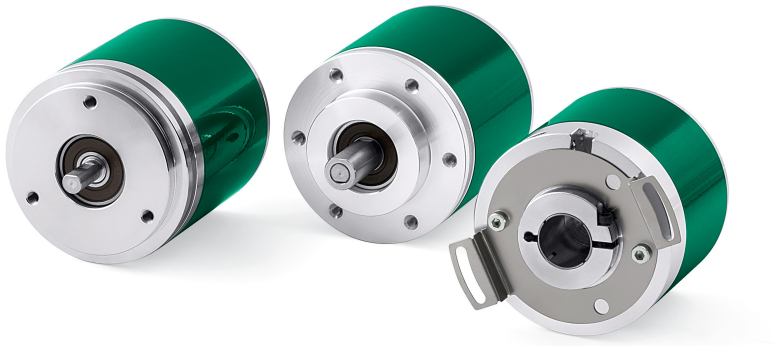
Smart encoders & actuators

Manuale d'uso

ES58, ES58S
EM58, EM58S
ESC58/59/60
EMC58/59/60



Modbus

Versione RS-485

- Encoder rotativo assoluto mono e multigiro
- Interfaccia MODBUS RTU (RS-485) con software di programmazione
- Risoluzione 4.096 cpr (ES58), 4.096 cpr x 16.384 giri (EM58)
- LED di diagnostica
- Grado di protezione IP67

Descrive i seguenti modelli:

- ES5812/MB-..., ES58S12/MB-...
- ESC5812/MB-..., ESC5912/MB-..., ESC6012/MB-...
- EM5812/16384MB-..., EM58S12/16384MB-...
- EMC5812/16384MB-..., EMC5912/16384MB-..., EMC6012/16384MB-...

Indice generale

Norme di sicurezza	13
Identificazione	15
Installazione meccanica	16
Connessioni elettriche	21
Quick reference	31
Interfaccia MODBUS®	50
Parametri di programmazione	65
Esempi di programmazione	85

Questa pubblicazione è edita da Lika Electronic s.r.l. 2021. All rights reserved. Tutti i diritti riservati. Alle Rechte vorbehalten. Todos los derechos reservados. Tous droits réservés.

Il presente manuale e le informazioni in esso contenute sono proprietà di Lika Electronic s.r.l. e non possono essere riprodotte né interamente né parzialmente senza una preventiva autorizzazione scritta di Lika Electronic s.r.l. La traduzione, la riproduzione e la modifica totale o parziale (incluse le copie fotostatiche, i film, i microfilm e ogni altro mezzo di riproduzione) sono vietate senza l'autorizzazione scritta di Lika Electronic s.r.l.

Le informazioni contenute nel presente manuale sono soggette a modifica senza preavviso e non devono essere in alcun modo ritenute vincolanti per Lika Electronic s.r.l. Lika Electronic s.r.l. si riserva il diritto di apportare delle modifiche al presente testo in qualunque momento e senza nessun obbligo di informazione a terzi.

Questo manuale è periodicamente rivisto e aggiornato. All'occorrenza si consiglia di verificare l'esistenza di aggiornamenti o nuove edizioni di questo manuale sul sito istituzionale di Lika Electronic s.r.l. Lika Electronic s.r.l. non si assume alcuna responsabilità per eventuali errori o omissioni riscontrabili in questo documento. Valutazioni critiche di questo manuale da parte degli utilizzatori sono gradite. Ogni eventuale osservazione ci è utile nella stesura della futura documentazione, al fine di redigere un prodotto che sia quanto più chiaro, utile e completo possibile. Per inviarci i Vostri commenti, suggerimenti e critiche mandate una e-mail all'indirizzo info@lika.it.

lika

Indice generale

Manuale d'uso **1**
Indice generale **3**
Indice analitico **6**
Convenzioni grafiche e iconografiche **7**
Informazioni preliminari **8**
Glossario dei termini MODBUS **9**
1 Norme di sicurezza **13**
1.1 Sicurezza 13
1.2 Avvertenze elettriche 13
1.3 Avvertenze meccaniche 14
2 Identificazione **15**
3 Installazione meccanica **16**
3.1 Encoder con asse sporgente (ES58, ES58S, EM58, EM58S) 16
3.1.1 Fissaggio standard 16
3.1.2 Fissaggio con graffe (codice LKM-386) 17
3.1.3 Fissaggio con campana (codice PF4256) 17
3.2 Encoder con asse cavo (ESC58, ESC59, ESC60, EMC58, EMC59, EMC60) 18
3.2.1 ESC58, EMC58 18
3.2.2 ESC59, EMC59 19
3.2.3 ESC60, EMC60 20
4 Connessioni elettriche **21**
4.1 Cavo CB 21
4.1.1 Caratteristiche del cavo CB 21
4.2 Connettore M12 5 pin 22
4.3 Collegamento messa a terra 22
4.4 Indicatori LED (Figura 1) 23
4.5 Selettori (Figura 2 e Figura 3) 26
4.5.1 Velocità di trasmissione dei dati: baud rate e bit di parità (Figura 3) 27
4.5.2 Indirizzo nodo (Figura 3) 28
4.5.3 Resistenza di terminazione (Figura 3) 30
5 Quick reference **31**
5.1 Getting started 31
5.2 Configurazione mediante software di Lika Electronic 32
5.3 Pagina principale dell'interfaccia 33
5.3.1 Configurazione porta seriale – Collegamento con il dispositivo 34
Leggi param. 36
Scrivi Holding 36
5.3.2 Lettura degli Input Register – Registri INPUT 37
Lettura continua 38
Posizione corrente 38
Angolo 38
Conti 38
Giri 38
Dip switch baud rate 38
Dip switch indirizzo 39

Versione SW ... 39
Versione HW ... 39
Status word ... 39
Registro allarmi ... 39
Dati macchina errati ... 40
5.3.3 Lettura delle exception response – Exception error ... 40
Exception error ... 40
5.3.4 Lettura / scrittura degli Holding Register – Registri HOLDING ... 41
Conti / giro ... 42
Risoluzione totale ... 42
Valore di preset ... 42
Valore di offset ... 42
Indirizzo nodo ... 42
Baud rate ... 43
Parametri operativi ... 43
Abilita funzione di scaling ... 43
Inverte direz. di conteggio ... 43
Control word ... 43
Abilita watchdog ... 43
Preset conteggio ... 44
Parametri di default ... 44
Salvataggio autom. ... 44
Salva parametri ... 45
5.4 Pagina Aggiorna FW - Upgrade del firmware ... 45
5.4.1 Informazioni sull'upgrade del firmware ... 46
5.4.2 Operazioni e connessioni preliminari ... 47
5.4.3 Esecuzione del processo di upgrade del firmware ... 47
5.5 Pagina Manual frame – Scrittura manuale PDU ... 48
6 Interfaccia MODBUS® ... 50
6.1 Principi guida del protocollo MODBUS Master / Slave ... 50
6.2 Frame MODBUS ... 51
6.3 Modalità di trasmissione ... 52
6.3.1 Modalità di trasmissione RTU ... 52
6.4 Codici funzione ... 55
6.4.1 Codici funzione implementati ... 55
03 Read Holding Registers ... 55
04 Read Input Register ... 57
06 Write Single Register ... 59
16 Write Multiple Registers ... 61
7 Parametri di programmazione ... 65
7.1 Parametri disponibili ... 65
7.1.1 Parametri Dati macchina (Holding register) ... 65
Informazioni per giro programmate [0000-0001 hex] ... 65
Risoluzione totale programmata [0002-0003 hex] ... 67
Valore di preset [0004-0005 hex] ... 69
Offset [0006-0007 hex] ... 71
Parametri operativi [0008 hex] ... 71
Funzione di scaling ... 71
Direzione di conteggio ... 72
Control Word [0009 hex] ... 73

Abilitazione Watchdog ... 73
Salva parametri ... 73
Carica parametri di default ... 74
Esegui preset conteggio ... 74
7.1.2 Parametri Input Register ... 76
Registro allarmi [0000 hex] ... 76
Dati macchina non validi ... 76
Errore memoria flash ... 76
Watchdog ... 76
Posizione corrente [0001-0002 hex] ... 77
Registro 4 [0003 hex] ... 77
Elenco DM errati [0004-0005 hex] ... 77
DIP switch baud rate [0006 hex] ... 78
DIP switch indirizzo [0007 hex] ... 79
Versione software [0008 hex] ... 79
Versione hardware [0009 hex] ... 79
Status word [000A hex] ... 80
Scaling ... 80
Direzione conteggio ... 80
Allarme ... 80
7.2 Codici eccezione ... 81
8 Esempi di programmazione ... 85
8.1 Utilizzo del codice funzione 03 Read Holding Registers ... 85
8.2 Utilizzo del codice funzione 04 Read Input Register ... 86
8.3 Utilizzo del codice funzione 06 Write Single Register ... 87
8.4 Utilizzo del codice funzione 16 Write Multiple Registers ... 88
9 Tabella registri ... 89
9.1 Lista Holding Register con valore di default ... 89
9.2 Lista Input Register ... 89

Indice analitico

A
Abilita funzione di scaling....43
Abilita watchdog....43
Abilitazione Watchdog....73
Allarme....80
Angolo....38
B
Baud rate....43
C
Carica parametri di default....74
Conti....38
Conti / giro....42
Control word....43
Control Word [0009 hex]....73
D
Dati macchina errati....40
Dati macchina non validi....76
Dip switch baud rate....38
DIP switch baud rate [0006 hex]....78
Dip switch indirizzo....39
DIP switch indirizzo [0007 hex]....79
Direzione conteggio....80
Direzione di conteggio....72
E
Elenco DM errati [0004-0005 hex]....77
Errore memoria flash....76
Esegui preset conteggio....74
Exception error....40
F
Funzione di scaling....71
G
Giri....38
I
Indirizzo nodo....42
Informazioni per giro programmate [0000-0001 hex]....65
Inverte direz. di conteggio....43
L
Leggi param....36
Lettura continua....38
M
MANUAL FRAME....48
O
Offset [0006-0007 hex]....71
P
Parametri di default....44
Parametri operativi....43
Parametri operativi [0008 hex]....71
Posizione corrente....38
Posizione corrente [0001-0002 hex]....77
Preset conteggio....44
R
Registro 4 [0003 hex]....77
Registro allarmi....39
Registro allarmi [0000 hex]....76
Risoluzione totale....42
Risoluzione totale programmata [0002-0003 hex]....67
S
Salva parametri....45, 73
Salvataggio autom....44
Scaling....80
Scrivi Holding....36
Status word....39
Status word [000A hex]....80
V
Valore di offset....42
Valore di preset....42
Valore di preset [0004-0005 hex]....69
Versione hardware [0009 hex]....79
Versione HW....39
Versione software [0008 hex]....79
Versione SW....39
W
Watchdog....76

Convenzioni grafiche 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 **ATTENZIONE**, 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 **NOTA**, 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 **ESEMPIO** quando le istruzioni di impostazione dei parametri siano seguite da esemplificazioni che ne chiarifichino l'utilizzo.

Informazioni preliminari

Questo manuale ha lo scopo di fornire tutte le informazioni necessarie per un'installazione e un utilizzo corretti e sicuri degli **encoder assoluti monogiro della serie ES58 e multigiro della serie EM58 con interfaccia MODBUS**.
In particolare:

- **ES5812/MB-...**
- **ES58S12/MB-...**
- **ESC5812/MB-...**
- **ESC5912/MB-...**
- **ESC6012/MB-...**
- **EM5812/16384MB-...**
- **EM58S12/16384MB-...**
- **EMC5812/16384MB-...**
- **EMC5912/16384MB-...**
- **EMC6012/16384MB-...**

Per le specifiche tecniche riferirsi al datasheet di prodotto.

Per una più agevole consultazione questo manuale può essere diviso in due parti.
Nella prima parte sono fornite le informazioni generali riguardanti l'encoder ES58 / EM58 MODBUS comprendenti le norme di sicurezza, le istruzioni di montaggio meccanico e le prescrizioni relative alle connessioni elettriche, nonché ulteriori informazioni sul funzionamento e la corretta messa a punto del dispositivo.
Nella seconda parte invece, intitolata **Interfaccia MODBUS**, sono fornite tutte le informazioni sia generali che specifiche relative all'interfaccia MODBUS. In questa sezione sono descritte le caratteristiche dell'interfaccia e i registri MODBUS che l'unità implementa.
Nella sezione "Quick reference" a pagina 31 è descritta l'interfaccia di configurazione semplificata dell'encoder via RS-485 predisposta da Lika Electronic.

Glossario dei termini MODBUS

MODBUS, 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. Sono elencati in ordine alfabetico.

Address field	Contiene l'indirizzo dello Slave.
Application Process	L'Application Process è il task sull'Application Layer.
Application protocol	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 mezzo di trasmissione.
Bus	Un bus è un mezzo di comunicazione per il collegamento di più nodi. I dati possono essere trasmessi attraverso circuiti seriali o paralleli, ossia mediante conduttori elettrici o fibra ottica.
Campo indirizzo	Contiene l'indirizzo dello Slave.
Client	Un Client è un qualsiasi dispositivo di rete che invia richieste di dati ai Server. MODBUS segue il modello Client/Server. I Master MODBUS sono intesi come Client, mentre gli Slave MODBUS sono intesi come Server.
Codice funzione	MODBUS è un protocollo di tipo richiesta/risposta e offre servizi specificati da codici funzione. Il codice funzione è inviato da un Client al Server e indica che tipo di azione deve eseguire il Server. 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. I dispositivi Lika implementano solamente codici funzione pubblici.
Codifica dati	MODBUS utilizza una rappresentazione 'big-Endian' per dati e indirizzi. Questo significa che, quando si invia una quantità numerica maggiore del singolo byte, il byte più significativo è inviato per primo.
Controllo a ridondanza ciclica (CRC)	Tecnica di verifica dell'errore mediante la quale il destinatario del frame calcola un valore dividendo il contenuto del frame per un divisore binario primo e confronta il valore calcolato con il valore memorizzato nel frame dal nodo trasmittente.
Cyclic Redundancy Check (CRC)	Tecnica di verifica dell'errore mediante la quale il destinatario del frame calcola un valore dividendo il contenuto del frame per un divisore binario primo e confronta il valore calcolato

	con il valore memorizzato nel frame dal nodo trasmittente.
Exception code	Codice che gli Slave devono restituire nel caso di problemi. Tutte le eccezioni sono segnalate mediante l'aggiunta di 0x80 al codice funzione della richiesta.
Exception response	MODBUS lavora secondo il comune 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).
Function code	MODBUS è un protocollo di tipo richiesta/risposta e offre servizi specificati da codici funzione. Il codice funzione è inviato da un Client al Server e indica che tipo di azione deve eseguire il Server. 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. I dispositivi Lika implementano solamente codici funzione pubblici.
Holding register	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.
IEEE 1588	Questo standard definisce un protocollo che abilita la sincronizzazione di clock nei dispositivi in una rete distribuita (per esempio connessi via Ethernet).
Input register	Nel modello dati MODBUS, un Input register è il dato in ingresso. Un Input register ha una dimensione di 16 bit, è fornito da un sistema I/O e permette un accesso in sola-lettura.
LRC Checking	In modalità ASCII, i messaggi includono un campo di verifica degli errori che si basa sul calcolo di un Longitudinal Redundancy Checking (LRC) realizzato sul contenuto del messaggio, senza i due punti iniziali e della coppia di caratteri CR (Carriage Return) e LF (Line Feed) finali. E' usato indipendentemente da qualsiasi metodo di controllo della parità che si applichi ai singoli caratteri del messaggio.
Master	Un Master è qualsiasi dispositivo di rete che invia richieste dati agli Slave.
Message	Il servizio di messaggistica MODBUS permette una comunicazione Client/Server tra i dispositivi connessi nella rete. Il modello Client / Server model si basa su quattro tipi di messaggi:

	• MODBUS Request • MODBUS Confirmation • MODBUS Indication • MODBUS Response I servizi di messaggistica MODBUS sono utilizzati per lo scambio di informazioni.
Messaggio	Il servizio di messaggistica MODBUS permette una comunicazione Client/Server tra i dispositivi connessi nella rete. Il modello Client / Server model si basa su quattro tipi di messaggi: • MODBUS Request • MODBUS Confirmation • MODBUS Indication • MODBUS Response I servizi di messaggistica MODBUS sono utilizzati per lo scambio di informazioni.
Modalità di trasmissione RTU	Remote Terminal Unit. Quando i dispositivi comunicano in una linea seriale MODBUS utilizzando la modalità RTU, ciascun byte a 8 bit nel messaggio contiene due caratteri esadecimali a 4 bit. Il vantaggio principale di questa modalità è il fatto che la densità del suo carattere più grande permette una migliore capacità di trattamento dati rispetto alla modalità ASCII a parità di baud rate. Ciascun messaggio deve essere trasmesso in un flusso continuo di caratteri.
MODBUS Confirmation	Un messaggio MODBUS Confirmation è il messaggio di risposta (Response message) ricevuto sul lato Client.
MODBUS Indication	Un messaggio MODBUS Indication è il messaggio di richiesta (Request message) ricevuto sul lato Server.
MODBUS Request	Un messaggio MODBUS Request è il messaggio inviato nella rete da un Client per avviare una transazione.
MODBUS Response	Un messaggio MODBUS Response è il messaggio di risposta (Response message) inviato dal Server.
PDU	La Protocol Data Unit (PDU) è il codice funzione MODBUS e il campo dati. E' associato al campo indirizzo (Address Field) e al CRC (o LRC) a formare la Modbus Serial Line PDU. Il protocollo MODBUS definisce tre PDU. Esse sono: • MODBUS Request PDU, mb_req_pdu • MODBUS Response PDU, mb_rsp_pdu • MODBUS Exception Response PDU, mb_excep_rsp_pdu
Protocollo a livello applicativo	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 mezzo di trasmissione.
Read Holding Register (03, 0003hex)	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 lavori disposti in sequenza.
Read Input Register (04, 0004hex)	Questo codice funzione è utilizzato 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.
Register	Le funzioni MODBUS operano sui registri di memoria per configurare, monitorare e controllare gli I/O del dispositivo.
Registro	Le funzioni MODBUS operano sui registri di memoria per configurare, monitorare e controllare gli I/O del dispositivo.
Rete	Una rete è un gruppo di computer in un singolo segmento di rete fisica.
Server	Un Server è un qualsiasi programma che attende che gli sia inviata una richiesta dati. I Server non avviano la comunicazione con i Client, ma inviano solo le risposte. MODBUS segue il modello Client/Server. MODBUS I Master devono essere intesi come Client, mentre gli Slave MODBUS devono essere intesi come Server.
Service request	E' la MODBUS Request, ossia il messaggio inviato nella rete da un Client per iniziare una transazione.
Slave	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.
Trasmissione ASCII	Quando i dispositivi sono impostati per comunicare su una linea seriale MODBUS utilizzando la modalità ASCII (American Standard Code for Information Interchange), ciascun byte (8 bit) nel messaggio è inviato sotto forma di due caratteri ASCII. Questa modalità è utilizzata quando il link di comunicazione fisica o le capacità del dispositivo non permettono di ottemperare ai requisiti della modalità RTU per quanto riguarda la gestione dei timer.
Velocità di trasmissione	Velocità di trasmissione dati (in bps).
Write Multiple Register (16, 0010hex)	Questo codice funzione è utilizzato per SCRIVERE un blocco di registri contigui (da 1 a 123 registri) in un dispositivo remoto.
Write Single Register (06, 0006hex)	Questo codice funzione è utilizzato per SCRIVERE un singolo holding register in un dispositivo remoto.

1 Norme di sicurezza

1.1 Sicurezza

- Durante l'installazione e l'utilizzo del dispositivo osservare 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;
- alte correnti, tensioni e parti meccaniche 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 "Connessioni elettriche" a pagina 21;
- 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 eventuali 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.

1.3 Avvertenze meccaniche

- Montare il dispositivo rispettando rigorosamente le istruzioni riportate nella sezione "Installazione meccanica" a pagina 16;
- effettuare il montaggio meccanico esclusivamente in assenza di parti meccaniche in movimento;
- non disassemblare il dispositivo;
- non eseguire lavorazioni meccaniche sul dispositivo;
- dispositivo elettronico delicato: maneggiare con cura;
- evitare urti o forti sollecitazioni sia all'albero 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 un **codice di ordinazione** e un **numero di serie** 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 e il numero di serie 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 Installazione meccanica

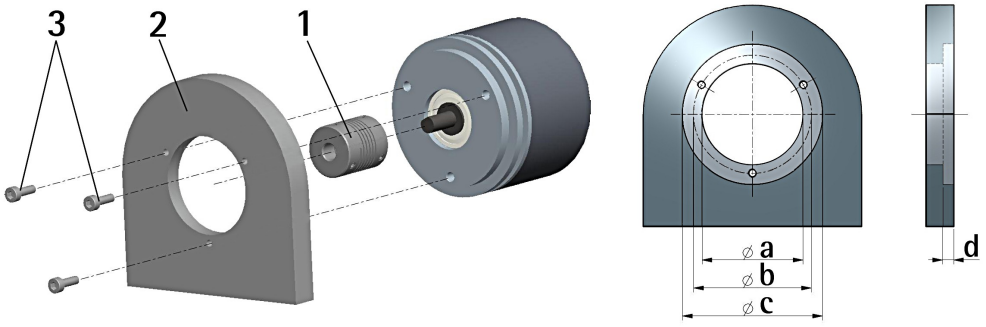
ATTENZIONE
L'installazione e le operazioni di manutenzione devono essere eseguite da personale qualificato, in assenza di tensione e movimenti dell'albero o di altre componenti meccaniche.

Per ogni informazione sulle caratteristiche meccaniche ed elettriche dell'encoder riferirsi al catalogo tecnico del prodotto.

3.1 Encoder con asse sporgente (ES58, ES58S, EM58, EM58S)

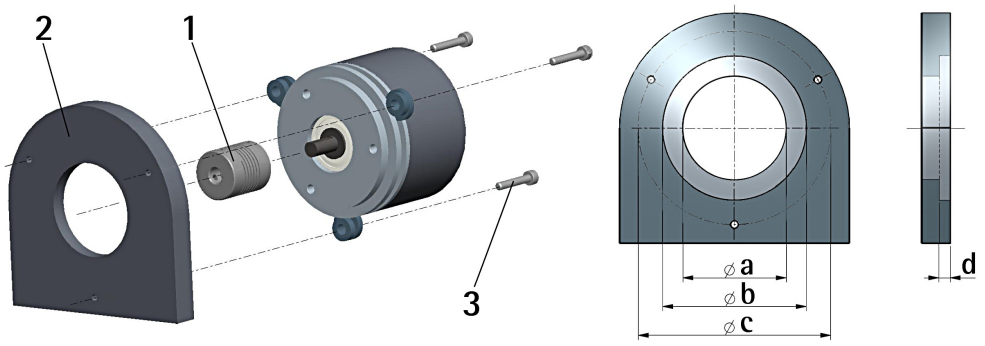
- Fissare il giunto elastico **1** all'encoder;
- fissare l'encoder alla flangia di fissaggio **2** o alla campana utilizzando le viti **3**;
- fissare la flangia **2** al supporto o la campana al motore;
- fissare il giunto elastico **1** al motore;
- assicurarsi che le tolleranze di disallineamento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

3.1.1 Fissaggio standard



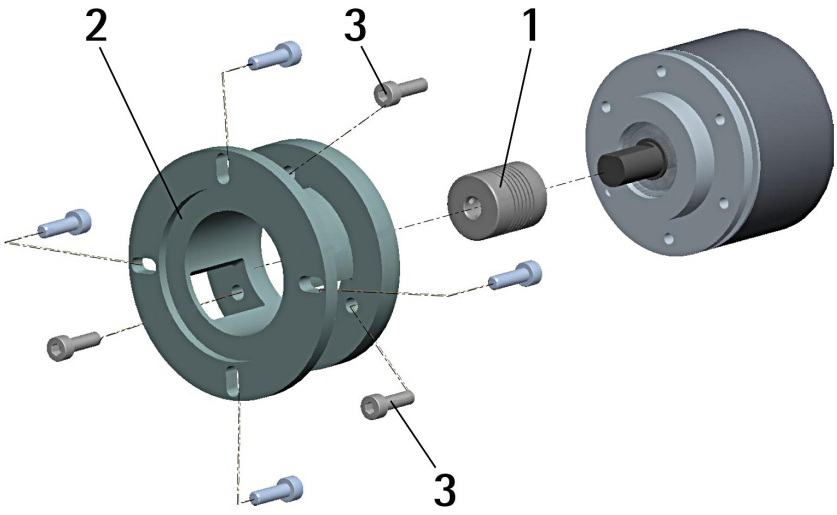
	a [mm]	b [mm]	c [mm]	d [mm]
ES58, EM58	-	42	50 F7	4
ES58S, EM58S	36 H7	48	-	-

3.1.2 Fissaggio con graffe (codice LKM-386)



	a [mm]	b [mm]	c [mm]	d [mm]
ES58, EM58	-	50 F7	67	4
ES58S, EM58S	36 H7	-	67	-

3.1.3 Fissaggio con campana (codice PF4256)
ES58S, EM58S

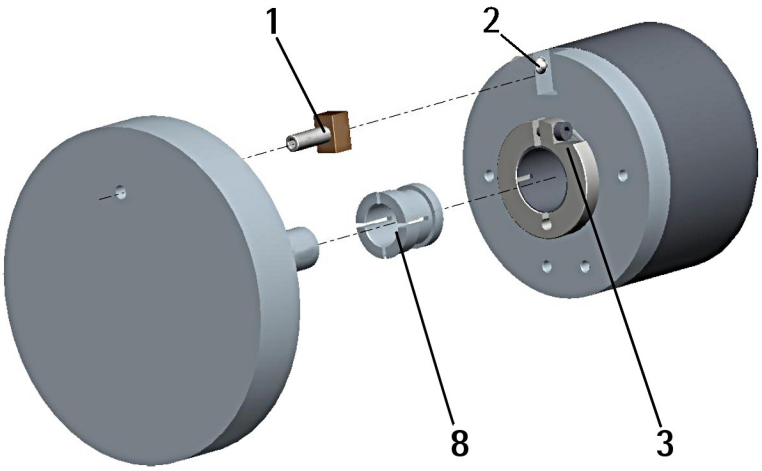


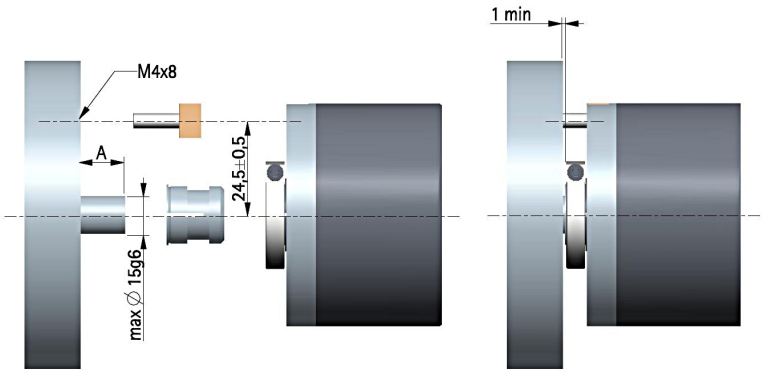
NOTA
Si raccomanda di utilizzare giunti elastici per collegare encoder ad asse sporgente e motore; rispettare le tolleranze di disallineamento ammesse dal giunto elastico.

3.2 Encoder con asse cavo (ESC58, ESC59, ESC60, EMC58, EMC59, EMC60)

3.2.1 ESC58, EMC58

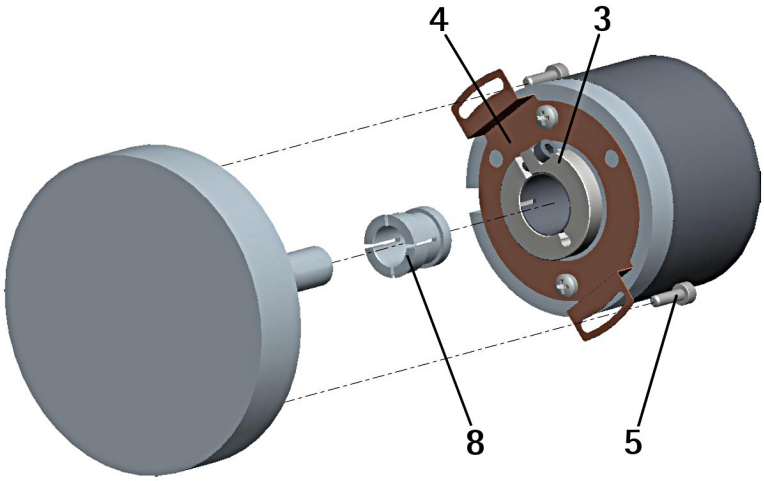
- Fissare il pin antirotazione **1** sul retro del motore (fissaggio con controdado);
- inserire l'encoder sull'albero del motore utilizzando la boccola di riduzione **8** (se fornita). Evitare sforzi sull'albero encoder;
- inserire il pin antirotazione **1** nella fresatura della flangia encoder; esso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

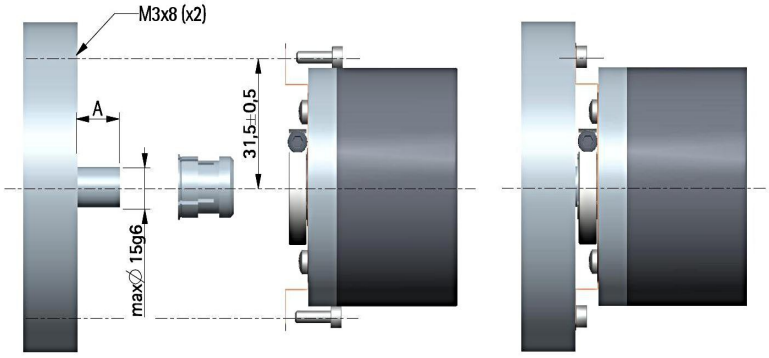




3.2.2 ESC59, EMC59

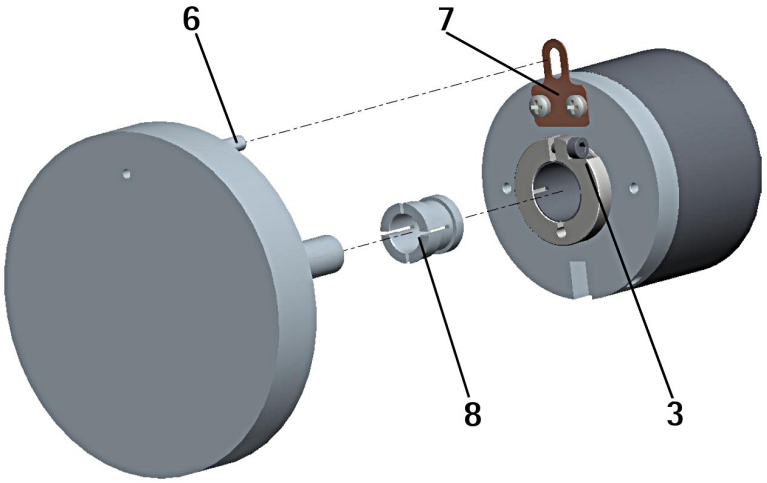
- Inserire l'encoder sull'albero del motore utilizzando la boccola di riduzione **8** (se fornita); evitare sforzi sull'albero encoder;
- fissare la molla di fissaggio **4** sul retro del motore utilizzando due viti M3 a testa cilindrica **5**;
- fissare il collare **3** dell'albero encoder mediante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

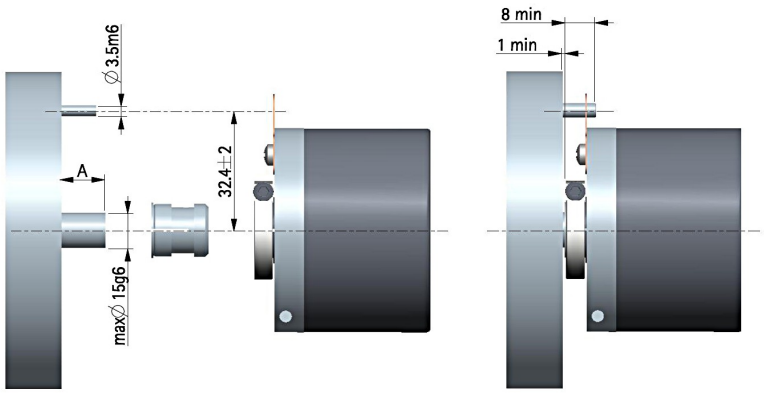




3.2.3 ESC60, EMC60

- 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.





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 ogni informazione sulla gamma disponibile di alberi "personalizzati".

4 Connessioni elettriche

ATTENZIONE
Le connessioni elettriche devono essere eseguite da personale qualificato, in assenza di tensione e componenti meccaniche in movimento.

Per ogni informazione sulle caratteristiche meccaniche ed elettriche dell'encoder riferirsi al catalogo tecnico del prodotto.

4.1 Cavo CB

Colore	Funzione
Rosso	+10Vdc +30Vdc Alimentazione
Nero[1]	0Vdc Alimentazione
Bianco	Modbus A (RS-485)
Blu	Modbus B (RS-485)

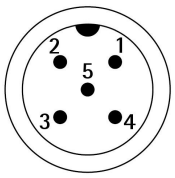
1 0Vdc anche della seriale RS-485.

4.1.1 Caratteristiche del cavo CB

Modello : cavo LIKA CB
Conduttori : due coppie 2 x 0,24 mm^2 (24/19AWG)+ 2 x 0,35 mm^2 (22/19AWG)
Guaina : PUR ritardante la fiamma privo di alogeni
Schermo : a treccia di rame stagnato copertura nom. 65%
Diametro esterno : 6,9 mm ±0,2 mm
Raggio di curvatura : Ø esterno x 6 (fisso); Ø esterno x 12 (dinamico)
Temperatura di lavoro : -40°C +80°C (fisso); -30°C +70°C (dinamico)
Resistenza elettrica : 78,0 Ω/km (24AWG) / ≤ 54,0 Ω/km (22AWG)
Velocità massima : 3,0 m/sec
Accelerazione massima : 5,0 m/sec^2

4.2 Connettore M12 5 pin

Connettore M12
5 pin maschio
codifica A
(vista lato contatti)

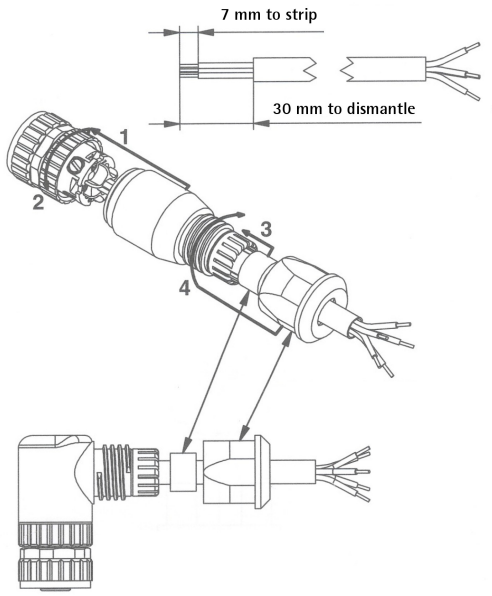


Pin	Funzione
1[1]	Shielding
2	+10Vdc +30Vdc Alimentazione
3[2]	0Vdc Alimentazione
4	Modbus A (RS-485)
5	Modbus B (RS-485)
Calza[3]	Custodia

1 Il piedino 1 permette il collegamento a terra della calza anche nel caso di connettore volante con case plastico.
2 0Vdc anche della seriale RS-485.
3 Solo con cavi intestati EC- Lika Electronic.

4.3 Collegamento messa a terra

Collegare il corpo del dispositivo e/o la calza del cavo e/o la custodia del connettore a un buon punto di terra. Collegare la calza del cavo sul lato utilizzatore. I cavi intestati 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 E- di Lika Electronic utilizzano un connettore plastico; pertanto non è possibile la raccolta calza. Tuttavia, nel caso del connettore M12, è possibile il collegamento a terra della calza mediante il pin 1 predisposto allo scopo. 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.



4.4 Indicatori LED (Figura 1)

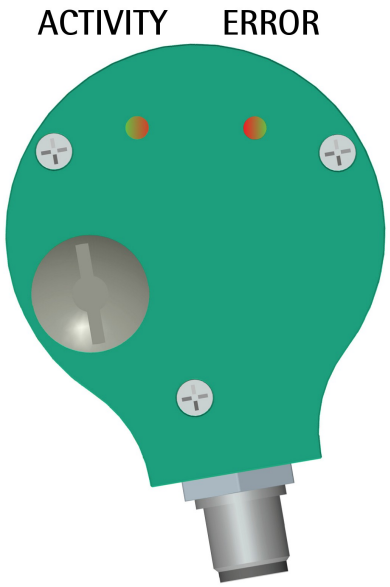


Figura 1: Indicatori LED

Due LED bicolori, posizionati nella parte posteriore dell'encoder (si veda la Figura qui sopra), segnalano visivamente la condizione di funzionamento del

dispositivo e dell'interfaccia MODBUS, come esplicitato nella seguente tabella. In caso di errore, per conoscere nei dettagli l'allarme generato, si veda la variabile **Registro allarmi [0000 hex]** a pagina 76.

LED ACTIVITY	Descrizione
VERDE Lampeggiante	Dispositivo in trasmissione o ricezione.
OFF	Nessuna trasmissione o ricezione attiva.

LED STATUS/ERROR	Descrizione
VERDE ON	Nessun allarme presente.
ROSSO Lampeggiante	Sono presenti allarmi, errore interno. Per maggiori informazioni consultare il **Registro allarmi [0000 hex]** a pagina 76.
ROSSO ON	Si è verificato un errore hardware che pregiudica il funzionamento del dispositivo. Spegnere e riaccendere il dispositivo. Se il problema permane, contattare il Servizio di Assistenza Tecnica di Lika Electronic.

Durante l'esecuzione dell'operazione di aggiornamento del firmware (bootloading; si veda la sezione "5.4 Pagina Aggiorna FW - Upgrade del firmware" a pagina 45), i due LED prevedono un funzionamento specifico, come spiegato nella seguente tabella.

LED ACTIVITY	LED STATUS/ERROR	Descrizione
VERDE Lampeggiante alla frequenza di 5 Hz con duty cycle = 50%	**VERDE** Lampeggiante alla frequenza di 5 Hz con duty cycle = 50%	Durante il trasferimento dei dati alla memoria flash per l'aggiornamento del firmware del dispositivo (si veda la sezione "5.4 Pagina Aggiorna FW - Upgrade del firmware" a pagina 45), i due led lampeggiano verdi a una frequenza di 5 Hz. con duty cycle = 50%
ROSSO Lampeggiante alla frequenza di 2 Hz con duty cycle = 50%	**ROSSO** Lampeggiante alla frequenza di 2 Hz con duty cycle = 50%	L'operatore ha premuto il pulsante **BOOT STATE** nella pagina del programma di upgrade del firmware, l'encoder è in attesa che abbia inizio l'operazione di aggiornamento (pressione del pulsante **DOWNLOAD**). Per ogni informazione riferirsi alla sezione "5.4 Pagina Aggiorna FW - Upgrade del firmware" a pagina 45.

LED ACTIVITY	LED STATUS/ERROR	Descrizione
ROSSO Lampeggiante alla frequenza di 5 Hz con duty cycle = 50%	ROSSO Lampeggiante alla frequenza di 5 Hz con duty cycle = 50%	Durante il trasferimento dei dati alla memoria flash per l'aggiornamento del firmware del dispositivo (si veda la sezione "5.4 Pagina Aggiorna FW - Upgrade del firmware" a pagina 45), se si verifica un errore che pregiudica l'upgrade (per esempio: la caduta di tensione e lo spegnimento dell'unità), l'assenza dello user program all'accensione procura l'attivazione dei due led che lampeggiano rossi a una frequenza di 5 Hz con duty cycle = 50%. Per ogni informazione sul ripristino dell'unità riferirsi alla sezione "5.4 Pagina Aggiorna FW - Upgrade del firmware" a pagina 45.
ROSSO ON	ROSSO ON	Durante il trasferimento dei dati alla memoria flash per l'aggiornamento del firmware del dispositivo (si veda la sezione "5.4 Pagina Aggiorna FW - Upgrade del firmware" a pagina 45), la mancata ricezione di record per un tempo superiore a 5 secondi (per esempio, a causa della disconnessione del cavo seriale) procura l'accensione dei due led rossi fissi. Per ogni informazione sul ripristino dell'unità riferirsi alla sezione "5.4 Pagina Aggiorna FW - Upgrade del firmware" a pagina 45.
OFF	VERDE ON	L'operazione di aggiornamento del firmware è stata portata a termine con successo, l'encoder è ora tornato regolarmente al normale funzionamento e non ci sono allarmi attivi. Per ogni informazione riferirsi alla sezione "5.4 Pagina Aggiorna FW - Upgrade del firmware" a pagina 45.

Durante l'inizializzazione del dispositivo, il sistema esegue un controllo sul corretto funzionamento degli indicatori led; pertanto gli indicatori led lampeggiano per un istante.

4.5 Selettori (Figura 2 e Figura 3)

ATTENZIONE
Questa operazione deve essere eseguita con dispositivo non alimentato!

NOTA
Eseguire questa operazione con estrema prudenza per non danneggiare i componenti e i circuiti interni.

Per accedere ai selettori svitare e rimuovere il tappo metallico M12 nella parte posteriore dell'encoder. Avere cura di ripristinare il tappo al termine dell'operazione.

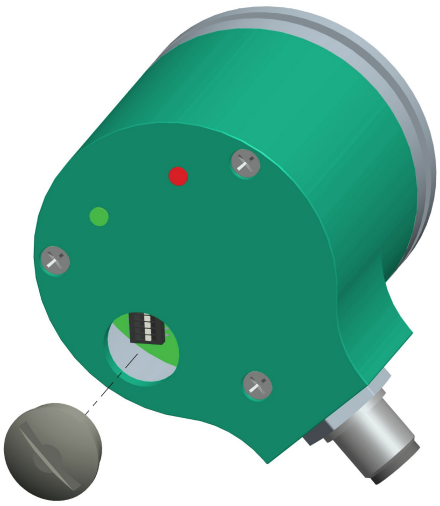


Figura 2: Accesso ai selettori

Accedere quindi ai sottostanti selettori.

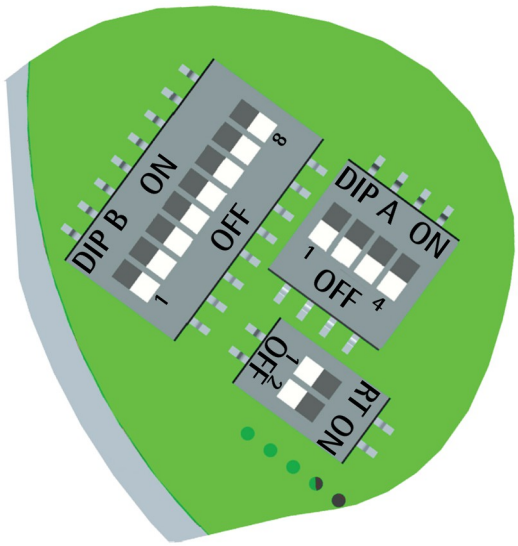


Figura 3: Selettori DIP switch

4.5.1 Velocità di trasmissione dei dati: baud rate e bit di parità (Figura 3)

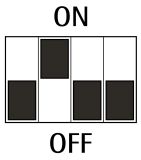
ATTENZIONE
Questa operazione deve essere eseguita con dispositivo non alimentato!

La velocità di trasmissione dei dati (baud rate e bit di parità) è impostata mediante il DIP switch A.
Impostare il valore binario della velocità di trasmissione dei dati secondo la seguente tabella, considerando che: ON = 1; OFF = 0.

Switch	Baud rate	Bit di parità
0000	9600 bit/s	No Parity
1000	9600 bit/s	Even
0100	9600 bit/s	Odd
1100	19200 bit/s	No Parity
0010 (default)	**19200 bit/s**	**Even**
1010	19200 bit/s	Odd
0110	115200 bit/s	No Parity
1110	115200 bit/s	Even
0001	115200 bit/s	Odd

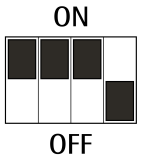
ESEMPIO
Impostare il baud rate a 9600 bit/s, bit di parità Odd:

Switch	1	2	3	4
Posizione	OFF	ON	OFF	OFF
Valore	0	1	0	0



Impostare il baud rate a 115200 bit/s, bit di parità Even:

Switch	1	2	3	4
Posizione	ON	ON	ON	OFF
Valore	1	1	1	0

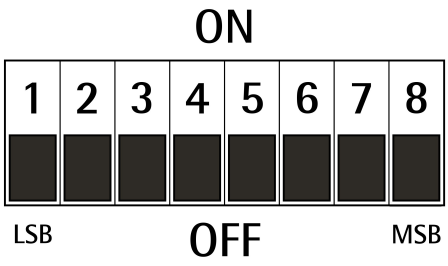


La velocità di trasmissione dei dati correntemente impostata nel dispositivo può essere letta al registro **DIP switch baud rate [0006 hex]**, si veda a pagina 78.

4.5.2 Indirizzo nodo (Figura 3)

ATTENZIONE
Questa impostazione deve essere eseguita con dispositivo non alimentato!

L'indirizzo del nodo è impostato mediante il DIP switch B.



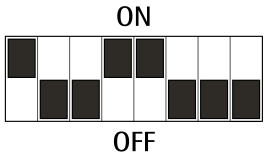
Impostare il valore binario dell'indirizzo del nodo considerando che ON=1, OFF=0.

bit	1 LSB	2	3	4	5	6	7	8 MSB
	2^0	2^1	2^2	2^3	2^4	2^5	2^6	2^7

L'indirizzo deve avere un valore compreso tra 1 e 247. L'indirizzo di default è 1.

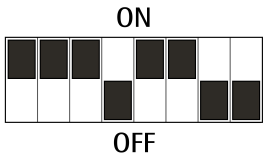
ESEMPIO
Impostare l'indirizzo 25:
25_{10} = 0001 1001$_2$ (valore binario)

Switch	1	2	3	4	5	6	7	8
Posizione	ON	OFF	OFF	ON	ON	OFF	OFF	OFF
Valore	1	0	0	1	1	0	0	0



Impostare l'indirizzo 55:
55_{10} = 0011 0111$_2$ (valore binario)

Switch	1	2	3	4	5	6	7	8
Posizione	ON	ON	ON	OFF	ON	ON	OFF	OFF
Valore	1	1	1	0	1	1	0	0

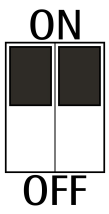
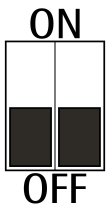


NOTA
L'indirizzo di default è 1.
L'indirizzo 0 è riservato per identificare uno scambio dati in modalità "broadcast" (il Master invia una richiesta contemporaneamente a tutti i nodi connessi nella rete Modbus). Si veda alla sezione "6.1 Principi guida del protocollo MODBUS Master / Slave" a pagina 50.
Il Master Modbus non ha un indirizzo specifico, solo i nodi Slave sono tenuti ad avere un indirizzo. L'indirizzo degli Slave deve essere univoco.
Gli indirizzi da 248 a 255 sono riservati.
Se si imposta l'indirizzo 0, il dispositivo utilizzerà automaticamente l'indirizzo 1.
Se si imposta un indirizzo maggiore di 247, il dispositivo utilizzerà automaticamente l'indirizzo 247.
L'indirizzo del nodo correntemente impostato nel dispositivo può essere letto al registro **DIP switch indirizzo [0007 hex]**, si veda a pagina 79.

4.5.3 Resistenza di terminazione (Figura 3)

ATTENZIONE
Questa impostazione deve essere eseguita con dispositivo non alimentato!

La resistenza di terminazione è impostata mediante il DIP switch RT. Essa deve essere utilizzata come elemento di terminazione del bus se il dispositivo è egli estremi della linea di trasmissione, ovverosia, se è il primo o l'ultimo della rete.

RT	Descrizione
1 = 2 = ON 	Attiva: se il dispositivo è il primo o l'ultimo della linea di trasmissione
1 = 2 = OFF 	Disattiva: se il dispositivo non è né il primo né l'ultimo della linea di trasmissione

5 Quick reference

5.1 Getting started

Le istruzioni che seguono sono fornite per permettere un set up rapido dell'unità in una modalità di funzionamento standard.

- Procedere all'installazione meccanica, si veda a pagina 16 segg.;
- effettuare le connessioni elettriche, si veda a pagina 21 segg.;
- impostare la velocità di trasmissione dei dati (baud rate e bit di parità, si veda la sezione "4.5.1 Velocità di trasmissione dei dati: baud rate e bit di parità (Figura 3)" a pagina 27); il valore di default preimpostato da Lika Electronic è "baud rate = 19200 bit/s, parità = Even";
- impostare l'indirizzo del nodo (node ID, si veda la sezione "4.5.2 Indirizzo nodo (Figura 3)" a pagina 28); il valore di default preimpostato da Lika Electronic è "1";
- alimentare l'unità con alimentazione +10Vdc ÷ +30Vdc;
- se si desidera utilizzare la risoluzione fisica del dispositivo (ES58: risoluzione monogiro fisica = 4.096 cpr; numero di giri fisici = 1 giro; risoluzione totale fisica = 4.096 info = 12 bit; EM58: risoluzione monogiro fisica = 4.096 cpr; numero di giri fisici = 16.384 giri; risoluzione totale fisica = 67.108.864 info = 26 bit), assicurarsi che la **Funzione di scaling** sia disabilitata (bit 0 del registro **Parametri operativi [0008 hex]** = 0; si veda a pagina 71);
- se invece si desidera una risoluzione specifica, abilitare la **Funzione di scaling** (bit 0 del registro **Parametri operativi [0008 hex]** = 1);
- impostare quindi il valore desiderato per la risoluzione monogiro in **Informazioni per giro programmate [0000-0001 hex]** (registri 1 e 2; si veda a pagina 65);
- impostare poi la risoluzione totale desiderata in **Risoluzione totale programmata [0002-0003 hex]** (registri 3 e 4; si veda a pagina 67);
- eventualmente impostare il preset al parametro **Valore di preset [0004-0005 hex]** e poi eseguire il comando **Esegui preset conteggio** bit 11 in **Control Word [0009 hex]**; si veda a pagina 74;
- procedere al salvataggio dei nuovi parametri (registro **Salva parametri** bit 9 in **Control Word [0009 hex]**; si veda a pagina 73).

5.2 Configurazione mediante software di Lika Electronic

Gli encoder rotativi della serie ES58 / EM58 sono forniti con un software sviluppato da Lika Electronic per la programmazione semplificata del dispositivo. Il programma permette di impostare i parametri di lavoro e testare manualmente e monitorare il funzionamento dei dispositivi. Il software è fornito gratuitamente e può essere installato in qualsiasi PC con sistema operativo Windows (Windows XP o successivo). Il file di esecuzione del programma **MODBUS-RTU.EXE** è disponibile al link SOFTWARE nella pagina del sito dedicata ai dispositivi. Il programma non richiede installazione; per lanciarlo è sufficiente fare un doppio click sull'icona del file. Per chiudere poi il programma, premere la X in alto a destra nella finestra.

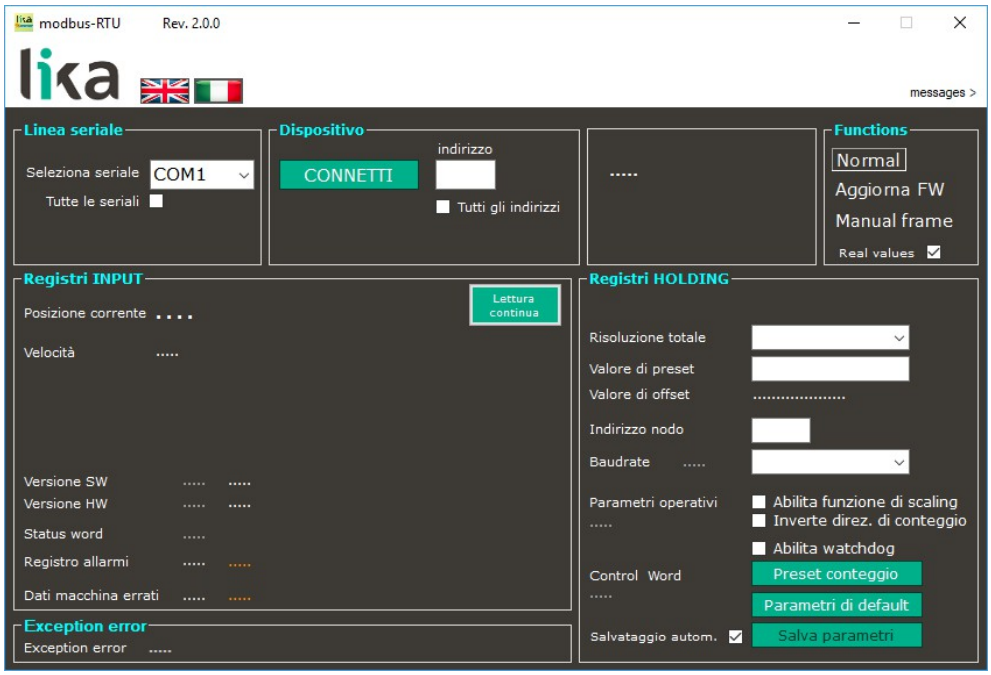
NOTA
Prima di connettersi al dispositivo è necessario collegarlo serialmente al personal computer. L'interfaccia seriale degli encoder della serie ES58 / EM58 è del tipo RS-485, mentre lo standard seriale dei personal computer (ove la porta sia prevista) è del tipo RS-232. Bisogna quindi prevedere l'utilizzo di un convertitore RS-485 / RS-232, facilmente reperibile in commercio. Nel caso in cui il personal computer non sia provvisto di porta seriale (RS-232 o RS-485), sarà necessario installare un convertitore USB / RS-485, anch'esso facilmente reperibile in commercio. Per ogni informazione sullo schema di collegamento e il pinout del cavo riferirsi al foglio informativo del convertitore utilizzato.
Sul lato ENCODER il cavo deve essere cablato come descritto nella sezione "Connessioni elettriche" a pagina 21.
Assicurarsi sempre che RX dell'ENCODER MODBUS sia collegato con TX del PC e che RX del PC sia collegato con TX dell'ENCODER MODBUS.

NOTA
E' disponibile su richiesta un cavo adattatore con convertitore RS-485 precablato con M12 5 poli / USB.
Il codice è il seguente:

- **KIT EM58 MB** Adattatore M12 / USB.

5.3 Pagina principale dell'interfaccia

Per lanciare il programma di configurazione dell'encoder MODBUS fare doppio click sull'eseguibile **MODBUS-RTU.EXE**.
L'interfaccia consta di una pagina principale e di due pagine secondarie.
All'avvio del programma, si visualizza la pagina principale.



La pagina principale dell'interfaccia si può dividere in sette parti.

1. Nel primo group box **LINEA SERIALE** in alto a sinistra sono disponibili i campi dedicati alla selezione della porta seriale per il collegamento con il dispositivo.
2. Nel riquadro **DISPOSITIVO** in alto al centro sono disponibili i campi dedicati all'impostazione dell'indirizzo del nodo, i comandi di ricerca del dispositivo e, una volta collegati, di lettura e scrittura dei parametri.
3. Nel riquadro appena più a destra sono visualizzate le informazioni sul dispositivo collegato, una volta che la connessione è stata realizzata.
4. Nel group box **FUNCTIONS** in alto a destra sono disponibili i comandi visualizzazione della schermata standard, di accesso alle finestre di aggiornamento del firmware e di immissione manuale dei byte da trasmettere al dispositivo (invio manuale delle Request PDU).
5. Nella riquadro **REGISTRI INPUT** in basso a sinistra sono disponibili gli Input Register, ossia i valori di risultato e gli stati / allarmi relativi al dispositivo. Questi elementi sono descritti nella sezione "7.1.2 Parametri Input Register" a pagina 76.

6. Appena più in basso, nel riquadro **EXCEPTION ERROR**, sono visualizzate le exception response inviate in risposta a errori di comunicazione o altro. Per maggiori informazioni sulle exception response riferirsi alla sezione "7.2 Codici eccezione" a pagina 81.
7. Nel riquadro **REGISTRI HOLDING** in basso a destra sono disponibili gli Holding Register; in quest'area è possibile la scrittura o la lettura dei valori impostati nei parametri di lavoro del dispositivo collegato.

E' possibile la scelta della lingua di visualizzazione dei testi. Cliccare sul pulsante **Bandiera italiana** per scegliere la lingua di visualizzazione italiana; cliccare sul pulsante **Bandiera inglese** per scegliere la lingua di visualizzazione inglese.

In alto a destra sopra il riquadro **FUNCTIONS** è disponibile il pulsante **MESSAGES >**. Premendo questo pulsante si ingrandisce la schermata principale e nella finestra a fianco vengono visualizzati i frame di comunicazione tra dispositivo e tool software. Per chiudere la finestra dei messaggi e ritornare alla visualizzazione normale della pagina principale premere il pulsante **< CLOSE**.

5.3.1 Configurazione porta seriale – Collegamento con il dispositivo

I primi quattro riquadri in alto sono tutti utilizzati al momento del collegamento con il dispositivo tramite la porta seriale per (in sequenza):

- scegliere la porta seriale cui abbiamo fisicamente collegato l'encoder (nel group box **LINEA SERIALE**);
- impostare l'indirizzo del nodo da ricercare e avviare la ricerca del dispositivo collegato (nel group box **DISPOSITIVO**);
- visualizzare i dati relativi al dispositivo collegato, una volta connesso;
- scegliere la pagina visualizzata e attivare la visualizzazione del valore dei byte (se il check box **Real values** è selezionato).

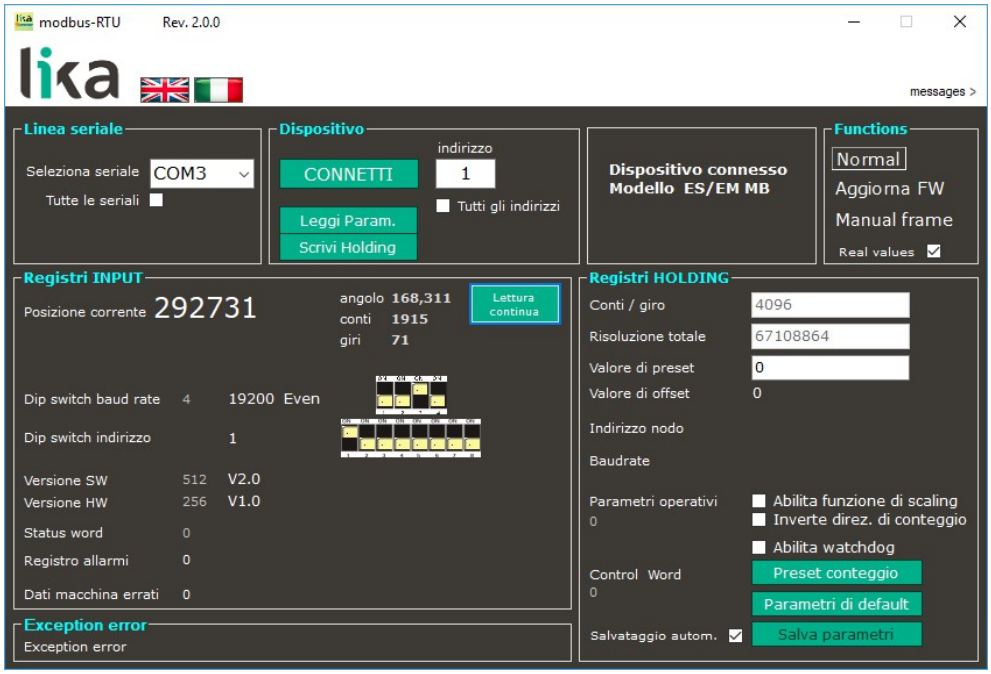
All'apertura della pagina dell'interfaccia, tramite il menu a tendina del campo **Seleziona seriale** nel group box **LINEA SERIALE** si sceglie la porta seriale cui è collegato il dispositivo.
Qualora non sia noto il numero della porta COM cui è stato collegato il dispositivo selezionare il check box **Tutte le seriali**.

Nel campo **Indirizzo** nel group box **DISPOSITIVO** si imposta invece l'indirizzo MODBUS del dispositivo collegato. L'indirizzo di default dei dispositivi Lika è "1".
Qualora non sia noto l'indirizzo MODBUS del dispositivo selezionare il check box **Tutti gli indirizzi**.

Selezionando i check box **Tutte le seriali** e **Tutti gli indirizzi** il tool andrà a interrogare tutte le porte seriali disponibili (COM1 prima, poi COM2, COM3, ecc., se presenti) testando per ciascuna le varie opzioni di baud rate e bit di parità (9600 No parity, 9600 Even, 9600 Odd, ecc.) e tutti gli indirizzi possibili (da 1 a 247, secondo il range MODBUS).
E' evidente che questa operazione di ricerca potrebbe richiedere anche del tempo.

Per avviare la ricerca e il collegamento al dispositivo premere il pulsante **CONNETTI** nel group box **DISPOSITIVO**. Durante il tentativo di connessione una barra di avanzamento sopra il pulsante indica che è attiva la ricerca del dispositivo; una barra di avanzamento e la label **Busy** sopra il menu a tendina del campo **Seleziona seriale** indicano che la porta è aperta e viene testata. Con dispositivo collegato, questa barra indica che la porta è aperta. In assenza di comunicazione la porta viene lasciata disponibile per altre applicazioni dopo un timeout di circa 1 secondo.

Qualora vi fossero problemi nel collegamento (porta seriale selezionata errata o occupata, indirizzo nodo impostato errato, ecc.), l'interfaccia eseguirà indefinitamente dei tentativi di connessione, fino alla nuova pressione del pulsante **CONNETTI**. Durante la ricerca a fianco del pulsante compare il messaggio **No device** e l'indicazione dell'indirizzo e della porta testati.



Se il collegamento ha esito positivo, nel group box compreso tra **DISPOSITIVO** e **FUNCTIONS** appaiono il messaggio **Dispositivo connesso** e il modello del dispositivo individuato. Il software infatti è in grado di riconoscere automaticamente il tipo di dispositivo collegato modificando di conseguenza la pagina visualizzata e la lista dei registri.

Nei group box **REGISTRI INPUT** e **REGISTRI HOLDING** vengono poi visualizzati tutti i valori dei registri dell'encoder.
Nel riquadro **DISPOSITIVO** appaiono inoltre i pulsanti **Leggi param.** e **Scrivi Holding**.

Leggi param.
Con la pressione del pulsante **Leggi param.** si invia un singolo comando di lettura degli Input Register e degli Holding Register. Il valore dei registri Input e Holding elencati nella pagina viene aggiornato a quello letto nel dispositivo al momento di invio della richiesta (lettura istantanea). Per una lettura continua dei registri occorre invece utilizzare il pulsante **Lettura continua**, si veda a pagina 38.

Scrivi Holding
La pressione di questo pulsante procura l'invio del comando di scrittura dei valori di tutti gli Holding Register. E' anche possibile premere il tasto **ENTER** nella tastiera del personal computer che però invia il comando di scrittura del

solo registro in cui è posizionato il cursore. Dopo aver impostato un nuovo valore in uno qualsiasi degli Holding Register, premere il pulsante **Scrivi Holding** per inviare all'encoder i dati di tutti gli Holding Register; oppure premere il tasto **ENTER** per inviare il solo dato dell'Holding Register dove è posizionato il cursore.

All'interno del group box **FUNCTIONS** sono disponibili i pulsanti **AGGIORNA FW** e **MANUAL FRAME**. Il primo permette di accedere alla finestra di aggiornamento del firmware; il secondo alla finestra di invio manuale delle PDU. Per ogni informazione sulla procedura di aggiornamento del firmware riferirsi alla sezione "5.4 Pagina Aggiorna FW - Upgrade del firmware" a pagina 45; per ogni informazione sull'invio manuale delle PDU riferirsi alla sezione "5.5 Pagina Manual frame – Scrittura manuale PDU" a pagina 48. Premendo il pulsante **NORMAL** si ritorna alla schermata principale.

5.3.2 Lettura degli Input Register – Registri INPUT

Nella riquadro più grande a sinistra nell'interfaccia di programmazione sono disponibili gli Input Register, ossia i valori di risultato e gli stati / allarmi relativi

al dispositivo. Questi elementi sono descritti nella sezione "7.1.2 Parametri Input Register" a pagina 76 di questo manuale.

In questo group box sono disponibili gli elementi elencati qui di seguito.

Lettura continua
Premendo il pulsante **Lettura continua** si attiva la modalità di lettura continua degli Input Register. Dopo la pressione il pulsante diventa arancione e una barra di avanzamento compare sotto il campo a indicare la condizione di lettura. Una barra di avanzamento e la label **Busy** sopra il menu a tendina del campo **Seleziona seriale** indicano che la porta seriale è aperta. Inoltre i campi dei group box **FUNCTIONS** e **REGISTRI HOLDING** sono resi indisponibili. I valori degli Input Register elencati nella pagina sono aggiornati in un ciclo continuo. Per arrestare la lettura continua premere nuovamente il pulsante **Lettura continua**. Per l'invio di un singolo comando di lettura dei registri, premere il pulsante **Leggi param.** nel group box **DISPOSITIVO**, si veda a pagina 36.

Posizione corrente
Visualizza la posizione corrente del dispositivo espressa in numero di informazioni. In presenza di errori il valore di posizione non viene visualizzato. Si veda il registro Posizione corrente [0001-0002 hex] a pagina 77.

Angolo
Visualizza la posizione angolare dell'albero espressa in gradi. In presenza di errori il valore di posizione non viene visualizzato.

Conti
Visualizza la posizione angolare dell'albero espressa in conti (numero di informazioni sul giro). In presenza di errori il valore di posizione non viene visualizzato.

Giri
Visualizza il numero di giri misurati (**Posizione corrente** / **Conti / giro**). In presenza di errori il numero di giri non viene visualizzato.

Dip switch baud rate
Visualizza il valore della velocità di trasmissione dei dati (baud rate e valore del bit di parità) correntemente impostato tramite il DIP switch A (come valore decimale se il check box **Real values** è selezionato -4 nella Figura = 0010hex-,

come stringa -19200 Even nella Figura- e graficamente). Si veda il registro **DIP switch baud rate [0006 hex]** a pagina 78. Per l'impostazione del baud rate riferirsi alla sezione "4.5.1 Velocità di trasmissione dei dati: baud rate e bit di parità (Figura 3)" a pagina 27.

Dip switch indirizzo
Visualizza l'indirizzo del nodo correntemente impostato tramite il DIP switch B (come valore decimale se il check box **Real values** è selezionato -1 nella Figura- e graficamente). Si veda il registro **DIP switch indirizzo [0007 hex]** a pagina 79. Per l'impostazione dell'indirizzo del nodo riferirsi alla sezione "4.5.2 Indirizzo nodo (Figura 3)" a pagina 28.

Versione SW
Visualizza la versione del software installato (come valore decimale se il check box **Real values** è selezionato -512 nella Figura- e come stringa -V2.0 nella Figura). Si veda il registro **Versione software [0008 hex]** a pagina 79.

Versione HW
Visualizza la versione dell'hardware (versione PCB) installato (come valore decimale se il check box **Real values** è selezionato -256 nella Figura- e come stringa -V1.0 nella Figura). Si veda il registro **Versione hardware [0009 hex]** a pagina 79.

Status word
Se il check box **Real values** è selezionato visualizza il valore decimale corrente -257 nella Figura- del registro **Status word [000A hex]** cui si rimanda (si veda a pagina 80). In presenza di allarmi, compare anche il messaggio **Allarme** e i valori di posizione e del numero di giri non vengono visualizzati. Un messaggio in giallo avvisa anche se la funzione di scaling e la direzione di conteggio sono abilitate.

Registro allarmi
Se il check box **Real values** è selezionato visualizza il valore decimale corrente -1 nella Figura- del **Registro allarmi [0000 hex]** cui si rimanda (si veda a pagina 76). In presenza di allarmi, a fianco viene visualizzato anche lo specifico allarme attivo -**Err. DM** nella Figura- e i valori di posizione e del numero di giri non vengono visualizzati.

Dati macchina errati
Se il check box **Real values** è selezionato visualizza il valore decimale corrente -2 nella Figura- del registro **Elenco DM errati [0004-0005 hex]** cui si rimanda (si veda a pagina 77). In presenza di dati macchina errati, a fianco viene visualizzato anche lo specifico parametro errato -**Info / giro** nella Figura- e i valori di posizione e del numero di giri non vengono visualizzati. Si attivano inoltre i registri **Status word** e **Registro allarmi**. Lo sfondo del campo dello specifico parametro errato nell'area degli Holding Register dell'interfaccia viene evidenziato in rosso.

5.3.3 Lettura delle exception response – Exception error

Nel riquadro **EXCEPTION ERROR** posto sotto il group box **REGISTRI INPUT**, sono visualizzate le exception response inviate in risposta a errori di comunicazione o altro.

Exception error
Visualizza i messaggi di exception response che il Server invia al Client in presenza di errori dovuti all'impossibilità da parte del Server di eseguire la richiesta del Client (per esempio, nel caso di scrittura di valori non ammessi oppure nel caso in cui si trattasse di una richiesta di lettura di un'uscita o di un registro inesistenti). Per maggiori informazioni sulle exception response e sui codici eccezione MODBUS (exception code) riferirsi alla sezione "7.2 Codici eccezione" a pagina 81.

5.3.4 Lettura / scrittura degli Holding Register – Registri HOLDING

Nella riquadro più grande a destra nell'interfaccia di programmazione sono disponibili gli Holding Register. E' qui possibile la lettura e la scrittura dei valori impostati nei parametri di lavoro del dispositivo, rispettivamente mediante la pressione dei pulsanti **Leggi param.** e **Scrivi Holding** disponibili nel group box **DISPOSITIVO**. Gli Holding Register sono descritti nella sezione "7.1.1 Parametri Dati macchina (Holding register)" a pagina 65 di questo manuale.

ATTENZIONE
Se il check box **Salvataggio autom.** in basso nella pagina è selezionato (si veda a pagina 44), tutte le impostazioni degli holding register sono memorizzate automaticamente e istantaneamente: i check box all'atto della selezione/deselezione, i campi in scrittura alla pressione del pulsante **ENTER** della tastiera oppure posizionando il cursore in qualunque posizione al di fuori del campo dopo l'impostazione.

Se invece il check box **Salvataggio autom.** non è selezionato, occorre premere il pulsante **Salva parametri** per effettuare il salvataggio dopo l'impostazione (si veda a pagina 45).

In questa area sono disponibili gli elementi elencati qui di seguito.

Conti / giro
Permette di impostare una risoluzione monogiro personalizzata del dispositivo e ne visualizza il valore corrente. Il valore impostato è espresso in numero di informazioni. Si veda il registro **Informazioni per giro programmate [0000-0001 hex]** a pagina 65. Per poter modificare la risoluzione monogiro del dispositivo occorre che il check box **Abilita funzione di scaling** in **Parametri operativi** sia selezionato.

Risoluzione totale
Permette di impostare una risoluzione totale personalizzata del dispositivo e ne visualizza il valore corrente. Il valore impostato è espresso in numero di informazioni. Si veda il registro **Risoluzione totale programmata [0002-0003 hex]** a pagina 67. Per poter modificare la risoluzione totale del dispositivo occorre che il check box **Abilita funzione di scaling** in **Parametri operativi** sia selezionato.

Valore di preset
Permette di impostare il valore di preset e ne visualizza il valore corrente. Per eseguire il preset occorre poi premere il pulsante **Preset conteggio** in basso nella sezione **Control word** che esegue l'intera sequenza dei comandi di preset (attivazione del bit **Esegui preset conteggio** e scrittura dei registri; disattivazione del bit **Esegui preset conteggio** e scrittura dei registri). Si veda anche a pagina 44. Per maggiori informazioni riferirsi al registro **Valore di preset [0004-0005 hex]** a pagina 69.

Valore di offset
Visualizza il valore di offset che consegue all'impostazione del **Valore di preset**. Per maggiori informazioni riferirsi al registro **Offset [0006-0007 hex]** a pagina 71.

Indirizzo nodo
Poiché il dispositivo permette l'impostazione dell'indirizzo del nodo solo tramite DIP switch, questo parametro non è utilizzato.

Baud rate
Poiché il dispositivo permette l'impostazione della velocità di trasmissione dei dati solo tramite DIP switch, questo parametro non è utilizzato.

Parametri operativi
Raggruppa le funzioni previste nel registro **Parametri operativi [0008 hex]** (si veda a pagina 71) e ne visualizza lo stato di abilitazione 1/disabilitazione 0 (se il check box **Real values** è selezionato il valore decimale risultante dallo stato di abilitazione 1/disabilitazione 0 compare nel campo sotto la label, 0 nella Figura sopra).

Abilita funzione di scaling
Permette di abilitare 1/disabilitare 0 la funzione di scaling e di visualizzarne lo stato. Utilizzare il check box per abilitare 1/disabilitare 0 la funzione, se il check box **Real values** è selezionato il risultante valore decimale del registro **Parametri operativi [0008 hex]** compare nel campo a sinistra. Con opzione abilitata, il testo della label è riportato in colore giallo. Per maggiori informazioni riferirsi al parametro **Funzione di scaling** a pagina 71.

Inverte direz. di conteggio
Permette di invertire la direzione di conteggio e di visualizzarne l'attuale impostazione. Utilizzare il check box per abilitare 1/disabilitare 0 la funzione, se il check box **Real values** è selezionato il risultante valore decimale del registro **Parametri operativi [0008 hex]** compare nel campo a sinistra. Con opzione abilitata, il testo della label è riportato in colore giallo. Per maggiori informazioni riferirsi al parametro **Direzione di conteggio** a pagina 72.

Control word
Raggruppa le funzioni previste nel registro **Control Word [0009 hex]** e ne visualizza lo stato di abilitazione-attivazione 1/disabilitazione-disattivazione 0. Utilizzare i check box / pulsanti a lato per impostare le funzioni come desiderato, se il check box **Real values** è selezionato il risultante valore decimale compare sotto la label a sinistra (0 nella Figura sopra). Per maggiori informazioni si veda il registro **Control Word [0009 hex]** a pagina 73.

Abilita watchdog
Permette di abilitare 1/disabilitare 0 la funzione di watchdog del protocollo MODBUS e di visualizzarne lo stato di abilitazione. Utilizzare il check box per abilitare 1/disabilitare 0 la funzione, se il check box **Real values** è selezionato il risultante valore decimale compare nel campo a sinistra. Con opzione abilitata, il testo della label è riportato in colore giallo. Per maggiori informazioni riferirsi al parametro **Abilitazione Watchdog** a pagina 73.

Preset conteggio

Questo pulsante invia il comando completo di esecuzione del preset. A seguito dell'invio di questo comando, il valore della posizione corrente dell'encoder assume quello impostato in **Valore di preset** e tutte le altre posizioni precedenti e successive assumono un valore conseguente. Il pulsante **Preset conteggio** esegue l'intera sequenza dei comandi di preset (attivazione del bit **Esegui preset conteggio** e scrittura dei registri; disattivazione del bit **Esegui preset conteggio** e scrittura dei registri; salvataggio dei parametri). Durante l'esecuzione del comando lo sfondo del pulsante diventa arancione e il valore decimale della **Control word** è aggiornato nel campo sotto la label se il check box **Real values** è selezionato. A operazione eseguita il valore della **Posizione corrente** coincide con il valore impostato in **Valore di preset** (non serve premere il pulsante **Leggi param.** per eseguire un refresh della posizione corrente). Per maggiori informazioni riferirsi al parametro **Esegui preset conteggio** a pagina 74.

Parametri di default

Il pulsante **Parametri di default** esegue l'intera sequenza dei comandi di upload dei parametri di default (attivazione del bit **Carica parametri di default** e scrittura dei registri; disattivazione del bit **Carica parametri di default** e scrittura dei registri; salvataggio dei parametri). Durante l'esecuzione del comando lo sfondo del pulsante diventa arancione e il valore decimale della **Control word** è aggiornato nel campo sotto la label se il check box **Real values** è selezionato. A operazione eseguita il valore dei parametri è aggiornato automaticamente (non serve premere il pulsante **Leggi param.** per eseguire un refresh dei valori impostati visualizzati). Per maggiori informazioni riferirsi al parametro **Carica parametri di default** a pagina 74.

ATTENZIONE

Con l'esecuzione di questo comando tutti i valori precedentemente impostati sono sovrascritti!

Salvataggio autom.

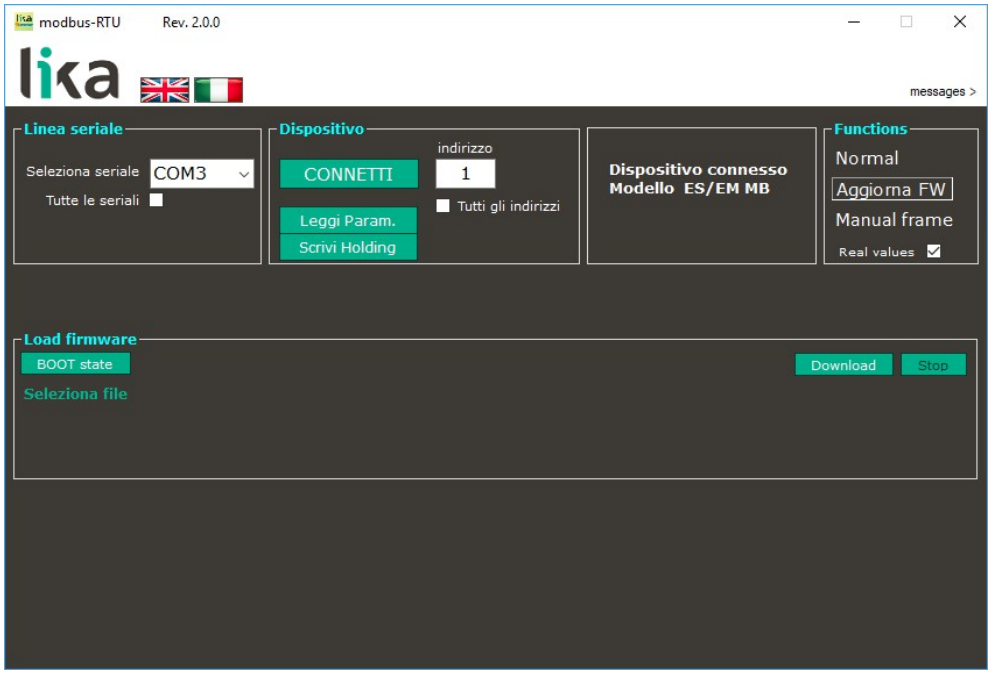
Se il check box **Salvataggio autom.** è selezionato, tutte le impostazioni degli holding register sono memorizzate automaticamente e istantaneamente: i check box all'atto della selezione/deselezione, i campi in scrittura alla pressione del pulsante **ENTER** della tastiera oppure posizionando il cursore in qualunque posizione al di fuori del campo dopo l'impostazione.
Se invece il check box **Salvataggio autom.** non è selezionato, occorre premere il pulsante **Salva parametri** per effettuare il salvataggio dopo l'impostazione.

Salva parametri
Questo pulsante è attivo solo se il check box **Salvataggio autom.** non è selezionato. Esegue l'intera sequenza dei comandi di memorizzazione in maniera permanente dei parametri nella EEPROM (attivazione del bit **Salva parametri** e scrittura dei registri; disattivazione del bit **Salva parametri** e scrittura dei registri). Durante l'esecuzione del comando lo sfondo del pulsante diventa arancione e il valore decimale della **Control word** è aggiornato nel campo sotto la label se il check box **Real values** è selezionato. Per maggiori informazioni si veda **Salva parametri** a pagina 73.

ATTENZIONE
Per attivare la funzione di salvataggio automatico dei parametri selezionare il check box **Salvataggio autom.**. Se il check box è selezionato, tutte le impostazioni sono memorizzate automaticamente e istantaneamente senza che sia mai necessario premere il pulsante **Salva parametri**: i check box all'atto della selezione/deselezione, i campi in scrittura alla pressione del pulsante **ENTER** della tastiera oppure posizionando il cursore in qualunque posizione al di fuori del campo dopo l'impostazione. Per maggiori informazioni si veda alla pagina precedente.

5.4 Pagina Aggiorna FW - Upgrade del firmware

Premendo il pulsante **AGGIORNA FW** nel group box **FUNCTIONS** a destra nella pagina principale dell'interfaccia si accede alla finestra di aggiornamento del firmware del dispositivo.



ATTENZIONE
L'operazione di upgrade del firmware deve essere eseguita da personale qualificato e competente. L'applicazione di un aggiornamento errato o incompatibile pregiudica il funzionamento del dispositivo stesso. E' fondamentale che il processo sia avviato nel rigoroso rispetto delle istruzioni fornite in questa sezione.
Prima di avviare l'installazione assicurarsi sempre che il file di aggiornamento firmware sia compatibile con l'hardware e il software del dispositivo. Non togliere mai tensione durante l'upgrade della flash.

5.4.1 Informazioni sull'upgrade del firmware

Questa operazione permette l'upgrade dell'unità mediante il download dei dati di aggiornamento alla memoria flash.
Il firmware è un programma software che permette la gestione e il controllo del funzionamento di un dispositivo; il programma firmware, talora chiamato anche "user program" o "programma utente", è memorizzato nella memoria flash integrata all'interno dell'unità. Questi encoder sono progettati in modo che il firmware possa essere aggiornato agevolmente e direttamente dall'utente finale. Questo permette a Lika Electronic di rendere disponibili nuovi e più aggiornati firmware durante tutto il corso di vita del prodotto.
Le tipiche motivazioni che procurano il rilascio di un nuovo firmware derivano dalla necessità di correggere, migliorare e talora aggiungere nuove funzionalità al dispositivo.
L'aggiornamento firmware consiste in un file con estensione .BIN da scaricare nel dispositivo mediante le funzioni disponibili in questa pagina. I file sono forniti direttamente dal Servizio di Assistenza Tecnica di Lika Electronic.

Se la versione firmware più recente è già installata nel dispositivo, non è necessario procedere con l'installazione di alcun aggiornamento. La versione firmware correntemente installata può essere verificata tramite il campo **Versione SW** dell'interfaccia (si veda a pagina 39) oppure al registro **Versione software [0008 hex]** (si veda a pagina 79) dopo la corretta connessione al dispositivo.

NOTA
Qualora sussistano dei dubbi sull'aggiornamento del firmware, si prega di contattare il Servizio di Assistenza Tecnica di Lika Electronic.

5.4.2 Operazioni e connessioni preliminari

Prima di procedere con l'upgrade del firmware assicurarsi che siano preventivamente soddisfatti i seguenti requisiti:

- l'encoder è correttamente collegato al PC mediante una port COM seriale RS-485;
- l'interfaccia **Modbus-RTU.exe** è aperta e la connessione è attiva;
- si è in possesso del file .BIN per l'upgrade del firmware.

5.4.3 Esecuzione del processo di upgrade del firmware

Per aggiornare il firmware del dispositivo procedere come segue:

1. aprire la pagina **Load firmware** mediante il pulsante **AGGIORNA FW** nel riquadro **FUNCTIONS**;
2. premere **SELEZIONA FILE**; una volta premuto il pulsante appare sullo schermo la finestra di dialogo **Apri**: accedere alla cartella in cui si trova il file .BIN di upgrade del firmware rilasciato da Lika Electronic, selezionare il file e confermare mediante il pulsante **APRI**;
3. premere il pulsante **BOOT STATE**; se l'encoder è collegato correttamente e il sistema è in grado di accedere con successo allo stato di boot (boot mode), il messaggio **BOOT OK** appare a destra del pulsante e i LED sul corpo dell'encoder iniziano a lampeggiare rossi a una frequenza di 5 Hz con duty cycle = 50%; l'encoder è ora pronto e in attesa che l'operazione di aggiornamento del firmware abbia inizio;

ATTENZIONE
Quando l'encoder è nello stato di boot, è possibile ripristinare lo stato di comunicazione normale semplicemente spegnendo e riaccendendo il dispositivo. Una volta avviato il download, invece, qualunque azione (pressione del pulsante **STOP**, evento inatteso, ecc.) richiederà non solo lo spegnimento e la riaccensione del dispositivo, ma anche necessariamente il download del firmware.

ATTENZIONE
Prima di avviare l'installazione assicurarsi sempre che il file di aggiornamento firmware sia compatibile con l'hardware e il software del dispositivo. Non togliere mai tensione durante l'upgrade della flash.

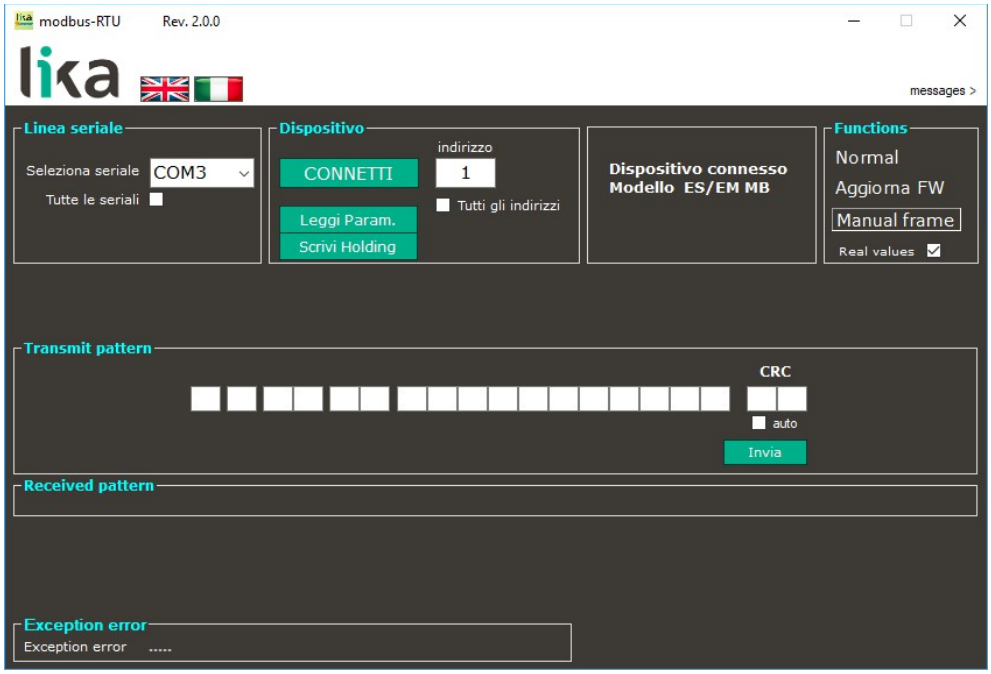
4. premere il pulsante **DOWNLOAD** per avviare il processo di upgrade del firmware; i due LED sul corpo dell'encoder iniziano a lampeggiare verdi a una frequenza di 5 Hz con duty cycle = 50% e nella pagina compare il messaggio **DOWNLOADING**;
5. non appena l'operazione si conclude positivamente, sullo schermo appare il messaggio **DOWNLOADED**;

6. togliere e ridare tensione all'encoder per completare l'operazione.

NOTA
Durante il download dell'aggiornamento firmware potrebbe essere necessario premere il pulsante **STOP** che arresta il processo di upgrade; oppure potrebbero verificarsi delle condizioni inattese che pregiudicano l'esito positivo del processo. In questo caso il download non può essere completato e la condizione di fault è segnalata mediante l'accensione dei LED sul corpo dell'encoder che iniziano a lampeggiare rossi a una frequenza di 5 Hz oppure si accendono fissi rossi; in questo caso è necessario togliere e ridare tensione per resettare il dispositivo e poter riavviare l'operazione.

5.5 Pagina Manual frame – Scrittura manuale PDU

Premendo il pulsante **MANUAL FRAME** nel group box **FUNCTIONS** a destra nella pagina principale dell'interfaccia si accede alla finestra di scrittura e invio manuali di Request PDU al dispositivo. Per alcuni esempi di scrittura di Request PDU e i relativi messaggi Response PDU si rimanda alla sezione "Esempi di programmazione" a pagina 85.



Per la trasmissione manuale di una Request PDU procedere come segue:

1. introdurre il messaggio PDU in notazione esadecimale nei campi a disposizione in **Transmit pattern**; per passare da un campo all'altro si può utilizzare il tasto TAB della tastiera;
2. introdurre il valore Cyclical Redundancy Check (CRC) nei due ultimi campi a destra;
3. selezionando il check box **CRC auto** è possibile evitare di introdurre manualmente il CRC che sarà calcolato automaticamente dal programma al momento dell'invio del messaggio;
4. premere il pulsante **INVIA** per trasmettere il messaggio Request PDU.

Nel campo **Received pattern** sarà visualizzata la Response PDU in formato esadecimale inviata dal Server.

In caso di impossibilità di ricezione della Response PDU, in corrispondenza del campo **Received pattern** compare il messaggio di errore **No pattern**.

In presenza di errori dovuti all'impossibilità da parte del Server di eseguire la richiesta del Client (per esempio, nel caso di scrittura di valori non ammessi oppure nel caso in cui si trattasse di una richiesta di lettura di un'uscita o di un registro inesistenti) i messaggi di exception response che il Server invia al Client sono visualizzati in corrispondenza del group box **Exception error** in basso nella pagina. Per maggiori informazioni sulle exception response e sui codici eccezione MODBUS (exception code) riferirsi alla sezione "7.2 Codici eccezione" a pagina 81.

6 Interfaccia MODBUS®

Gli encoder della serie EMx58 MODBUS di Lika sono dispositivi Slave e implementano il protocollo applicativo MODBUS (livello OSI 7) e il protocollo "Modbus over Serial Line" (livelli OSI 1 & 2).
Per ogni informazione e specifica omessa fare riferimento ai documenti "Modbus Application Protocol Specification V1.1b" e "Modbus over Serial Line. Specification and Implementation Guide V1.02" disponibili sul sito www.modbus.org.

6.1 Principi guida del protocollo MODBUS Master / Slave

Il protocollo seriale Modbus è un protocollo Master – Slave. Un solo Master può essere connesso contemporaneamente alla rete Modbus; mentre il numero di Slave connessi allo stesso bus seriale può essere compreso tra 1 e 247. Una comunicazione Modbus è sempre iniziata dal Master. I nodi Slave non sono abilitati alla trasmissione di dati se non a seguito di una richiesta da parte del nodo Master. Inoltre i nodi Slave non solo abilitati a comunicare l'uno con l'altro. Il Master può attivare una sola transazione Modbus per volta.

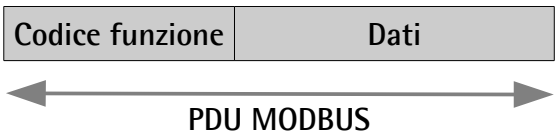
Il nodo Master può inviare una richiesta Modbus ai nodi Slave in due modalità:

- **modalità UNICAST**: il Master invia la richiesta a un singolo Slave. Dopo aver ricevuto e processato la richiesta, lo Slave invia una risposta al Master. In questa modalità, la transazione Modbus consiste di due messaggi: una richiesta da parte del Master e una risposta da parte dello Slave. Ogni Slave deve avere il proprio indirizzo (da 1 a 247) di modo che la richiesta possa essere inviata specificatamente. Nei dispositivi Lika sono previsti comandi solo in modalità "unicast".
- **modalità BROADCAST**: il Master può inviare una richiesta a tutti gli Slave contemporaneamente. Gli Slave non inviano nessuna risposta a seguito di una richiesta di tipo "broadcast". Ne consegue che le richieste di tipo "broadcast" sono necessariamente dei comandi di scrittura. L'indirizzo 0 è riservato per identificare uno scambio dati in modalità "broadcast". Nei dispositivi Lika non sono previsti comandi in modalità "broadcast".

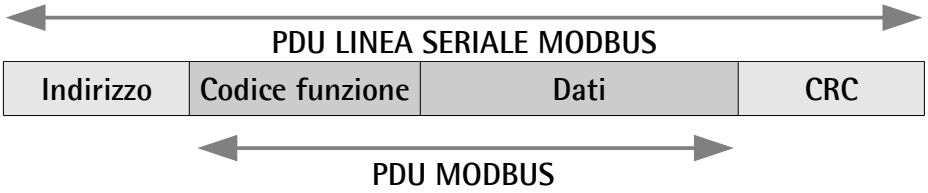
NOTA
Nei dispositivi Lika non sono previsti comandi in modalità "broadcast".

6.2 Frame MODBUS

Il protocollo applicativo Modbus definisce una semplice Protocol Data Unit (PDU) indipendentemente dal livello di comunicazione:



La mappatura del protocollo Modbus in specifici bus o reti introduce ulteriori campi nella PDU. Il client che avvia una transazione Modbus prepara la PDU Modbus, quindi aggiunge i campi al fine di ottenere la PDU di comunicazione appropriata.



- **INDIRIZZO**: contiene l'indirizzo dello Slave. Come detto in precedenza (sezione "4.5.2 Indirizzo nodo (Figura 3)" a pagina 28), gli indirizzi dei nodi Slave devono essere compresi tra 1 e 247. Il Master invia un messaggio a uno Slave impostando l'indirizzo nel campo INDIRIZZO del messaggio. Nella risposta, lo Slave pone a sua volta il proprio indirizzo, per far sì che il Master possa riconoscere da chi proviene il messaggio.
- **CODICE FUNZIONE**: indica al server il tipo di azione da eseguire. Si deve impostare il valore esadecimale del codice funzione voluto. Il codice funzione può essere seguito da una campo **DATI** che contiene i parametri di interrogazione e risposta. Per maggiori informazioni sui codici funzione implementati riferirsi alla sezione "6.4 Codici funzione" a pagina 55.
- **DATI**: byte dedicati alle informazioni addizionali e alla trasmissione dei dati, il numero di byte e la struttura dipendono da ciascun codice funzione. Il campo **DATI** include valori come per esempio indirizzi di registro, numero di registri da processare, numero di data byte presenti nel campo, ecc. (si veda alla sezione "6.4 Codici funzione" a pagina 55).
- **CRC (Cyclic Redundancy Check, controllo a ridondanza ciclica)**: campo di verifica della corretta trasmissione del frame, basato sul metodo del controllo a ridondanza ciclica. E' utilizzato per verificare se la trasmissione è stata realizzata correttamente. Il campo CRC ha una dimensione di 2 byte, contenenti un valore binario di 16 bit. Il valore CRC è calcolato dal dispositivo trasmittente che lo allega al messaggio. Il dispositivo che riceve il messaggio ricalcola il valore del CRC alla

ricezione e lo confronta con quello ricevuto. Se i due valori non sono uguali, il dispositivo attiva un allarme.

Il protocollo Modbus definisce tre PDU. Esse sono:

- **Modbus Request PDU**;
- **Modbus Response PDU**;
- **Modbus Exception Response PDU**.

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 + 80 hex, 1 byte;
exception_code = Modbus Exception code, riferirsi alla tabella "Modbus Exception Codes" nel documento "Modbus Application Protocol Specification V1.1b".

6.3 Modalità di trasmissione

Il protocollo seriale Modbus prevede due modalità di trasmissione: la **modalità RTU** (**Remote Terminal Unit**) e la **modalità ASCII**. La modalità di trasmissione definisce la sequenza dei bit nei campi messaggio trasmessi serialmente. Definisce cioè come le informazioni sono ordinate all'interno dei campi messaggio e codificate. La modalità di trasmissione e i parametri della porta seriale devono essere gli stessi per tutti i dispositivi della linea seriale Modbus. Tutti i dispositivi devono implementare la modalità di trasmissione RTU, mentre la modalità di trasmissione ASCII è opzionale. I dispositivi Lika implementano solamente la modalità di trasmissione RTU, descritta nel successivo paragrafo.

6.3.1 Modalità di trasmissione RTU

Quando dei dispositivi comunicano in una linea seriale Modbus utilizzando la modalità di trasmissione RTU, ogni byte di 8 bit del messaggio contiene due caratteri esadecimali a 4 bit. Ogni messaggio deve essere inviato in una

sequenza ininterrotta di caratteri. La sincronizzazione dei messaggi tra trasmettitore e ricevitore è ottenuta interponendo un intervallo tra messaggi successivi (chiamato "silent interval") pari ad almeno 3,5 volte il tempo di un carattere. Se quindi il ricevitore non riceve un messaggio per un tempo di 4 caratteri, ritiene completato il messaggio precedente e considera che il successivo byte ricevuto sarà il primo del nuovo messaggio e quindi un indirizzo.
Il "silent interval" con velocità di trasmissione = 9600 bit/s è pari a 4 ms.
Il "silent interval" con velocità di trasmissione = 19200 bit/s è pari a 2 ms.
Il "silent interval" con velocità di trasmissione = 115200 bit/s è pari a 3,5 ms.

Il formato (11 bit) per ogni byte in modalità RTU sarà il seguente:

Sistema di codifica: binario a 8 bit

Bit per byte: 1 bit di start;
8 bit di dati, lsb inviato per primo;
1 bit di parità (= Even);
1 bit di stop.

Il protocollo Modbus utilizza il formato Big Endian, questo significa che quando è trasmessa una quantità numerica più grande di un singolo byte, l'MSB è trasmesso per primo.
Ogni carattere o byte è trasmesso nel seguente ordine (da sinistra a destra):

lsb (Least Significant Bit) ... msb (Most Significant Bit)

Start	1	2	3	4	5	6	7	8	Parità*	Stop

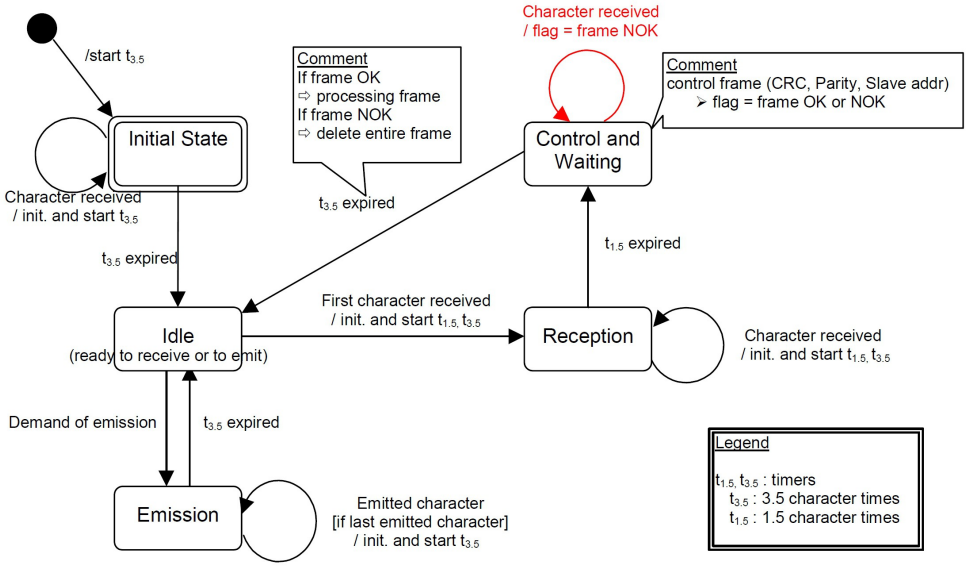
* Con "No parity", il bit di parità è sostituito da un bit di stop.

Come default bisogna impostare il bit di parità = EVEN.

Il frame, che avrà una dimensione massima di 256 byte, sarà così composto:

Indirizzo	Codice funzione	Dati	CRC
1 byte	1 byte	da 0 a 252 byte	2 byte CRC Low | CRC Hi

L'immagine che segue visualizza il diagramma degli stati nella modalità di trasmissione RTU.



- La transizione dallo stato **Initiate** allo stato **Idle** necessita di un intervallo di almeno 3,5 volte il tempo di un carattere ($t_{3,5}$).
- Lo stato **Idle** è lo stato normale quando non sono attivi né invii né ricezioni, quando cioè non è presente attività di trasmissione dopo un intervallo di tempo pari ad almeno 3,5 volte il tempo di un carattere ($t_{3,5}$).
- Una richiesta (request) può essere inviata solamente nello stato **Idle**. Dopo aver inviato una richiesta, il Master abbandona lo stato **Idle** e non può inviare una seconda richiesta nello stesso tempo.
- In una condizione **Idle**, la ricezione di un carattere trasmesso è considerata l'inizio di un frame e il link passa allo stato **Active**. Si considera il frame concluso quando non c'è trasmissione di caratteri per un intervallo almeno pari a $t_{3,5}$.
- Dopo che il frame è considerato concluso, si calcola e controlla il CRC. Successivamente si analizza il campo INDIRIZZO per determinare se il frame sia indirizzato al dispositivo. Se non è così, il frame viene scartato. Per ridurre il tempo di elaborazione della ricezione, l'INDIRIZZO può essere analizzato immediatamente alla ricezione senza attendere il completamento del frame. In questo caso il CRC viene calcolato e controllato solamente se il frame è effettivamente indirizzato allo Slave.

6.4 Codici funzione

Come detto in precedenza, i codici funzione esplicitano al server il tipo di azione da eseguire. Ogni codice funzione è codificato in un byte e i valori disponibili sono compresi tra 1 e 255 (ma i valori tra 128 e 255 sono riservati e utilizzati per le Exception response). Quando un messaggio è inviato dal Client al Server, il codice funzione esplicita al Server il tipo di azione da eseguire. Il codice funzione 0 non è ammesso.

Modbus prevede tre tipologie di codici funzione: i **codici funzione pubblici**, i **codici funzione definiti dall'utente** e i **codici funzione riservati**.

I **codici funzione pubblici** (compresi tra 1 e 64, tra 73 e 99 e tra 111 e 127) sono definiti e approvati da MODBUS-IDA.org che ne gestisce la conformità e ne garantisce l'unicità. I codici funzioni compresi tra 65 e 72 e tra 100 e 110 sono a disposizione e possono essere definiti a piacimento dall'utilizzatore (**codici funzione utente**). Naturalmente non c'è alcuna garanzia che un codice utente sia univoco nella rete. I **codici funzione riservati** non sono invece disponibili in alcun modo agli utilizzatori.

6.4.1 Codici funzione implementati

Gli encoder Lika della serie EM58 Modbus implementano esclusivamente i codici funzione pubblici descritti qui di seguito.

03 Read Holding Registers

FC = 03 (03 hex) ro

Questo codice funzione è utilizzato per LEGGERE i valori in un blocco di holding register contigui di un dispositivo remoto; in altri termini, permette la lettura dei valori impostati in alcuni parametri di lavoro in successione nel dispositivo. Il Request PDU specifica l'indirizzo del primo registro del gruppo e il numero di registri del gruppo. Nel PDU i Register sono indirizzati a partire da 0. Ne consegue che i registri numerati 1-16 sono indirizzati come 0-15.

Il valore del registro nel Response PDU è inviato in due byte per ciascun registro con il valore binario allineato a destra in ogni byte. Per ogni registro, il primo byte contiene i bit msb, mentre il secondo contiene i bit lsb.

Per un elenco dei parametri accessibili mediante il codice funzione **03 Read Holding Registers**, si veda la sezione "7.1.1 Parametri Dati macchina (Holding register)" a pagina 65.

Request PDU

Codice funzione	1 byte	**03 hex**
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	**03 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	**83 hex (=03 hex + 80 hex)**
Codice eccezione	1 byte	01 o 02 o 03 o 04

Esempio di richiesta di lettura del parametro **Valore di preset [0004-0005 hex]** (registri 5 e 6).

Request	
Nome campo	(Hex)
Codice funzione	**03**
Indirizzo iniziale Hi	**00**
Indirizzo iniziale Lo	**04**
Numero di registri Hi	**00**
Numero di registri Lo	**02**

Response	
Nome campo	(Hex)
Codice funzione	**03**
Numero byte	**04**
Valore registro 5 Hi	**00**
Valore registro 5 Lo	**00**
Valore registro 6 Hi	**05**
Valore registro 6 Lo	**DC**

Come si evince dalla tabella, i due registri 5 e 6 del parametro **Valore di preset [0004-0005 hex]** hanno valore rispettivamente 00 00 hex e 05 DC hex, il valore impostato nel parametro è cioè 1500 in notazione decimale.

Il frame completo di richiesta di lettura del parametro **Valore di preset [0004-0005 hex]** (registri 5 e 6) allo Slave con indirizzo 1 è il seguente:

Request PDU (in formato esadecimale)
[01][03][00][04][00][02][85][CA]
dove:
[01] = indirizzo dello Slave
[03] = codice funzione **03 Read Holding Registers**
[00][04] = indirizzo iniziale (parametro **Valore di preset [0004-0005 hex]**, registro 5)
[00][02] = numero di registri richiesti
[85][CA] = CRC

Il frame completo di invio dei valori nei due registri 5 e 6 del parametro **Valore di preset [0004-0005 hex]** da parte dello Slave con indirizzo 1 è il seguente:

Response PDU (in formato esadecimale)
[01][03][04][00][00][05][DC][F8][FA]
dove:
[01] = indirizzo dello Slave
[03] = codice funzione **03 Read Holding Registers**
[04] = numero di byte (2 byte per ciascun registro)
[00][00] = valore del registro 5, 00 00 hex = 0 dec
[05][DC] = valore del registro 6, 05 DC hex = 1500 dec
[F8][FA] = CRC

04 Read Input Register

FC = 04 (04 hex)

Questo codice funzione è utilizzato per LEGGERE registri di input contigui compresi tra 1 e 125 in un dispositivo remoto; in altri termini permette di leggere alcuni valori di risultato e gli stati / allarmi relativi al dispositivo. Il Request PDU specifica l'indirizzo del primo registro da leggere e il numero di registri da leggere. Nel PDU i registri sono indirizzati a partire da 0. Ne consegue che i registri di ingresso numerati 1-16 sono indirizzati come 0-15.
Il valore del registro nel Response PDU è inviato in due byte per ciascun registro con il valore binario allineato a destra in ogni byte. Per ogni registro, il primo byte contiene i bit msb, mentre il secondo contiene i bit lsb.
Per un elenco dei parametri accessibili mediante il codice funzione **04 Read Input Register**, si veda la sezione "7.1.2 Parametri Input Register" a pagina 76.

Request PDU

Codice funzione	1 byte	**04 hex**
Indirizzo iniziale	2 byte	da 0000 hex a FFFF hex
Numero di registri	2 byte	da 0000 hex a 007D hex

Response PDU

Codice funzione	1 byte	**04 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	**84 hex (=04 hex + 80 hex)**
Codice eccezione	1 byte	01 o 02 o 03 o 04

Esempio di richiesta di lettura del parametro **Posizione corrente [0001-0002 hex]** (registri 2 e 3).

Request	
Nome campo	(Hex)
Codice funzione	**04**
Indirizzo iniziale Hi	**00**
Indirizzo iniziale Lo	**01**
Numero di registri Hi	**00**
Numero di registri Lo	**02**

Response	
Nome campo	(Hex)
Codice funzione	**04**
Numero byte	**04**
Valore registro 2 Hi	**00**
Valore registro 2 Lo	**00**
Valore registro 3 Hi	**2F**
Valore registro 3 Lo	**F0**

Come si evince dalla tabella, i registri 2 e 3 del parametro **Posizione corrente [0001-0002 hex]** hanno valore rispettivamente 00 00 hex e 2F F0 hex, cioè 12.272 in notazione decimale.

Il frame completo di richiesta di lettura del parametro **Posizione corrente [0001-0002 hex]** (registri 2 e 3) allo Slave con indirizzo 1 è il seguente:

Request PDU (in formato esadecimale)
[01][04][00][01][00][02][20][0B]
dove:
[01] = indirizzo dello Slave
[04] = codice funzione **04 Read Input Register**
[00][01] = indirizzo iniziale (parametro **Posizione corrente [0001-0002 hex]**, registro 2)
[00][02] = numero di registri richiesti
[20][0B] = CRC

Il frame completo di invio del valore del parametro **Posizione corrente [0001-0002 hex]** (registri 2 e 3) da parte dello Slave con indirizzo 1 è il seguente:

Response PDU (in formato esadecimale)
[01][04][04][00][00][2F][F0][E7][F0]
dove:
[01] = indirizzo dello Slave
[04] = codice funzione **04 Read Input Register**
[04] = numero di byte (2 byte per ciascun registro)
[00][00] = valore del registro 2 **Posizione corrente [0001-0002 hex]**, 00 00 hex = 0 dec
[2F][F0] = valore del registro 3 **Posizione corrente [0001-0002 hex]**, 2F F0 hex = 12.272 dec
[E7][F0] = CRC

06 Write Single Register

FC = 06 (06 hex)

Questo codice funzione è utilizzato per ASSEGNARE UN VALORE a un singolo holding register in un dispositivo remoto. Il Request PDU specifica l'indirizzo del registro da scrivere. I registri sono indirizzati a partire da 0. Ne consegue che i registri numerati 1-16 sono indirizzati come 0-15.
La risposta positiva rispecchia nella sua struttura la domanda ed è inviata dopo che il valore richiesto è stato scritto.
Per un elenco dei parametri accessibili mediante il codice funzione **06 Write Single Register**, si veda la sezione "7.1.1 Parametri Dati macchina (Holding register)" a pagina 65.

Request PDU

Codice funzione	1 byte	**06 hex**
Indirizzo del registro	2 byte	da 0000 hex a FFFF hex
Valore del registro	2 byte	da 0000 hex a FFFF hex

Response PDU

Codice funzione	1 byte	**06 hex**
Indirizzo del registro	2 byte	da 0000 hex a FFFF hex
Valore del registro	2 byte	da 0000 hex a FFFF hex

Exception Response PDU

Codice errore	1 byte	**86 hex (=06 hex + 80 hex)**
Codice eccezione	1 byte	01 o 02 o 03 o 04

Esempio di impostazione del parametro **Parametri operativi [0008 hex]** (registro 9): abilitazione dello scaling (**Funzione di scaling** = 1) e conteggio crescente con rotazione oraria dell'albero dell'encoder (**Direzione di conteggio** = 0).

Request	
Nome campo	(Hex)
Codice funzione	**06**
Indirizzo registro Hi	**00**
Indirizzo registro Lo	**08**
Valore del registro Hi	**00**
Valore del registro Lo	**01**

Response	
Nome campo	(Hex)
Codice funzione	**06**
Indirizzo registro Hi	**00**
Indirizzo registro Lo	**08**
Valore del registro Hi	**00**
Valore del registro Lo	**01**

Come si evince dalla tabella, nel parametro **Parametri operativi [0008 hex]** (registro 9) si imposta il valore 00 01 hex, cioè 0000 0000 0000 0001 in

notazione binaria: il bit 0 **Funzione di scaling** = 1; il bit 1 **Direzione di conteggio** = 0; tutti gli altri bit non sono utilizzati e perciò posti a 0.

Il frame completo di richiesta di scrittura del valore 00 01 hex nel parametro **Parametri operativi [0008 hex]** (registro 9) dello Slave con indirizzo 1 è il seguente:

Request PDU (in formato esadecimale)
[01][06][00][08][00][01][C9][C8]
dove:
[01] = indirizzo dello Slave
[06] = codice funzione **06 Write Single Register**
[00][08] = indirizzo del registro (parametro **Parametri operativi [0008 hex]**, registro 9)
[00][01] = valore da impostare nel registro
[C9][C8] = CRC

Il frame completo di risposta alla richiesta di scrittura nel parametro **Parametri operativi [0008 hex]** (registro 9) da parte dello Slave con indirizzo 1 è il seguente:

Response PDU (in formato esadecimale)
[01][06][00][08][00][01][C9][C8]
dove:
[01] = indirizzo dello Slave
[06] = codice funzione **06 Write Single Register**
[00][08] = indirizzo del registro (parametro **Parametri operativi [0008 hex]**, registro 9)
[00][01] = valore impostato nel registro
[C9][C8] = CRC

16 Write Multiple Registers

FC = 16 (10 hex)

Questo codice funzione è utilizzato per ASSEGNARE DEI VALORI a un blocco di registri contigui (registri da 1 a 123) in un dispositivo remoto.
I valori da impostare sono specificati nel campo dati della richiesta. Il valore da assegnare a ogni registro è inviato in due byte per ciascun registro.
La risposta positiva restituisce il codice funzione, l'indirizzo iniziale e il numero di registri su cui si è scritto.

Per un elenco dei parametri accessibili mediante il codice funzione **16 Write Multiple Registers**, si veda la sezione "7.1.1 Parametri Dati macchina (Holding register)" a pagina 65.

Request PDU

Codice funzione	1 byte	**10 hex**
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 **N***
Valore dei registri	**N*** x 2 byte	valore

*N = Numero di registri

Response PDU

Codice funzione	1 byte	**10 hex**
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	**90 hex (= 10 hex + 80 hex)**
Codice eccezione	1 byte	01 o 02 o 03 o 04

Esempio di scrittura del valore 00 00 08 00 hex (= 2.048 dec) nel parametro **Informazioni per giro programmate [0000-0001 hex]** (registri 1 e 2) e del valore 00 80 00 00 hex (= 8.388.608 dec) nel parametro **Risoluzione totale programmata [0002-0003 hex]** (registri 3 e 4).

Request	
Nome campo	(Hex)
Codice funzione	**10**

Response	
Nome campo	(Hex)
Codice funzione	**10**

Indirizzo iniziale Hi	**00**
Indirizzo iniziale Lo	**00**
Numero di registri Hi	**00**
Numero di registri Lo	**04**
Numero byte	**08**
Valore registro 1 Hi	**00**
Valore registro 1 Lo	**00**
Valore registro 2 Hi	**08**
Valore registro 2 Lo	**00**
Valore registro 3 Hi	**00**
Valore registro 3 Lo	**80**
Valore registro 4 Hi	**00**
Valore registro 4 Lo	**00**

Indirizzo iniziale Hi	**00**
Indirizzo iniziale Lo	**00**
Numero di registri Hi	**00**
Numero di registri Lo	**04**

Come si evince dalla tabella, nei due registri 1 e 2 del parametro **Informazioni per giro programmate [0000-0001 hex]** si impostano rispettivamente i valori 00 00 hex e 08 00 hex, cioè 2.048 in notazione decimale; nei due registri 3 e 4 del parametro **Risoluzione totale programmata [0002-0003 hex]** si impostano rispettivamente i valori 00 80 hex e 00 00 hex, cioè 8.388.608 in notazione decimale. L'encoder avrà quindi una risoluzione monogiro di 2.048 informazioni/giro e una risoluzione multigiro di 4096 giri (8.388.608/2.048).

Il frame completo di richiesta di scrittura del valore 2.048 nel parametro **Informazioni per giro programmate [0000-0001 hex]** (registri 1 e 2) e del valore 8.388.608 nel parametro **Risoluzione totale programmata [0002-0003 hex]** (registri 3 e 4) dello Slave con indirizzo 1 è il seguente:

Request PDU (in formato esadecimale)
[01][10][00][00][00][04][08][00][00][08][00][00][80][00][00][B6][DA]
dove:
[01] = indirizzo dello Slave
[10] = codice funzione **16 Write Multiple Registers**
[00][00] = indirizzo iniziale (parametro **Informazioni per giro programmate [0000-0001 hex]**, registro 1)
[00][04] = numero di registri richiesti

[08] = numero di byte (2 byte per ciascun registro)
[00][00] = valore da impostare nel registro 1
[08][00] = valore da impostare nel registro 2, 00 00 08 00 hex = 2.048 dec
[00][80] = valore da impostare nel registro 3
[00][00] = valore da impostare nel registro 4, 00 80 00 00 hex = 8.388.608 dec
[B6][DA] = CRC

Il frame completo di risposta alla richiesta di impostazione del valore 2.048 nel parametro **Informazioni per giro programmate [0000-0001 hex]** (registri 1 e 2) e del valore 8.388.608 nel parametro **Risoluzione totale programmata [0002-0003 hex]** (registri 3 e 4) da parte dello Slave con indirizzo 1 è il seguente:

Response PDU (in formato esadecimale)
[01][10][00][00][00][04][C1][CA]
dove:
[01] = indirizzo dello Slave
[10] = codice funzione **16 Write Multiple Registers**
[00][00] = indirizzo iniziale (parametro **Informazioni per giro programmate [0000-0001 hex]**, registro 1)
[00][04] = numero di registri su cui si è scritto
[C1][CA] = CRC

IMPORTANTE
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 presenza di comunicazione; questo si rende necessario per evitare situazioni di pericolo nel caso in cui fossero presenti guasti nella rete di comunicazione.
A questo scopo il sistema prevede l'attivazione opzionale della funzione Watchdog. Il Watchdog è un sistema 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 seriale mentre è attivo un comando, il Watchdog interverrebbe comandando l'attivazione di un allarme. Per abilitare il Watchdog impostare a "=1" il bit **Abilitazione Watchdog** in **Control Word [0009 hex]**. Se impostato a "=0" il Watchdog non è attivo; se impostato a "=1" il Watchdog è attivo. Con Watchdog attivo, se il dispositivo non riceve un messaggio dal Server entro 1 secondo, il sistema forza una condizione di allarme (visualizzazione dell'allarme **Watchdog** al ripristino della comunicazione con la rete Modbus).

7 Parametri di programmazione

7.1 Parametri disponibili

Di seguito sono riportati i parametri disponibili per il dispositivo, per ognuno è indicato:

Nome parametro [Indirizzo registro]
[Numero registro, tipo variabile, attributo]

- L'indirizzo del registro è espresso in valore esadecimale.
- Il numero del registro è espresso in valore decimale.
- Attributo:
  ro = variabile accessibile in sola lettura
  rw = variabile accessibile in lettura e scrittura

7.1.1 Parametri Dati macchina (Holding register)

I parametri **Dati macchina** sono accessibili sia in lettura che in scrittura; per leggere il valore di un parametro utilizzare il codice funzione **03 Read Holding Registers** (lettura multipla dei registri); per scrivere il valore in un parametro utilizzare il codice funzione **06 Write Single Register** (scrittura di un singolo registro) oppure il codice funzione **16 Write Multiple Registers** (scrittura di più registri); per ogni informazione sui codici funzione implementati riferirsi alla sezione "6.4.1 Codici funzione implementati" a pagina 55.

Informazioni per giro programmate [0000-0001 hex]
[Registri 1-2, Unsigned32, rw]

ATTENZIONE
Questo parametro è attivo solamente se il bit 0 **Funzione di scaling** del registro **Parametri operativi [0008 hex]** è impostato a "1". Se **Funzione di scaling** = 1 (ABILITATA), i valori della risoluzione monogiro impostati in questo parametro sono attivati e utilizzati dall'encoder; al contrario, se **Funzione di scaling** = 0 (DISABILITATA), il sistema utilizza i valori fisici della risoluzione:
ES58: risoluzione fisica monogiro = 4.096 cpr; numero di giri fisici = 1 giro; risoluzione fisica totale = 4.096 = 12 bit.
EM58: risoluzione fisica monogiro = 4.096 cpr; numero di giri fisici = 16.384 giri; risoluzione fisica totale = 67.108.864 = 26 bit.

Questo parametro imposta il numero di informazioni per giro desiderate, ossia un valore specifico della risoluzione monogiro.
E' possibile impostare qualsiasi valore intero minore o uguale al **numero di informazioni per giro fisiche** (4.096 cpr). Impostando un valore maggiore del numero di informazioni per giro fisiche, a seguito dell'invio della Request PDU sarà restituito il messaggio di errore **Dati macchina non validi** con impostazione a 1 del bit relativo in **Elenco DM errati [0004-0005 hex]**.
Per ogni informazione sul numero di informazioni per giro fisiche e il numero di giri fisici riferirsi ai dati di targa dello specifico dispositivo.
Default = 4.096 (min. = 1, max. = 4.096)

NOTA
Per evitare possibili salti di quota si consiglia di assicurarsi sempre che:

$$\frac{\text{Risoluzione totale programmata [0002-0003 hex]}}{\text{Informazioni per giro programmate [0000-0001 hex]}} = \textbf{potenza di 2.}$$

ATTENZIONE
Quando si modifica il valore del registro **Informazioni per giro programmate [0000-0001 hex]**, verificare sempre anche il valore della **Risoluzione totale programmata [0002-0003 hex]** e che la seguente relazione sia comunque soddisfatta:

$$\frac{\text{Risoluzione totale programmata [0002-0003 hex]}}{\text{Informazioni per giro programmate [0000-0001 hex]}} \leq \textbf{Numero di giri fisici}$$

Immaginiamo per esempio che l'encoder EM5812/16384MB utilizzi la risoluzione fisica, ossia:

- **Informazioni per giro fisiche** = 4.096
- **Numero di giri fisici** = 16.384
- **Risoluzione totale fisica** = 67.108.864

Impostiamo ora una risoluzione monogiro personalizzata, per esempio: **Informazioni per giro programmate [0000-0001 hex]** = 2.048.
Se non modifichiamo contestualmente anche il valore della risoluzione totale, risulterà che:

$$\textbf{Numero di giri} = \frac{67.108.864\ (\text{Risoluzione totale programmata [0002-0003 hex]})}{2.048\ (\text{Informazioni per giro programmate [0000-0001 hex]})} = 32.768$$

Sarebbero cioè richiesti all'encoder 32.768 giri, il che non può essere dato che il numero di giri fisici massimo è, come detto, 16.384. In questo caso l'encoder segnalerebbe l'errore tramite i bit di allarme e i LED.

ATTENZIONE
Con funzione di preset, quando si modifica il valore in **Informazioni per giro programmate [0000-0001 hex]**, bisogna poi verificare il valore del registro **Valore di preset [0004-0005 hex]** ed eseguire uno zero macchina (bit 11 **Esegui preset conteggio** in **Control Word [0009 hex]** = 1).

Risoluzione totale programmata [0002-0003 hex]

[Registri 3-4, Unsigned32, rw]

ATTENZIONE
Questo parametro è attivo solamente se il bit 0 **Funzione di scaling** del registro **Parametri operativi [0008 hex]** è impostato a "1". Se **Funzione di scaling** = 1 (ABILITATA), i valori della risoluzione totale impostati in questo parametro sono attivati e utilizzati dall'encoder; al contrario, se **Funzione di scaling** = 0 (DISABILITATA), il sistema utilizza i valori fisici della risoluzione:
ES58: risoluzione fisica monogiro = 4.096 cpr; numero di giri fisici = 1 giro; risoluzione fisica totale = 4.096 = 12 bit.
EM58: risoluzione fisica monogiro = 4.096 cpr; numero di giri fisici = 16.384 giri; risoluzione fisica totale = 67.108.864 = 26 bit.

Questo registro imposta la risoluzione totale desiderata. La **Risoluzione totale programmata [0002-0003 hex]** dell'encoder risulta dal prodotto delle **Informazioni per giro programmate [0000-0001 hex]** per il **numero di giri** richiesti.

Il rapporto $\frac{\text{Risoluzione totale programmata [0002-0003 hex]}}{\text{Informazioni per giro programmate [0000-0001 hex]}}$

definisce il **Numero di giri programmato**.
E' possibile impostare qualsiasi valore intero minore o uguale alla **Risoluzione totale fisica** (ES58: 4.096 info = 12 bit; EM58: 67.108.864 info = 26 bit). Impostando un valore maggiore della risoluzione totale fisica, a seguito dell'invio della Request PDU sarà restituito il messaggio di errore **Dati macchina non validi** con impostazione a 1 del bit relativo in **Elenco DM errati [0004-0005 hex]**.
Per ogni informazione sul numero di informazioni per giro fisiche e il numero di giri fisici riferirsi ai dati di targa dello specifico dispositivo.

Default = 4.096 (min. = 1, max. = 4.096) ES58
67.108.864 (min. = 1, max. = 67.108.864) EM58

NOTA
Per evitare possibili salti di quota si consiglia di assicurarsi sempre che:

$$\frac{\text{Risoluzione totale programmata [0002-0003 hex]}}{\text{Informazioni per giro programmate [0000-0001 hex]}} = \textbf{potenza di 2.}$$

ATTENZIONE
Quando si modifica il valore della **Risoluzione totale programmata [0002-0003 hex]**, verificare sempre anche il valore delle **Informazioni per giro programmate [0000-0001 hex]** e assicurarsi che il numero di giri che ne consegue sia congruo con il **Numero di giri fisici** del dispositivo (ES58 = 1; EM58 = 16.384).
Immaginiamo per esempio che l'encoder EM5812/16384MB sia programmato come segue:
Informazioni per giro programmate [0000-0001 hex] = 4.096 cpr
Risoluzione totale programmata [0002-0003 hex] = $16.777.216_{10}$ = 4.096 (info/giro) * 4.096 (giri)
Impostiamo ora una nuova risoluzione complessiva, per esempio: **Risoluzione totale programmata [0002-0003 hex]** = 360.
In questo caso la **Risoluzione totale programmata [0002-0003 hex]** sarebbe minore delle **Informazioni per giro programmate [0000-0001 hex]**, il valore è comunque considerato lecito.

ATTENZIONE
Con funzione di preset, quando si modifica il valore in **Risoluzione totale programmata [0002-0003 hex]** bisogna poi verificare il valore del registro **Valore di preset [0004-0005 hex]** ed eseguire uno zero macchina (bit 11 **Esegui preset conteggio** in **Control Word [0009 hex]** = 1).

ESEMPIO
Encoder multigiro EM58**12/16384MB-...**.

Risoluzione fisica:

- **Informazioni per giro fisiche** = 4.096 inf./giro (2^{12})
- **Numero di giri fisici** = 16.384 giri (2^{14})
- **Risoluzione totale fisica** = 67.108.864 (2^{26})

Si desidera impostare la risoluzione di **2.048 inf./giro * 1.024 giri**:

- Attivare la **Funzione di scaling**: **Parametri operativi [0008 hex]**, bit 0 = 1
- Impostare le informazioni per giro: **Informazioni per giro programmate [0000-0001 hex]** = 2.048 (0000 0800 hex)
- Impostare la risoluzione totale: **Risoluzione totale programmata [0002-0003 hex]** = 2.048 * 1.024 = 2.097.152 (0020 0000 hex)
- Eseguire un salvataggio dei parametri (comando **Salva parametri** del registro **Control Word [0009 hex]**; si veda a pagina 73)

NOTA
Per evitare possibili salti di quota si consiglia di impostare sempre valori di potenza di due (2^n: es. 1, 2, 4, ..., 2048, 4096, 8192,...) nei registri **Informazioni per giro programmate [0000-0001 hex]** e **Risoluzione totale programmata [0002-0003 hex]**.
Se si modificano i registri **Informazioni per giro programmate [0000-0001 hex]** e/o **Risoluzione totale programmata [0002-0003 hex]** bisogna reimpostare eventuali azzeramenti (**Valore di preset [0004-0005 hex]**).

Valore di preset [0004-0005 hex]

[Registri 5-6, Unsigned32, rw]
Questo registro permette di impostare un valore di Preset. Con la funzione di preset è possibile assegnare un valore desiderato a una definita posizione dell'encoder. Tale posizione assumerà perciò il valore impostato in questo parametro e tutte le altre posizioni precedenti e successive assumeranno un valore conseguente. Questa funzione si rivela utile, per esempio, per far sì che lo zero dell'encoder corrisponda allo zero dell'applicazione. Il valore di preset sarà assegnato alla posizione dell'asse al momento dell'invio del comando **Esegui preset conteggio** in **Control Word [0009 hex]**. Consigliamo di impostare il preset con encoder fermo.
Default = 0 (min. = 0, max. = 67.108.864)

Default = 0 (min. = 0, max. = 4.096) ES58
0 (min. = 0, max. = 67.108.864) EM58

ESEMPIO

Forniamo qui di seguito un esempio per comprendere meglio la funzione di preset e il significato dei registri e dei comandi correlati: **Valore di preset [0004-0005 hex]**, **Offset [0006-0007 hex]** e **Esegui preset conteggio**.

La posizione encoder trasmessa è il risultato del seguente calcolo:
Quota trasmessa = **posizione letta** (sia essa fisica o "scalata") + **Valore di preset [0004-0005 hex]** - **Offset [0006-0007 hex]**.
Se non si è mai impostato un **Valore di preset [0004-0005 hex]** e comunque eseguito un comando di zero macchina (comando **Esegui preset conteggio**), quota trasmessa e posizione letta coincidono, in quanto **Valore di preset [0004-0005 hex]** = 0 e **Offset [0006-0007 hex]** = 0.
Quando si imposta un **Valore di preset [0004-0005 hex]** e si esegue poi il comando **Esegui preset conteggio** in **Control Word [0009 hex]**, il sistema memorizza in **Offset [0006-0007 hex]** l'attuale posizione dell'encoder. Ne risulta che quota trasmessa e **Valore di preset [0004-0005 hex]** coincidono in quanto posizione letta - **Offset [0006-0007 hex]** = 0; in altri termini, a una desiderata posizione dell'encoder corrisponde il valore impostato in **Valore di preset [0004-0005 hex]**.
Per esempio, supponiamo di impostare il valore "50" nel registro **Valore di preset [0004-0005 hex]** e di eseguire il comando **Esegui preset conteggio** 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") + **Valore di preset [0004-0005 hex]** (="50") - **Offset [0006-0007 hex]** (="1000") = **50**.
La successiva quota trasmessa sarà poi:
Quota trasmessa = **posizione letta** (="1001") + **Valore di preset [0004-0005 hex]** (="50") - **Offset [0006-0007 hex]** (="1000") = **51**.
E così via.

NOTA

- Se **Funzione di scaling** è disabilitato (il bit 0 del registro **Parametri operativi [0008 hex]** = 0), **Valore di preset [0004-0005 hex]** deve essere minore o uguale alla risoluzione totale fisica (informazioni per giro fisiche * numero di giri fisici).

- Se **Funzione di scaling** è abilitato (il bit 0 del registro **Parametri operativi [0008 hex]** = 1), **Valore di preset [0004-0005 hex]** deve essere minore o uguale a **Risoluzione totale programmata [0002-0003 hex]**).

ATTENZIONE
Verificare il parametro **Valore di preset [0004-0005 hex]** ed eseguire uno zero macchina (bit 11 **Esegui preset conteggio** in **Control Word [0009 hex]** = 1) ogniqualvolta vengono modificati i parametri **Informazioni per giro programmate [0000-0001 hex]** e/o **Risoluzione totale programmata [0002-0003 hex]**.

Offset [0006-0007 hex]

[Registri 7-8, Unsigned32, ro]
In questo registro viene memorizzata la posizione dell'encoder nel momento in cui si esegue il comando **Esegui preset conteggio**. Questo valore è poi utilizzato nella funzione di preset per il calcolo della posizione trasmessa. Per riportare a 0 il valore in questo parametro è necessario caricare i valori di default (comando **Carica parametri di default** in **Control Word [0009 hex]**). Per maggiori informazioni sulla funzione di preset e il significato dei registri e comandi correlati (**Valore di preset [0004-0005 hex]**, **Offset [0006-0007 hex]** e **Esegui preset conteggio**) riferirsi alla pagina 69.

Parametri operativi [0008 hex]

[Registro 9, Unsigned16, rw]

Struttura byte del registro **Parametri operativi [0008 hex]**:

byte	MSB			LSB		
bit	15	...	8	7	...	0
	msb		lsb	msb		lsb

Byte 0
Funzione di scaling
bit 0 Questo bit è utilizzato per abilitare / disabilitare i parametri **Informazioni per giro programmate [0000-0001 hex]** e **Risoluzione totale programmata [0002-0003 hex]**. Se disabilitato (bit 0 = 0), l'encoder utilizza la propria risoluzione fisica (vale a dire: numero di informazioni per

giro fisiche e numero di giri fisici, si vedano i dati di targa del dispositivo); se abilitato (bit 0 = 1), l'encoder utilizza la risoluzione impostata nei parametri **Informazioni per giro programmate [0000-0001 hex]** e **Risoluzione totale programmata [0002-0003 hex]** con la seguente relazione:

$$\text{Posizione trasmessa} = \frac{\text{Informazioni per giro programmate [0000-0001 hex]}}{\text{Informazioni per giro fisiche}} \times \text{Posizione reale} \leq \text{Risoluzione totale programmata [0002-0003 hex]}$$

E' possibile leggere il valore correntemente impostato tramite il bit 0 **Scaling** del registro **Status word [000A hex]**, si veda a pagina 80.

ATTENZIONE
Ogniqualvolta si abilita la **Funzione di scaling** e/o si modificano i valori di scaling (si vedano i registri **Informazioni per giro programmate [0000-0001 hex]** e **Risoluzione totale programmata [0002-0003 hex]**) sarà poi necessario verificare ed eventualmente impostare anche il valore di preset (si veda il registro **Valore di preset [0004-0005 hex]**) e quindi salvare i nuovi parametri (si veda il comando **Salva parametri**).

Direzione di conteggio

bit 1

Mediante il comando **Direzione di conteggio** si imposta se il valore trasmesso dall'encoder è crescente quando l'albero ruota in senso orario oppure quando l'albero ruota in senso antiorario. Il senso di rotazione è stabilito guardando l'encoder dal lato dell'albero. Impostando 0 (bit 1 = 0) il valore è crescente con rotazione oraria dell'albero dell'encoder; impostando 1 (bit 1 = 1) il valore è crescente con rotazione antioraria dell'albero dell'encoder.
E' possibile leggere il valore correntemente impostato tramite il bit 1 **Direzione conteggio** del registro **Status word [000A hex]**, si veda a pagina 80.

ATTENZIONE
Ogniqualvolta si abilita la **Direzione di conteggio** sarà poi necessario verificare ed eventualmente impostare il valore di preset (si veda il registro **Valore di preset [0004-0005**

hex]) e quindi salvare i nuovi parametri (si veda il comando **Salva parametri**).

bit 2 ... 7 Non utilizzati.

Byte 1 Non utilizzato.

Control Word [0009 hex]

[Registro 10, Unsigned16, rw]
Contiene i comandi da inviare in tempo reale al dispositivo per controllarlo.

Struttura byte del registro **Control Word [0009 hex]**:

byte	MSB			LSB		
bit	15	...	8	7	...	0
	msb		lsb	msb		lsb

Byte 0 Non utilizzato.

Byte 1

Abilitazione Watchdog

bit 8 Se impostato a "=0" il Watchdog non è attivo; se impostato a "=1" il Watchdog è attivo. Con Watchdog attivo, se il dispositivo non riceve un messaggio dal Server entro 1 secondo, il sistema forza una condizione di allarme (attivazione dell'allarme **Watchdog** al ripristino della comunicazione con la rete Modbus). Il Watchdog è un sistema 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 seriale mentre è attivo un comando il Watchdog interverrebbe comandando l'attivazione dell'allarme.

Salva parametri

bit 9 Il salvataggio nella memoria non volatile dei parametri impostati è eseguito in corrispondenza del fronte di salita di questo bit; in altri termini il salvataggio dei dati macchina impostati è richiesto alla variazione del bit dal livello logico basso (0) al livello logico alto (1).

Carica parametri di default

bit 10

Il caricamento dei parametri di default (parametri impostati durante la messa a punto in azienda del dispositivo che permettono un funzionamento standard e sicuro dell'encoder) è eseguito in corrispondenza del fronte di salita di questo bit; in altri termini il caricamento dei valori di default è richiesto alla variazione del bit dal livello logico basso (0) al livello logico alto (1). A pagina 89 è disponibile l'elenco dei dati macchina e il rispettivo valore di default preimpostato da Lika Electronic.

ATTENZIONE
Con l'esecuzione di questo comando tutti i valori precedentemente impostati sono sovrascritti!

Esegui preset conteggio

bit 11

Permette di eseguire un cosiddetto "zero macchina". A seguito dell'invio di questo comando, il valore di posizione trasmesso per la corrente posizione dell'encoder assume il valore impostato nella variabile **Valore di preset [0004-0005 hex]** e tutte le altre posizioni assumono un valore conseguente. L'operazione è eseguita in corrispondenza del fronte di salita di questo bit, cioè alla variazione del bit dal livello logico basso (0) al livello logico alto (1). Con l'invio di questo comando la posizione corrente dell'encoder è temporaneamente memorizzata nel registro **Offset [0006-0007 hex]**. Per maggiori informazioni sulla funzione di preset e il significato dei registri e comandi correlati (**Valore di preset [0004-0005 hex]**, **Offset [0006-0007 hex]** e **Esegui preset conteggio**) riferirsi alla pagina 69.

ATTENZIONE
Per memorizzare permanentemente la posizione corrente dell'encoder nel registro **Offset [0006-0007 hex]** eseguire il comando **Salva parametri**. Diversamente, nel caso di spegnimento del dispositivo il dato andrà perso!

bit 12 ... 15

Non utilizzati.

NOTA

Per salvare i parametri modificati eseguire il comando **Salva parametri** bit 9 del registro **Control Word [0009 hex]**.
Diversamente, nel caso di spegnimento del dispositivo i dati non salvati andranno persi!

7.1.2 Parametri Input Register

I parametri **Input Register** sono accessibili in sola lettura; per leggere il valore di un parametro utilizzare il codice funzione **04 Read Input Register** (lettura multipla degli input register); per ogni informazione sui codici funzione implementati riferirsi alla sezione "6.4.1 Codici funzione implementati" a pagina 55.

Registro allarmi [0000 hex]

[Registro 1, Unsigned16, ro]
Questa variabile fornisce informazioni sugli allarmi attivi nel dispositivo. L'attivazione di un allarme è notificata anche visivamente tramite i LED, si veda la sezione "4.4 Indicatori LED (Figura 1)" a pagina 23.
Struttura byte allarmi:

byte	MSB			LSB		
bit	15	...	8	7	...	0
	msb		lsb	msb		lsb

Codici registro allarmi previsti:

Byte 0

Dati macchina non validi

bit 0 — Uno o più parametri non sono validi, impostare valori corretti per ristabilire la normale condizione di lavoro. Per conoscere in dettaglio quale parametro sia errato. controllare la lista dei parametri errati in **Elenco DM errati [0004-0005 hex]**.

Errore memoria flash

bit 1 — Errore nei valori della memoria flash dati (checksum errata, ecc.). Errore interno non ripristinabile.

bit 2 ... 7 — Non utilizzati.

Byte 1

bit 8 ... 10 — Non utilizzati.

Watchdog

bit 11 — Con Watchdog attivo (**Abilitazione Watchdog** in **Control Word [0009 hex]** impostato a "=1"), se il dispositivo non riceve un messaggio dal Server entro 1 secondo, il sistema

forza una condizione di allarme (attivazione del bit di allarme **Watchdog**). L'allarme è segnalato al ripristino della comunicazione con la rete Modbus. Il Watchdog è un sistema 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 seriale mentre è attivo un comando il Watchdog interverrebbe comandando l'attivazione dell'allarme.

bit 12 ... 15 Non utilizzati.

NOTA

Si badi che, se l'allarme è relativo a dati macchina non validi (si veda il messaggio **Dati macchina non validi** e il registro **Elenco DM errati [0004-0005 hex]**), si può tornare allo stato normale solo impostando dati macchina validi. L'allarme **Watchdog** è automaticamente resettato al ripristino della comunicazione. L'allarme **Errore memoria flash** non è invece ripristinabile.

Posizione corrente [0001-0002 hex]

[Registri 2-3, Integer32, ro]

Questi registri visualizzano il valore di posizione al momento dell'invio della richiesta, eventualmente ridefinito sulla base dello scaling impostato, si veda **Funzione di scaling** a pagina 71. Il valore è espresso in numero di informazioni.

Registro 4 [0003 hex]

[Registro 4, Integer16, ro]

Registro non utilizzato e disponibile per un uso futuro.

Elenco DM errati [0004-0005 hex]

[Registri 5-6, Unsigned32, ro]

L'operatore ha impostato valori non consoni e il sistema ha visualizzato il messaggio di allarme **Dati macchina non validi**. Questa variabile indica (valore del bit = ALTO) quali parametri contengono dati errati, secondo la lista riportata nella tabella che segue.

Si badi che si può ripristinare il normale stato di lavoro del dispositivo solo impostando dati macchina validi.

Bit	Nome parametro
0	Non utilizzato
1	Informazioni per giro programmate [0000-0001 hex]
2	Risoluzione totale programmata [0002-0003 hex]
3	Valore di preset [0004-0005 hex]
4	Offset [0006-0007 hex]
5	Parametri operativi [0008 hex]
6	DIP switch indirizzo [0007 hex]
7	DIP switch baud rate [0006 hex]
8 ... 15	Non utilizzati

DIP switch baud rate [0006 hex]

[Registro 7, Unsigned16, ro]

Visualizza il valore della velocità di trasmissione dei dati (baud rate e bit di parità) della porta seriale del dispositivo impostata mediante il DIP switch previsto, secondo quanto indicato nella tabella che segue. Per ogni informazione sull'impostazione del baud rate e del bit di parità riferirsi alla sezione "4.5.1 Velocità di trasmissione dei dati: baud rate e bit di parità (Figura 3)" a pagina 27.

Valore (hex)	Baud rate	Bit di parità
0000	9600 bit/s	No Parity
0001	9600 bit/s	Even
0002	9600 bit/s	Odd
0003	19200 bit/s	No Parity
0004	19200 bit/s	Even
0005	19200 bit/s	Odd
0006	115200 bit/s	No Parity
0007	115200 bit/s	Even
0008	115200 bit/s	Odd

DIP switch indirizzo [0007 hex]

[Registro 8, Unsigned16, ro]
Visualizza l'indirizzo del nodo del dispositivo impostato mediante il DIP switch previsto. Per ogni informazione sull'impostazione dell'indirizzo del nodo riferirsi alla sezione "4.5.2 Indirizzo nodo (Figura 3)" a pagina 28.

Versione software [0008 hex]

[Registro 9, Unsigned16, ro]
Visualizza la versione software del dispositivo.
Il numero major indica l'edizione del firmware, mentre il numero minor indica la revisione del firmware.
Il significato dei 16 bit che compongono il registro è il seguente:

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Ms bit								Ls bit							
Numero major								Numero minor							

Il valore 01 02 hex, che corrisponde alla rappresentazione binaria 00000001 00000010, deve perciò essere interpretato come: versione "1.2": edizione firmware 1, revisione firmware 2.

Versione hardware [0009 hex]

[Registro 10, Unsigned16, ro]
Visualizza la versione hardware del dispositivo.
Il numero major indica l'edizione dell'hardware, mentre il numero minor indica la revisione dell'hardware.
Il significato dei 16 bit che compongono il registro è il seguente:

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Ms bit								Ls bit							
Numero major								Numero minor							

Il valore 01 00 hex, che corrisponde alla rappresentazione binaria 00000001 00000000, deve perciò essere interpretato come: versione "1.0": edizione hardware 1, revisione hardware 0.

Status word [000A hex]

[Registro 11, Unsigned16, ro]

Questo registro contiene le informazioni relative allo stato del dispositivo.

Gli otto bit di LSB visualizzano le impostazioni eseguite tramite gli otto bit meno significativi del registro **Parametri operativi [0008 hex]**; il bit 8 di MSB segnala la presenza di allarmi.

Struttura byte del registro **Status word [000A hex]**:

byte	MSB			LSB		
bit	15	...	8	7	...	0
	msb		lsb	msb		lsb

Byte 0

Scaling

bit 0 — Visualizza l'attuale impostazione del parametro **Funzione di scaling** in **Parametri operativi [0008 hex]**. In altre parole, visualizza lo stato di abilitazione / disabilitazione della funzione di scaling. Se "=0" la funzione di scaling è disabilitata; se invece è "=1" la funzione di scaling è abilitata. Per maggiori informazioni sull'impostazione e il significato della funzione riferirsi al parametro **Funzione di scaling** a pagina 71.

Direzione conteggio

bit 1 — Visualizza l'attuale impostazione del parametro **Direzione di conteggio** in **Parametri operativi [0008 hex]**. Se "=0" si è impostato che il valore trasmesso dall'encoder sia crescente quando l'albero ruota in senso orario; se invece è "=1" si è impostato che il valore trasmesso dall'encoder sia crescente quando l'albero ruota in senso antiorario. Per maggiori informazioni sull'impostazione e il significato della funzione riferirsi al parametro **Direzione di conteggio** a pagina 72.

bit 2 ... 7 — Non utilizzati.

Byte 1

Allarme

bit 8 — Se "=1" si è verificato un allarme, si vedano i dettagli in **Registro allarmi [0000 hex]** a pagina 76.

bit 9 ... 15 — Non utilizzati.

7.2 Codici eccezione

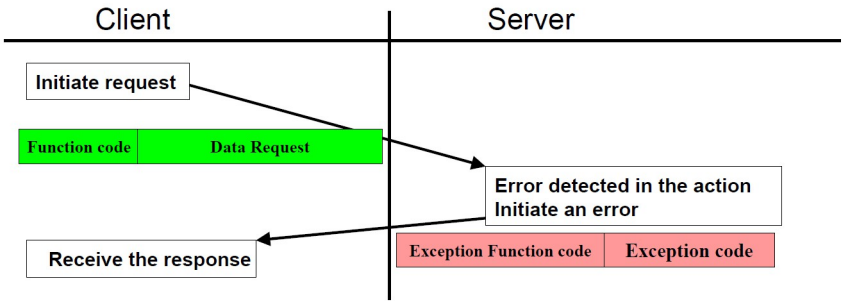
Quando un dispositivo Client invia una richiesta a un dispositivo Server, si attende poi una risposta normale. In realtà si possono verificare uno dei quattro possibili eventi descritti qui di seguito.

- 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 (bit di parità, CRC, ...), 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 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 most significant bit (msb) di 0 (i loro valori sono tutti inferiori a 80 esadecimale). In una exception response, il Server imposta il most significant bit del codice funzione a 1. Questo fa sì che il valore del codice funzione di una exception response sia maggiore del valore che esso avrebbe assunto in una risposta normale esattamente di 80 esadecimale. Con il most significant bit del function code impostato, il programma di applicazione client può riconoscere la exception response e valutare il campo dati al fine di inviare un exception code.

CAMPO DATI: in una risposta normale, il Server può ritornare dati o statistiche nel campo dati (qualsiasi informazione che fosse richiesta nella interrogazione). In un exception code, il Server scrive nel campo dati il 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.

Codici eccezione MODBUS (exception code)		
Codice	**Nome**	**Significato**
01	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.
02	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 registro 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 registri 96, 97, 98, 99. Se invece viene inviata una richiesta dove l'indirizzo del registro iniziale è 96 e la quantità di 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.

03	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 un qualunque valore di un qualunque registro.
04	SERVER DEVICE FAILURE	Si è verificato un errore non recuperabile durante il tentativo da parte del server di eseguire una azione richiesta.
05	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.
06	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.
08	MEMORY PARITY ERROR	Utilizzo specifico con i codici funzione 20 e 21 e il 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.
0A	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 è mal configurato o sovraccarico.
0B	GATEWAY TARGET DEVICE FAILED TO	Utilizzo specifico con gateway, indica che non si è avuta risposta dal dispositivo indicato. Solitamente significa che il dispositivo non è

	RESPOND	presente nella rete.

Per ogni informazione sui codici eccezione (exception code) disponibili e sul loro significato riferirsi al capitolo "MODBUS Exception Responses" alla pagina 48 del documento "MODBUS Application Protocol Specification V1.1b3".

8 Esempi di programmazione

Di seguito sono riportati alcuni esempi di lettura e impostazione dei parametri. Tutti i valori sono espressi in notazione esadecimale.

8.1 Utilizzo del codice funzione 03 Read Holding Registers

ESEMPIO 1
Richiesta di lettura del parametro **Valore di preset [0004-0005 hex]** (registri 5 e 6) allo Slave con indirizzo 1.

Request PDU (in formato esadecimale)
[01][03][00][04][00][02][85][CA]
dove:
[01] = indirizzo dello Slave
[03] = codice funzione **03 Read Holding Registers**
[00][04] = indirizzo iniziale (parametro **Valore di preset [0004-0005 hex]**, registro 5)
[00][02] = numero di registri richiesti
[85][CA] = CRC

Response PDU (in formato esadecimale)
[01][03][04][00][00][05][DC][F8][FA]
dove:
[01] = indirizzo dello Slave
[03] = codice funzione **03 Read Holding Registers**
[04] = numero di byte (2 byte per ciascun registro)
[00][00] = valore del registro 5, 00 00 hex = 0 dec
[05][DC] = valore del registro 6, 05 DC hex = 1500 dec
[F8][FA] = CRC

Come si evince, i due registri 5 e 6 del parametro **Valore di preset [0004-0005 hex]** hanno valore rispettivamente 00 00 hex e 05 DC hex, il valore impostato nel parametro è cioè 1.500 in notazione decimale.

8.2 Utilizzo del codice funzione 04 Read Input Register

ESEMPIO 1
Richiesta di lettura del parametro **Posizione corrente [0001-0002 hex]** (registri 2 e 3) allo Slave con indirizzo 1.

Request PDU (in formato esadecimale)
[01][04][00][01][00][02][20][0B]
dove:
[01] = indirizzo dello Slave
[04] = codice funzione **04 Read Input Register**
[00][01] = indirizzo iniziale (parametro **Posizione corrente [0001-0002 hex]**, registro 2)
[00][02] = numero di registri richiesti
[20][0B] = CRC

Response PDU (in formato esadecimale)
[01][04][04][00][00][2F][F0][E7][F0]
dove:
[01] = indirizzo dello Slave
[04] = codice funzione **04 Read Input Register**
[04] = numero di byte (2 byte per ciascun registro)
[00][00] = valore del registro 2 **Posizione corrente [0001-0002 hex]**, 00 00 hex = 0 dec
[2F][F0] = valore del registro 3 **Posizione corrente [0001-0002 hex]**, 2F F0 hex = 12.272 dec
[E7][F0] = CRC

I due registri 2 e 3 del parametro **Posizione corrente [0001-0002 hex]** hanno valore rispettivamente 00 00 hex e 2F F0 hex, cioè 12.272 in notazione decimale.

8.3 Utilizzo del codice funzione 06 Write Single Register

ESEMPIO 1

Richiesta di scrittura nel parametro **Parametri operativi [0008 hex]** (registro 9) dello Slave con indirizzo 1: abilitazione dello scaling (**Funzione di scaling** = 1) e conteggio crescente con rotazione oraria dell'albero dell'encoder (**Direzione di conteggio** = 0). Il valore da impostare è 00 01 hex (= 0000 0000 0000 0001 in formato binario: bit 0 **Funzione di scaling** = 1; bit 1 **Direzione di conteggio** = 0; tutti gli altri bit non sono utilizzati e perciò posti a 0).

Request PDU (in formato esadecimale)
[01][06][00][08][00][01][C9][C8]
dove:
[01] = indirizzo dello Slave
[06] = codice funzione **06 Write Single Register**
[00][08] = indirizzo del registro (parametro **Parametri operativi [0008 hex]**, registro 9)
[00][01] = valore da impostare nel registro
[C9][C8] = CRC

Response PDU (in formato esadecimale)
[01][06][00][08][00][01][C9][C8]
dove:
[01] = indirizzo dello Slave
[06] = codice funzione **06 Write Single Register**
[00][08] = indirizzo del registro (parametro **Parametri operativi [0008 hex]**, registro 9)
[00][01] = valore impostato nel registro
[C9][C8] = CRC

Come si evince, nel parametro **Parametri operativi [0008 hex]** (registro 9) si imposta il valore 00 01 hex, cioè 0000 0000 0000 0001 in notazione binaria: il bit 0 **Funzione di scaling** = 1; il bit 1 **Direzione di conteggio** = 0; tutti gli altri bit non sono utilizzati e perciò posti a 0.

8.4 Utilizzo del codice funzione 16 Write Multiple Registers

ESEMPIO 1

Richiesta di scrittura del valore 00 00 08 00 hex (= 2.048 dec) nel parametro **Informazioni per giro programmate [0000-0001 hex]** (registri 1 e 2) e del valore 00 80 00 00 hex (= 8.388.608 dec) nel parametro **Risoluzione totale programmata [0002-0003 hex]** (registri 3 e 4) dello Slave con indirizzo 1.

Request PDU (in formato esadecimale)
[01][10][00][00][00][04][08][00][00][08][00][00][80][00][00][B6][DA]
dove:
[01] = indirizzo dello Slave
[10] = codice funzione **16 Write Multiple Registers**
[00][00] = indirizzo iniziale (parametro **Informazioni per giro programmate [0000-0001 hex]**, registro 1)
[00][04] = numero di registri richiesti
[08] = numero di byte (2 byte per ciascun registro)
[00][00] = valore da impostare nel registro 1
[08][00] = valore da impostare nel registro 2, 00 00 08 00 hex = 2.048 dec
[00][80] = valore da impostare nel registro 3
[00][00] = valore da impostare nel registro 4, 00 80 00 00 hex = 8.388.608 dec
[B6][DA] = CRC

Response PDU (in formato esadecimale)
[01][10][00][00][00][04][C1][CA]
dove:
[01] = indirizzo dello Slave
[10] = codice funzione **16 Write Multiple Registers**
[00][00] = indirizzo iniziale (parametro **Informazioni per giro programmate [0000-0001 hex]**, registro 1)
[00][04] = numero di registri su cui si è scritto
[C1][CA] = CRC

Come si evince, nei due registri 1 e 2 del parametro **Informazioni per giro programmate [0000-0001 hex]** si impostano rispettivamente i valori 00 00 hex e 08 00 hex, cioè 2.048 in notazione decimale; nei due registri 3 e 4 del parametro **Risoluzione totale programmata [0002-0003 hex]** si impostano rispettivamente i valori 00 80 hex e 00 00 hex, cioè 8.388.608 in notazione decimale. L'encoder avrà quindi una risoluzione monogiro di 2.048 informazioni/giro e una risoluzione multigiro di 4.096 giri.

9 Tabella registri

9.1 Lista Holding Register con valore di default

Descrizione registro e indirizzo	Valore di default		
Informazioni per giro programmate [0000-0001 hex] imp/giro	4.096		
Risoluzione totale programmata [0002-0003 hex] imp	67.108.864		
Valore di preset [0004-0005 hex]	0		
Offset [0006-0007 hex]	0		
Funzione di scaling in Parametri operativi [0008 hex]	0		
Direzione di conteggio in Parametri operativi [0008 hex]	0		
Abilitazione Watchdog in Control Word [0009 hex]	0		
Salva parametri in Control Word [0009 hex]	-		
Carica parametri di default in Control Word [0009 hex]	-		
Esegui preset conteggio in Control Word [0009 hex]	-		

9.2 Lista Input Register

Descrizione registro e indirizzo	Descrizione dei bit
Registro allarmi [0000 hex]	0 Dati macchina non validi 1 Errore memoria flash 11 Watchdog
Posizione corrente [0001-0002 hex]	-
Registro 4 [0003 hex]	-
Elenco DM errati [0004-0005 hex]	1 Informazioni per giro programmate [0000-0001 hex] 2 Risoluzione totale programmata [0002-0003 hex] 3 Valore di preset [0004-0005 hex] 4 Offset [0006-0007 hex] 5 Parametri operativi [0008 hex] 6 DIP switch indirizzo [0007 hex] 7 DIP switch baud rate [0006 hex]
DIP switch baud rate [0006 hex]	-
DIP switch indirizzo [0007 hex]	-
Versione software [0008 hex]	-
Versione hardware [0009 hex]	-
Status word [000A hex]	0 Scaling 1 Direzione conteggio 8 Allarme

Pagina lasciata intenzionalmente bianca

Pagina lasciata intenzionalmente bianca

Versione documento	Data release	Descrizione	HW	SW	Versione interfaccia
1.0	06.12.2013	Prima stampa	V1.0	V1.0	-
1.1	16.04.2014	Aggiornamento sezioni "Informazioni preliminari" e "Quick reference"	V1.0	V2.0	-
1.2	15.02.2019	Aggiunto modello monogiro, introduzione interfaccia di configurazione, revisione generale	V1.0	V2.0	1.1.0.0
1.3	20.04.2021	Aggiornamento interfaccia di configurazione	V1.0	V2.0	2.0.0

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

Tel. +39 0445 806600
Fax +39 0445 806699

info@lika.biz • www.lika.biz