi di allarme disponibili sono descritti qui di seguito.

**Byte 0** Non usato.

**Byte 1**

bit 8 ... 11 Non usati.

#### **Machine data not valid**

bit 12 Uno o più parametri non sono validi, impostare valori appropriati per ripristinare le normali condizioni di lavoro. Si veda la lista dei parametri errati nei registri **Wrong parameters list [125-126]**.

**Setting data not valid**

bit 13 Questo messaggio di allarme è al momento disabilitato in questa versione firmware.

**Flash memory error**

bit 14 Errore interno memoria flash, non può essere ripristinato (checksum errata, ecc.).

bit 15 Non usato.

Byte 2 e 3 Non usati.



**NOTA**

Si badi che se l'allarme è relativo ai dati macchina non validi (si veda l'allarme **Machine data not valid** e i registri **Wrong parameters list [125-126]**), è possibile ripristinare le normali condizioni di lavoro solo dopo l'impostazione di valori corretti. L'allarme **Flash memory error** non può essere cancellato.

**Warnings register [123-124]**

[122-123, Unsigned16, ro]

Questa variabile visualizza le avvertenze correntemente attive nel dispositivo. Non ci sono avvertenze supportate in questo encoder.

**Wrong parameters list [125-126]**

[124-125, Unsigned16, ro]

L'operatore ha impostato dati non validi e ha procurato l'attivazione dell'allarme **Machine data not valid**. Questa variabile visualizza (valore del bit = ALTO) la lista dei parametri errati, secondo la seguente tabella.

Si badi che le normali condizioni di lavoro possono essere ripristinate solo dopo l'impostazione di valori corretti.

Bit	Funzione	bit = 0	bit = 1
0	<b>Counts per revolution error</b>	Allarme non attivo	Allarme attivo
1	<b>Total resolution error</b>	Allarme non attivo	Allarme attivo
2	<b>Preset value error</b>	Allarme non attivo	Allarme attivo
3	<b>Offset value error</b>	Allarme non attivo	Allarme attivo
4 ... 31	non usati		

Byte 0

**Counts per revolution error**

bit 0 Sono stati impostati valori errati nei registri **Counts per revolution [101-102]**. Impostare valori consoni per ripristinare le normali condizioni di lavoro.

**Total resolution error**

bit 1 Sono stati impostati valori errati nei registri **Total Resolution [103-104]**. Impostare valori consoni per ripristinare le normali condizioni di lavoro.

**Preset value error**

bit 2 Sono stati impostati valori errati nei registri **Preset value [105-106]**. Impostare valori consoni per ripristinare le normali condizioni di lavoro.

**Offset value error**

bit 3 Sono stati impostati valori errati nei registri **Offset value [127-128]**. Impostare valori consoni per ripristinare le normali condizioni di lavoro.

bit 4 ... 7 Non usati.

Byte 1 ... 3 Non usati.

**Offset value [127-128]**

[126-127, Signed32, ro]

All'atto dell'invio del comando **Perform counting preset** (si veda il bit 11 dei registri **Control Word [111-112]**), la posizione corrente dell'encoder viene salvata in questi registri. Il valore di offset sarà poi utilizzato nella funzione di preset per calcolare la posizione dell'encoder che dovrà essere trasmessa. Per riportare a zero il valore memorizzato in questi registri è necessario caricare i valori di default impostati in fabbrica (si veda il bit 10, comando **Restore default parameters**, dei registri **Control Word [111-112]** a pagina 62).

Per ogni ulteriore informazione sulla funzione di preset e sul significato e l'uso dei relativi registri e comandi **Preset value [105-106]**, **Offset value [127-128]** e **Perform counting preset** riferirsi a pagina 59.

Default = 0

**DSC Firmware Version [129-130]**

[128-129, Unsigned32, ro]

Questi registri visualizzano la versione firmware del DSC (Digital Signal Controller, controllore del segnale digitale).

Il significato dei 32 bit che compongono questi registri è il seguente:

Word	MS Word			LS Word		
bit	31	...	16	15	...	0
	msb		lsb	msb		Lsb
	Versione major			Versione minor		



Per esempio, il valore 0001 0001 hex in notazione esadecimale corrisponde alla rappresentazione binaria 0000 0000 0000 0001 0000 0000 0000 0001 e deve essere interpretato come: versione firmware 1.1.

**PCB Hardware Version [131-132]**

[130-131, Unsigned32, ro]

Questi registri visualizzano la versione hardware della PCB (Printed Circuit Board, scheda a circuito stampato).

Il significato dei 32 bit che compongono questi registri è il seguente:

Word	MS Word			LS Word		
bit	31	...	16	15	...	0
	msb		lsb	msb		Lsb
	Versione major			Versione minor		



Per esempio, il valore 0001 0001 hex in notazione esadecimale corrisponde alla rappresentazione binaria 0000 0000 0000 0001 0000 0000 0000 0001 e deve essere interpretato come: versione hardware 1.1.

### 7.1.2 Parametri Input Register

Gli **Input Register** sono registri con riferimento 3X e accessibili in sola lettura; per leggere i valori impostati in un registro di input utilizzare il codice funzione **04 Read Input Register** (lettura di registri di input multipli); per ogni ulteriore informazione sui codici funzione implementati riferirsi alla sezione "6.5.1 Codici funzione implementati" a pagina 41.

#### Current position [1-2]

[000-001, Unsigned32, ro]

Questi registri visualizzano la posizione corrente del dispositivo nel momento in cui viene inviata la richiesta. Il valore di output è scalato secondo i parametri di scaling impostati, si veda **Scaling function** a pagina 60. Il valore è espresso in conteggi.

I registri di input **Current position [1-2]** sono disponibili anche come holding register all'indirizzo 94-95 e accessibili utilizzando il codice funzione **03 Read Holding Register**. Per ogni informazione riferirsi alla pagina 54.

#### Speed value [3-4]

[002-003, Signed32, ro]

Questo attributo visualizza il valore della velocità di corrente rilevata dall'encoder e calcolata ogni 100 ms.

Il valore può essere espresso sia in conteggi al secondo che in giri al minuto secondo l'impostazione dei registri **Speed format [107-108]** a pagina 60.

I registri di input **Speed value [3-4]** sono disponibili anche come holding register all'indirizzo 96-97 e accessibili utilizzando il codice funzione **03 Read Holding Register**. Per ogni informazione riferirsi alla pagina 55.

#### Status word [5-6]

[004-005, Unsigned16, ro]

Questi registri contengono l'informazione sullo stato corrente del dispositivo. Gli otto bit del byte 0 mostrano i valori correntemente impostati nei registri **Operating parameters [109-110]**; mentre gli otto bit del byte 1 sono usati per segnalare se ci sono allarmi attivi. I byte 2 e 3 non sono utilizzati.

Struttura dei registri **Status word [5-6]**:

Word	MS Word			LS Word		
bit	31	...	16	15	...	0
	msb		lsb	msb		Lsb

Byte 0

Scaling function

bit 0

Indica se la funzione di scaling (si veda il bit 0 **Scaling function** dei registri **Operating parameters [109-110]**) è correntemente disabilitata o abilitata. Se il valore è "0" la funzione di scaling è disabilitata (quindi il sistema utilizza i valori fisici -**Singleturn resolution [113-114]** e **Number of revolutions [115-116]**- per calcolare l'informazione di posizione); se il valore è "1" la funzione di scaling è abilitata (quindi il sistema utilizza i valori della risoluzione specifica -**Counts per revolution [101-102]** e **Total Resolution [103-104]**- per calcolare l'informazione di posizione). Per disabilitare / abilitare la funzione di scaling occorre impostare il bit 0 **Scaling function** dei registri **Operating parameters [109-110]** a 0 / 1. Per ogni ulteriore informazione sull'impostazione e l'utilizzo della funzione di scaling riferirsi al parametro **Scaling function** a pagina 60.

Code sequence

bit 1

Indica se la direzione di conteggio (si veda il bit 1 **Code sequence** dei registri **Operating parameters [109-110]**) è correntemente impostata come oraria o antioraria. Se il bit è "0" il valore di posizione dell'encoder che viene trasmesso è impostato per incrementare (informazione di conteggio crescente) quando l'albero dell'encoder ruota in senso orario; se il bit è "1" il valore di posizione dell'encoder che viene trasmesso è impostato per incrementare quando l'albero dell'encoder ruota in senso antiorario. Per selezionare la direzione di conteggio oraria o antioraria occorre impostare il bit 1 **Code sequence** dei registri **Operating parameters [109-110]** a 0 / 1. Per ogni ulteriore informazione sull'impostazione e l'utilizzo della funzione della direzione di conteggio riferirsi al parametro **Code sequence** a pagina 61.

bit 2 ... 7

Non usati.

Byte 1

Alarm

bit 8

Se il valore è "1" si è verificato un allarme, si vedano i dettagli nella variabile **Alarm registers [121-122]** a pagina 66.

bit 9 ... 15

Non usati.

Byte 2 e 3          Non usati.

**NOTA**

Gli input register **Status word [5-6]** sono disponibili anche come holding register all'indirizzo 98-99 e accessibili utilizzando il codice funzione **03 Read Holding Register**. Per ogni informazione riferirsi a pagina 55.



## 7.2 Exception response e codici eccezione

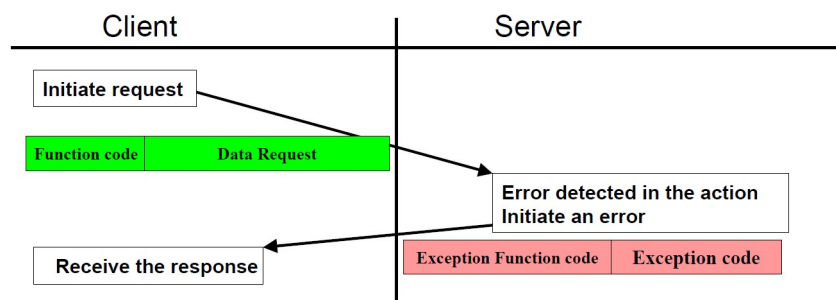
Quando un dispositivo Client invia una richiesta a un dispositivo Server, si attende poi una risposta normale. In realtà, a seguito della richiesta del Master, si può verificare uno dei quattro possibili eventi descritti qui a seguire.

- Se il dispositivo Server riceve la richiesta senza che siano presenti errori di comunicazione e di conseguenza può gestire normalmente l'interrogazione, invia di ritorno una risposta normale.
- Se il dispositivo Server non riceve la richiesta a causa di un errore di comunicazione, non sarà possibile ritornare alcuna risposta. Eventualmente il programma Client potrà gestire un timeout per la richiesta.
- Se il Server riceve la richiesta, ma rileva un errore di comunicazione, non sarà possibile ritornare alcuna risposta. Eventualmente il programma Client potrà gestire un timeout per la richiesta.
- Se il Server riceve la richiesta senza che siano presenti errori di comunicazione, ma non è in grado di processarla (per esempio, nel caso in cui si trattasse di una richiesta di lettura di un'uscita o di un registro inesistenti), il Server invierà una **exception response** (risposta d eccezione) informando il Client sulla natura dell'errore.

Il messaggio di exception response consta di due campi che lo differenziano da un messaggio normale di risposta:

**CAMPO CODICE FUNZIONE:** in una risposta normale, il Server ripete il codice funzione della richiesta originale riportandolo nel campo codice funzione della risposta. Tutti i codici funzione hanno un bit più significativo (msb, most significant bit) impostato a 0 (i loro valori sono tutti inferiori a 80 esadecimale). In una exception response, il Server imposta il bit più significativo del codice funzione a 1. Questo fa sì che il valore del codice funzione di una exception response sia maggiore esattamente di 80 esadecimale rispetto al valore che esso avrebbe assunto in una risposta normale. Con il bit più significativo del codice funzione impostato, il programma di applicazione Client può riconoscere la exception response e valutare il campo dati al fine di inviare un codice eccezione.

**CAMPO DATI:** in una risposta normale, il Server può ritornare dati o statistiche nel campo dati (qualsiasi informazione che fosse richiesta nella interrogazione). In una exception response invece, il Server scrive nel campo dati un codice eccezione. Esso specifica la condizione Server che ha procurato l'eccezione.




**NOTA**

Si badi che quello che segue è un elenco dei codici eccezione (exception code) indicati da MODBUS, ma non necessariamente supportati dal costruttore.

<b>Codici eccezione MODBUS</b>		
<b>Codice</b>	<b>Nome</b>	<b>Significato</b>
<b>01</b>	ILLEGAL FUNCTION	Il codice funzione ricevuto nell'interrogazione si configura in un'azione inammissibile per il server. Il motivo potrebbe risiedere nel fatto che il codice funzione è ammesso solo per nuovi dispositivi e non è implementato nell'unità selezionata. Potrebbe anche indicare che il server si trova in uno stato non compatibile per la processazione di una richiesta di questo tipo, per esempio perché non è configurato e riceve una richiesta di restituzione dei valori di registro.
<b>02</b>	ILLEGAL DATA ADDRESS	L'indirizzo dati ricevuto nell'interrogazione non è un indirizzo ammissibile per il server. Più specificamente, la combinazione di reference number e registri richiesti non è valida. Per un controller con 100 registri, la PDU assegna al primo indirizzo il valore 0 e all'ultimo il valore 99. Se viene fatta una richiesta dove l'indirizzo del registro iniziale è 96 e la quantità di registri richiesti è 4, la richiesta viene assolta con successo (indirizzamento corretto) operando sui registro 96, 97, 98, 99. Se invece viene inviata una richiesta dove l'indirizzo del registro iniziale è 96 e la quantità dei registri richiesti è 5, in questo caso la richiesta non potrà essere portata a termine e causerà il codice eccezione 0x02 "Illegal Data Address" dato che viene richiesto di operare sui registri 96, 97, 98, 99 e 100, ma non esiste nessun registro con indirizzo 100.
<b>03</b>	ILLEGAL DATA VALUE	Un valore contenuto nel campo dati dell'interrogazione non è ammissibile per il server. Indica un errore nella struttura del remainder di una richiesta complessa, come per esempio che la lunghezza indicata non è corretta. Non significa nello specifico che un data item inviato per la memorizzazione in un registro ha un valore fuori dal range atteso dal programma applicativo, dato che il protocollo MODBUS non tiene conto del significato di qualsivoglia valore di un qualunque registro.
<b>04</b>	SERVER DEVICE FAILURE	Si è verificato un errore non recuperabile durante il tentativo da parte del server di eseguire una

		azione richiesta.
<b>05</b>	ACKNOWLEDGE	Utilizzo specifico con comandi di programmazione. Il server ha accettato la richiesta e la sta eseguendo, ma per fare questo sarà necessario diverso tempo. Questa risposta viene restituita per evitare che si possa registrare nel Client un errore di timeout. Il Client può poi inviare un messaggio Poll Program Complete per determinare se la processazione è completata.
<b>06</b>	SERVER DEVICE BUSY	Utilizzo specifico con comandi di programmazione. Il server è impegnato nella processazione di un comando di programmazione che richiede diverso tempo. Pertanto il client è invitato a ritrasmettere il messaggio più tardi quando il server sarà disponibile.
<b>08</b>	MEMORY PARITY ERROR	Utilizzo specifico con codici funzione 20 e 21 e reference type 6, indica che l'area di file estesa non ha superato un consistency check. Il server ha tentato di leggere il file in memoria, ma ha rilevato un errore di parità nella memoria. Il client può ritentare la richiesta, ma potrebbe essere necessario un intervento sul dispositivo server.
<b>0A</b>	GATEWAY PATH UNAVAILABLE	Utilizzo specifico con gateway, indica che il gateway non è stato in grado di assegnare un percorso di comunicazione interna dalla porta d'ingresso alla porta d'uscita per la processazione della richiesta. Solitamente significa che il gateway non è configurato correttamente o è sovraccarico.
<b>0B</b>	GATEWAY TARGET DEVICE FAILED TO RESPOND	Utilizzo specifico con gateway, indica che non si è avuta risposta dl dispositivo target. Solitamente significa che il dispositivo non è presente nella rete.

Per ogni informazione sui codici eccezione (exception code) disponibili e sul loro significato riferirsi al capitolo "MODBUS Exception Responses" alla pagina 47 del documento "MODBUS Application Protocol Specification V1.1b3".

## 8 Web Server Integrato

### 8.1 Web server integrato – Informazioni preliminari

Gli encoder MODBUS/TCP di Lika Electronic integrano un web server. Questa interfaccia utente basata su una connessione di tipo web è progettata per offrire funzioni di assistenza e informazioni complete sul dispositivo al quale si può accedere tramite un connessione Internet.

In particolare permette di:

- visualizzare e monitorare i parametri correntemente impostati;
- impostare i parametri di comunicazione della rete;
- impostare alcuni parametri come il preset e la direzione di conteggio;
- aggiornare il firmware;
- monitorare l'encoder e accedere ad alcune funzioni di manutenzione avanzate.

Al web server si può accedere tramite un qualsiasi PC sul quale sia installato un browser web. Dato che il suo solo requisito è la presenza di una connessione HTTP tra il browser web e il web server attiva nel dispositivo, è una soluzione valida anche nei casi di accesso remoto.

Prima di aprire il web server dell'encoder MODBUS/TCP assicurarsi di soddisfare completamente i seguenti requisiti:

- l'encoder è collegato alla rete;
- l'encoder ha un indirizzo IP valido;
- il PC è collegato alla rete;
- nel PC o nel dispositivo utilizzato per la connessione è installato un browser web (Internet Explorer, Mozilla Firefox, Google Chrome, Opera, ...).



#### NOTA

Questo web server è stato testato e verificato utilizzando i seguenti browser web:

- Internet Explorer IE11 versione 11.1593.14393.0
- Mozilla Firefox versione 73.0.1
- Google Chrome versione 83.0.4103.97
- Opera versione 68.0.3618.165



#### NOTA

Si badi che l'aspetto delle schermate può variare a seconda del browser web usato. Gli snapshot che seguono sono stati acquisiti utilizzando Mozilla Firefox.

## 8.2 Pagina Home del Web server

Per aprire il web server dell'encoder MODBUS/TCP procedere come segue:

1. digitare l'indirizzo IP dell'encoder cui ci si vuole collegare (nell'esempio: 192.168.1.10, questo è l'indirizzo IP di default assegnato da Lika, si veda a pagina 29) nella barra dell'indirizzo del browser web e confermare premendo **ENTER**;

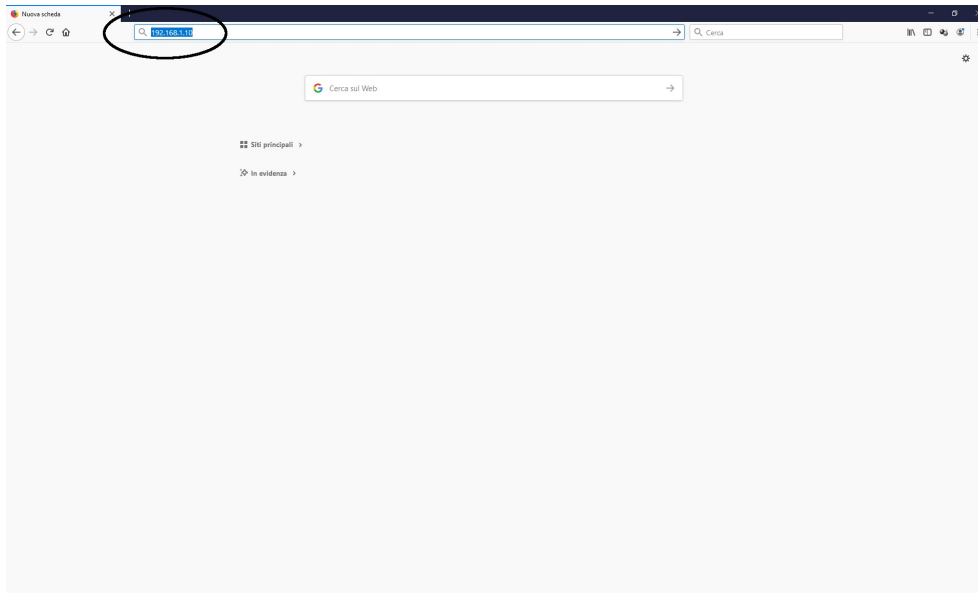


Figura 3 – Apertura del web server

2. non appena la connessione è stabilita, appare sullo schermo la pagina **Home** del web server;

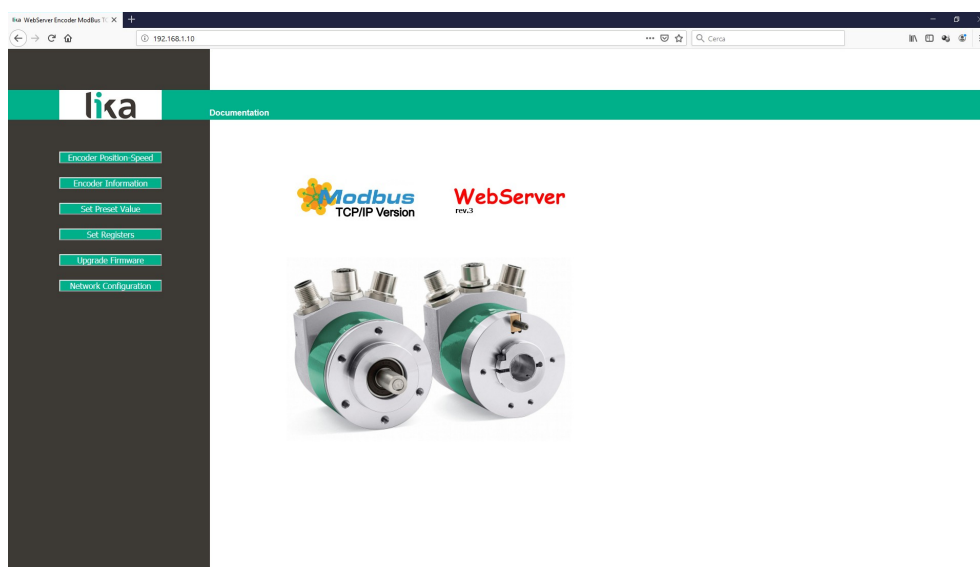


Figura 4 – Pagina Home del Web server

Sulla barra di menu della pagina **Home** sono disponibili alcuni comandi.

Premere sul **logo Lika** per accedere al sito web di Lika ([www.lika.biz](http://www.lika.biz)).

Premere il comando **Documentation** per accedere alla pagina della documentazione tecnica dell'encoder MODBUS/TCP disponibile sul sito internet di Lika (<https://www.lika.it/ita/prodotti/encoder-rotativi/assoluti/ethernet/>) dove è possibile trovare informazioni tecniche specifiche e la documentazione dell'encoder MODBUS/TCP.

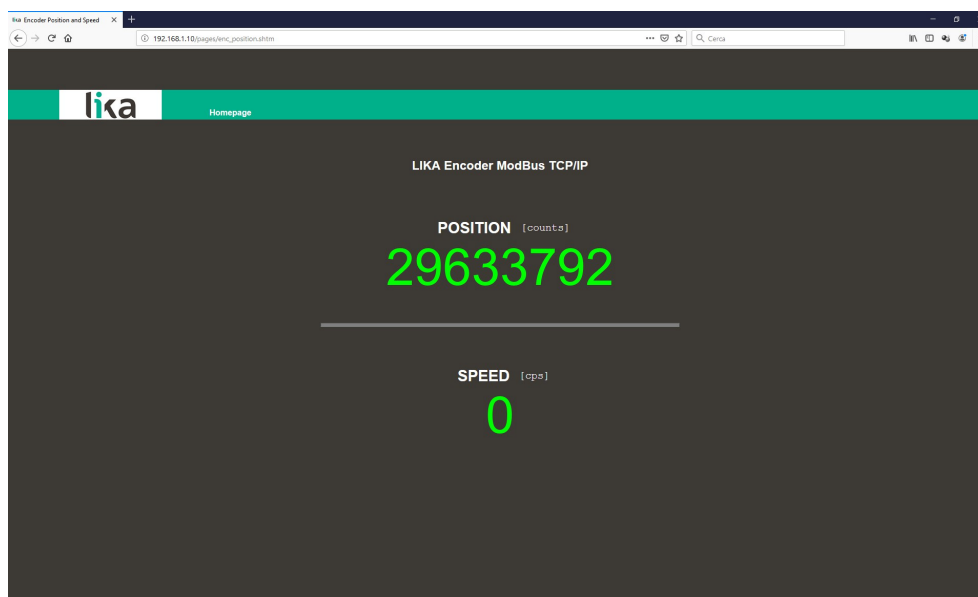
Alcuni comandi sono poi disponibili nella barra di navigazione laterale, a sinistra. Tutte le pagine, eccetto la pagina **Upgrade firmware**, sono liberamente accessibili tramite i comandi nella barra. La pagina **Upgrade firmware** è protetta e richiede l'inserimento di una password per l'accesso.

Questi comandi permettono l'accesso a pagine specifiche dove è possibile trovare informazioni di configurazione e di diagnostica sull'encoder collegato nonché funzioni utili all'utilizzatore.

Esse sono descritte nelle sezioni che seguono.

### 8.3 Posizione e velocità dell'encoder

Premere il comando **Encoder Position-Speed** nella barra di navigazione laterale a sinistra della pagina **Home** del Web server per accedere alla pagina dove sono visualizzate le informazioni sulla posizione corrente dell'encoder e sulla velocità corrente dell'encoder.



**Figura 5 - Pagina della posizione e velocità dell'encoder**

La posizione corrente dell'encoder è espressa in conteggi. Per ogni informazione riferirsi ai registri **Current position [1-2]** a pagina 70.

La velocità corrente dell'encoder è espressa nell'unità di misura impostata nei registri **Speed format [107-108]** a pagina 60 (numero di conteggi al secondo o giri al minuto). Per ogni informazione riferirsi ai registri **Speed value [3-4]** a pagina 70.



#### NOTA

I valori della posizione corrente e della velocità corrente dell'encoder sono processati in tempo reale e aggiornati in maniera continua (ogni 200 msec. sullo schermo).

Premere il comando **Homepage** per ritornare alla pagina **Home** del Web server.

### 8.3.1 Note specifiche sull'utilizzo di Internet Explorer

Al fine di ottenere un aggiornamento continuo della pagina **Encoder position and speed**, in Internet Explorer devono essere impostate opportunamente le seguenti opzioni.

- Aprire il menu **Settings**;
- aprire la scheda delle proprietà **Internet Options**;
- nella pagina a schede **General**, premere il pulsante **Setting** disponibile nella sezione **History Browsing**;
- sotto la voce **Check for newer versions of stored pages**, premere **Every time I visit the webpage**;
- ogniqualvolta richiesto, premere il pulsante **OK** per confermare.

## 8.4 Encoder information (registri MODBUS)

Premere il comando **Encoder information** nella barra di navigazione laterale a sinistra della pagina **Home** del Web server per accedere alla pagina **Encoder Information**. In questa pagina è visualizzata la lista completa dei registri MODBUS disponibili.

The screenshot shows the 'Encoder Information' page with the following data:

Input Registers			Holding Registers		
Addr.	Description	Value	Addr.	Description	Value
000	Position:	29633792	110	Control Word:	00000000h
002	Speed:	0	112	Single Turn Resolution:	8192
004	Status Word:	00000002h	114	Number of Revolutions:	16384
== ***** ==			116	Supported Alarms:	00007000h
			118	Supported Warnings:	00000000h
			120	Alarms Register:	00000000h
			122	Warning Register:	00000000h
			124	DM Errors Register:	00000000h
Holding Registers			126	Offset Value:	0
Addr.	Description	Value	128	Application-DSC FW Version:	00030000h
081	Watchdog Value (ms)	0	130	Hardware Version:	00010000h
100	Counts per Revolution:	8192	134	Encoder Serial Number:	0
102	Total Resolution:	134217728	138	Network-DSC FW Version:	00010401h
104	Preset Value:	0	140	Network-DSC Serial Number:	A0237C5Fh
106	Speed Format:	0			
108	Operating Parameters:	00000002h			

Figura 6 – Pagina Encoder Information

I registri elencati sotto la sezione **Input registers** sono i dati di processo e il loro valore è accessibile in sola lettura. Per una descrizione completa degli Input register riferirsi alla sezione "7.1.2 Parametri Input Register" a pagina 70.

I registri elencati sotto la sezione **Holding registers** sono i parametri di configurazione dell'encoder; possono essere parametri con accesso in lettura-scrittura o in sola lettura. Per una descrizione completa degli Holding register riferirsi alla sezione "7.1.1 Parametri Holding Register" a pagina 53.



### NOTA

I parametri sono costituiti da due registri a 16 bit, fatta eccezione per **Watchdog Value (ms)** che è costituito da un solo registro a 16 bit, si veda il registro **Watchdog timeout [82]** a pagina 54). Per tale ragione sotto la colonna **ADDR.** Compare solo l'indirizzo iniziale. Per leggere -per esempio- l'elemento **114 Number of Revolutions**, occorre leggere il registro all'indirizzo 114 (MSWord) e il registro all'indirizzo 115 (LSWord).



### NOTA

Si badi che i valori che appaiono nella pagina **Encoder Information** sono "congelati" al momento in cui si visualizza la pagina. Per aggiornare i valori occorre fare un refresh della pagina web.



**NOTA**

I registri nella pagina **Encoder Information** non possono essere modificati anche quando fossero registri con accesso in lettura-scrittura. Accedere alla pagina **Set Registers** per modificarli (si veda a pagina 83).

Premere il comando **Homepage** per ritornare alla pagina **Home** del Web server.

### 8.5 Impostazione del valore di Preset

Premere il comando **Set Preset Value** nella barra di navigazione laterale a sinistra della pagina **Home** del Web server per accedere alla pagina **Set Encoder Preset** e impostare/attivare un valore di Preset. Per informazioni complete sulla funzione di preset riferirsi ai registri **Preset value [105-106]** a pagina 59.

Non appena si preme il comando **Set Preset Value** sullo schermo appare un messaggio di avvertenza (**Are you sure you want to change Preset Value? / Siete sicuri di voler modificare il valore di Preset?**): avvisa l'operatore della delicatezza dell'operazione, richiedendogli pertanto di confermare la procedura prima di proseguire.

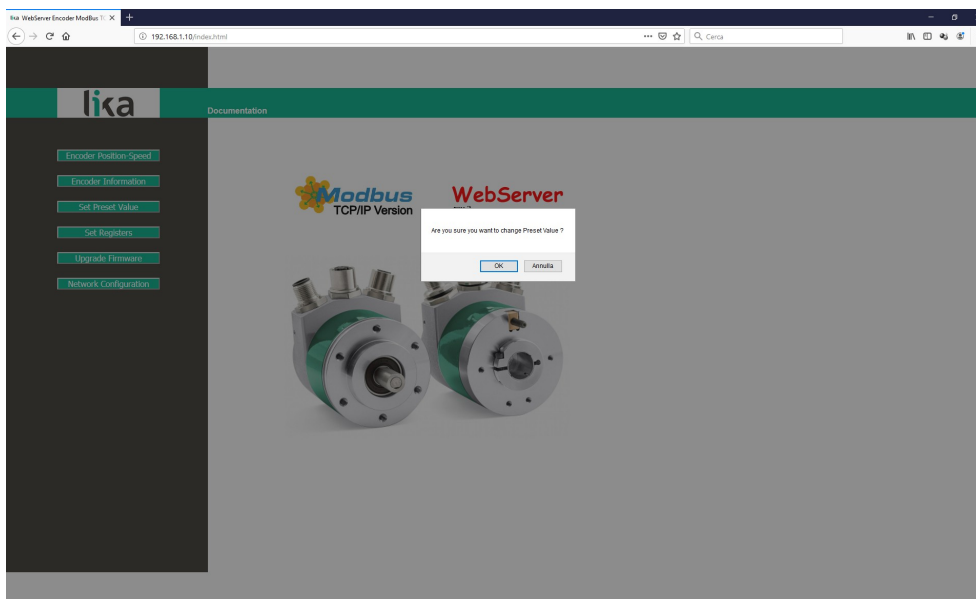


Figura 7 – Accesso alla pagina Set Encoder Preset

Premere il pulsante **OK** per procedere.

Altrimenti premere il pulsante **EXIT** per interrompere la procedura. Sullo schermo apparirà il messaggio **Set Preset cancelled! / Impostazione Preset abortita!**. Premere il pulsante **OK** per ritornare alla pagina **Home** del Web server.

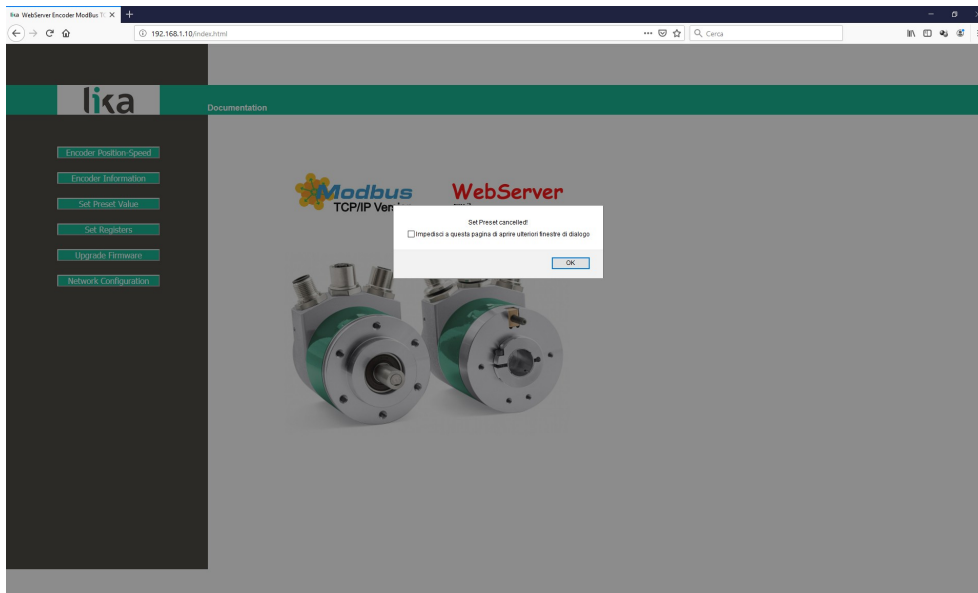


Figura 8 - Operazione di Preset abortita

Se si conferma la procedura, sullo schermo appare la pagina **Set Encoder Preset**:

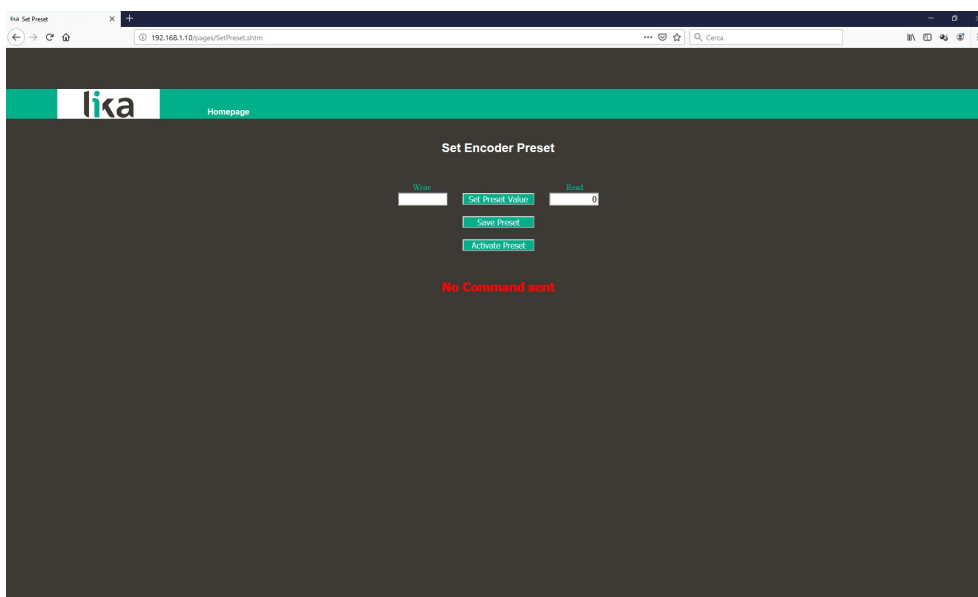


Figura 9 - Pagina Set Encoder Preset

Il valore di Preset correntemente impostato nell'encoder (si veda il registro **Preset value [105-106]** a pagina 59) sarà visualizzato nella casella **READ**.

Per modificare il Preset impostare un valore consono nella casella **WRITE** e premere poi il pulsante **Set Preset Value** per confermare. Il valore deve essere impostato in notazione decimale.

**NOTA**

Si badi che il valore di Preset è ora salvato temporaneamente nei registri **Preset value [105-106]**. Per memorizzare in maniera permanente nei registri **Preset value [105-106]** il valore di Preset impostato, premere il pulsante **Save Preset**. Qualora fosse tolta l'alimentazione all'encoder senza salvare i dati, il valore di Preset che non è stato memorizzato nella EEPROM Flash sarebbe perso! Per maggiori informazioni riferirsi al comando **Save parameters** dei registri **Control Word [111-112]** a pagina 62.

Dopo il salvataggio, occorre poi attivare il nuovo Preset (si veda il comando **Perform counting preset** dei registri **Control Word [111-112]** a pagina 63). Premere il pulsante **Activate Preset** per attivare il valore di preset. Il valore di Preset è attivato per la posizione dell'encoder nel momento in cui è premuto il pulsante **Activate Preset**. Consigliamo di attivare il valore di preset quando l'encoder è in stop.

**NOTA**

A ogni conferma e/o attivazione del valore di Preset, apparirà un messaggio in corrispondenza della riga **No Command sent / Nessun Comando inviato**. Informa sull'esito positivo dell'operazione o sull'occorrenza di un errore (per esempio **Command was set correctly / Comando impostato correttamente** se tutto è andato a buon fine; oppure **Command Error! / Errore Comando!** se qualcosa è andato storto).

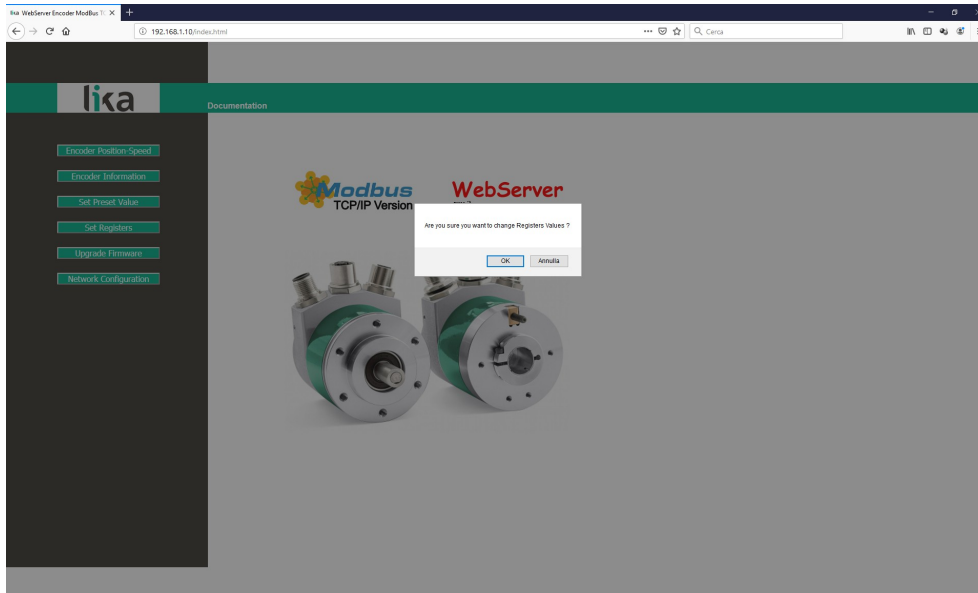
Premere il comando **Homepage** per ritornare alla pagina **Home** del Web server.

## 8.6 Impostazione dei registri

Premere il comando **Set Registers** nella barra di navigazione laterale a sinistra nella pagina **Home** del Web server per accedere alla pagina **Set Encoder Registers**. In questa pagina sono visualizzati i registri MODBUS ad accesso in lettura-scrittura; è possibile modificarne il valore.

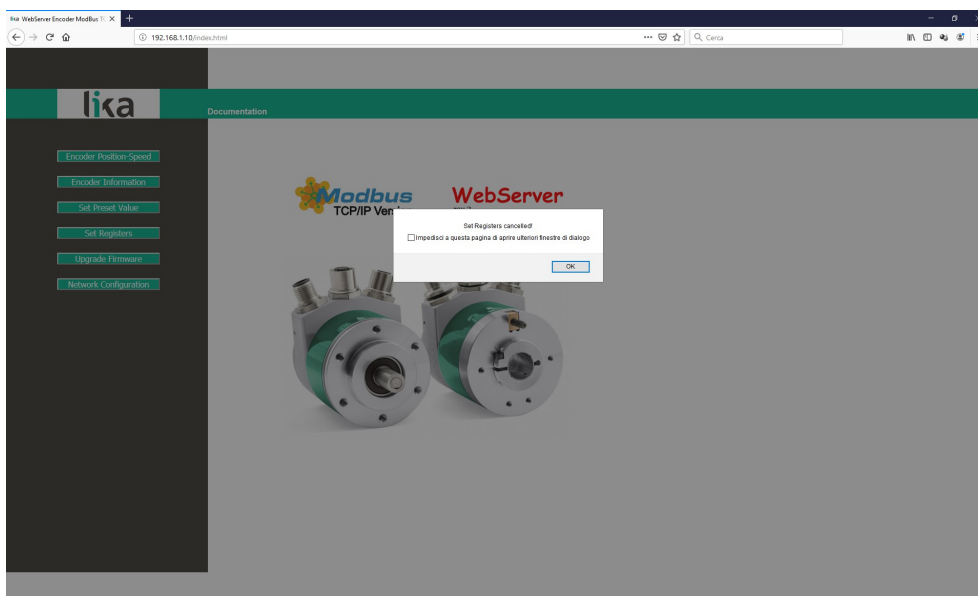
Per informazioni complete sugli holding register disponibili riferirsi alla sezione "7.1.1 Parametri Holding Register" a pagina 53.

Non appena si preme il comando **Set Registers** appare sullo schermo un messaggio di avvertimento (**Are you sure you want to change Registers Values? / Siete sicuri di voler modificare i valori dei registri?**): avvisa l'operatore della delicatezza dell'operazione, richiedendogli pertanto di confermare la procedura prima di proseguire.



**Figura 10 - Accesso alla pagina Set Encoder Registers**

Premere il pulsante **OK** per procedere, altrimenti premere il pulsante **EXIT** per interrompere la procedura. Sullo schermo apparirà il messaggio **Set Registers cancelled! / Impostazione registri abortita!**. Premere il pulsante **OK** per ritornare alla pagina **Home** del Web server.



**Figura 11 - Operazione di impostazione dei registri abortita**

Se si conferma la procedura, sullo schermo appare la pagina **Set Encoder Registers**:

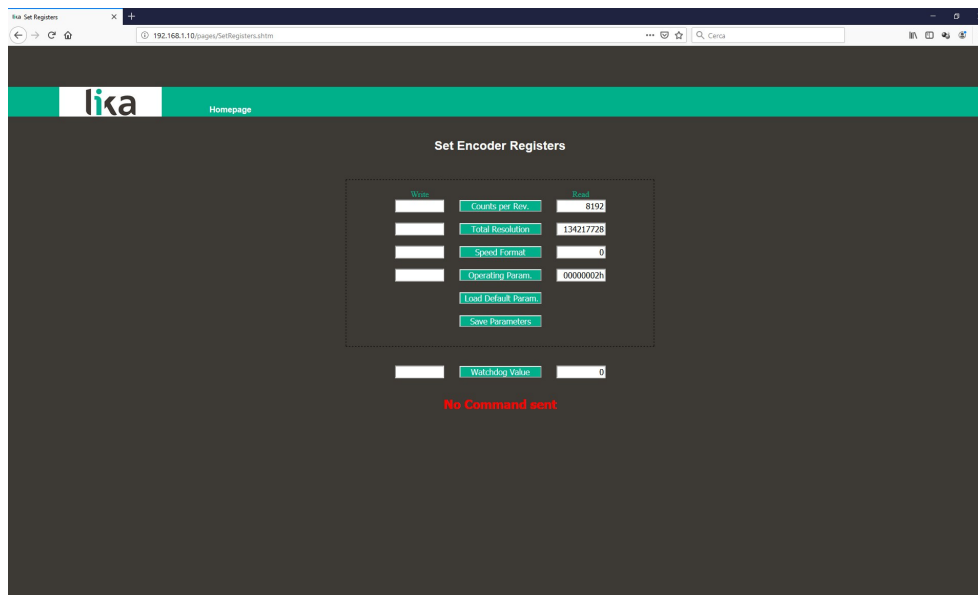


Figura 12 – Pagina Set Encoder Registers

Nella colonna **READ** vengono visualizzati i valori che sono impostati correntemente nell'encoder.

Per modificare un parametro impostare un valore adeguato nella colonna **WRITE** a fianco del parametro desiderato e quindi premere il pulsante tra le due caselle per confermare. I valori devono essere impostati in notazione decimale. Per informazioni complete sui registri disponibili riferirsi alla sezione "7.1.1 Parametri Holding Register" a pagina 53.



#### ESEMPIO

I registri **Counts per revolution [101-102]** sono attualmente impostati a "8192" (si veda la casella **READ** nella prima riga della Figura sopra). Per modificare il valore impostato inserire un valore adeguato nella corrispondente casella **WRITE** della stessa riga e premere poi il pulsante **COUNTS PER REV.** Per confermare.



#### NOTA

Si badi che, dopo la pressione del pulsante tra le caselle, il valore impostato è salvato temporaneamente nei registri. Per salvarlo in maniera permanente, premere il pulsante **Save Parameters**. Qualora fosse tolta l'alimentazione all'encoder senza salvare i dati, i valori non memorizzati nella EEPROM Flash sarebbero persi! Per maggiori informazioni riferirsi al comando **Save parameters** nei registri **Control Word [111-112]** a pagina 62.

**NOTA**

Solo il parametro **Watchdog Value** è salvato automaticamente dopo l'impostazione.

Premere il pulsante **Load Default Param.** per ripristinare tutti i parametri ai valori di default. I valori di default sono impostati in fabbrica dagli ingegneri di Lika Electronic per permettere all'operatore un funzionamento standard e sicuro del dispositivo. Questa funzione può essere utile, per esempio, per ripristinare i valori di fabbrica nel caso in cui l'encoder sia programmato in maniera non corretta e non si sia in grado di ripristinarne un funzionamento regolare. Per maggiori informazioni riferirsi al comando **Restore default parameters** dei registri **Control Word [111–112]** a pagina 62.

**ATTENZIONE**

L'esecuzione di questo comando causa la sovrascrittura di tutti i parametri impostati in precedenza!

**NOTA**

Dopo ciascuna conferma dei parametri impostati, apparirà un messaggio in corrispondenza della riga **No Command sent / Nessun Comando inviato**. Informa sull'esito positivo dell'operazione o sull'occorrenza di un errore (per esempio **Command was set correctly / Comando impostato correttamente** se tutto è andato a buon fine; oppure **Command Error! / Errore Comando!** se qualcosa è andato storto).

Premere il comando **Homepage** per ritornare alla pagina **Home** del Web server.

## 8.7 Upgrade del firmware

Premere il comando **Upgrade Firmware** nella barra di navigazione laterale a sinistra nella pagina **Home** del Web server per accedere alla pagina **Firmware Upgrade**. Si badi che la pagina è protetta da password, occorre pertanto la password per accedere alla pagina.



### ATTENZIONE

Il processo di upgrade del firmware deve essere eseguito da personale formato e competente. E' obbligatorio eseguire l'upgrade rispettando le istruzioni riportate in questa sezione.

Prima dell'installazione accertarsi sempre che il programma firmware sia compatibile con l'hardware e il software del dispositivo. Inoltre non togliere mai l'alimentazione durante l'upgrade della flash.

Questa operazione permette l'upgrade del firmware dell'unità tramite il download dei dati di upgrade alla memoria flash.

Il firmware è un programma software che controlla le funzioni e l'operatività del dispositivo; il programma firmware, talora detto anche "user program / programma utente", è memorizzato nella memoria flash integrata all'interno dell'unità. Questi encoder sono progettati in modo che il firmware possa essere aggiornato facilmente dallo stesso utilizzatore. Questo permette a Lika Electronic di approntare nuovi programmi firmware che aggiungono miglioramenti per tutta la durata della vita del prodotto.

Ragioni tipiche per il rilascio di nuovi programmi firmware sono la necessità di fare delle correzioni, ma anche migliorare e aggiungere nuove funzionalità al dispositivo.

Il programma di upgrade del firmware consiste di un singolo file con estensione .BIN. Viene rilasciato dal Servizio di Assistenza Tecnica e Post-Vendita di Lika Electronic.

Se la versione più recente del firmware è già installata sull'unità, non è necessario procedere all'installazione di alcun nuovo firmware. La versione firmware correntemente installata può essere letta nel parametro **Firmware Version** nella pagina **Encoder Information** dopo il collegamento al web server (si veda a pagina 80; si vedano anche i registri **DSC Firmware Version [129-130]** a pagina 69).



### NOTA

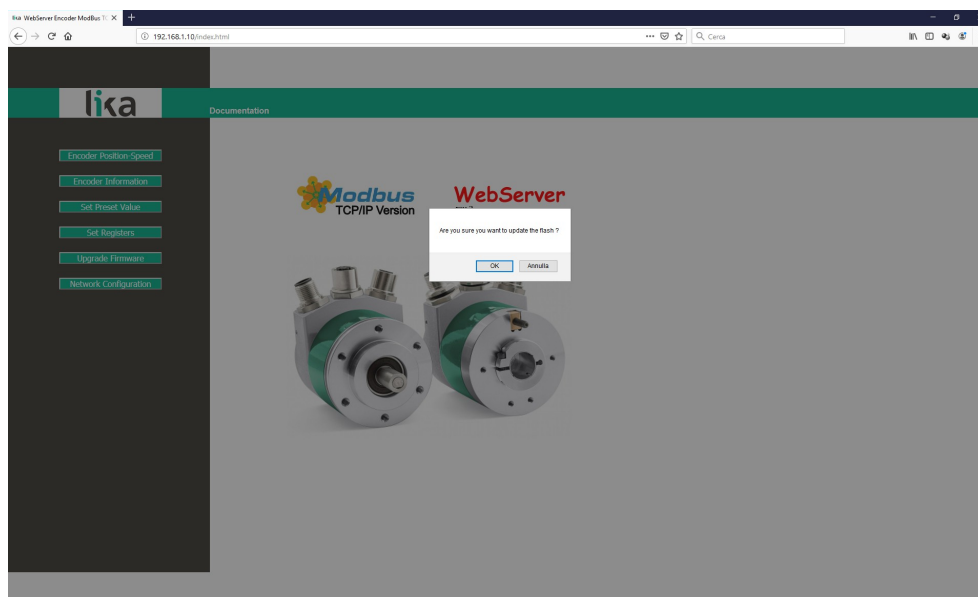
Se non si è sicuri di essere in grado di eseguire l'aggiornamento positivamente, contattare il Servizio di Assistenza Tecnica e Post-Vendita di Lika Electronic.

Prima di procedere all'aggiornamento del firmware accertarsi che i seguenti requisiti siano pienamente soddisfatti:

- l'encoder è collegato alla rete Ethernet;
- l'encoder ha indirizzo IP valido;
- il PC è collegato sia alla rete che al controllore IO;
- nel PC o nel dispositivo utilizzato per la connessione è installato un browser web (Internet Explorer, Mozilla Firefox, Google Chrome, Opera, ...);
- è a disposizione l'eseguibile SW\_ETH\_revX\_Y.exe;
- è a disposizione il file .BIN per l'upgrade del firmware.

Per fare l'upgrade del programma firmware procedere come segue.

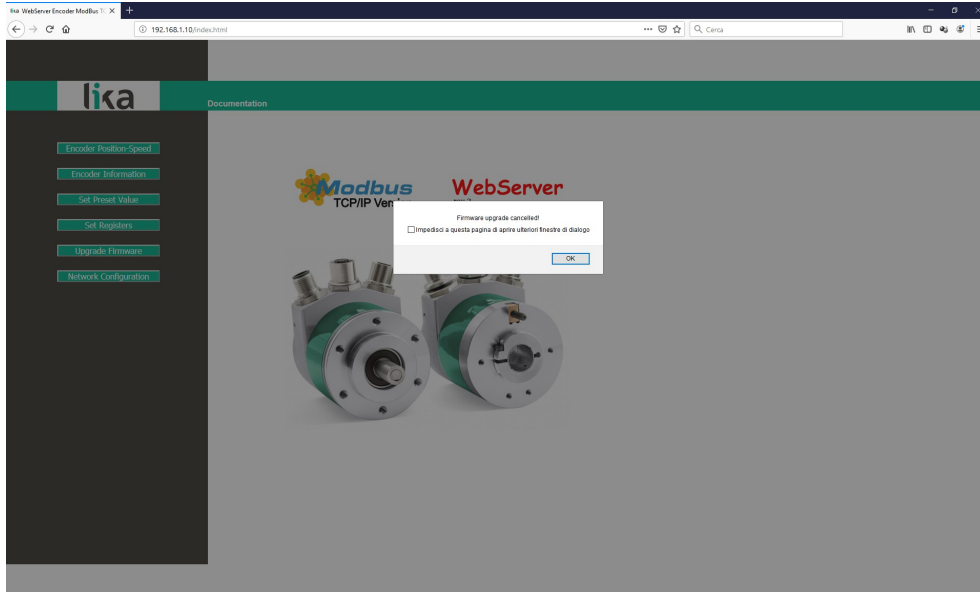
1. Premere il comando **Upgrade Firmware** nella barra di navigazione laterale a sinistra nella pagina **Home** del Web server per accedere alla pagina **Firmware Upgrade**.
2. Non appena si preme il comando **Upgrade Firmware** sullo schermo appare un messaggio di avvertimento (**Are you sure you want to update the flash? / Siete sicuri di voler aggiornare la flash?**): avvisa l'operatore della delicatezza dell'operazione, richiedendogli pertanto di confermare la procedura prima di proseguire.



**Figura 13 – Accesso alla pagina Upgrade Firmware**

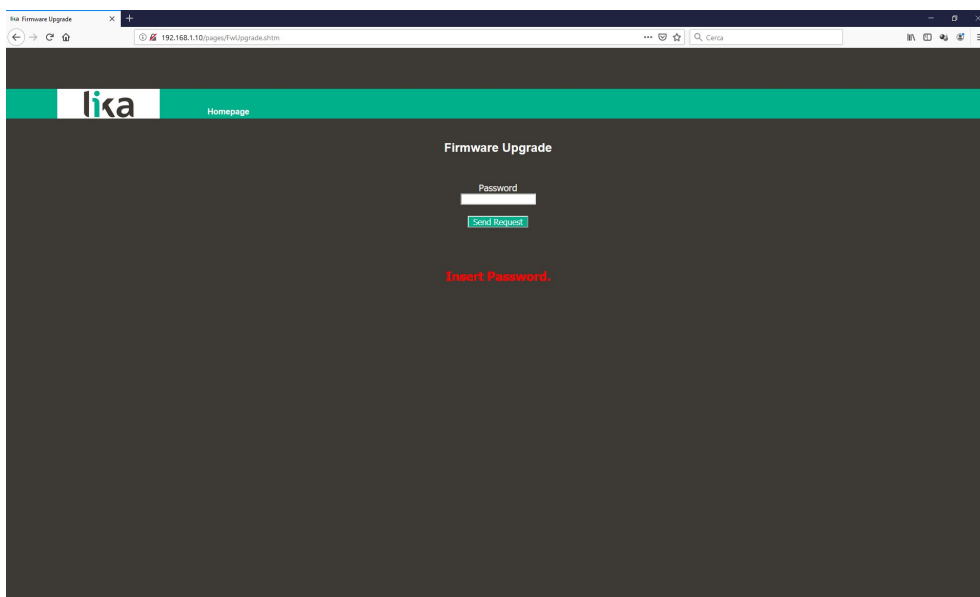


3. Premere il pulsante **OK** per procedere, altrimenti premere il pulsante **EXIT** per interrompere la procedura. Sullo schermo apparirà il messaggio **Firmware upgrade cancelled! / Aggiornamento firmware interrotto!**. Premere il pulsante **OK** per ritornare alla pagina **Home** del Web server.



**Figura 14 – Operazione di upgrade del firmware abortita**

4. Se si conferma la procedura, sullo schermo appare la pagina **Firmware Upgrade**: all'operatore è richiesto l'inserimento di una password prima di avviare la procedura di upgrade del firmware.
5. Nella casella di testo **Password** digitare la password **LIKA** (tutte le lettere maiuscole) e premere poi il pulsante **Send Request**.



**Figura 15 – Pagina Firmware Upgrade**

6. Se la password inserita è sbagliata, sullo schermo appare il seguente messaggio di avvertimento: **THE PASSWORD INSERTED IS INCORRECT. PLEASE RETRY! / LA PASSWORD INSERITA E' ERRATA. RIPROVARE!**. Digitare la password corretta e confermare.
7. Se la password inserita è corretta, sullo schermo appare il seguente messaggio: **THE PASSWORD INSERTED IS CORRECT. THE WEB SERVER OF THE ENCODER IS STOPPED. NOW LAUNCH THE PROGRAM SW\_ETH\_REVX\_Y.EXE / LA PASSWORD INSERITA E' CORRETTA. IL WEB SERVER DELL'ENCODER SARA' ARRESTATO. LANCIARE ORA IL PROGRAMMA SW\_ETH\_REVX\_Y.EXE.**
8. L'encoder si trova ora in modalità Bootloader ed è pronto per accettare il programma firmware: il web server viene arrestato e la comunicazione con l'encoder attraverso il browser web interrotta; se fosse necessario uscire dalla procedura e ripristinare la comunicazione, occorre togliere e ridare tensione all'encoder.
9. Bisogna ora lanciare l'eseguibile SW\_ETH\_REVX\_Y.EXE fornito da Lika Electronic per continuare nella procedura; X e Y indicano la versione del programma di upgrade del firmware: REV1\_0 è la versione 1.0.
10. Lanciare l'eseguibile SW\_ETH\_REVX\_Y.EXE; appare la seguente pagina:

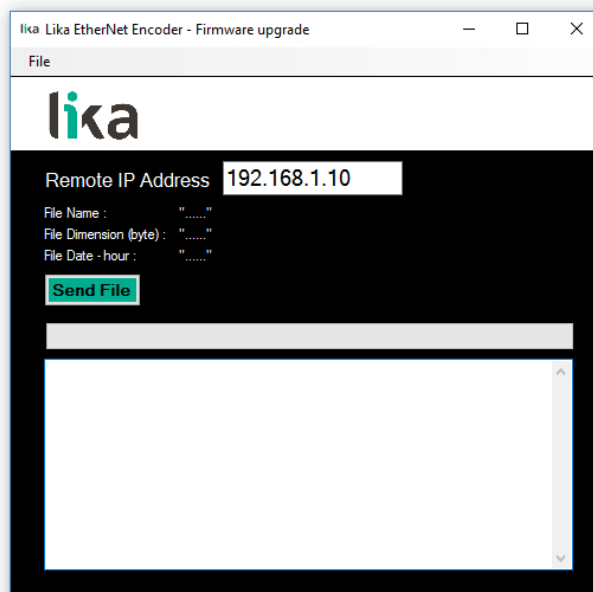


Figura 16 – Eseguibile di upgrade del firmware

11. Digitare l'indirizzo IP dell'encoder nella casella **Remote IP Address**. L'indirizzo IP di default impostato da Lika Electronic è 192.168.1.10.
12. Premere il comando **FILE** e poi il comando **OPEN** nella barra di menu; non appena si preme il comando **OPEN** sullo schermo appare la finestra di dialogo **OPEN**: aprire la cartella dove è salvato il file .BIN di upgrade del firmware rilasciato da Lika Electronic, selezionare il file e confermare.

Hx nel nome del file indica la versione hardware del PCB; Sx indica la versione software del file di upgrade del firmware.



**ATTENZIONE**

Fare attenzione a installare il file BIN che corrisponde perfettamente alla serie dell'encoder da aggiornare.

EM58\_HMS\_MT\_Hx\_Sx.bin per gli encoder EM/HS/HM MODBUS/TCP

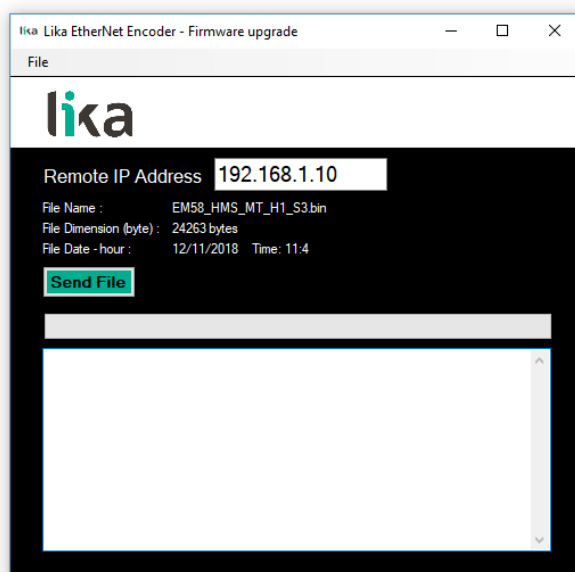


Figura 17 - Selezione del file .BIN di upgrade del firmware

13. A fianco delle rispettive etichette nella pagina vengono visualizzate alcune proprietà del file selezionato: **File Name / Nome file**, **File Dimension (byte) / Dimensione file (byte)**, **File Date – hour / Data – ora file**. Verificare le proprietà del file e assicurarsi di accingersi all'installazione del file di upgrade corretto.



**ATTENZIONE**

Prima dell'installazione accertarsi sempre che il programma firmware sia compatibile con l'hardware e il software del dispositivo.

Non togliere mai l'alimentazione durante l'operazione di upgrade della flash.

14. Premere il pulsante **Send File** per avviare il processo di upgrade del firmware.

15. Una barra di avanzamento del download, insieme ad altre informazioni aggiuntive, apparirà nella pagina durante l'upgrade del firmware.

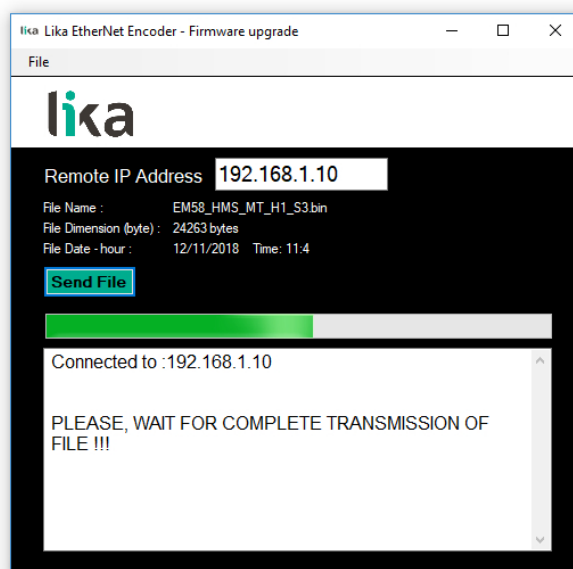


Figura 18 - Firmware in aggiornamento

16. Non appena l'operazione si è conclusa con successo, sullo schermo appare il messaggio **FILE SENT CORRECTLY / FILE INVIATO CORRETTAMENTE**.

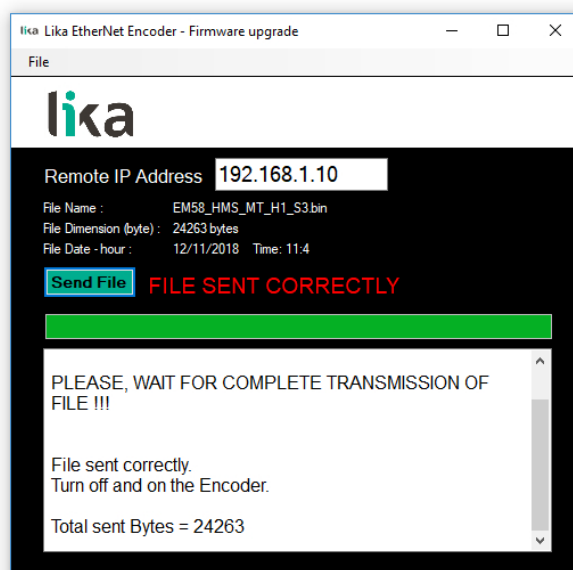


Figura 19 - Processo di upgrade del firmware completato

17. Ora all'operatore è richiesto di spegnere e riaccendere l'encoder.  
Chiudere il programma
18. Spegnere e riaccendere l'encoder per completare l'operazione.

**NOTA**

Durante il download del programma di upgrade del firmware, potrebbero verificarsi delle condizioni inaspettate che potrebbero portare al fallimento del processo di installazione. Quando si verifica questa evenienza, il processo di download non può essere portato a termine con successo e l'operazione è così abortita; sono visualizzati dei messaggi di errore. In caso di errore di upgrade della flash, spegnere e riaccendere l'encoder e riprovare l'operazione.

Premere il comando **Homepage** per ritornare alla pagina Home del Web server.

## 8.8 Configurazione della rete

Premere il comando **Network Configuration** nella barra di navigazione laterale a sinistra nella pagina **Home** del Web server per accedere alla pagina **Network Configuration**. Questa pagina permette all'operatore di configurare le proprietà TCP/IP, ossia come l'encoder comunica con gli altri dispositivi nella rete. Per ulteriori informazioni sui parametri di comunicazione della rete riferirsi alla sezione "4.5 Impostazione dell'indirizzo IP e dei parametri di configurazione della rete" a pagina 29.

**ATTENZIONE**

La configurazione della rete deve essere eseguita da personale istruito e competente.

Non appena si preme il comando **Network Configuration** sullo schermo appare un messaggio di avvertenza (**Are you sure you want to change Network Parameters? / Siete sicuri di voler modificare i parametri di rete?**): avvisa l'operatore della delicatezza dell'operazione, richiedendogli pertanto di confermare la procedura prima di proseguire.

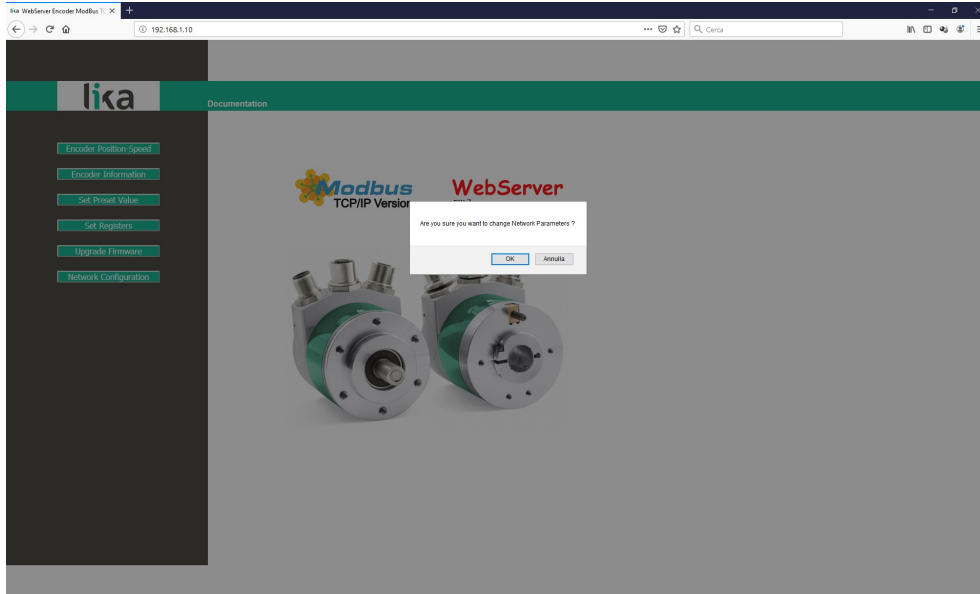


Figura 20 – Accesso alla pagina Network Configuration

Premere il pulsante **OK** per procedere, altrimenti premere il pulsante **EXIT** per interrompere la procedura. Sullo schermo apparirà il messaggio **Set Network parameters cancelled / Impostazione parametri di rete interrotta!**. Premere il pulsante **OK** per ritornare alla pagina **Home** del Web server.

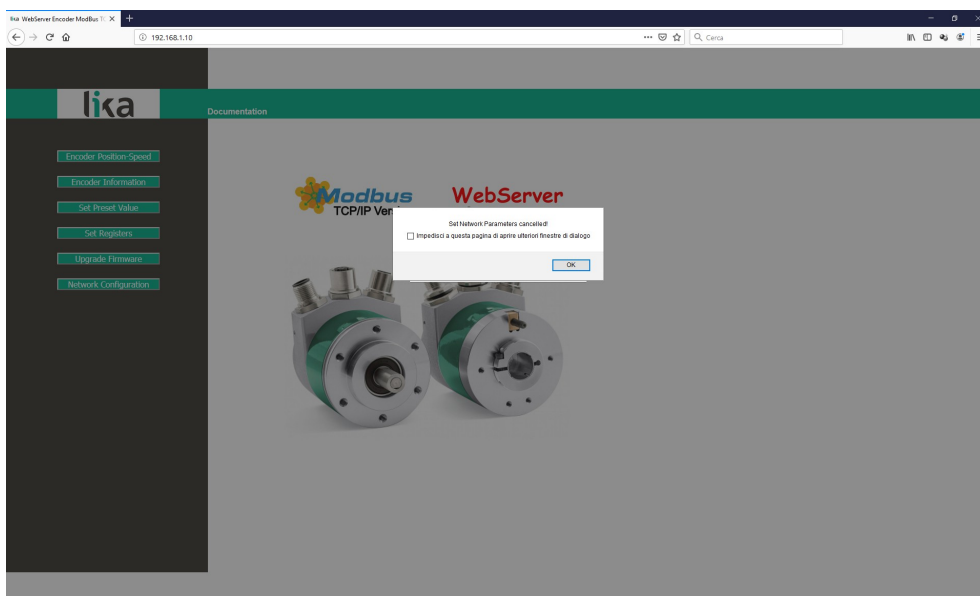
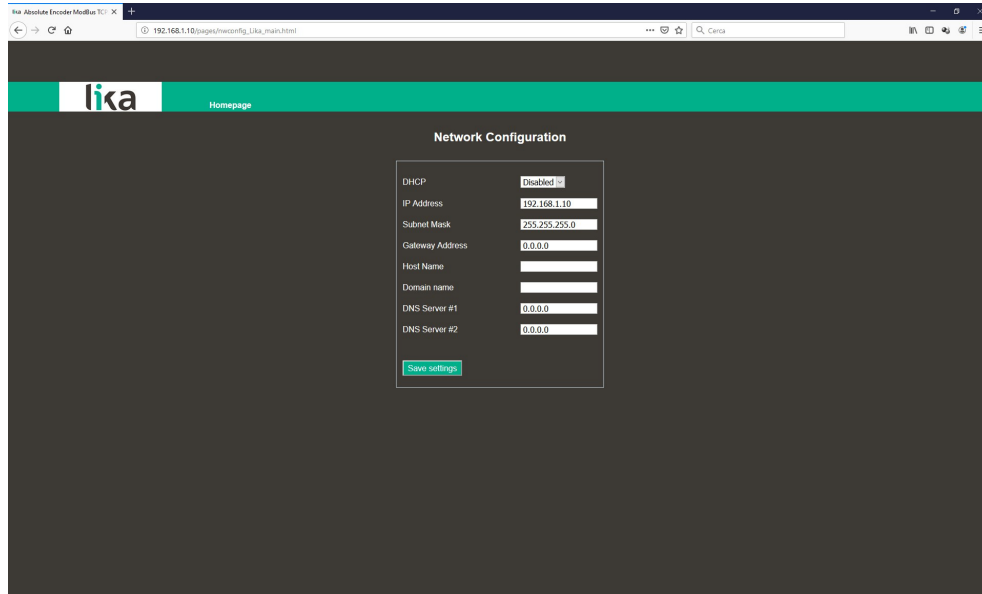


Figura 21 – Configurazione della rete interrotta

Se si conferma la procedura, sullo schermo appare la pagina **Network Configuration**.



**Figura 22 – Pagina Network Configuration**



**ATTENZIONE**

Solo tecnici competenti, ben istruiti e dotati di adeguata esperienza, esperti di architettura dei calcolatori, di progettazione di reti e di sistemi operativi devono avere accesso alla configurazione dei parametri di configurazione della rete. L'impostazione inappropriata dei parametri di rete si traduce in un funzionamento errato del sistema.

In questa pagina è possibile impostare i parametri che influenzano la comunicazione corretta di un encoder nella rete TCP/IP: indirizzo IP, Subnet mask, DHCP, DNS, ecc.

La seguente tabella riassume l'indirizzo IP di default e i parametri di configurazione della rete.

<b>Parametri IP</b>	<b>Valore</b>
DHCP	Disabilitato
Indirizzo IP	192.168.1.10
Subnet mask	255.255.255.0
Indirizzo Gateway	0.0.0.0
Nome Host	
Nome Dominio	
DNS Server #1	0.0.0.0
DNS Server #2	0.0.0.0

Per salvare i valori impostati in maniera permanente, premere il pulsante **Save Settings**. Qualora fosse tolta l'alimentazione all'encoder senza salvare i dati, i valori non memorizzati nella EEPROM Flash sarebbero persi!

**ATTENZIONE**

Dopo ogni impostazione annotare i valori di configurazione per avere accesso all'encoder e alle pagine del Web server in futuro. Se per qualsiasi ragione non si fosse più in grado di stabilire la comunicazione con l'encoder e di accedere al Web server sarà necessario ripristinare i valori di fabbrica (valori di default) dei parametri di configurazione della rete. Per fare questo accedere al DIP switch DIP A installato all'interno del coperchio di connessione. Per informazioni complete riferirsi alla sezione "4.8 DIP A: reset dei parametri di configurazione della rete ai valori di fabbrica" a pagina 32.

**ATTENZIONE**

Se si abilita il protocollo di rete DHCP (DHCP = ENABLED), i seguenti parametri di default vengono impostati nell'encoder:

INDIRIZZO IP = 0.0.0.0

SUBNET MASK = 0.0.0.0

Verificare che queste impostazioni siano ammesse dal server DHCP e che si tratti di valori di indirizzi validi.

**NOTA**

Se per qualsiasi ragione si dovessero ripristinare i valori di fabbrica (valori di default) dei parametri di configurazione della rete, occorrerebbe accedere al DIP switch DIP A installato all'interno del coperchio di connessione. Per informazioni complete riferirsi alla sezione "4.8 DIP A: reset dei parametri di configurazione della rete ai valori di fabbrica" a pagina 32.

Premere il comando **Homepage** per ritornare alla pagina **Home** del Web server.



## 9 Esempi di programmazione

Qui di seguito sono riportati alcuni esempi di lettura e scrittura dei parametri. Tutti i valori sono espressi in notazione esadecimale. Per ogni informazione sulla ADU di MODBUS/TCP (MBAP Header + PDU) riferirsi alla sezione "6.3 MODBUS su Application Data Unit TCP/IP" a pagina 38.

### 9.1 Utilizzo del codice funzione 03 Read Holding Register



#### ESEMPIO 1

Richiesta di lettura dei registri **Preset value [105-106]** (indirizzo 104-105).

**MBAP Header + Request PDU** (in notazione esadecimale)

[00][01][00][00][00][06][00][03][00][68][00][02]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][06] = Dimensione

[00] = Identificatore di unità

[03] = codice funzione **03 Read Holding Register**

[00][68] = indirizzo iniziale (registri **Preset value [105-106]**, indirizzo 104-105)

[00][02] = numero di registri richiesti

**MBAP Header + Response PDU** (in notazione esadecimale)

[00][01][00][00][00][07][00][03][04][00][00][05][DC]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][07] = Dimensione

[00] = Identificatore di unità

[03] = codice funzione **03 Read Holding Register**

[04] = numero di byte (2 byte per ciascun registro)

[00][00] = valore del registro 105, 00 00 hex = 0 dec

[05][DC] = valore del registro 106, 05 DC hex = 1500 dec

I registri **Preset value [105-106]** (indirizzo 104-105) contengono il valore 00 00 hex e 05 DC hex, ossia 1500 in notazione decimale; in altre parole il valore impostato nei registri **Preset value [105-106]** è 1500 dec.

## 9.2 Utilizzo del codice funzione 04 Read Input Register



### ESEMPIO 1

Richiesta di lettura dei registri **Current position [1-2]** (indirizzo 0-1).

**MBAP Header + Request PDU** (in notazione esadecimale)

[00][01][00][00][00][06][00][04][00][00][00][02]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][06] = Dimensione

[00] = Identificatore di unità

[04] = codice funzione **04 Read Input Register**

[00][00] = indirizzo iniziale (registri **Current position [1-2]**, indirizzo 0-1)

[00][02] = numero di registri richiesti

**MBAP Header + Response PDU** (in notazione esadecimale)

[00][01][00][00][00][07][00][04][04][00][00][2F][F0]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][07] = Dimensione

[00] = Identificatore di unità

[04] = codice funzione **04 Read Input Register**

[04] = numero di byte (2 byte per ciascun registro)

[00][00] = valore del registro 1, 00 00 hex = 0 dec

[2F][F0] = valore del registro 2, 2F F0 hex = 12272 dec

I registri **Current position [1-2]** (indirizzo 0-1) contengono il valore 00 00 2F F0 hex, ossia 12272 in notazione decimale.

### 9.3 Utilizzo del codice funzione 06 Write Single Register



#### ESEMPIO 1

Richiesta di scrittura del registro **Watchdog timeout [82]** (indirizzo 81): si vuole abilitare la funzione di Watchdog e impostare un timeout di 10 ms.

#### MBAP Header + Request PDU (in notazione esadecimale)

[00][01][00][00][00][06][00][06][00][51][00][0A]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][06] = Dimensione

[00] = Identificatore di unità

[06] = codice funzione **06 Write Single Register**

[00][51] = indirizzo del registro **Watchdog timeout [82]**, 51 hex = 81 dec

[00][0A] = valore da impostare nel registro

#### MBAP Header + Response PDU (in notazione esadecimale)

[00][01][00][00][00][06][00][06][00][51][00][0A]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][06] = Dimensione

[00] = Identificatore di unità

[06] = codice funzione **06 Write Single Register**

[00][51] = indirizzo del registro **Watchdog timeout [82]**, 51 hex = 81 dec

[00][0A] = valore impostato nel registro

Nel registro **Watchdog timeout [82]** (indirizzo 81) è impostato il valore 00 0A hex (10 dec): la funzione di Watchdog è abilitata e il timeout è impostato a 10 ms.

## 9.4 Utilizzo del codice funzione 16 Write Multiple Register



### ESEMPIO 1

Richiesta di scrittura del valore 00 00 08 00 hex (=2048 dec) nei registri **Counts per revolution [101-102]** (indirizzo 100-101) e del valore 00 80 00 00 hex (= 8388608 dec) nei registri **Total Resolution [103-104]** (indirizzo 102-103).

#### MBAP Header + Request PDU (in notazione esadecimale)

[00][01][00][00][00][0F][00][10][00][64][00][04][08][00][00][08][00][00][80][00][00]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][0F] = Dimensione

[00] = Identificatore di unità

[10] = codice funzione **16 Write Multiple Register**

[00][64] = indirizzo iniziale (registri **Counts per revolution [101-102]**, indirizzo 100-101)

[00][04] = numero di registri richiesti

[08] = numero di byte (2 byte per ciascun registro)

[00][00] = valore da impostare nel registro 101, 00 00 hex

[08][00] = valore da impostare nel registro 102, 08 00 hex (00 00 08 00 hex = 2048 dec)

[00][80] = valore da impostare nel registro 103, 00 80 hex

[00][00] = valore da impostare nel registro 104, 00 00 hex (00 80 00 00 hex = 8388608 dec)

#### MBAP Header + Response PDU (in notazione esadecimale)

[00][01][00][00][00][06][00][10][00][64][00][04]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][06] = Dimensione

[00] = Identificatore di unità

[10] = codice funzione **16 Write Multiple Register**

[00][64] = indirizzo iniziale (registri **Counts per revolution [101-102]**, indirizzo 100-101)

[00][04] = numero di registri scritti

Nei registri **Counts per revolution [101-102]** all'indirizzo 100-101 sono impostati i valori 00 00 hex e 08 00 hex, ossia 2048 in notazione decimale; mentre nei registri **Total Resolution [103-104]** all'indirizzo 102-103 sono impostati i valori 00 80 hex e 00 00 hex, ossia 8388608 in notazione decimale. Pertanto l'encoder sarà programmato per avere una risoluzione monogiro di 2048 cpr e 4096 giri (= 8388608/2048).

**ESEMPIO 2**

Richiesta di scrittura dei registri **Operating parameters [109-110]** (indirizzo 108-109): si vogliono impostare la funzione di scaling (bit 0 **Scaling function** = 1) e il conteggio crescente con rotazione oraria dell'albero dell'encoder (bit 1 **Code sequence** = 0). Il valore da impostare è 00 00 00 01 hex (= 0000 0000 0000 0000 0000 0000 0000 0001 in notazione binaria: il bit 0 **Scaling function** = 1; il bit 1 **Code sequence** = 0; i rimanenti bit non sono usati, pertanto il loro valore è 0).

**MBAP Header + Request PDU** (in notazione esadecimale)

[00][01][00][00][00][0B][00][10][00][6C][00][02][04][00][00][00][01]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][0B] = Dimensione

[00] = Identificatore di unità

[10] = codice funzione **16 Write Multiple Register**

[00][6C] = indirizzo iniziale (registri **Operating parameters [109-110]**, indirizzo 108-109)

[00][02] = numero di registri richiesti

[04] = numero di byte (2 byte per ciascun registro)

[00][00] = valore da impostare nel registro 109, 00 00 hex

[00][01] = valore da impostare nel registro 110, 00 01 hex

**MBAP Header + Response PDU** (in notazione esadecimale)

[00][01][00][00][00][06][00][10][00][6C][00][02]

dove:

[00][01] = Identificatore di transazione

[00][00] = Identificatore di protocollo

[00][06] = Dimensione

[00] = Identificatore di unità

[10] = codice funzione **16 Write Multiple Register**

[00][6C] = indirizzo iniziale (registri **Operating parameters [109-110]**, indirizzo 108-109)

[00][02] = numero di registri scritti

Nei registri **Operating parameters [109-110]** (indirizzo 108-109) è impostato 00 00 00 01 hex, ossia 0000 0000 0000 0000 0000 0000 0000 0001 in notazione binaria: il bit 0 **Scaling function** = 1; il bit 1 **Code sequence** = 0; i rimanenti bit non sono usati, pertanto il loro valore è 0.

## 10 Lista parametri di default

I valori di default sono espressi in notazione decimale, salvo diversa indicazione.

Lista parametri	Valori di default		
Watchdog timeout [82]	0		
Counts per revolution [101-102]	8192	per serie EM58	
	262144	per serie HS58	
	65536	per serie HM58	
Total Resolution [103-104]	134217728	per serie EM58	
	262144	per serie HS58	
	1073741824	per serie HM58	
Preset value [105-106]	0		
Speed format [107-108]	0 = steps/s		
Operating parameters [109-110]	0000 0000 hex		
	bit 0 Scaling function = 0		
	bit 1 Code sequence = 0		
Singleturn resolution [113-114]	8192	per serie EM58	
	262144	per serie HS58	
	65536	per serie HM58	
Number of revolutions [115-116]	16384	per serie EM58	
	1	per serie HS58	
	16384	per serie HM58	
Supported alarms [117-118]	0000 7000 hex		
Supported warnings [119-120]	0		

Pagina lasciata bianca intenzionalmente

Versione documento	Data release	Descrizione	HW	SW	Interfaccia
1.0	29.11.2016	Prima edizione	1.0	1.0	-
1.1	23.07.2019	Aggiornamento web server, revisione generale	1.0	2.0, 3.0	-
1.2	18.09.2023	Aggiunta informazione connettori assiali, revisione generale, prima edizione in italiano ottobre 2022, nuovi codici di ordinazione, informazione su prodotto in via di dismissione	1.0	3.0	-



This device is to be supplied by a Class 2 Circuit or Low-Voltage Limited Energy or Energy Source not exceeding 30 Vdc. Refer to the order code for supply voltage rate.

Ce dispositif doit être alimenté par un circuit de Classe 2 ou à très basse tension ou bien en appliquant une tension maxi de 30Vcc. Voir le code de commande pour la tension d'alimentation.



Smaltire separatamente

# 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