

Manuale d'uso

RD1A

RD12A



Versione RS-232



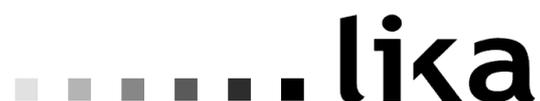
Smart encoders & actuators

Questa pubblicazione è edita da Lika Electronic s.r.l. 2014. 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 (includere 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.



Indice generale

Manuale d'uso	1
Indice generale.....	3
Indice analitico.....	8
Convenzioni grafiche e iconografiche.....	10
Informazioni preliminari.....	11
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
4 Connessioni elettriche	20
4.1 Collegamento messa a terra (Figura 1 e Figura 2).....	20
4.2 Connettori (Figura 4).....	22
4.3 Indicatori LED (Figura 4).....	24
4.4 Selettori e pulsanti (Figura 5).....	27
4.4.1 Indirizzo nodo: Node ID (Figura 5).....	28
4.4.2 Resistenza di terminazione: RT (Figura 5).....	29
4.4.3 Pulsanti JOG + e JOG - (Figura 5).....	29
4.4.4 Pulsante PRESET (Figura 5).....	29
5 Quick reference	31
6 Funzioni	33
6.1 Principio di funzionamento.....	33
6.2 Tipi di movimento: jog e posizione.....	34
Jog: controllo di velocità.....	34
Posizionamento: controllo di posizione e velocità.....	34
6.3 Ingressi e uscite digitali.....	35
6.4 Distanza_giro, Velocità Jog, Velocità di lavoro, Valore di preset e valori di finecorsa.....	36
7 Tabella parametri di default	40
Interfaccia Profibus®	41
1 Programmazione con Siemens STEP7	42
1.1 Configurazione tramite Siemens STEP7.....	42
1.1.1 Importazione file GSD.....	42
1.1.2 Aggiungere il nodo al progetto.....	44
1.1.3 Configurazione dei parametri "Dati macchina".....	45
1.2 Lettura della diagnostica.....	47
1.3 Impostazione e lettura parametri.....	49
1.3.1 Impostazione parametro 28 Valore di preset.....	49
1.3.2 Lettura parametro 29 Velocità attuale.....	51
2 Interfaccia Profibus®	53
2.1 File GSD.....	53
2.2 Baud rate.....	55
2.3 Funzionamento a stati.....	56
Tipi di messaggi.....	56
2.4 DDLM_Set_Prm.....	57
2.5 DDLM_Chk_Cfg.....	58

2.6 DDLM_Data_Exchange.....	59
2.6.1 Funzione Master → Slave.....	59
Control Word (Byte 0 e 1).....	59
Jog +.....	59
Jog -.....	60
Stop.....	60
Reset allarmi.....	60
Jog incrementale.....	60
Start.....	61
Emergenza.....	61
Salva parametri.....	61
Carica parametri di default.....	61
Asse in coppia.....	62
OUT 1.....	62
Sblocco freno.....	62
Target position (Byte 4 ... 7).....	63
Numero parametro (Byte 8).....	64
Valore parametro (Byte 9 ... 12).....	65
2.6.2 Funzione Slave → Master.....	67
Status word (Byte 0 e 1).....	67
Asse in posizione.....	67
Stato Profibus.....	67
Asse abilitato.....	67
Finecorsa SW +.....	67
Finecorsa SW -.....	67
Allarme.....	67
Asse in movimento.....	68
Comando in corso.....	68
Target raggiunto.....	68
Pulsante 1 Jog +.....	68
Pulsante 2 Jog -.....	68
Pulsante 3 Preset.....	68
Saturazione DAC.....	69
IN 1.....	69
IN 2.....	69
IN 3.....	69
Allarmi (Byte 2 e 3).....	69
Dati macchina non validi.....	69
Errore memoria flash.....	69
Errore di inseguimento.....	70
Asse non sincronizzato.....	70
Target non valido.....	70
Emergenza.....	70
Sovracorrente.....	70
Sovratemperatura.....	70
Indirizzo non valido.....	70
Sottotensione.....	70
Sola lettura.....	70
Assenza comunicazione Profibus.....	70
Posizione (Byte 4 ... 7).....	71
Numero parametro (Byte 8).....	71

Valore parametro (Byte 9..12).....	71
2.7 DDLM_Slave_Diag.....	74
3 Parametri di programmazione Profibus®.....	75
Parametri Dati macchina.....	76
01 Distanza_giro.....	76
02 Tolleranza di posizione.....	77
03 Tempo asse in tolleranza.....	77
04 Max errore di inseguimento.....	77
05 Kp anello di posizione.....	78
06 Ki anello di posizione.....	78
07 Accelerazione.....	78
08 Decelerazione.....	78
09 Delta positivo.....	78
0A Delta negativo.....	79
0B Velocità Jog.....	80
0C Velocità di lavoro.....	80
0D Durata corrente di stacco.....	81
0E Direzione conteggio.....	81
0F Kp anello di corrente.....	81
10 Ki anello di corrente.....	82
11 Corrente massima.....	82
12 Corrente di stacco.....	82
13 Rapporto di riduzione.....	82
14 Ampiezza passo jog.....	82
Parametri dati operativi.....	83
28 Valore di preset.....	83
29 Velocità attuale.....	84
2A Temperatura.....	84
2B Corrente attuale.....	84
2C Errore di posizione.....	84
2D Versione software.....	84
2E Versione hardware.....	85
2F Valore di offset.....	85
30 Finecorsa assoluto positivo.....	85
31 Finecorsa assoluto negativo.....	86
32 Lista parametri errati.....	86
Interfaccia Modbus®.....	88
1 Utilizzo con seriale di servizio.....	89
1.1 Configurazione mediante software di Lika Electronic.....	89
1.2 Pagina "Configurazione seriale".....	92
1.3 Pagina "Modalità operativa".....	95
1.4 Pagina "Dati macchina".....	100
1.5 Pagina "Monitor messaggi".....	103
1.6 Pagina "Test Lika".....	104
1.7 Pagina "Upgrade Firmware".....	104
1.7.1 In caso di errore.....	107
2 Interfaccia Modbus®.....	109
2.1 Principi guida del protocollo Modbus Master / Slave.....	109
2.2 Frame Modbus.....	110
2.3 Modalità di trasmissione.....	111
2.3.1 Modalità di trasmissione RTU.....	111

2.4 Codici funzione.....	114
2.4.1 Codici funzione implementati.....	114
03 Read Holding Registers.....	114
04 Read Input Register.....	116
06 Write Single Register.....	118
16 Write Multiple Registers.....	120
3 Parametri di programmazione Modbus®.....	124
3.1 Parametri disponibili.....	124
3.1.1 Parametri Dati macchina.....	124
Distanza_giro [0x00].....	125
Tolleranza di posizione [0x01].....	126
Tempo asse in tolleranza [0x02].....	126
Max errore di inseguimento [0x03].....	127
Kp anello di posizione [0x04].....	127
Ki anello di posizione [0x05].....	127
Accelerazione [0x06].....	127
Decelerazione [0x07].....	127
Delta spazio positivo [0x08-0x09].....	128
Delta spazio negativo [0x0A-0x0B].....	129
Velocità Jog [0x0C].....	130
Velocità di lavoro [0x0D].....	130
Durata corrente di stacco [0x0E].....	130
Direzione conteggio [0x0F].....	131
Kp anello di corrente [0x10].....	131
Ki anello di corrente [0x11].....	131
Corrente massima [0x12].....	131
Corrente di stacco [0x13].....	132
Valore di offset [0x14-0x15].....	132
Valore di preset [0x16-0x17].....	132
Rapporto di riduzione [0x18].....	133
Ampiezza passo jog [0x19].....	133
Registro comandi Extra [0x29].....	133
Lettura assoluta.....	133
Controllo da PC.....	133
Control Word [0x2A].....	134
Jog +.....	134
Jog -.....	134
Stop.....	134
Reset allarmi.....	135
Jog incrementale.....	135
Start.....	135
Emergenza.....	135
Abilitazione Watch dog.....	136
Salva parametri.....	136
Carica parametri di default.....	136
Esegui preset conteggio.....	136
Asse in coppia.....	136
OUT 1.....	137
Sblocco freno.....	137
Target position [0x2B-0x2C].....	137
3.1.2 Parametri Input Register.....	139

Registro allarmi [0x00].....	139
Dati macchina non validi.....	139
Errore memoria flash.....	139
Errore di inseguimento.....	139
Asse non sincronizzato.....	139
Target non valido.....	139
Emergenza.....	140
Sovracorrente.....	140
Sovratemperatura.....	140
Sottotensione.....	140
Watch dog.....	140
Status word [0x01].....	141
Asse in posizione.....	141
Asse abilitato.....	141
Finecorsa SW +.....	141
Finecorsa SW -.....	141
Allarme.....	142
Asse in movimento.....	142
Comando in corso.....	142
Target raggiunto.....	142
Pulsante 1 Jog +.....	142
Pulsante 2 Jog -.....	142
Pulsante 3 Preset.....	142
Saturazione DAC.....	143
IN 1.....	143
IN 2.....	143
IN 3.....	143
Posizione corrente [0x02-0x03].....	143
Velocità corrente [0x04].....	143
Errore di inseguimento [0x05-0x06].....	144
Corrente attuale [0x07].....	144
Temperatura [0x08].....	144
Elenco DM errati [0x09-0x0A].....	144
I2t [0x0B].....	145
Versione software [0x0E].....	145
Versione hardware [0x0F].....	146
3.2 Codici eccezione.....	148
4 Esempi di programmazione Modbus®.....	149
4.1 Utilizzo del codice funzione 03 Read Holding Registers.....	149
4.2 Utilizzo del codice funzione 04 Read Input Register.....	150
4.3 Utilizzo del codice funzione 06 Write Single Register.....	152
4.4 Utilizzo del codice funzione 16 Write Multiple Registers.....	154

Indice analitico

0

01 Distanza_giro.....	76
02 Tolleranza di posizione.....	77
03 Tempo asse in tolleranza.....	77
04 Max errore di inseguimento.....	77
05 Kp anello di posizione.....	78
06 Ki anello di posizione.....	78
07 Accelerazione.....	78
08 Decelerazione.....	78
09 Delta positivo.....	78
0A Delta negativo.....	79
0B Velocità Jog.....	80
0C Velocità di lavoro.....	80
0D Durata corrente di stacco.....	81
0E Direzione conteggio.....	81
0F Kp anello di corrente.....	81

1

10 Ki anello di corrente.....	82
11 Corrente massima.....	82
12 Corrente di stacco.....	82
13 Rapporto di riduzione.....	82
14 Ampiezza passo jog.....	82

2

28 Valore di preset.....	83
29 Velocità attuale.....	84
2A Temperatura.....	84
2B Corrente attuale.....	84
2C Errore di posizione.....	84
2D Versione software.....	84
2E Versione hardware.....	85
2F Valore di offset.....	85

3

30 Finecorsa assoluto positivo.....	85
31 Finecorsa assoluto negativo.....	86
32 Lista parametri errati.....	86

A

Abilitazione Watch dog.....	136
Accelerazione [0x06].....	127
Allarme.....	67, 142
Allarmi (Byte 2 e 3).....	69
Ampiezza passo jog [0x19].....	133
Asse abilitato.....	67, 141
Asse in coppia.....	62, 136
Asse in movimento.....	68, 142
Asse in posizione.....	67, 141
Asse non sincronizzato.....	70, 139

Assenza comunicazione Profibus.....	70
-------------------------------------	----

C

Carica parametri di default.....	61, 136
Comando in corso.....	68, 142
Control Word (Byte 0 e 1).....	59
Control Word [0x2A].....	134
Controllo da PC.....	133
Corrente attuale [0x07].....	144
Corrente di stacco [0x13].....	132
Corrente massima [0x12].....	131

D

Dati macchina non validi.....	69, 139
Decelerazione [0x07].....	127
Delta spazio negativo [0x0A-0x0B].....	129
Delta spazio positivo [0x08-0x09].....	128
Direzione conteggio [0x0F].....	131
Distanza_giro [0x00].....	125
Durata corrente di stacco [0x0E].....	130

E

Elenco DM errati [0x09-0x0A].....	144
Emergenza.....	61, 70, 135, 140
Errore di inseguimento.....	70, 139
Errore di inseguimento [0x05-0x06].....	144
Errore memoria flash.....	69, 139
Esegui preset conteggio.....	136

F

Finecorsa SW -.....	67, 141
Finecorsa SW +.....	67, 141

I

I2t [0x0B].....	145
IN 1.....	69, 143
IN 2.....	69, 143
IN 3.....	69, 143
Indirizzo non valido.....	70

J

Jog -.....	60, 134
Jog +.....	59, 134
Jog incrementale.....	60, 135

K

Ki anello di corrente [0x11].....	131
Ki anello di posizione [0x05].....	127
Kp anello di corrente [0x10].....	131
Kp anello di posizione [0x04].....	127

L

Lettura assoluta.....	133
-----------------------	-----

M

Max errore di inseguimento [0x03].....	127
N	
Numero parametro (Byte 8).....	64, 71
O	
OUT 1.....	62, 137
P	
Posizione (Byte 4 ... 7).....	71
Posizione corrente [0x02-0x03].....	143
Pulsante 1 Jog +.....	68, 142
Pulsante 2 Jog -.....	68, 142
Pulsante 3 Preset.....	68, 142
R	
Rapporto di riduzione [0x18].....	133
Registro allarmi [0x00].....	139
Registro comandi Extra [0x29].....	133
Reset allarmi.....	60, 135
S	
Salva parametri.....	61, 136
Saturazione DAC.....	69, 143
Sblocco freno.....	62, 137
Sola lettura.....	70
Sottotensione.....	70, 140
Sovracorrente.....	70, 140
Sovratemperatura.....	70, 140

Start.....	61, 135
Stato Profibus.....	67
Status word (Byte 0 e 1).....	67
Status word [0x01].....	141
Stop.....	60, 134
T	
Target non valido.....	70, 139
Target position (Byte 4 ... 7).....	63
Target position [0x2B-0x2C].....	137
Target raggiunto.....	68, 142
Temperatura [0x08].....	144
Tempo asse in tolleranza [0x02].....	126
Tolleranza di posizione [0x01].....	126
V	
Valore di offset [0x14-0x15].....	132
Valore di preset [0x16-0x17].....	132
Valore parametro (Byte 9 ... 12).....	65
Valore parametro (Byte 9...12).....	71
Velocità corrente [0x04].....	143
Velocità di lavoro [0x0D].....	130
Velocità Jog [0x0C].....	130
Versione hardware [0x0F].....	146
Versione software [0x0E].....	145
W	
Watch dog.....	140

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 **ARANCIONE**;
- 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 delle **unità di posizionamento ROTADRIVE modelli RD1A e RD12A**.

Le unità RD1A e RD12A sono dispositivi di movimentazione che integrano in uno un motore brushless con riduttore, un azionamento, un encoder assoluto multigiro e la logica per il posizionamento. Esse trovano impiego in ogni settore industriale e sono adatte a svolgere funzioni di posizionamento di assi secondari come per esempio, tra le tante, cambioformati, battute mobili, cambio utensili, movimento ventose.

La versione RD12A si differenzia dalla versione RD1A per la presenza di un freno di stazionamento integrato. Il freno ha lo scopo di inibire la possibilità di qualsiasi movimento dell'asse all'arresto del dispositivo.

Le interfacce disponibili per la comunicazione in bus di campo sono le seguenti: **Modbus RTU, Profibus-DP e CANopen DS 301**.

Nella versione Modbus la parametrizzazione e la messa in funzione possono essere effettuate tramite un software progettato da Lika Electronic e compreso nella fornitura che consente l'impostazione dei parametri di lavoro dell'unità ROTADRIVE, l'attivazione e il controllo dei movimenti programmati e il monitoraggio del funzionamento da qualsiasi PC con sistema operativo Windows (Windows XP o successivo). Nelle versioni Profibus e CANopen la parametrizzazione e la messa in funzione possono essere effettuate utilizzando lo stesso software, tramite un'**interfaccia seriale di servizio RS-232 in conformità con il protocollo Modbus**.

Per una più agevole consultazione questo manuale è diviso in tre parti.

Nella prima parte sono fornite le informazioni generali riguardanti l'unità di posizionamento 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. In questa sezione i loghi Profibus e Modbus che dovessero comparire in prossimità del testo identificano le informazioni che riguardano specificatamente una delle due interfacce. Eventuali ulteriori differenziazioni sono sottolineate all'interno del testo.

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

Nella terza parte infine, intitolata **Interfaccia Modbus**, sono fornite tutte le informazioni sia generali che specifiche relative all'interfaccia Modbus. Come detto in precedenza, le versioni Profibus e CANopen dell'unità ROTADRIVE sono fornite di un'interfaccia seriale di servizio RS-232 in protocollo Modbus. Tramite un software specifico progettato da Lika Electronic e compreso nella fornitura essa permette la parametrizzazione e la messa a punto del dispositivo prima dell'installazione in una rete Profibus o CANopen. Nella sezione **Interfaccia Modbus** sono descritte le caratteristiche dell'interfaccia e i registri Modbus che l'unità implementa.



ATTENZIONE

Utilizzando i moduli **Lika RD1xA-T12, Lika RD1xA-T24, Lika RD1xA-T48 e Lika RD1xA-T92** il valore di tutti i parametri è caricato all'accensione prendendo i dati memorizzati sul PLC a partire dal file GSD. Ogni modifica locale tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** è perciò temporanea: allo spegnimento del dispositivo ogni valore impostato viene perso (a eccezione del preset, unico parametro non compreso

nel file GSD; e, per tutti i parametri, nel caso in cui si esegua precedentemente un'impostazione del preset, si veda alla pagina 83) e alla successiva riaccensione viene caricato il valore presente nel PLC (per cui anche i parametri salvati a seguito di un preset sono poi comunque sovrascritti).

Questo dispositivo prevede altresì la possibilità di installazione del modulo **Lika RD1xA-no param** (disponibile a partire dalla versione H3S3, file GSD V5). Con questo modulo è possibile bypassare il trasferimento dei parametri dal PLC all'accensione (i parametri infatti vengono letti dalla memoria flash) e memorizzare in flash (tramite il bit 9 **Salva parametri** in **Control Word (Byte 0 e 1)**) ogni modifica locale effettuata mediante **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)**. Con il modulo **Lika RD1xA-no param** NON è possibile leggere e modificare i parametri tramite la pagina **Parametrizza** della finestra **Proprietà slave DP** di STEP7 (si veda il paragrafo "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). La modifica dei parametri è quindi possibile tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** oppure tramite l'interfaccia Modbus. Il modulo **Lika RD1xA-no param** è indipendente dal rapporto di riduzione e può essere installato quale che sia quello del dispositivo che si va a programmare. Naturalmente i parametri da impostare dovranno tenere conto delle caratteristiche meccaniche ed elettriche del dispositivo.

Utilizzando l'interfaccia seriale RS-232 di servizio Modbus è possibile impostare e salvare i parametri di modo che, anche a seguito di uno spegnimento, i valori impostati siano mantenuti. Questo però vale solamente fino a che sia mantenuta attiva l'interfaccia seriale: infatti nel momento in cui il dispositivo sia collegato alla rete Profibus, saranno caricati i dati memorizzati nel PLC e di conseguenza i dati impostati tramite la seriale Modbus saranno sovrascritti.

Utilizzando STEP7 di Siemens invece è possibile memorizzare in maniera permanente ogni modifica al valore di un parametro (eccetto che nel modulo -no param). Per fare questo modificare il valore dei parametri nella pagina **Proprietà slave DP** di STEP7 (si veda alla sezione "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). Si badi che, al contrario, le modifiche realizzate mediante la tabella delle variabili (si veda alla sezione "1.3.1 Impostazione parametro 28 Valore di preset" a pagina 49) sono temporanee.

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";
- è necessario prevedere l'installazione di un pulsante di emergenza per l'interruzione dell'alimentazione al motore in caso di necessità;
- in conformità alla normativa 2004/108/CE 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 "Montaggio meccanico";
- 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;
- dispositivo con albero sporgente: utilizzare giunti elastici per collegare ROTADRIVE e asse utilizzatore; rispettare le tolleranze di allineamento ammesse dal giunto elastico;
- dispositivo con albero cavo: l'unità ROTADRIVE può essere montata direttamente su un albero che rispetti le caratteristiche definite nel foglio d'ordine e fissata mediante il collare e l'asola per l'introduzione di un pin antirotazione.



ATTENZIONE

La taratura dell'unità è stata realizzata testandone il funzionamento a vuoto; i valori di default impostati si riferiscono pertanto a un dispositivo operante in questa modalità. Essi sono altresì previsti per garantire un funzionamento a vuoto e sicuro del dispositivo, che potrà talora non risultare ottimale né prestazionale. Si badi quindi che nella specifica applicazione può essere consigliabile, se non necessaria, la modifica dei parametri di fabbrica e in particolare dei valori di velocità, accelerazione, decelerazione e guadagno.



ATTENZIONE

La tensione controelettromotrice generata dal motore a seguito di una movimentazione manuale dell'asse forzata dall'esterno può provocare danni irreparabili alla circuiteria interna.

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 s.r.l. 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.



3 Installazione meccanica



ATTENZIONE

L'installazione e le operazioni di manutenzione devono essere eseguite da personale qualificato, in assenza di tensione e movimento del motore e dell'albero.

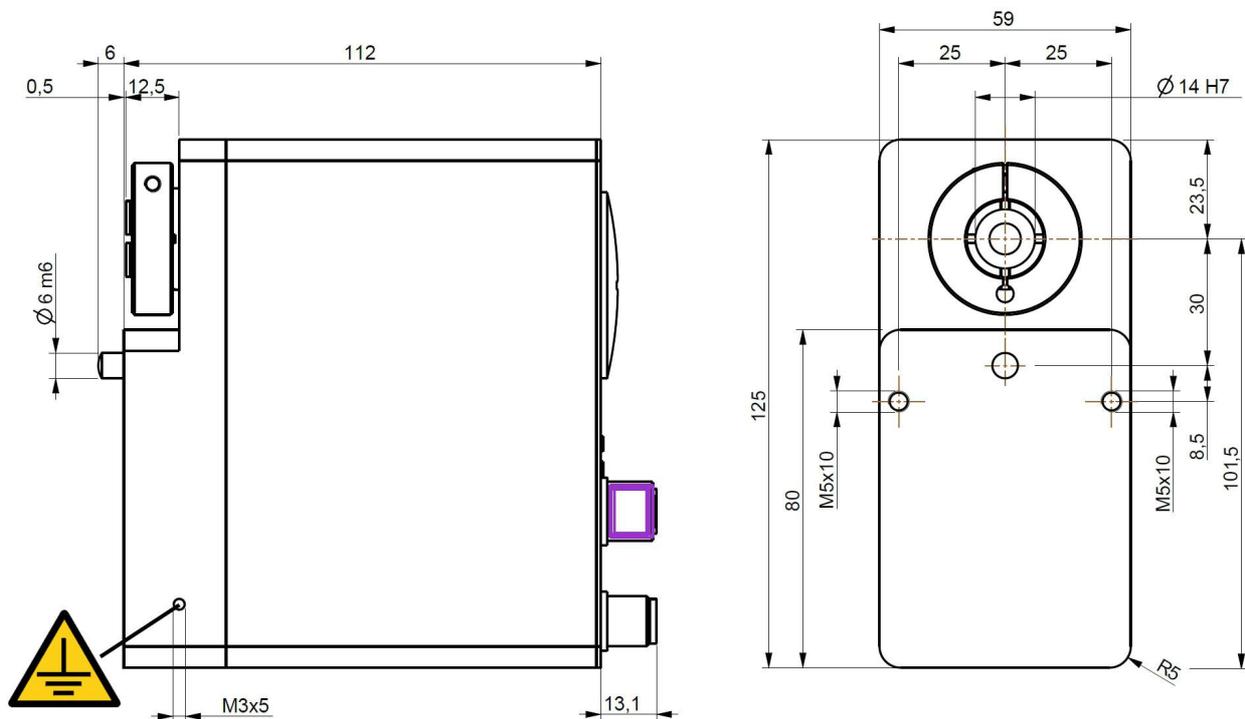


Figura 1 - Lay-out unità RD1A

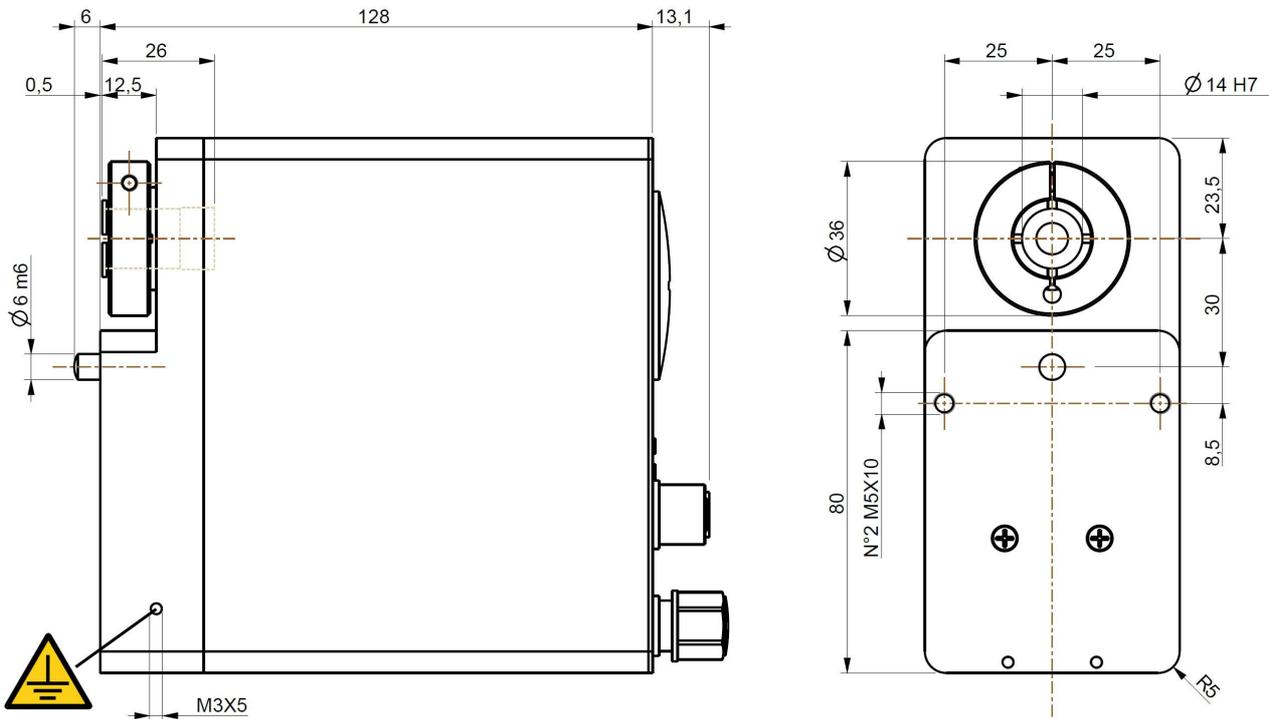


Figura 2 - Lay-out unità RD12A



L'unità ROTADRIVE deve essere fissata in maniera solidale esclusivamente all'asse utilizzatore mediante il collare. E' fornita in dotazione di un gommino antivibrazione sul quale andrà a inserirsi il pin antirotazione, anch'esso fornito in dotazione. Questo assicura al dispositivo stabilità e contemporaneamente la mobilità necessaria per assorbire le tensioni meccaniche sviluppate durante il funzionamento. Non fissare in maniera rigida il pin antirotazione al supporto fisso lato utilizzatore senza il gommino antivibrazione! Ugualmente non portare in battuta l'unità ROTARIVE contro il supporto fisso lato utilizzatore. Se questo avvenisse le tensioni meccaniche sarebbero completamente concentrate sull'albero motore con conseguente danneggiamento dei cuscinetti!

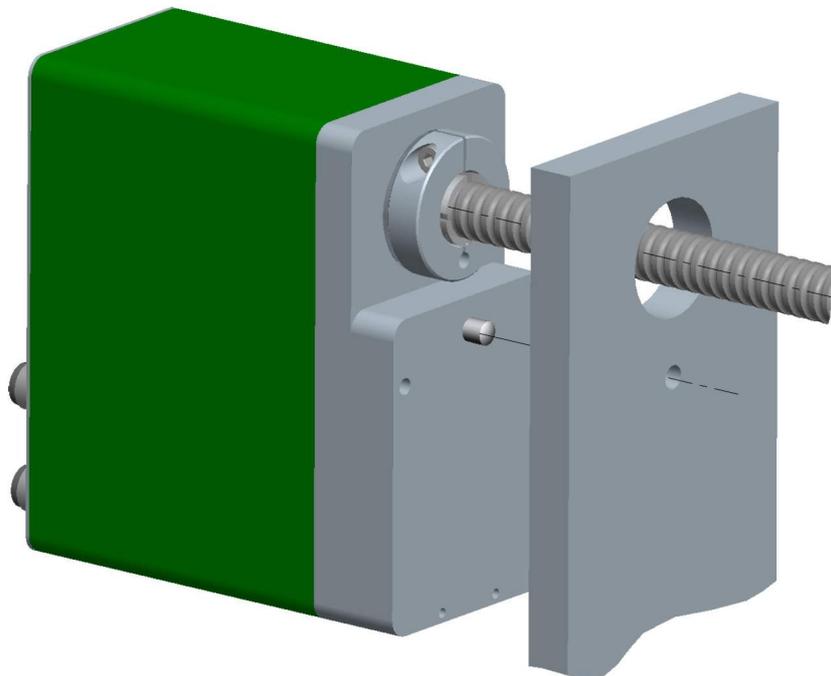
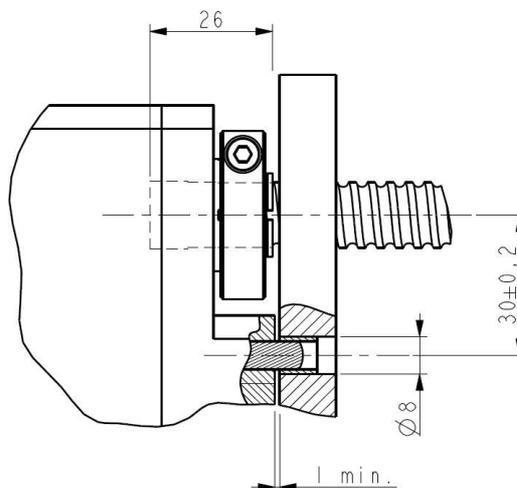


Figura 3 - Esempio di installazione di unità RD1xA su vite senza fine

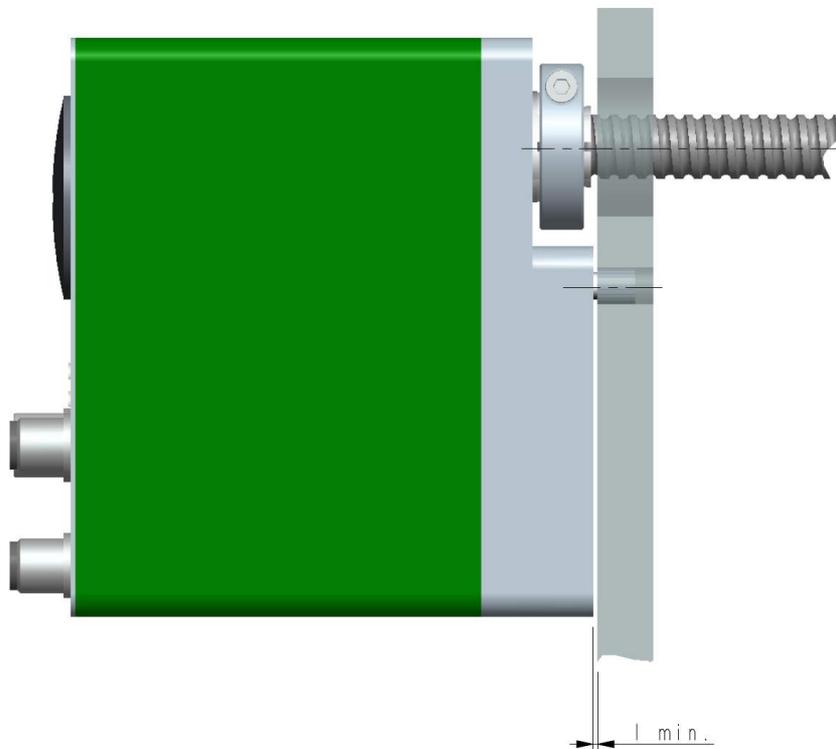
Per installare correttamente l'unità ROTADRIVE seguire scrupolosamente le istruzioni riportate qui di seguito, tenendo conto a ogni modo che le possibilità di installazione possono essere molteplici e dipendenti dalla specifica applicazione.

- Eseguire un foro $\varnothing 8$ mm sulla flangia o sul supporto lato utilizzatore per l'inserimento del gommino antivibrazione e del pin antirotazione. La distanza tra l'asse dell'albero e l'asse del foro deve essere di $30 \pm 0,2$ mm. Assicurarsi che il foro e l'albero siano perfettamente centrati sull'asse verticale. Eventuali disallineamenti risulterebbero in tensioni



meccaniche sull'albero motore con conseguente danneggiamento dei cuscinetti!

- inserire il gommino antivibrazione nel foro realizzato;
- inserire l'albero utilizzatore nell'asse cavo dell'unità ROTADRIVE per una profondità massima di 26 mm e assicurarsi che anche il pin antirotazione si vada a inserire nel gommino antivibrazione;
- la distanza minima tra la flangia del dispositivo ROTADRIVE e il supporto fisso sul lato utilizzatore deve essere di almeno 1 mm per evitare ogni possibile contatto;
- fissare l'albero utilizzatore mediante il collare e la relativa vite di fissaggio.



ATTENZIONE

Non forzare manualmente la rotazione dell'albero al fine di evitare danni permanenti! La tensione controelettromotrice generata dal motore a seguito di una movimentazione manuale dell'asse forzata dall'esterno può provocare danni irreparabili alla circuiteria interna.

4 Connessioni elettriche



ATTENZIONE

Questa operazione deve essere eseguita in assenza di tensione!

La trasmissione dei comandi **Start**, **Jog +** e **Jog -** procura l'avvio del movimento dell'unità e dell'asse. Assicurarsi che in conseguenza di questo non possano verificarsi lesioni alle persone o danneggiamenti meccanici.

Ogni routine di **Start** deve essere preventivamente valutata con scrupolo!

Non forzare manualmente la rotazione dell'asse al fine di evitare danni permanenti!

4.1 Collegamento messa a terra (Figura 1 e Figura 2)

Collegare il corpo del dispositivo a un buon punto di terra; si consiglia di utilizzare il punto di collegamento previsto (si vedano la Figura 1 e la Figura 2). Collegare la calza del cavo a un buon punto di terra sul lato utilizzatore. Collegare la calza del cavo al pressacavo del connettore metallico; si veda anche la nota al paragrafo successivo. In tutti i casi assicurarsi che il punto di terra sia privo di disturbi e il più vicino possibile al dispositivo.

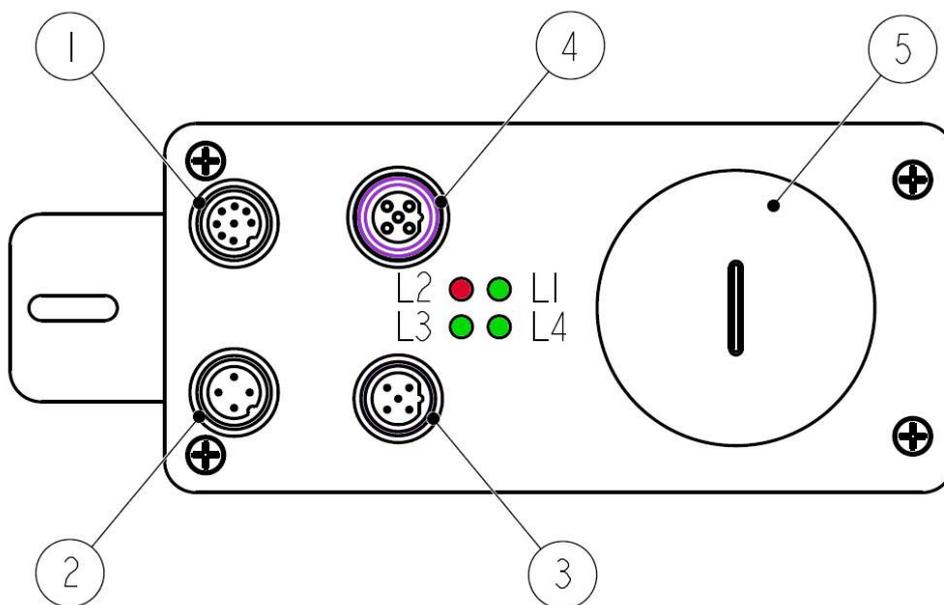


Figura 4: Connessioni elettriche

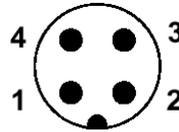
Legenda

1	Connettore M12 maschio 8 pin ingressi / uscite + RS-232 Modbus
2	Connettore M12 maschio 4 pin alimentazione
3	Connettore M12 maschio 5 pin BUS IN Profibus
4	Connettore M12 femmina 5 pin BUS OUT Profibus
5	Alloggiamento interno selettori e pulsanti
L1	LED 1 informazione alimentazione controller
L2	LED 2 presenza errori
L3	LED 3 stato bus di campo
L4	LED 4 informazione alimentazione motore

4.2 Connettori (Figura 4)

Alimentazione

connettore M12 maschio 4 pin
 codifica A
 (vista lato contatti)

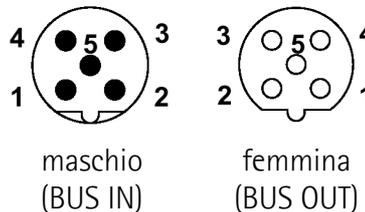


Pin	Funzione
1	+24VDC ±10% motore
2	+24VDC ±10% controllore
3	0 VDC motore e controllore
4	n.c.

n.c. = non connesso

Interfaccia Profibus

connettore M12 5 pin
 codifica B
 (vista lato contatti)



Pin	Funzione
1	n.c.
2	Profibus A (Verde)
3	n.c.
4	Profibus B (Rosso)
5	n.c.
Custodia	Calza ¹

n.c. = non connesso

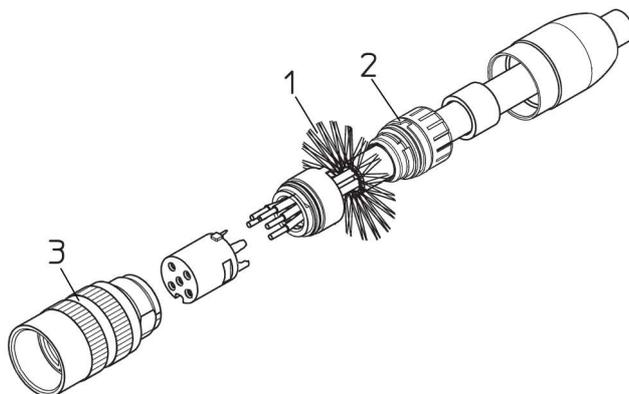
1 Collegare la calza del cavo al pressacavo del connettore metallico

Per il collegamento del bus si raccomanda di usare connettori e cavi certificati Profibus-DP.

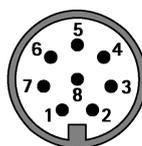


NOTA

E' fondamentale che per la trasmissione dei segnali nella rete Profibus si utilizzino cavi schermati e che la calza dei cavi sia opportunamente collegata alla ghiera metallica del connettore per una efficace messa a terra attraverso il corpo del dispositivo. Per questo bisogna districare la calza 1 e tagliarla alla giusta misura; quindi piegarla sul particolare 2; infine posizionare la ghiera 3 assicurandosi che la calza 1 e la ghiera 3 siano adeguatamente in contatto.



Ingressi / uscite + RS-232 Modbus connettore M12 maschio 8 pin



(vista lato contatti)

Pin	Funzione
1	0 VDC
2	Ingresso 1
3	Ingresso 2
4	Ingresso 3
5	Uscita 1
6	TD (RS-232) ¹
7	RD (RS-232) ¹
8	0 VDC (RS-232) ¹

¹ Solo per porta seriale di servizio Modbus. Per ogni informazione sulla configurazione e l'utilizzo della seriale di servizio RS-232 riferirsi alla sezione "Interfaccia Modbus®" a pagina 88.

4.3 Indicatori LED (Figura 4)

Quattro led, posizionati in prossimità dei due connettori BUS (si veda la Figura 4), segnalano visivamente la condizione di funzionamento del bus di campo attivo e del dispositivo, come esplicitato nelle seguenti tabelle.



Si badi che i led hanno significato diverso a seconda che sia attiva la rete Profibus oppure la rete Modbus.



LED 1 VERDE		Descrizione
ON		Elettronica di controllo alimentata, presenza di tensione
OFF		Elettronica di controllo non alimentata, assenza di tensione

LED 2 ROSSO	LED 3 VERDE	Descrizione
FAULT	STATUS	Led rete Profibus
OFF	OFF	Dispositivo non alimentato o anomalia hardware non diagnosticabile
OFF	ON	Funzionamento normale in modalità Data_Exchange
ON	Lampeggiante	Presenza di allarmi (per una lista completa degli allarmi riferirsi a pagina 69); o parametri di configurazione non validi
Lampeggiante	ON	Mancanza di comunicazione con il bus
Lampeggiante	Lampeggiante	Errore memoria interna, errore non recuperabile

LED 2	LED 3	Descrizione
VERDE Lampeggiante	VERDE Lampeggiante	Durante il trasferimento dei dati alla memoria flash per l'aggiornamento del firmware del dispositivo (si veda la sezione "1.7 Pagina "Upgrade Firmware"" a pagina 104), i due led lampeggiano verdi a una frequenza di 5 Hz.
ROSSO Lampeggiante	ROSSO Lampeggiante	Durante il trasferimento dei dati alla memoria flash per l'aggiornamento del firmware del dispositivo (si veda la sezione "1.7 Pagina "Upgrade Firmware"" a pagina 104), se si verifica un errore che pregiudica l'upgrade (per esempio: la caduta di tensione e lo spegnimento dell'unità ROTADRIVE), l'assenza dello user program all'accensione procura l'attivazione dei due led che lampeggiano rossi a una frequenza di 5 Hz. Per ogni informazione sul ripristino dell'unità riferirsi alla sezione "1.7 Pagina "Upgrade Firmware"" a pagina 104.
ROSSO ON	ROSSO ON	Durante il trasferimento dei dati alla memoria flash per l'aggiornamento del firmware del dispositivo (si

		veda la sezione "1.7 Pagina "Upgrade Firmware"" a pagina 104), 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 "1.7 Pagina "Upgrade Firmware"" a pagina 104.
LED 4 VERDE		Descrizione
ON		Motore alimentato, presenza di tensione
OFF		Motore non alimentato, assenza di tensione



LED 1 VERDE		Descrizione
ON		Elettronica di controllo alimentata, presenza di tensione
OFF		Elettronica di controllo non alimentata, assenza di tensione
LED 2 ROSSO		Descrizione
ON		Sono presenti allarmi, errore interno
OFF		Nessun allarme presente
LED 3 VERDE		Descrizione
Lampeggiante		Dispositivo in trasmissione o ricezione
OFF		Nessuna trasmissione o ricezione attiva
LED 2	LED 3	Descrizione
VERDE Lampeggiante	VERDE Lampeggiante	Durante il trasferimento dei dati alla memoria flash per l'aggiornamento del firmware del dispositivo (si veda la sezione "1.7 Pagina "Upgrade Firmware"" a pagina 104), i due led lampeggiano verdi a una frequenza di 5 Hz.
ROSSO Lampeggiante	ROSSO Lampeggiante	Durante il trasferimento dei dati alla memoria flash per l'aggiornamento del firmware del dispositivo (si veda la sezione "1.7 Pagina "Upgrade Firmware"" a pagina 104), se si verifica un errore che pregiudica l'upgrade (per esempio: la caduta di tensione e lo spegnimento dell'unità ROTADRIVE), l'assenza dello user program all'accensione procura l'attivazione dei due led che lampeggiano rossi a una frequenza di 5 Hz. Per ogni informazione sul ripristino dell'unità riferirsi alla sezione "1.7 Pagina "Upgrade Firmware"" a pagina 104.
ROSSO ON	ROSSO ON	Durante il trasferimento dei dati alla memoria flash per l'aggiornamento del firmware del dispositivo (si veda la sezione "1.7 Pagina "Upgrade Firmware"" a pagina 104), 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 "1.7 Pagina "Upgrade Firmware"" a pagina 104.
LED 4 VERDE		Descrizione
ON		Motore alimentato, presenza di tensione
OFF		Motore non alimentato, assenza di tensione

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

4.4 Selettori e pulsanti (Figura 5)



ATTENZIONE

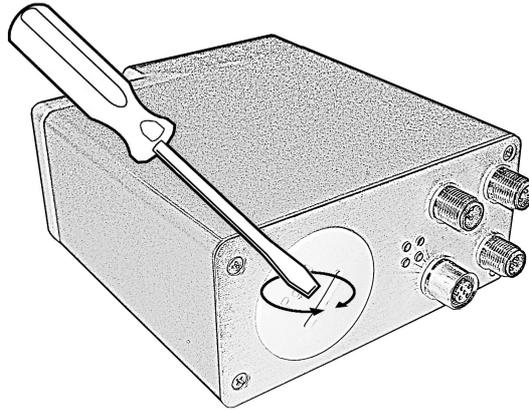
Questa operazione deve essere eseguita con dispositivo non alimentato!



NOTA

Eeguire questa operazione con estrema prudenza per non danneggiare le connessioni.

Per accedere ai selettori e ai pulsanti svitare e rimuovere il tappo a vite con filettatura PG 29. Avere cura di ripristinare il coperchio al termine dell'operazione.



Accedere quindi ai sottostanti selettori e pulsanti.

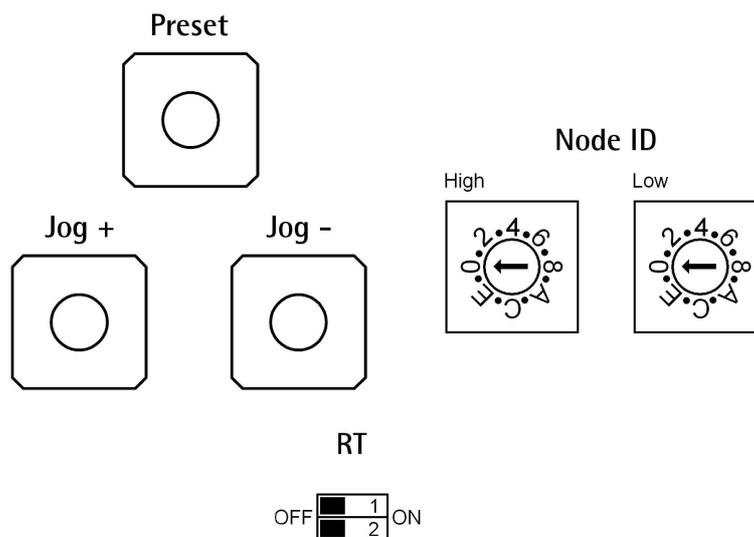


Figura 5: Selettori e pulsanti

4.4.1 Indirizzo nodo: Node ID (Figura 5)



ATTENZIONE

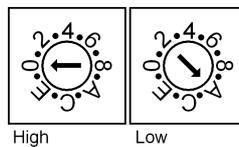
Questa operazione deve essere eseguita con dispositivo non alimentato!

Impostare il valore esadecimale dell'indirizzo del nodo.
L'indirizzo del nodo deve avere un valore compreso tra 1 e 125 (125 = 7D hex).

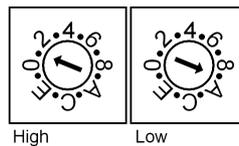


Esempio

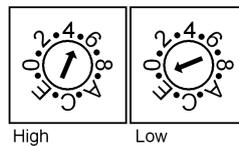
Indirizzo 10 = 0A hex:



Indirizzo 25 = 19 hex:



Indirizzo 95 = 5F hex:



NOTA

L'indirizzo assegnato tramite questo selettore è valido per entrambi i bus di campo Profibus e Modbus.

L'indirizzo di default è 1.

Se si imposta l'indirizzo = 0, il dispositivo utilizzerà automaticamente l'indirizzo 1 (l'indirizzo 0 è riservato al Master).

Se si imposta un indirizzo maggiore di 125, il dispositivo utilizzerà automaticamente l'indirizzo 125.

4.4.2 Resistenza di terminazione: RT (Figura 5)

La resistenza di terminazione RT deve essere utilizzata come linea di terminazione sull'ultimo dispositivo della rete. Per attivarla si agisce sullo switch RT.

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

4.4.3 Pulsanti JOG + e JOG - (Figura 5)

Premere questi pulsanti per comandare i movimenti in direzione positiva (JOG +) e negativa (JOG -) nella modalità di funzionamento manuale. Per ogni informazione si consultino i comandi **Jog +** e **Jog -** a pagina 59 (interfaccia Profibus) o a pagina 134 (interfaccia Modbus).



NOTA

Si badi che nell'utilizzo dei pulsanti manuali la funzione di jog incrementale (si veda **Jog incrementale** in **Control Word (Byte 0 e 1)** a pagina 60 -versione Profibus; **Jog incrementale** in **Control Word [0x2A]** a pagina 135 -versione Modbus) è disabilitata; non è cioè possibile eseguire passi jog utilizzando i pulsanti manuali. Pertanto i movimenti positivo e negativo sono comandati solo mediante la pressione continuata dei pulsanti e hanno termine in corrispondenza dei rispettivi finecorsa.



ATTENZIONE

I pulsanti JOG e PRESET sono sempre attivi, anche con dispositivo in allarme o in condizione di emergenza. Prima di premere questi pulsanti assicurarsi che il movimento del dispositivo si possa realizzare in condizioni di assoluta sicurezza e che non sussista alcun pericolo di procurare lesioni a persone o danni alle apparecchiature.

4.4.4 Pulsante PRESET (Figura 5)

Utilizzare questo pulsante per assegnare alla posizione attuale dell'asse il valore impostato al parametro **Valore di preset**. Il pulsante deve essere mantenuto premuto per almeno 3 secondi. Per ogni informazione si veda il parametro

Valore di preset a pagina 83 (interfaccia Profibus) o a pagina 132 (interfaccia Modbus).



ATTENZIONE

I pulsanti JOG e PRESET sono sempre attivi, anche con dispositivo in allarme o in condizione di emergenza. Prima di premere questi pulsanti assicurarsi che il movimento del dispositivo si possa realizzare in condizioni di assoluta sicurezza e che non sussista alcun pericolo di procurare lesioni a persone o danni alle apparecchiature.

5 Quick reference

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

- Procedere all'installazione meccanica;
- effettuare le connessioni elettriche;
- impostare l'indirizzo del nodo (node ID, l'indirizzo impostato di default è 1, si veda a pagina 28);
- alimentare l'unità con alimentazione +24VDC (sia il motore che il controllore);
- impostare il valore in **Distanza_giro** (si veda a pagina 76 -interfaccia Profibus- o a pagina 125 - interfaccia Modbus);
- impostare il valore in **Velocità Jog** (si veda alla pagina 80 - interfaccia Profibus- o a pagina 130 - interfaccia Modbus);
- impostare il valore in **Velocità di lavoro** (si veda alla pagina 80 - interfaccia Profibus- o a pagina 130 - interfaccia Modbus);
- impostare il valore in **Valore di preset** (si veda a pagina 83 - interfaccia Profibus- o a pagina 132 - interfaccia Modbus);
- impostare i valori dei finecorsa in **09 Delta positivo** e **0A Delta negativo**; si veda a pagina 78 - interfaccia Profibus; oppure in **Delta spazio positivo [0x08-0x09]** e **Delta spazio negativo [0x0A-0x0B]** a pagina 128 - interfaccia Modbus;
- solo interfaccia Modbus: procedere al salvataggio dei nuovi valori tramite il bit 9 **Salva parametri** del registro **Control Word [0x2A]** (si veda a pagina 136).



ATTENZIONE

Utilizzando i moduli **Lika RD1xA-T12**, **Lika RD1xA-T24**, **Lika RD1xA-T48** e **Lika RD1xA-T92** il valore di tutti i parametri è caricato all'accensione prendendo i dati memorizzati sul PLC a partire dal file GSD. Ogni modifica locale tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** è perciò temporanea: allo spegnimento del dispositivo ogni valore impostato viene perso (a eccezione del preset, unico parametro non compreso nel file GSD; e, per tutti i parametri, nel caso in cui si esegua precedentemente un'impostazione del preset, si veda alla pagina 83) e alla successiva riaccensione viene caricato il valore presente nel PLC (per cui anche i parametri salvati a seguito di un preset sono poi comunque sovrascritti).

Questo dispositivo prevede altresì la possibilità di installazione del modulo **Lika RD1xA-no param** (disponibile a partire dalla versione H3S3, file GSD V5). Con questo modulo è possibile bypassare il trasferimento dei parametri dal PLC all'accensione (i parametri infatti vengono letti dalla memoria flash) e

memorizzare in flash (tramite il bit 9 **Salva parametri** in **Control Word (Byte 0 e 1)**) ogni modifica locale effettuata mediante **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)**. Con il modulo **Lika RD1xA-no param** NON è possibile leggere e modificare i parametri tramite la pagina **Parametrizza** della finestra **Proprietà slave DP** di STEP7 (si veda il paragrafo "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). La modifica dei parametri è quindi possibile tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** oppure tramite l'interfaccia Modbus. Il modulo **Lika RD1xA-no param** è indipendente dal rapporto di riduzione e può essere installato quale che sia quello del dispositivo che si va a programmare. Naturalmente i parametri da impostare dovranno tenere conto delle caratteristiche meccaniche ed elettriche del dispositivo.

Utilizzando l'interfaccia seriale RS-232 di servizio Modbus è possibile impostare e salvare i parametri di modo che, anche a seguito di uno spegnimento, i valori impostati siano mantenuti. Questo però vale solamente fino a che sia mantenuta attiva l'interfaccia seriale: infatti nel momento in cui il dispositivo sia collegato alla rete Profibus, saranno caricati i dati memorizzati nel PLC e di conseguenza i dati impostati tramite la seriale Modbus saranno sovrascritti.

Utilizzando STEP7 di Siemens invece è possibile memorizzare in maniera permanente ogni modifica al valore di un parametro (eccetto che nel modulo -no param). Per fare questo modificare il valore dei parametri nella pagina **Proprietà slave DP** di STEP7 (si veda alla sezione "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). Si badi che, al contrario, le modifiche realizzate mediante la tabella delle variabili (si veda alla sezione "1.3.1 Impostazione parametro 28 Valore di preset" a pagina 49) sono temporanee.



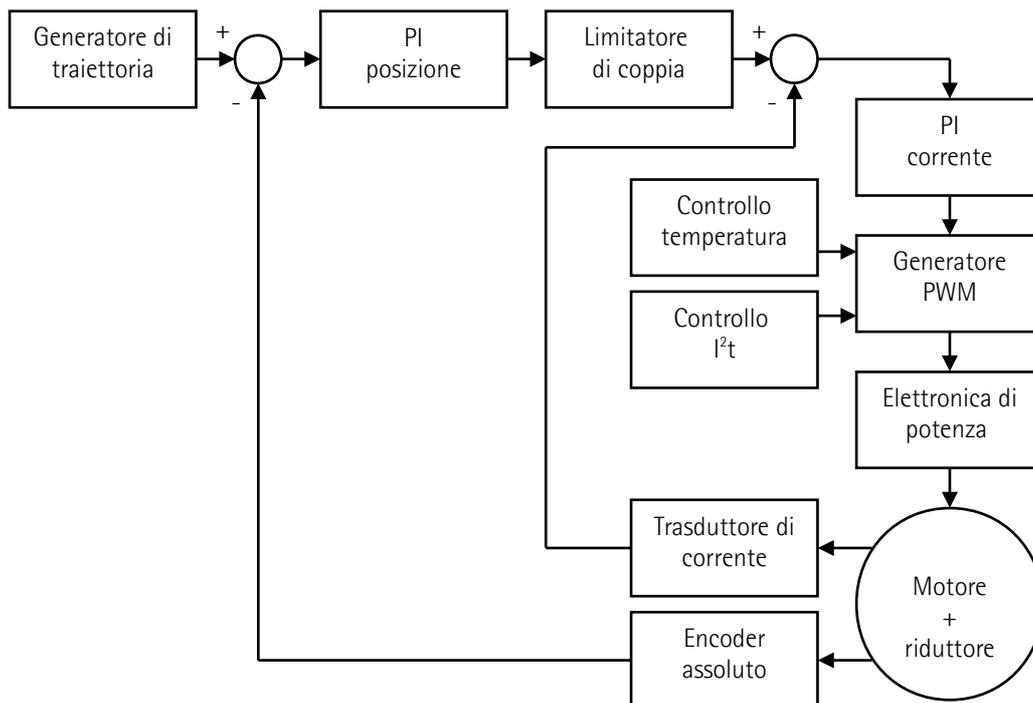
NOTA

I dati macchina **Distanza_giro**, **Velocità Jog**, **Velocità di lavoro** e **Valore di preset**, nonché i valori dei finecorsa (**09 Delta positivo / 0A Delta negativo** - interfaccia Profibus - e **Delta spazio positivo [0x08-0x09] / Delta spazio negativo [0x0A-0x0B]** - interfaccia Modbus) sono strettamente relazionati tra loro; si deve quindi prestare particolare attenzione quando si procede alla modifica anche di uno solo di essi. Per maggiori informazioni riferirsi a pagina 36.

6 Funzioni

6.1 Principio di funzionamento

Il seguente schema a blocchi illustra la logica di controllo del sistema:



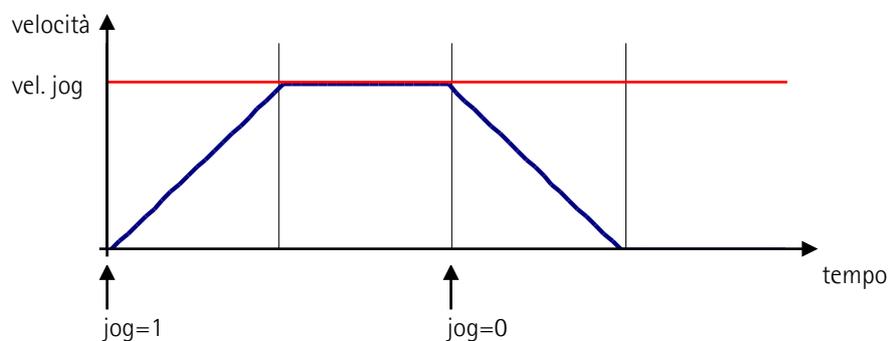
6.2 Tipi di movimento: jog e posizione

I tipi di movimento previsti nell'unità ROTADRIVE sono i seguenti:

- Jog: controllo di velocità;
- Posizionamento: controllo di posizione e di velocità.

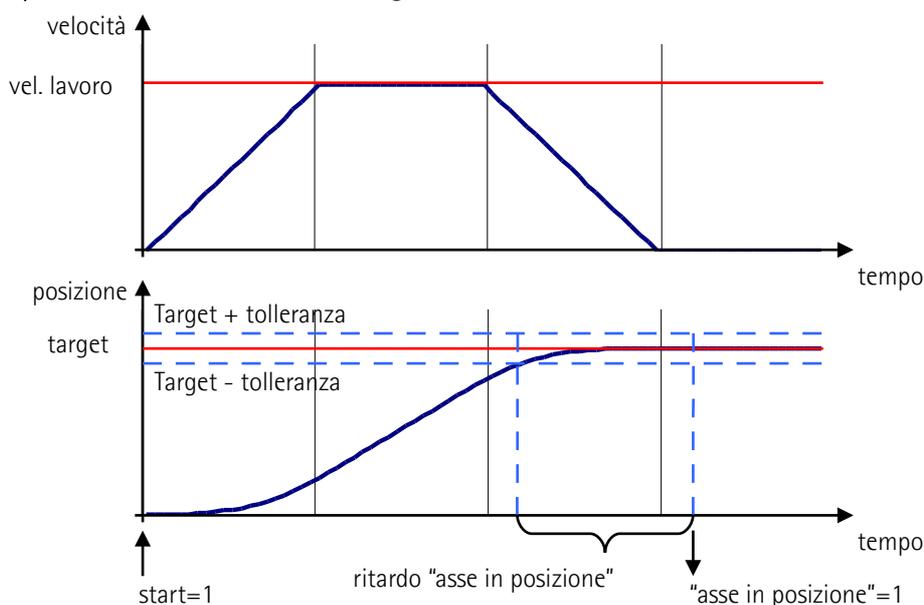
Jog: controllo di velocità

Questo tipo di controllo genera una traiettoria di velocità tale da imporre a regime una velocità di rotazione dell'asse del dispositivo uguale a **Velocità Jog** (si veda a pagina 80 -interfaccia Profibus-, pagina 130 -interfaccia Modbus).



Posizionamento: controllo di posizione e velocità

Questo tipo di controllo è un movimento punto-punto, nel quale la massima velocità raggiunta è pari a **Velocità di lavoro** (si veda a pagina 80 -interfaccia Profibus-, pagina 130 -interfaccia Modbus) che verrà raggiunta solo se lo spazio da percorrere è sufficientemente grande.



6.3 Ingressi e uscite digitali

Il dispositivo è provvisto di **tre ingressi e un'uscita digitali**.

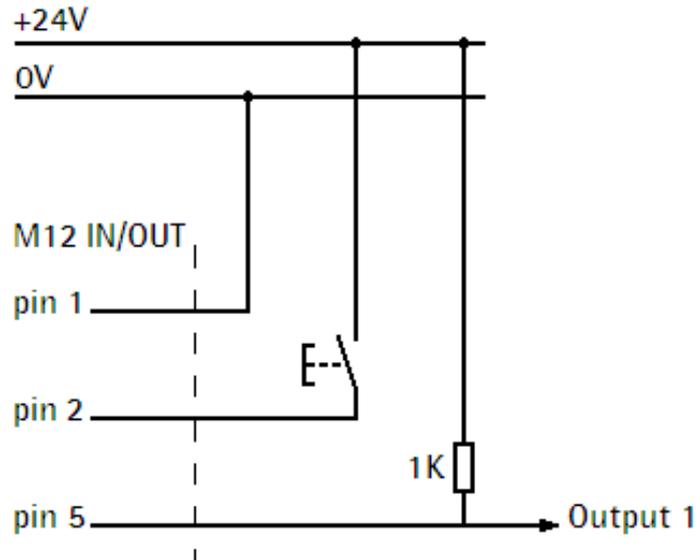
Gli ingressi sono letti dal dispositivo e trasmessi al Master attraverso **Status Word** (bit 13-15, si veda alla pagina 69 -interfaccia Profibus- o a pagina 143 -interfaccia Modbus) quando il dispositivo si trova nello stato **Data_Exchange** -interfaccia Profibus- / **Idle** -interfaccia Modbus.

Il valore logico "Alto" è letto con tensione pari a $+24\text{VDC} \pm 10\%$.

L'uscita è comandata dal Master allo Slave attraverso **Control Word** (bit 13, si veda alla pagina 62 -interfaccia Profibus- o a pagina 137 -interfaccia Modbus) quando il dispositivo si trova nello stato **Data_Exchange** -interfaccia Profibus- / **Idle** -interfaccia Modbus.

E' un'uscita open collector con $I_{\text{max}} = 150\text{mA}$.

Esempio schema di collegamento:



6.4 Distanza_giro, Velocità Jog, Velocità di lavoro, Valore di preset e valori di finecorsa

I dati macchina **Distanza_giro**, **Velocità Jog**, **Velocità di lavoro** e **Valore di preset**, nonché i valori dei finecorsa (**09 Delta positivo / 0A Delta negativo** - interfaccia Profibus - e **Delta spazio positivo [0x08-0x09] / Delta spazio negativo [0x0A-0x0B]** - interfaccia Modbus) sono strettamente relazionati tra loro; si deve quindi prestare particolare attenzione quando si procede alla modifica anche di uno solo di essi.

La sequenza corretta per la modifica dei parametri è la seguente:

- impostazione del valore di **Distanza_giro** (si veda a pagina 76 -interfaccia Profibus- o a pagina 125 - interfaccia Modbus);
- impostazione del valore di **Velocità Jog** (si veda alla pagina 80 - interfaccia Profibus- o a pagina 130 - interfaccia Modbus);
- impostazione del valore di **Velocità di lavoro** (si veda alla pagina 80 - interfaccia Profibus- o a pagina 130 - interfaccia Modbus);
- impostazione di **Valore di preset** (si veda a pagina 83 - interfaccia Profibus- o a pagina 132 - interfaccia Modbus);
- verifica della correttezza del valore di finecorsa positivo (**09 Delta positivo**, si veda alla pagina 78 -interfaccia Profibus; **Delta spazio positivo [0x08-0x09]**, si veda alla pagina 128 -interfaccia Modbus);
- verifica della correttezza del valore di finecorsa negativo (**0A Delta negativo**, si veda alla pagina 79 -interfaccia Profibus; **Delta spazio negativo [0x0A-0x0B]**, si veda alla pagina 129 -interfaccia Modbus);
- solo interfaccia Modbus: salvataggio dei nuovi valori tramite il bit 9 **Salva parametri** del registro **Control Word [0x2A]** (si veda a pagina 136).



ATTENZIONE

A ogni modifica del parametro **Distanza_giro** si devono poi reimpostare **Velocità Jog** e **Velocità di lavoro** in quanto le velocità sono espresse in impulsi al secondo. Nel calcolo delle velocità si deve sempre rispettare la seguente relazione:

$$\frac{vel_{\min} * Distanza / giro}{1024} \leq Velocità \leq \frac{vel_{\max} * Distanza / giro}{1024}$$

dove:

- **Distanza/giro**: nuovo valore di **Distanza_giro** impostato dall'utilizzatore ed espresso in impulsi
- **vel_{min}**: velocità minima 1 [imp/s] per tutti i dispositivi RD1xA

- **vel_{max}**: velocità massima
 - 4266 [imp/s] per RD1xA-...-T12-...
 - 2133 [imp/s] per RD1xA-...-T24-...
 - 1066 [imp/s] per RD1xA-...-T48-...
 - 556 [imp/s] per RD1xA-...-T92-...
- **1024**: è il valore massimo ammesso per **Distanza_giro** (espresso in impulsi).

Dopo ogni modifica del parametro **Distanza_giro** si deve reimpostare anche il **Valore di preset** in modo da definire lo zero asse in quanto il sistema di riferimento è variato.

Dopo la modifica del parametro **Valore di preset** non occorre invece reimpostare il valore dei finecorsa in quanto la funzione di Preset provvede a ricalcolarli automaticamente reinizializzando i limiti positivo e negativo sulla base dei valori impostati in **09 Delta positivo** e **0A Delta negativo** (interfaccia Profibus) oppure in **Delta spazio positivo [0x08-0x09]** e **Delta spazio negativo [0x0A-0x0B]** (interfaccia Modbus).

Il numero di giri gestiti dal dispositivo è di 511 in direzione negativa e 511 in direzione positiva rispetto al preset.

Il valore del finecorsa positivo (parametro **09 Delta positivo** -interfaccia Profibus; **Delta spazio positivo [0x08-0x09]** -interfaccia Modbus) sommato a **Valore di preset** definisce il massimo spostamento in avanti (positivo) rispetto al preset (valore espresso in impulsi).

Il valore del finecorsa negativo (parametro **0A Delta negativo** -interfaccia Profibus; **Delta spazio negativo [0x0A-0x0B]** -interfaccia Modbus) sottratto a **Valore di preset** definisce il massimo spostamento all'indietro (negativo) rispetto al preset (valore espresso in impulsi).



ATTENZIONE

Si badi inoltre che i parametri di seguito elencati sono tutti espressi in relazione al parametro **Distanza_giro**; di conseguenza la modifica del valore nel parametro **Distanza_giro** comporta necessariamente una ridefinizione dei valori da essi espressi. I parametri sono: **Accelerazione**, **Decelerazione**, **Max errore di inseguimento**, **Tolleranza di posizione**, **09 Delta positivo / 0A Delta negativo**, (interfaccia Profibus) e **Delta spazio positivo [0x08-0x09] / Delta spazio negativo [0x0A-0x0B]** (interfaccia Modbus), **Velocità massima**, **Fincorsa assoluto positivo** (interfaccia Profibus), **Fincorsa assoluto negativo** (interfaccia Profibus), **Posizione attuale**, **Velocità attuale** e **Target position**. Si veda per esempio la relazione che intercorre tra **Distanza_giro** e i valori di velocità, illustrata in questo paragrafo.



Esempio 1

Valori di default:

Distanza_giro = 1024 impulsi/giro

Velocità di lavoro massima:

= 4266 impulsi/secondo per RD1xA-...T12-... ($4266 \cdot 1024 / 1024 = 4266$)

= 2133 impulsi/secondo per RD1xA-...T24-... ($2133 \cdot 1024 / 1024 = 2133$)

= 1066 impulsi/secondo per RD1xA-...T48-... ($1066 \cdot 1024 / 1024 = 1066$)

= 556 impulsi/secondo per RD1xA-...T92-... ($556 \cdot 1024 / 1024 = 556$)

Valore di preset = 0

Finecorsa positivo (**09 Delta positivo** -interfaccia Profibus; **Delta spazio positivo [0x08-0x09]** -interfaccia Modbus) e finecorsa negativo (**0A Delta negativo** -interfaccia Profibus; **Delta spazio negativo [0x0A-0x0B]** -interfaccia Modbus) massimi = 523263 = (1024 impulsi/giro x 511 giri) - 1 con **Valore di preset** = 0

Finecorsa SW + massimo = 0 + 523263 = + 523263 impulsi (in avanti)

Finecorsa SW - massimo = 0 - 523263 = - 523263 impulsi (indietro).

Con **Valore di preset** = 0, la corsa utile dell'asse sarà perciò compresa tra i due limiti **Finecorsa SW +** massimo + 523263 e **Finecorsa SW -** massimo - 523263.





Esempio 2

L'unità RD1xA-...T12-... è montata su una vite senza fine con passo 1 mm e si desidera mantenere una risoluzione al centesimo di millimetro.

Distanza_giro = 100 impulsi/giro

Velocità di lavoro massima = 417 impulsi/secondo ($4266 \cdot 100 / 1024 = 417$, arrotondato al numero intero)

Valore di preset = -500 (es. spessore utensile)

Fincorsa positivo (**09 Delta positivo** -interfaccia Profibus; **Delta spazio positivo [0x08-0x09]** -interfaccia Modbus) e fincorsa negativo (**0A Delta negativo** -interfaccia Profibus; **Delta spazio negativo [0x0A-0x0B]** -interfaccia Modbus) massimi = 100 impulsi/giro x 511 giri = 51100 impulsi

Fincorsa SW + massimo = (-500) + 51100 = 50600 impulsi (in avanti)

Fincorsa SW - massimo = (-500) - 51100 = -51600 impulsi (indietro)

Con **Valore di preset** = - 500, la corsa utile dell'asse sarà perciò compresa tra i due limiti **Fincorsa SW +** massimo + 50600 e **Fincorsa SW -** massimo - 51600.



7 Tabella parametri di default

Lista parametri	Indice Profibus	Indirizzo registro Modbus	Valore di default
Distanza_giro imp/giro	01	0x00	1024
Tolleranza di posizione imp	02	0x01	0
Tempo asse in tolleranza ms	03	0x02	0
Max errore di inseguimento imp	04	0x03	1024
Kp anello di posizione	05	0x04	400
Ki anello di posizione	06	0x05	100
Accelerazione imp/s ²	07	0x06	5000 (RD1xA-...T12-...) 2500 (RD1xA-...T24-...) 1000 (RD1xA-...T48-...) 500 (RD1xA-...T92-...)
Decelerazione imp/s ²	08	0x07	5000 (RD1xA-...T12-...) 2500 (RD1xA-...T24-...) 1000 (RD1xA-...T48-...) 500 (RD1xA-...T92-...)
Delta spazio positivo imp	09	0x08-0x09	523263
Delta spazio negativo imp	0A	0x0A-0x0B	523263
Velocità jog imp/s	0B	0x0C	4266 (RD1xA-...T12-...) 2133 (RD1xA-...T24-...) 1066 (RD1xA-...T48-...) 556 (RD1xA-...T92-...)
Velocità di lavoro imp/s	0C	0x0D	4266 (RD1xA-...T12-...) 2133 (RD1xA-...T24-...) 1066 (RD1xA-...T48-...) 556 (RD1xA-...T92-...)
Durata corrente di stacco ms	0D	0x0E	2000
Direzione conteggio	0E	0x0F	0
Kp anello di corrente	0F	0x10	200
Ki anello di corrente	10	0x11	30
Corrente massima mA	11	0x12	2000
Corrente di stacco mA	12	0x13	4000
Valore di preset imp	28	0x16-0x17	0
Rapporto di riduzione	13	0x18	12 (RD1xA-...T12-...) 24 (RD1xA-...T24-...) 48 (RD1xA-...T48-...) 92 (RD1xA-...T92-...)
Ampiezza passo jog imp	14	0x19	100

Interfaccia Profibus®

RD1A

RD12A



Smart encoders & actuators

1 Programmazione con Siemens STEP7

1.1 Configurazione tramite Siemens STEP7

1.1.1. Importazione file GSD

I dispositivi RD1A - RD12A con interfaccia Profibus sono forniti di un loro proprio file GSD **RD1xAVx.GSD** (si veda il supporto informatico allegato oppure all'indirizzo www.lika.it > **ATTUATORI ROTATIVI** > **CAMBIAFORMATO (DRIVECOD)** > **RD1A / RD12A**). Il file è disponibile sia in versione inglese (**RD1xAVx.GSE**) che in versione italiana (**RD1xAVx.GSI**).



ATTENZIONE

La versione HW3 SW2 dell'unità RD1xA-Profibus introduce il modulo con rapporto di riduzione -T92. A questo modulo si associa il file GSD V4. E' possibile installare anche la versione V3, ma naturalmente solo per i moduli con rapporto di riduzione -T12, -T24 e -T48, in quanto le versioni precedenti a V4 non prevedono la parametrizzazione per il modulo -T92. Per versioni HW-SW precedenti la 3-2 non è ammessa l'installazione del file GSD V4.



ATTENZIONE

La versione HW3 SW3 dell'unità RD1xA-Profibus introduce il modulo **Lika RD1xA-no param**. Con questo modulo è possibile bypassare il trasferimento dei parametri dal PLC all'accensione (i parametri infatti vengono letti dalla memoria flash) e memorizzare in flash (tramite il bit 9 **Salva parametri** in **Control Word (Byte 0 e 1)**) ogni modifica locale effettuata mediante **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)**. A questo modulo si associa il file GSD V5. E' possibile installare anche la versione GSD V4, ma naturalmente solo per i moduli con rapporto di riduzione -T12, -T24, -T48 e -T92, in quanto le versioni firmware precedenti la H3S3 non prevedono la parametrizzazione per il modulo -no param.



ATTENZIONE

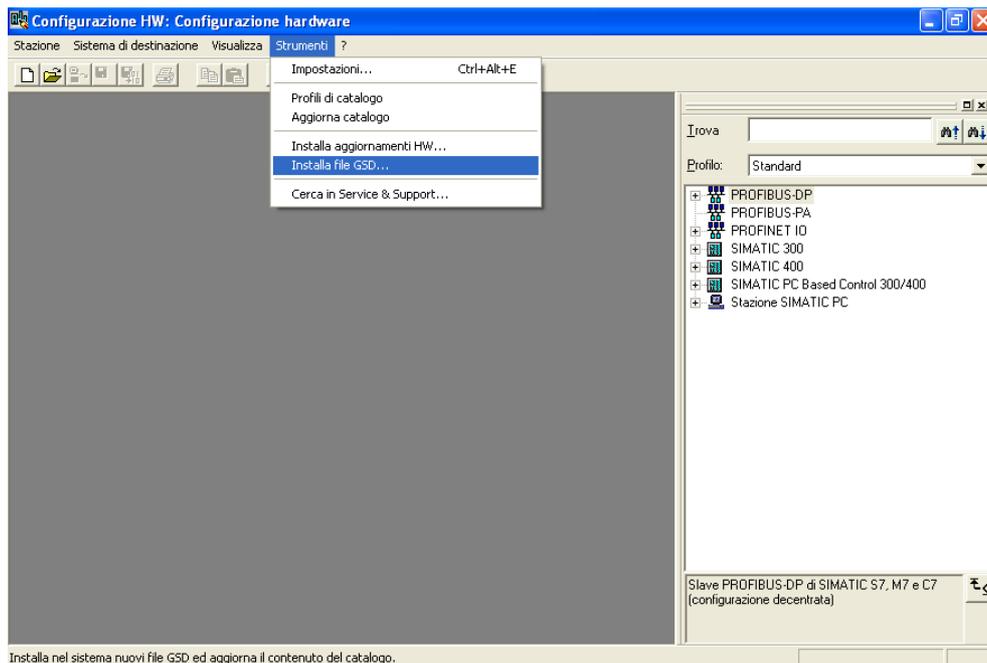
Si badi di rispettare sempre le seguenti compatibilità tra la versione del file GSD e la versione software dell'eseguibile Modbus.

Compatibilità	File GSD	EXE Modbus
	V1	fino a 2.1
	V2	fino a 2.4
	V3-V4-V5	da 2.5 a ...

Per installare il dispositivo RD1xA su Siemens STEP7 procedere nel modo seguente.

Nella finestra principale **Configurazione HW** di STEP7 selezionare il comando **Installa file GSD...** dal menù **Strumenti**.

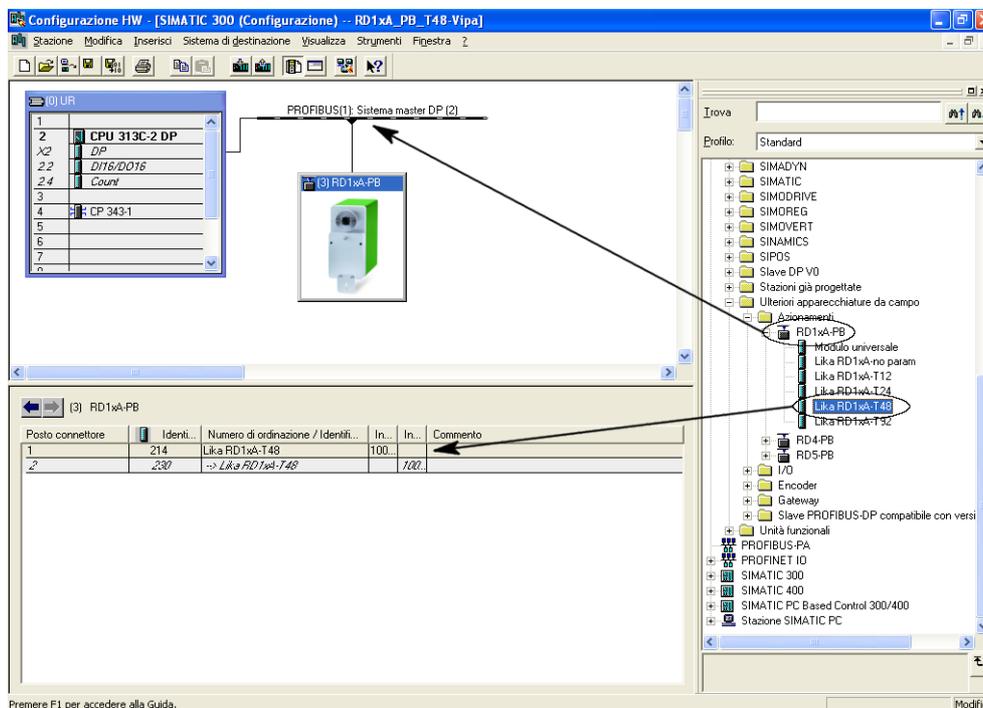
Si aprirà quindi una finestra che permetterà di selezionare il file GSD associato al dispositivo da caricare nel sistema di controllo.



1.1.2 Aggiungere il nodo al progetto

Per aggiungere il nodo al progetto accedere alla finestra principale **Configurazione HW** di STEP7 e selezionare tramite l'albero nella finestra a destra il modulo **RD1xA-PB** disponibile in **Catalogo > PROFIBUS-DP > Ulteriori apparecchiature da campo > Azionamenti**; trascinare il modulo nella finestra a sinistra e collegarlo al "BUS".

Trascinare poi il modulo **Lika RD1xA-no param** o **Lika RD1xA-T12** o **Lika RD1xA-T24** o **Lika RD1xA-T48** o **Lika RD1xA-T92** corrispondente al modello che si vuole installare nella tabella in basso a sinistra dedicata alle variabili, per esempio il modulo **Lika RD1xA-T48**.



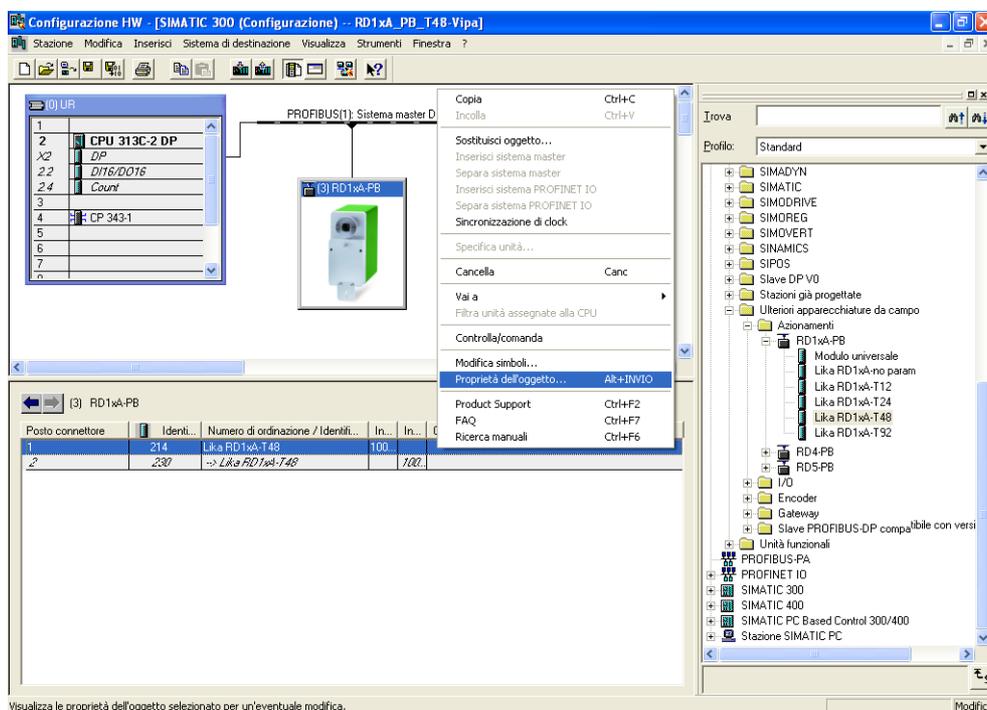
1.1.3 Configurazione dei parametri "Dati macchina"



NOTA

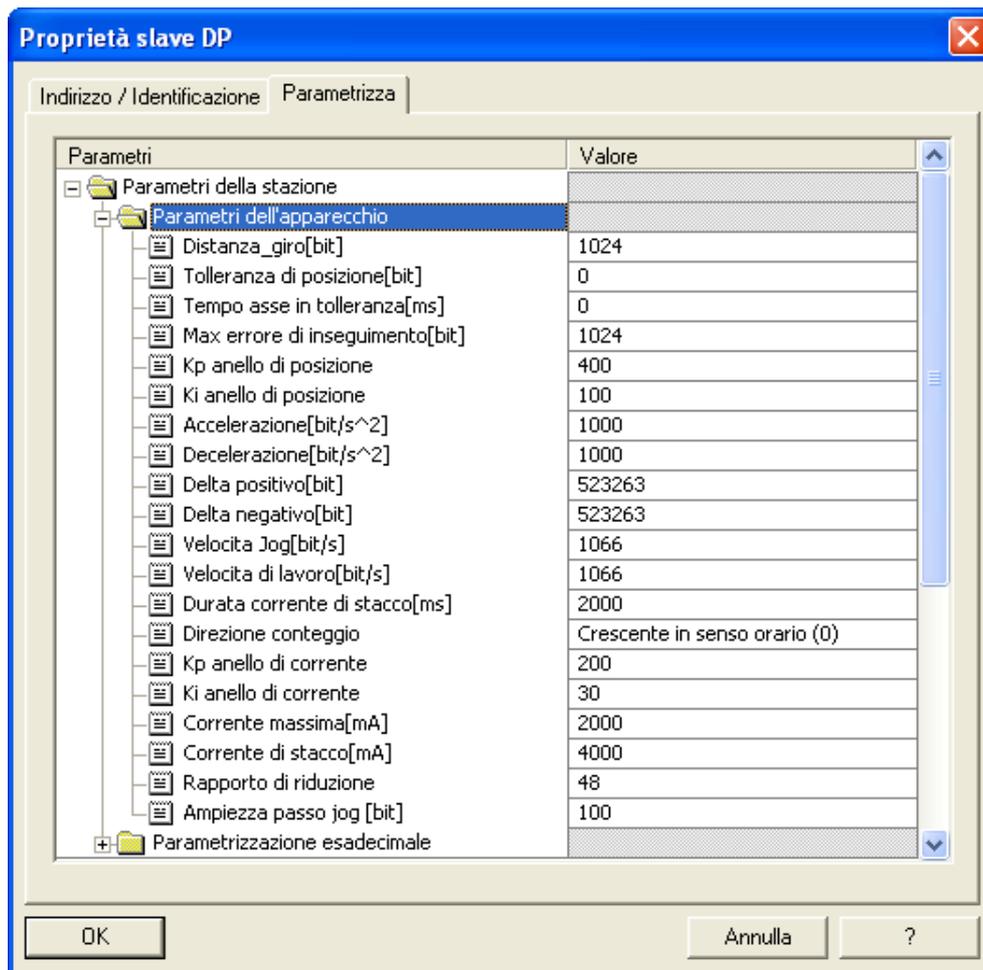
La pagina di configurazione dei parametri "Dati macchina" (pagina **Parametrizza** nella finestra **Proprietà slave DP**) non è disponibile se si installa il modulo **Lika RD1xA-no param**. Con questo modulo è possibile bypassare il trasferimento dei parametri dal PLC all'accensione (i parametri infatti vengono letti dalla memoria flash) e memorizzare in flash (tramite il bit 9 **Salva parametri** in **Control Word (Byte 0 e 1)**) ogni modifica locale effettuata mediante **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)**. Con il modulo **Lika RD1xA-no param** la modifica dei parametri è quindi possibile tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** oppure tramite l'interfaccia Modbus.

Per aprire la finestra di impostazione dei parametri, accedere alla finestra principale **Configurazione HW** di STEP7 e selezionare il dispositivo **Lika RD1xA** appena installato (nel nostro esempio il modello **-T48**) nella tabella in basso a sinistra; premere quindi il tasto destro del mouse e, nel menù a tendina che si apre, selezionare il comando **Proprietà dell'oggetto...**



Si aprirà quindi la finestra **Proprietà slave DP** dove, nella pagina **Parametrizza**, sono elencati i parametri **Dati macchina** del dispositivo.

Per una descrizione e un uso corretto dei parametri si veda la spiegazione al capitolo "Parametri di programmazione Profibus®" a pagina 75.



Dopo aver impostato i parametri, premere il pulsante **OK** per chiudere la finestra **Proprietà slave DP**, quindi premere il pulsante **Download** (icona a lato) nella barra degli strumenti della finestra **Configurazione HW** per scaricare i dati al dispositivo.

1.2 Lettura della diagnostica



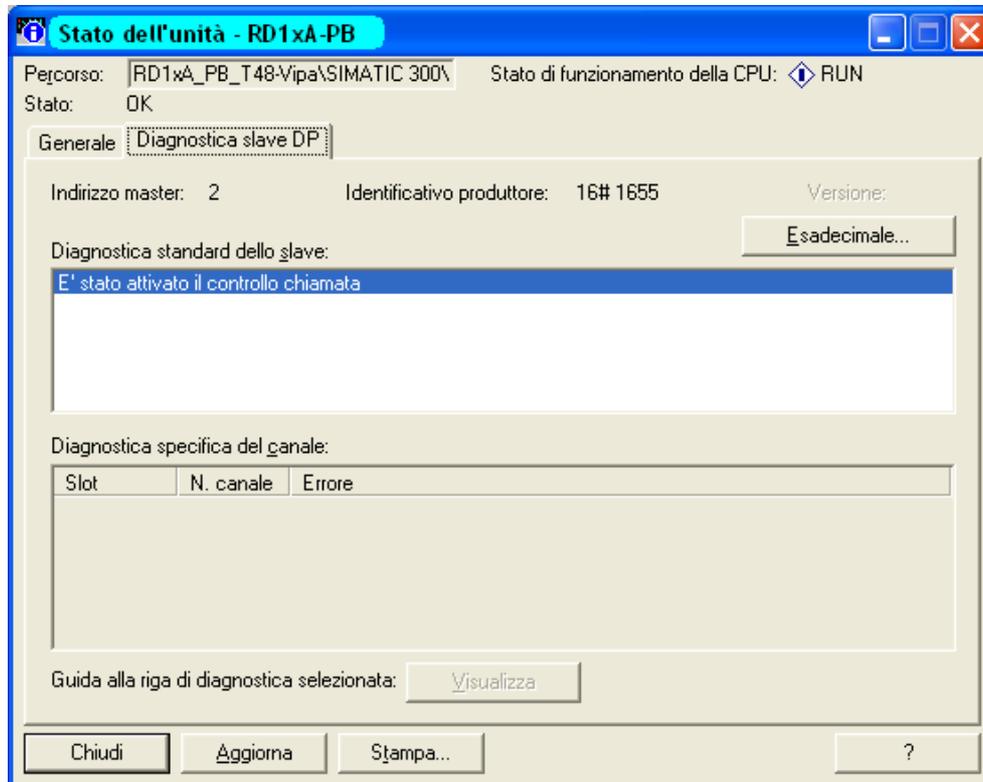
Prima di accedere alla finestra di diagnostica, occorre portare online il sistema. Per fare questo, nella finestra principale **Configurazione HW** di STEP7 selezionare il comando **Apri online** nella barra di menu **Stazione**; oppure premere il pulsante **Online / Offline** nella barra degli strumenti della stessa finestra (icona a lato).

Quindi selezionare il modulo RD1xA-PB collegato al bus e poi il comando **Stato dell'unità...** nella barra di menu **Sistema di destinazione** per accedere alla finestra **Stato dell'unità**.

Posto connettore	Identi...	Numero di ordinazione / Identificazione	Indirizzo E	Indirizzo A	Commento
1	214	Lika RD1xA-T48	100...113		
2	237	-> Lika RD1xA-T48		100...113	

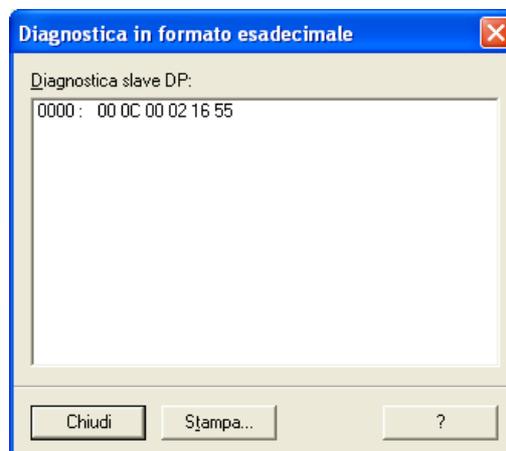
Visualizza lo stato dell'unità attuale (buffer di diagnostica, memoria, tempi di ciclo, stack).

Nella finestra **Stato dell'unità** aprire poi la pagina **Diagnostica slave DP**.



Per visualizzare i dati relativi alla diagnostica premere il pulsante **Esadecimale...** in questa pagina.

Diagnostica a 6 byte:



1.3 Impostazione e lettura parametri

Quando il dispositivo si trova in modalità **Data_Exchange** è possibile leggere o modificare sia i **dati macchina** che i **parametri operativi**; nella sezione "2.6 DDLM_Data_Exchange" a pagina 59 è indicato il significato delle variabili scambiate tra Master e Slave.

Per la modifica di un parametro riferirsi all'esempio riportato nella sezione "**Impostazione parametro 28 Valore di preset**" a seguire qui sotto.

Per la lettura di un parametro riferirsi all'esempio riportato nella sezione "**Letture parametro 29 Velocità attuale**" qui sotto più in basso.



1.3.1 Impostazione parametro 28 Valore di preset

In questo esempio il dispositivo con indirizzo 3 trasmette al Master la posizione sulla variabile all'indirizzo ED 104...107 -**Posizione (Byte 4 ... 7)**- e riceve il valore preset tramite le variabili AB 108 (**Numero parametro (Byte 8)**) e AD 109...112 (**Valore parametro (Byte 9 ... 12)**).

Operando	Simbolo	Forma	Valore di stato	Valore di comando
1	// ===== VARIABILI DI INGRESSO =====			
2	// STATUS WORD (16 bit)			
3	EW 100 "STATUS WORD"	BIN	2#0000_0000_0000_0010	
4	// ALLARMI (16 bit)			
5	EW 102 "ALLARMI"	BIN	2#0000_0000_0000_0000	
6	// POSIZIONE REALE(32 bit)			
7	ED 104 "POZIZIONE REALE"	DEC	L#3652	
8	// N. PARAMETRO(8 bit)			
9	EB 108	HEX	B#16#28	
10	// VALORE PARAMETRO LETTO (32 bit)			
11	ED 109	DEC	L#3652	
12	// not used(8 bit)			
13	//			
14	// ===== VARIABILI DI USCITA =====			
15	// CONTROL WORD (16 bit)			
16	AW 100 "CTRL_WRD"	BIN	2#0000_0000_1000_0100	
17	// TARGET POSITION (32 bit)			
18	AD 104 "TARGET POSITION"	DEC	L#0	L#0
19	// N. PARAMETRO (8 bit)			
20	AB 108 "Num. Parametro"	HEX	B#16#A8	B#16#A8
21	// VALORE PARAMETRO SCRITTO (32 bit)			
22	AD 109 "Valore Parametro"	DEC	L#10000	L#10000
23	// not used (8 bit)			
24	//			
25				
26				
27				
28				

Numero parametro = A8hex l'indice del parametro **28 Valore di preset** è 28h (si veda alla pagina 83), cui va sommato il comando di scrittura 80h (28h + 80h = A8h).

Valore parametro = 10000 (esempio di preset)



Per inviare il valore di preset premere il pulsante **Comanda variabile** nella barra degli strumenti (icona a lato, a destra del pulsante **Controlla variabile** simboleggiato dagli occhiali).

Ora il dispositivo trasmette la posizione "10000".

Per concludere la procedura di preset, riportare a 0 il bit 7 della stessa variabile e premere nuovamente il pulsante **Comanda variabile** (**Numero parametro** 28h).



1.3.2 Lettura parametro 29 Velocità attuale

In questo esempio il dispositivo con indirizzo 3 trasmette al Master la propria posizione sulla variabile all'indirizzo ED 104...107 (4 byte) e il valore di velocità tramite la variabile ED 109...112 (**Valore parametro**).

	Operando	Simbolo	Forma	Valore di stato	Valore di comando
1	//	===== VARIABILI DI INGRESSO =====			
2	//	STATUS WORD (16 bit)			
3	EW 100	"STATUS WORD"	BIN	2#0000_0010_1100_0110	
4	//	ALLARMI (16 bit)			
5	EW 102	"ALLARMI"	BIN	2#0000_0000_0000_0000	
6	//	POSIZIONE REALE(32 bit)			
7	ED 104	"POSIZIONE REALE"	DEC	L#32579	
8	//	N. PARAMETRO(8 bit)			
9	EB 108		HEX	B#16#29	
10	//	VALORE PARAMETRO LETTO (32 bit)			
11	ED 109		DEC	L#1066	
12	//	not used(8 bit)			
13	//				
14	//	===== VARIABILI DI USCITA =====			
15	//	CONTROL WORD (16 bit)			
16	AWV 100	"CTRL_WVRD"	BIN	2#0000_0000_1000_0100	
17	//	TARGET POSITION (32 bit)			
18	AD 104	"TARGET POSITION"	DEC	L#0	L#0
19	//	N. PARAMETRO (8 bit)			
20	AB 108	"Num. Parametro"	HEX	B#16#29	B#16#29
21	//	VALORE PARAMETRO SCRITTO (32 bit)			
22	AD 109	"Valore Parametro"	DEC	L#0	L#0
23	//	not used (8 bit)			
24	//				
25					
26					
27					
28					

Numero parametro = 29hex indice del parametro **29 Velocità attuale** (si veda a pagina 84)

Valore parametro = 1066 velocità = 1066 impulsi/secondo



Per richiedere il valore di velocità premere il pulsante **Comanda variabile** nella barra degli strumenti (icona a lato, a destra del pulsante **Controllo variabile** simboleggiato dagli occhiali).

Ora il dispositivo trasmette il valore della velocità: "1066" impulsi/ secondo.



NOTA

Qualora in STEP7 si presentassero anomalie di funzionamento delle variabili di Ingresso e Uscita con indice maggiore di 127 o con dati superiori a 4 byte, si consiglia di usare variabili di appoggio "MD" (puntatori) per la gestione delle variabili di scambio tra Master e Slave.

2 Interfaccia Profibus®

Le unità ROTADRIVE Lika sono dispositivi Slave e soddisfano il "PROFIBUS-DP Profile".

Per ogni informazione e specifica omessa fare riferimento ai documenti rilasciati da Profibus International disponibili sul sito www.profibus.com.

2.1 File GSD

I dispositivi RD1A-RD12A con interfaccia Profibus sono forniti di un loro proprio file GSD **RD1xAVx.GSD** (si veda il supporto informatico allegato oppure all'indirizzo www.lika.it > **ATTUATORI ROTATIVI** > **CAMBIAFORMATI (DRIVECOD)** > **RD1A / RD12A**). Il file GSD deve essere installato sul dispositivo Master Profibus.

Il file è disponibile sia in versione inglese (**RD1xAVx.GSE**) che in versione italiana (**RD1xAVx.GSI**).



ATTENZIONE

La versione HW3 SW2 dell'unità RD1xA-Profibus introduce il modulo con rapporto di riduzione -T92. A questo modulo si associa il file GSD V4. E' possibile installare anche la versione V3, ma naturalmente solo per i moduli con rapporto di riduzione -T12, -T24 e -T48, in quanto le versioni firmware precedenti la H3S2 non prevedono la parametrizzazione per il modulo -T92. Per versioni HW-SW precedenti la 3-2 non è ammessa l'installazione del file GSD V4.



ATTENZIONE

La versione HW3 SW3 dell'unità RD1xA-Profibus introduce il modulo **Lika RD1xA-no param**. Con questo modulo è possibile bypassare il trasferimento dei parametri dal PLC all'accensione (i parametri infatti vengono letti dalla memoria flash) e memorizzare in flash (tramite il bit 9 **Salva parametri** in **Control Word (Byte 0 e 1)**) ogni modifica locale effettuata mediante **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)**. A questo modulo si associa il file GSD V5. E' possibile installare anche la versione GSD V4, ma naturalmente solo per i moduli con rapporto di riduzione -T12, -T24, -T48 e -T92, in quanto le versioni firmware precedenti la H3S3 non prevedono la parametrizzazione per il modulo -no param.



ATTENZIONE

Si badi di rispettare sempre le seguenti compatibilità tra la versione del file GSD e la versione software dell'eseguibile Modbus.

Compatibilità	File GSD	EXE Modbus
	V1	fino a 2.1
	V2	fino a 2.4
	V3-V4-V5	da 2.5 a ...



ATTENZIONE

Utilizzando i moduli **Lika RD1xA-T12**, **Lika RD1xA-T24**, **Lika RD1xA-T48** e **Lika RD1xA-T92** il valore di tutti i parametri è caricato all'accensione prendendo i dati memorizzati sul PLC a partire dal file GSD. Ogni modifica locale tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** è perciò temporanea: allo spegnimento del dispositivo ogni valore impostato viene perso (a eccezione del preset, unico parametro non compreso nel file GSD; e, per tutti i parametri, nel caso in cui si esegua precedentemente un'impostazione del preset, si veda alla pagina 83) e alla successiva riaccensione viene caricato il valore presente nel PLC (per cui anche i parametri salvati a seguito di un preset sono poi comunque sovrascritti).

Questo dispositivo prevede altresì la possibilità di installazione del modulo **Lika RD1xA-no param** (disponibile a partire dalla versione H3S3, file GSD V5). Con questo modulo è possibile bypassare il trasferimento dei parametri dal PLC all'accensione (i parametri infatti vengono letti dalla memoria flash) e memorizzare in flash (tramite il bit 9 **Salva parametri** in **Control Word (Byte 0 e 1)**) ogni modifica locale effettuata mediante **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)**. Con il modulo **Lika RD1xA-no param** NON è possibile modificare i parametri tramite la pagina **Parametrizza** della finestra **Proprietà slave DP** di STEP7 (si veda il paragrafo "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). La modifica dei parametri è quindi possibile tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** oppure tramite l'interfaccia Modbus. Il modulo **Lika RD1xA-no param** è indipendente dal rapporto di riduzione e può essere installato quale che sia quello del dispositivo che si va a programmare. Naturalmente i parametri da impostare dovranno tenere conto delle caratteristiche meccaniche ed elettriche del dispositivo.

Utilizzando l'interfaccia seriale RS-232 di servizio Modbus è possibile impostare e salvare i parametri di modo che, anche a seguito di uno spegnimento, i valori impostati siano mantenuti. Questo però vale solamente fino a che sia

mantenuta attiva l'interfaccia seriale: infatti nel momento in cui il dispositivo sia collegato alla rete Profibus, saranno caricati i dati memorizzati nel PLC e di conseguenza i dati impostati tramite la seriale Modbus saranno sovrascritti.

Utilizzando STEP7 di Siemens invece è possibile memorizzare in maniera permanente ogni modifica al valore di un parametro (eccetto che nel modulo -no param). Per fare questo modificare il valore dei parametri nella pagina **Proprietà slave DP** di STEP7 (si veda alla sezione "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). Si badi che, al contrario, le modifiche realizzate mediante la tabella delle variabili (si veda alla sezione "1.3.1 Impostazione parametro 28 Valore di preset" a pagina 49) sono temporanee.

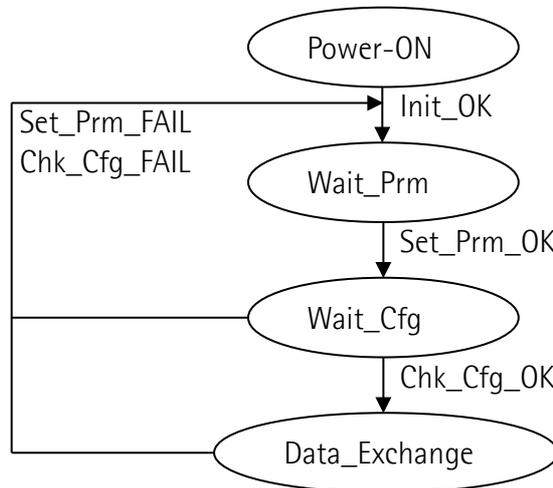
2.2 Baud rate

I dispositivi RD1A - RD12A con interfaccia Profibus riconoscono automaticamente la velocità di trasmissione definita dal Master Profibus-DP. La velocità di trasmissione è univoca per tutti i dispositivi nella rete. I baud rate supportati sono i seguenti:

9.6Kbps - 19.2Kbps - 93.75Kbps - 187.5Kbps - 500Kbps - 1.5Mbps - 3Mbps - 6Mbps - 12Mbps.

2.3 Funzionamento a stati

I dispositivi Profibus prevedono un funzionamento a stati. Lo schema è il seguente:



NOTA

I parametri dati macchina sono trasmessi in fase Set_Prm e nello stato **Data_Exchange**; i parametri operativi sono trasmessi solo quando il dispositivo si trova nello stato **Data_Exchange**.

Tipi di messaggi

Lo scambio dati tra Master e Slave avviene nei seguenti modi:

- **DDL_M_Set_Prm**: fase di configurazione e parametrizzazione. In questa modalità, attiva subito dopo l'accensione del sistema, vengono inviati i dati macchina al dispositivo. Per ogni informazione si veda alla sezione "2.4 DDL_M_Set_Prm" a pagina 57.
- **DDL_M_Chk_Cfg**: definisce il numero di byte utilizzati in ingresso e uscita nello stato **Data_Exchange**. Per ogni informazione si veda alla sezione "2.5 DDL_M_Chk_Cfg" a pagina 58.
- **DDL_M_Data_Exchange**: "Standard operation". In questa modalità il Master può inviare allo Slave un eventuale preset e lo Slave trasmette al Master il valore della posizione attuale (e della velocità). Per ogni informazione si veda alla sezione "2.6 DDL_M_Data_Exchange" a pagina 59.
- **DDL_M_Slave_Diag**: usato durante la fase di accensione e ogni volta che il Master vuole acquisire le informazioni di diagnostica relative allo Slave. Per ogni informazione si veda alla sezione "2.7 DDL_M_Slave_Diag" a pagina 74.

2.4 DDLM_Set_Prm

Quando il sistema viene attivato, i dati macchina impostati dall'utilizzatore sono trasferiti dal controllore al dispositivo. Generalmente il trasferimento dei parametri avviene automaticamente e i dati sono inseriti attraverso un'interfaccia utente presente nel software del dispositivo di controllo (per esempio "Step 7" di Siemens su PLC; si veda a pagina 42).

La descrizione dettagliata dei parametri è riportata al capitolo "Parametri di programmazione Profibus®" a pagina 75.

Il trasferimento dati rispetta la struttura descritta nella seguente tabella:

Byte	Parametro
0...6	Riservati Profibus
7...9	Riservati Profibus
10...13	01 Distanza_giro
14...17	02 Tolleranza di posizione
18...21	03 Tempo asse in tolleranza
22...25	04 Max errore di inseguimento
26...29	05 Kp anello di posizione
30...33	06 Ki anello di posizione
34...37	07 Accelerazione
38...41	08 Decelerazione
42...45	09 Delta positivo
46...49	0A Delta negativo
50...53	0B Velocità Jog
54...57	0C Velocità di lavoro
58...61	0D Durata corrente di stacco
62...65	0E Direzione conteggio
66...69	0F Kp anello di corrente
70...73	10 Ki anello di corrente
74...77	11 Corrente massima
78...81	12 Corrente di stacco
82...85	13 Rapporto di riduzione
86...89	14 Ampiezza passo jog

2.5 DDLM_Chk_Cfg

Questa configurazione definisce il numero di byte utilizzati in ingresso e uscita nello stato **Data_Exchange**; ogni riferimento è considerato dal punto di vista del Master.

Struttura messaggio **Chk_Cfg** (1 byte):

bit 7 = Consistency ("1")

bit 6 = Word format ("0"=byte, "1"=word=2byte)

bit 5...4 = In/out data ("01"=Input, "10"=output)

bit 3...0 = Code length

Valori validi per il dispositivo:

bit	7	6	5	4	3	2	1	0	
Data	1	1	0	1	0	1	1	0	D6h
	1	1	1	0	0	1	1	0	E6h

D6hex = 14 byte input

E6hex = 14 byte output

2.6 DDLM_Data_Exchange

Questo è il normale stato di funzionamento del sistema.

Attraverso i **messaggi Data_Exchange** si gestiscono tutti i parametri del dispositivo e si comandano i movimenti dell'asse.

Struttura messaggi Data_Exchange:

Byte	Funzione Master → Slave	Funzione Slave → Master
0	Control Word (Byte 0 e 1)	Status word (Byte 0 e 1)
1		
2	Non utilizzati	Allarmi (Byte 2 e 3)
3		
4	Target position (Byte 4 ... 7)	Posizione (Byte 4 ... 7)
5		
6		
7		
8	Numero parametro (Byte 8)	Numero parametro (Byte 8)
9	Valore parametro (Byte 9 ... 12)	Valore parametro (Byte 9...12)
10		
11		
12		
13	Non utilizzato	Non utilizzato

2.6.1 Funzione Master → Slave

In questa sezione sono descritti i messaggi Data_Exchange inviati dal Master allo Slave.

Control Word (Byte 0 e 1)

Contiene i comandi da inviare in tempo reale allo Slave per controllarlo.

Byte 0

Jog +

bit 0

Se il bit 4 **Jog incrementale** = 0, lo Slave si muove in direzione positiva per tutto il tempo in cui **Jog +** = 1; se invece il bit 4 **Jog incrementale** = 1 (abilitazione jog a passo), lo slave esegue un singolo passo in direzione positiva in corrispondenza del fronte di salita di **Jog +** la cui ampiezza, espressa in impulsi, è impostata al parametro **14 Ampiezza passo jog**; poi si arresta in attesa di un nuovo

comando. La velocità, l'accelerazione e la decelerazione sono definiti dai dati macchina **0B Velocità Jog**, **07 Accelerazione** e **08 Decelerazione**. Per una descrizione più dettagliata del controllo jog si veda a pagina 34.

Jog - bit 1

Se il bit 4 **Jog incrementale** = 0, lo Slave si muove in direzione negativa per tutto il tempo in cui **Jog -** = 1; se invece il bit 4 **Jog incrementale** = 1 (abilitazione jog a passo), lo slave esegue un singolo passo in direzione negativa in corrispondenza del fronte di salita di **Jog -** la cui ampiezza, espressa in impulsi, è impostata al parametro **14 Ampiezza passo jog**; poi si arresta in attesa di un nuovo comando. La velocità, l'accelerazione e la decelerazione sono definiti dai dati macchina **0B Velocità Jog**, **07 Accelerazione** e **08 Decelerazione**. Per una descrizione più dettagliata del controllo jog si veda a pagina 34.

Stop bit 2

Se impostato a "1" lo Slave è libero di eseguire i comandi di movimento ricevuti. Se durante il movimento questo bit diventa "0" allora lo Slave si ferma seguendo la decelerazione prevista nel dato macchina **08 Decelerazione**. Per un arresto immediato del movimento, utilizzare il bit 7 **Emergenza**.

Reset allarmi bit 3

Bit normalmente impostato a "0". Lo stato normale del dispositivo è ristabilito nel cambio di stato da "0" a "1" di questo bit. Questo comando toglie lo Slave dalla condizione di allarme solo se non sono più presenti le condizioni che hanno causato l'errore.

Si badi che, se l'allarme è relativo a dati macchina non validi (si veda **Dati macchina non validi** e **32 Lista parametri errati**), si può ripristinare il normale stato di lavoro solo impostando dati macchina validi. L'allarme **Errore memoria flash** non è ripristinabile.



Jog incrementale bit 4

Se il bit 4 " = 0", l'attivazione dei bit **Jog +** e **Jog -** procura il movimento manuale dello slave per tutto il tempo in cui **Jog + / Jog -** = 1. Impostando questo bit a 1 si abilita la funzione di jog a passo. L'attivazione dei bit **Jog +** e **Jog -** procura in corrispondenza del fronte di salita l'esecuzione di un singolo passo in direzione positiva o negativa la cui



ampiezza, espressa in impulsi, è impostata al parametro **14 Ampiezza passo jog**; quindi lo slave si arresta in attesa di un nuovo comando.

Si badi che nell'utilizzo dei pulsanti manuali di jog (si veda "4.4.3 Pulsanti JOG + e JOG - (Figura 5)" a pagina 29) la funzione di jog incrementale è disabilitata; non è cioè possibile eseguire passi jog utilizzando i pulsanti manuali.

bit 5

Non utilizzato.

Start

bit 6

Al cambio di stato da "0" a "1" il dispositivo si muove allo scopo di raggiungere la posizione di target specificata. Per la descrizione del controllo di posizione si veda a pagina 33. Per ogni informazione sulla posizione di target si veda **Target position (Byte 4 ... 7)** a pagina 63. Dopo l'avvio riportare a "0" questo bit.

Emergenza

bit 7

Questo bit deve essere normalmente alto ("=1") altrimenti il dispositivo bloccherà istantaneamente ogni suo movimento. Per procurare un arresto non istantaneo, che utilizzi la decelerazione programmata, usare il bit 2 **Stop**.

Byte 1

bit 8

Non utilizzato.

Salva parametri

bit 9

Disponibile solo se si installa il modulo **Lika RD1xA-no param** (si veda la sezione "1.1.2 Aggiungere il nodo al progetto" a pagina 44). Questo bit NON deve essere settato via Profibus se non si utilizza il modulo -no param. 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 valori impostati è richiesto alla variazione del bit dal livello logico basso (0) al livello logico alto (1). Per maggiori informazioni sul modulo **Lika RD1xA-no param** si veda la sezione "Informazioni preliminari" a pagina 11.

Carica parametri di default

bit 10

Disponibile solo se si installa il modulo **Lika RD1xA-no param** (si veda la sezione "1.1.2 Aggiungere il nodo al progetto" a pagina 44). Questo bit NON deve essere settato via Profibus se non si utilizza il modulo -no param. Il caricamento dei parametri di default (parametri impostati

durante la messa a punto in azienda del dispositivo che permettono un funzionamento a vuoto e sicuro del dispositivo) 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 40 è disponibile l'elenco dei dati macchina e il rispettivo valore di default preimpostato da Lika Electronic. All'atto del caricamento dei valori di default, il sistema esegue contestualmente anche un salvataggio automatico di tali valori nella memoria flash. Per maggiori informazioni sul modulo **Lika RD1xA-no param** si veda la sezione "Informazioni preliminari" a pagina 11.

bit 11

Non utilizzato.

Asse in coppia

bit 12

Mantenimento asse in coppia a fine posizionamento. Funzione disponibile solo per la versione RD1A (versione senza freno), nella versione RD12A (versione con freno) il bit 12 non è gestito. Se impostato a "=0", il PWM si disattiva (se non ci sono jog o posizionamenti in corso). Se impostato a "=1", il PWM diventa attivo: l'asse entra in controllo di spazio (anche se non ci sono jog o posizionamenti in corso).

OUT 1

bit 13

Attiva / disattiva l'uscita digitale 1 del dispositivo. Il significato delle uscite è esplicitato nel capitolo "Parametri di programmazione Profibus®" a pagina 75.

OUT 1 = 0 uscita 1 bassa (non attiva)

OUT 1 = 1 uscita 1 alta (attiva)

Sblocco freno

bit 14

Disponibile solo per la versione RD12A (versione con freno), nella versione RD1A (versione senza freno) il bit 14 non è gestito. Il modello RD12A è provvisto di un freno che, all'arresto del dispositivo, si attiva inibendo la possibilità di qualsiasi movimento dell'asse. Se si imposta questo bit "=1", il freno è disattivato; se si imposta "=0", il freno è gestito in modo automatico dal sistema.

Si badi che lo sblocco del freno è ammesso solo in assenza di allarmi.



bit 15 Non utilizzato.

Byte 2 e 3 Non utilizzati

Target position (Byte 4 ... 7)

Posizione di arrivo programmata. Il dispositivo si muove allo scopo di raggiungere questa posizione quando viene inviato il comando **Start**, i bit di **Stop** e **Emergenza** sono "1" e il dispositivo non è in stato di allarme.

byte 4	byte 5	byte 6	byte 7
Low	High



Funzione override di posizione

Durante il posizionamento è possibile modificare la posizione target, per fare ciò è sufficiente inviare nuovamente il comando **Start** con il nuovo valore di **Target position (Byte 4 ... 7)**.



NOTA

Non è possibile attivare le funzioni **Jog +**, **Jog -** e **Start** contemporaneamente. Per esempio: se viene inviato allo Slave il comando **Jog +** durante il movimento verso la posizione target, il comando di jog sarà ignorato; se si inviano i comandi **Jog +** e **Jog -** contemporaneamente il dispositivo non si muove o, se è già in movimento, arresta la sua corsa.

Se durante un movimento la comunicazione nella rete Profibus dovesse interrompersi (per esempio a causa della rottura del cavo), il dispositivo arresta ogni suo movimento ed entra in stato di emergenza. Al ripristino della comunicazione viene segnalato l'allarme **Assenza comunicazione Profibus**.

Numero parametro (Byte 8)

Usato per impostare l'indice del parametro di cui si vuole leggere o scrivere il valore nei byte successivi.

L'elenco dei parametri e il significato di ciascuno di essi sono riportati nel capitolo "Parametri di programmazione Profibus®" a pagina 75.

Struttura del byte:

bit 7	bit 6	bit 5 ... 0
R/W	0	Indice parametro

R/W = 0 lettura parametro

R/W = 1 scrittura parametro



Esempio 1

Per scrivere un valore nel parametro **28 Valore di preset** (indice 28h) impostare 0xA8 = 1010 1000:

	R/W	-	Indice parametro					
bit	7	6	5	4	3	2	1	0
binario	1	0	1	0	1	0	0	0

dove:

bit 7 = R/W = 1, cioè "scrittura parametro"

bit 6 = bit sempre impostato a 0

bit 5 ... 0 = indice parametro = 101000 binario = 28h = **28 Valore di preset**

Il valore da impostare è indicato nei successivi quattro byte 9...12 -**Valore parametro (Byte 9 ... 12)**.



Esempio 2

Per leggere il valore nel parametro **2A Temperatura** (indice 2Ah) impostare 0x2A = 0010 1010:

	R/W	-	Indice parametro					
bit	7	6	5	4	3	2	1	0
binario	0	0	1	0	1	0	1	0

dove:

bit 7 = R/W = 0, cioè "lettura parametro"

bit 6 = bit sempre impostato a 0

bit 5 ... 0 = indice parametro = 101010 binario = 2Ah = **2A Temperatura**

I successivi quattro byte 9...12 -**Valore parametro (Byte 9 ... 12)**- sono ignorati.

Valore parametro (Byte 9 ... 12)

Valore del parametro il cui indice è stato impostato al byte precedente. Tutti i parametri utilizzano 4 byte dati. L'elenco dei parametri e il significato di ciascuno di essi sono riportati nel capitolo "Parametri di programmazione Profibus®" a pagina 75.

byte 9	byte 10	byte 11	byte 12
Low	High



ATTENZIONE

Utilizzando i moduli **Lika RD1xA-T12**, **Lika RD1xA-T24**, **Lika RD1xA-T48** e **Lika RD1xA-T92** il valore di tutti i parametri è caricato all'accensione prendendo i dati memorizzati sul PLC a partire dal file GSD. Ogni modifica locale tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** è perciò temporanea: allo spegnimento del dispositivo ogni valore impostato viene perso (a eccezione del preset, unico parametro non compreso nel file GSD; e, per tutti i parametri, nel caso in cui si esegua precedentemente un'impostazione del preset, si veda alla pagina 83) e alla successiva riaccensione viene caricato il valore presente nel PLC (per cui anche i parametri salvati a seguito di un preset sono poi comunque sovrascritti).

Questo dispositivo prevede altresì la possibilità di installazione del modulo **Lika RD1xA-no param** (disponibile a partire dalla versione H3S3, file GSD V5). Con questo modulo è possibile bypassare il trasferimento dei parametri dal PLC all'accensione (i parametri infatti vengono letti dalla memoria flash) e memorizzare in flash (tramite il bit 9 **Salva parametri** in **Control Word (Byte 0 e 1)**) ogni modifica locale effettuata mediante **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)**. Con il modulo **Lika RD1xA-no param** NON è possibile leggere e modificare i parametri tramite la pagina **Parametrizza** della finestra **Proprietà slave DP** di STEP7 (si veda il paragrafo "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). La modifica dei parametri è quindi possibile tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** oppure tramite l'interfaccia Modbus. Il modulo **Lika RD1xA-no param** è indipendente dal rapporto di riduzione e può essere installato quale che sia quello del dispositivo che si va a programmare. Naturalmente i parametri da impostare dovranno tenere conto delle caratteristiche meccaniche ed elettriche del dispositivo.

Utilizzando l'interfaccia seriale RS-232 di servizio Modbus è possibile impostare e salvare i parametri di modo che, anche a seguito di uno spegnimento, i valori impostati siano mantenuti. Questo però vale solamente fino a che sia mantenuta attiva l'interfaccia seriale: infatti nel momento in cui il dispositivo sia collegato alla rete Profibus, saranno caricati i dati memorizzati nel PLC e di conseguenza i dati impostati tramite la seriale Modbus saranno sovrascritti.

Utilizzando STEP7 di Siemens invece è possibile memorizzare in maniera permanente ogni modifica al valore di un parametro (eccetto che nel modulo -no param). Per fare questo modificare il valore dei parametri nella pagina **Proprietà slave DP** di STEP7 (si veda alla sezione "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). Si badi che, al contrario, le modifiche realizzate mediante la tabella delle variabili (si veda alla sezione "1.3.1 Impostazione parametro 28 Valore di preset" a pagina 49) sono temporanee.

Byte 13 Non utilizzato

2.6.2 Funzione Slave → Master

In questa sezione sono descritti i messaggi Data_Exchange inviati dallo Slave al Master.

Status word (Byte 0 e 1)

In questi byte è indicato lo stato del controllore PI quando il dispositivo si trova nella modalità **Data_Exchange**.

Byte 0

Asse in posizione

bit 0

Se "=1" il dispositivo ha raggiunto la posizione programmata per il tempo impostato in **03 Tempo asse in tolleranza**. Rimane attivo fino a quando l'errore di posizione è inferiore a **02 Tolleranza di posizione**.

Stato Profibus

bit 1

Se "=1" il dispositivo si trova nello stato **Data_Exchange**.

Se "=0" il dispositivo si trova in uno degli altri stati Profibus previsti (si veda a pagina 56).

Asse abilitato

bit 2

Riporta lo stato di abilitazione del motore. Il bit è "=1" quando il motore è abilitato, cioè il PWM è attivo e l'asse in controllo ad anello chiuso (per esempio, durante un posizionamento o un jog). E' "=0" quando il motore è disabilitato, vale a dire quando l'azionamento che controlla il motore viene spento al termine di un posizionamento o di un jog o a seguito di un allarme.

Finecorsa SW +

bit 3

Se "=1" il dispositivo ha raggiunto la posizione programmata come finecorsa positivo. Si veda il dato macchina **09 Delta positivo**.

Finecorsa SW -

bit 4

Se "=1" il dispositivo ha raggiunto la posizione programmata come finecorsa negativo. Si veda il dato macchina **0A Delta negativo**.

Allarme

bit 5

Se "=1" si è verificato un allarme, si vedano i dettagli in **Allarmi (Byte 2 e 3)**.

Asse in movimento

bit 6

Stato teorico dell'asse.

Se "=0" il dispositivo è fermo.

Se "=1" il dispositivo è in movimento.

Comando in corso

bit 7

Se "=0" il controllore non sta eseguendo nessun comando.

Se "=1" il controllore sta eseguendo un comando.

Byte 1

Target raggiunto

bit 8

Se "=1" il dispositivo ha raggiunto la posizione programmata in **Target position (Byte 4 ... 7)**. Rimane attivo fino ai successivi comandi **Target position (Byte 4 ... 7)** o **Reset allarmi**.

Pulsante 1 Jog +

bit 9

L'unità RD1xA prevede l'installazione di tre pulsanti alloggiati all'interno del corpo del dispositivo e accessibili mediante la rimozione di un coperchio PG. Quando si preme il pulsante 1 JOG + il bit 9 è forzato alto "=1"; se il pulsante 1 non è attivo il bit è "=0". Per maggiori informazioni si veda la sezione "4.4 Selettori e pulsanti (Figura 5)" a pagina 27.

Pulsante 2 Jog -

bit 10

L'unità RD1xA prevede l'installazione di tre pulsanti alloggiati all'interno del corpo del dispositivo e accessibili mediante la rimozione di un coperchio PG. Quando si preme il pulsante 2 JOG - il bit 10 è forzato alto "=1"; se il pulsante 2 non è attivo il bit è "=0". Per maggiori informazioni si veda la sezione "4.4 Selettori e pulsanti (Figura 5)" a pagina 27.

Pulsante 3 Preset

bit 11

L'unità RD1xA prevede l'installazione di tre pulsanti alloggiati all'interno del corpo del dispositivo e accessibili mediante la rimozione di un coperchio PG. Quando si preme il pulsante 3 PRESET il bit 11 è forzato alto "=1"; se il pulsante 3 non è attivo il bit è "=0". Per maggiori informazioni si veda la sezione "4.4 Selettori e pulsanti (Figura 5)" a pagina 27.

Saturazione DAC

bit 12

La corrente erogata dall'elettronica di potenza per il controllo del motore ha raggiunto il livello massimo e non può essere ulteriormente aumentata.

IN 1

bit 13

Stato dell'ingresso digitale 1 del dispositivo. Il significato degli ingressi è esplicitato nel capitolo "Parametri di programmazione Profibus®" a pagina 75.

IN 1 = 0 ingresso 1 basso (non attivo)

IN 1 = 1 ingresso 1 alto (attivo)

IN 2

bit 14

Stato dell'ingresso digitale 2 del dispositivo. Il significato degli ingressi è esplicitato nel capitolo "Parametri di programmazione Profibus®" a pagina 75.

IN 2 = 0 ingresso 2 basso (non attivo)

IN 2 = 1 ingresso 2 alto (attivo)

IN 3

bit 15

Stato dell'ingresso digitale 3 del dispositivo. Il significato degli ingressi è esplicitato nel capitolo "Parametri di programmazione Profibus®" a pagina 75.

IN 3 = 0 ingresso 3 basso (non attivo)

IN 3 = 1 ingresso 3 alto (attivo)

Allarmi (Byte 2 e 3)

In questi byte sono indicati gli allarmi attivi nel dispositivo.

Byte 2

Dati macchina non validi

bit 0 0001h Uno o più parametri non sono validi, impostare valori corretti per ristabilire la normale condizione di lavoro. Controllare la lista dei parametri errati in **32 Lista parametri errati**.

Errore memoria flash

bit 1 0002h Errore interno non ripristinabile.

bit 2 Non utilizzato.

Errore di inseguimento

bit 3 0008h La differenza tra la posizione reale e quella teorica calcolata è superiore al valore del parametro **04 Max errore di inseguimento**; si consiglia di ridurre la velocità di lavoro.

Asse non sincronizzato

bit 4 0010h Errore interno non ripristinabile.

Target non valido

bit 5 0020h Posizione comandata oltre i finecorsa.

Emergenza

bit 6 0040h Il bit 7 **Emergenza** in **Control Word (Byte 0 e 1)** è stato forzato a 0; oppure sono attivi allarmi nel dispositivo.

Sovracorrente

bit 7 0080h Il valore della corrente di alimentazione è superiore al massimo consentito.

Byte 3

Sovratemperatura

bit 8 0100h Superamento della temperatura interna accettabile rilevata per mezzo di una sonda (si veda il parametro **2A Temperatura**).

Indirizzo non valido

bit 9 0200h Indirizzamento numero parametro non valido.

Sottotensione

bit 10 0400h Il valore della tensione di alimentazione è inferiore al minimo consentito.

bit 11 Non utilizzato.

Sola lettura

bit 12 1000h Tentativo di scrittura di un parametro accessibile in sola lettura (parametro "ro").

Assenza comunicazione Profibus

bit 13 2000h Questo bit segnala (valore 1) che è venuta a mancare la comunicazione nella rete Profibus (cavo disconnesso? Mancanza di tensione? ...). L'allarme è immediatamente segnalato al ripristino della comunicazione. Questo bit è considerato solo se non è attivo il bit **Controllo da PC** nel **Registro comandi Extra [0x29]** -interfaccia Modbus (se il bit **Controllo**

da PC è attivo l'assenza di comunicazione Profibus viene ignorata).

bit 14 e 15 Non utilizzati.

Per resettare una condizione di allarme utilizzare il comando **Reset allarmi, Control Word (Byte 0 e 1)** bit 3. Questo bit è normalmente impostato a "0". Lo stato normale del dispositivo è ristabilito nel cambio di stato da "0" a "1" di questo bit. Questo comando toglie lo Slave dalla condizione di allarme solo se non sono più presenti le condizioni che hanno causato l'errore.



Si badi che, se l'allarme è relativo a dati macchina non validi (si veda **Dati macchina non validi** e **32 Lista parametri errati**), si può tornare allo stato normale solo impostando dati macchina validi. L'allarme **Errore memoria flash** non è ripristinabile.

Posizione (Byte 4 ... 7)

Posizione assoluta del dispositivo al momento dell'invio del messaggio.

Numero parametro (Byte 8)

Indica l'indice del parametro cui si riferiscono i valori dei byte successivi. L'elenco degli indici è riportato nel capitolo "Parametri di programmazione Profibus®" a pagina 75.

Valore parametro (Byte 9...12)

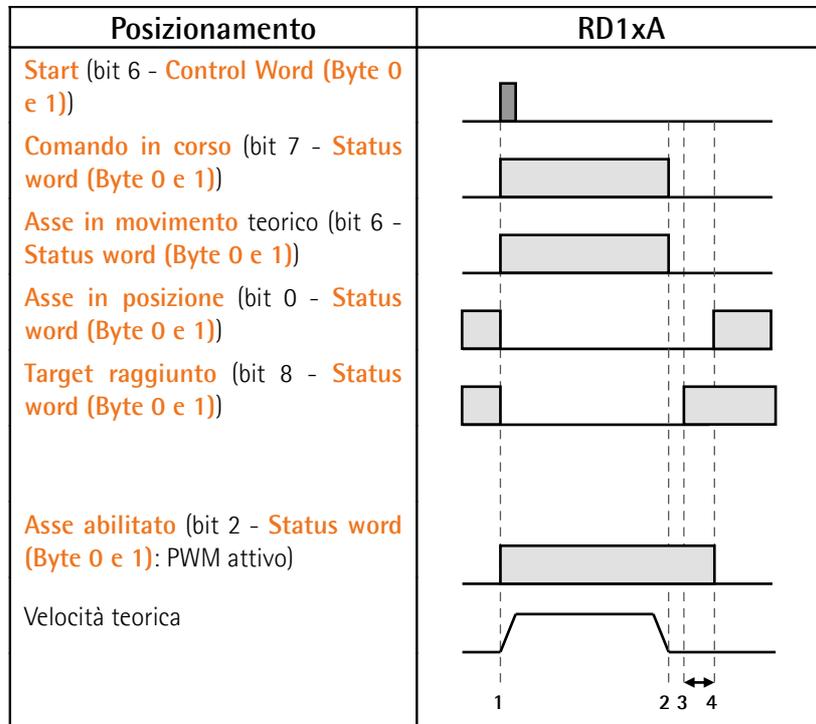
Valore del parametro il cui indice è indicato al byte precedente. Tutti i parametri utilizzano 4 byte dati, il significato è specificato nel capitolo "Parametri di programmazione Profibus®" a pagina 75.

byte 9	byte 10	byte 11	byte 12
Low	High

Byte 13 Non utilizzato



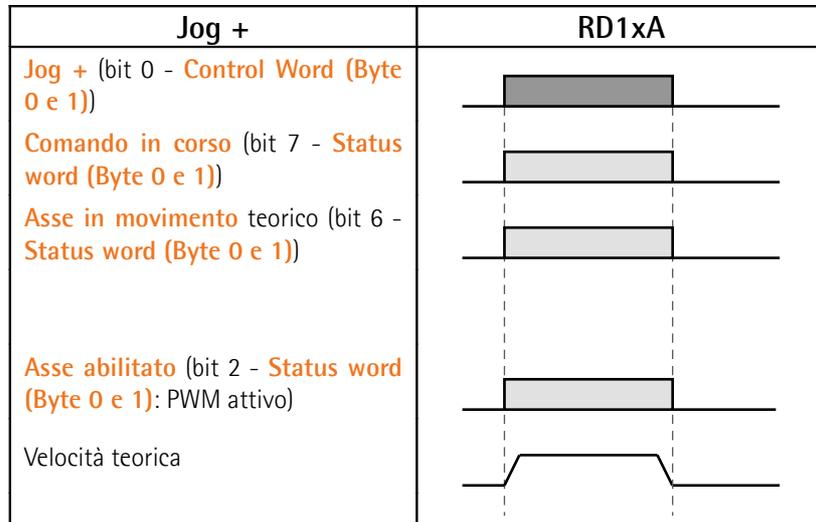
Esempio 1



- Punto 1: comando **Start** inviato dall'utilizzatore al dispositivo, il controllore attiva il PWM interno e genera il profilo di velocità; i bit **Comando in corso** e **Asse in movimento** sono a "1".
- Punto 2: il controllore ha raggiunto in modo teorico il target.
- Punto 3: l'asse è arrivato realmente all'interno della finestra tolleranza di posizione.
- Punto 4: è trascorso il tempo impostato in **03 Tempo asse in tolleranza** e quindi è dichiarata la condizione **Asse in posizione**.



Esempio 2



2.7 DDLM_Slave_Diag

Il Master può richiedere al dispositivo la diagnostica in qualsiasi momento. I dispositivi Lika prevedono la diagnostica ridotta (6 byte).

Diagnostica:

Byte	Descrizione
0	Status 1
1	Status 2
2	Status 3
3	Master ID
4	Codice costruttore
5	

3 Parametri di programmazione Profibus®

Di seguito sono riportati i parametri implementati nel dispositivo, per ognuno è indicato:

Indice Nome parametro

[tipo variabile, attributo]

- Indice: è espresso in esadecimale.
- Attributo:
 - ro = parametro accessibile in sola lettura
 - rw = parametro accessibile in lettura e scrittura

Struttura parametro Unsigned32:

Data byte			
byte 9	byte 10	byte 11	byte 12
da 2^{31} a 2^{24}	da 2^{23} a 2^{16}	da 2^{15} a 2^8	da 2^7 a 2^0
MSByte	LSByte



ATTENZIONE

Utilizzando i moduli **Lika RD1xA-T12**, **Lika RD1xA-T24**, **Lika RD1xA-T48** e **Lika RD1xA-T92** il valore di tutti i parametri è caricato all'accensione prendendo i dati memorizzati sul PLC a partire dal file GSD. Ogni modifica locale tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** è perciò temporanea: allo spegnimento del dispositivo ogni valore impostato viene perso (a eccezione del preset, unico parametro non compreso nel file GSD; e, per tutti i parametri, nel caso in cui si esegua precedentemente un'impostazione del preset, si veda alla pagina 83) e alla successiva riaccensione viene caricato il valore presente nel PLC (per cui anche i parametri salvati a seguito di un preset sono poi comunque sovrascritti).

Questo dispositivo prevede altresì la possibilità di installazione del modulo **Lika RD1xA-no param** (disponibile a partire dalla versione H3S3, file GSD V5). Con questo modulo è possibile bypassare il trasferimento dei parametri dal PLC all'accensione (i parametri infatti vengono letti dalla memoria flash) e memorizzare in flash (tramite il bit 9 **Salva parametri** in **Control Word (Byte 0 e 1)**) ogni modifica locale effettuata mediante **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)**. Con il modulo **Lika RD1xA-no param** NON è possibile leggere e modificare i parametri tramite la pagina **Parametrizza** della

finestra **Proprietà slave DP** di STEP7 (si veda il paragrafo "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). La modifica dei parametri è quindi possibile tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** oppure tramite l'interfaccia Modbus. Il modulo **Lika RD1xA-no param** è indipendente dal rapporto di riduzione e può essere installato quale che sia quello del dispositivo che si va a programmare. Naturalmente i parametri da impostare dovranno tenere conto delle caratteristiche meccaniche ed elettriche del dispositivo.

Utilizzando l'interfaccia seriale RS-232 di servizio Modbus è possibile impostare e salvare i parametri di modo che, anche a seguito di uno spegnimento, i valori impostati siano mantenuti. Questo però vale solamente fino a che sia mantenuta attiva l'interfaccia seriale: infatti nel momento in cui il dispositivo sia collegato alla rete Profibus, saranno caricati i dati memorizzati nel PLC e di conseguenza i dati impostati tramite la seriale Modbus saranno sovrascritti.

Utilizzando STEP7 di Siemens invece è possibile memorizzare in maniera permanente ogni modifica al valore di un parametro (eccetto che nel modulo -no param). Per fare questo modificare il valore dei parametri nella pagina **Proprietà slave DP** di STEP7 (si veda alla sezione "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). Si badi che, al contrario, le modifiche realizzate mediante la tabella delle variabili (si veda alla sezione "1.3.1 Impostazione parametro 28 Valore di preset" a pagina 49) sono temporanee.

Parametri Dati macchina

01 Distanza_giro

[Unsigned32, rw]

01 Distanza_giro definisce il numero di impulsi per ogni giro completo dell'albero. Questo parametro è utile per relazionare un giro dell'asse con una grandezza lineare; per esempio, se il dispositivo è montato su una vite senza fine di passo 5 mm, impostando **01 Distanza_giro** = 500 si ottiene che a ogni giro dell'asse il sistema trasla di 5 mm con una risoluzione al centesimo di millimetro. Default = 1024 (min. = 1, max. = 1024).



ATTENZIONE

Dopo la modifica di questo parametro è necessario reimpostare i parametri **0B Velocità Jog**, **0C Velocità di lavoro** e **28 Valore di preset**. Per maggiori informazioni riferirsi a pagina 36 e ai rispettivi parametri.

Si badi inoltre che i parametri di seguito elencati sono tutti espressi in relazione al parametro **01 Distanza_giro**; di conseguenza la modifica del valore nel parametro **01 Distanza_giro** comporta necessariamente una ridefinizione dei valori da essi espressi. I parametri sono: **02 Tolleranza di posizione**, **04 Max errore di inseguimento**, **07 Accelerazione**, **08 Decelerazione**, **09 Delta positivo**, **0A Delta negativo**, **29 Velocità attuale**, **30 Finecorsa assoluto positivo**, **31 Finecorsa assoluto negativo**, **Target position (Byte 4 ... 7)** e **Posizione (Byte 4 ... 7)**. Si veda per esempio la relazione che intercorre tra **01 Distanza_giro** e i valori di velocità, illustrata a pagina 80.



NOTA

Se **01 Distanza_giro** non è una potenza di due (2, ..., 512, 1024), durante il controllo di posizione potrebbe verificarsi un errore di posizionamento pari a un impulso.

02 Tolleranza di posizione

[Unsigned32, rw]

Questo parametro definisce la finestra di tolleranza da applicare a **Target position (Byte 4 ... 7)**. Se l'asse è all'interno di questa finestra per il tempo impostato nel parametro **03 Tempo asse in tolleranza**, allora è segnalato lo stato mediante il bit **Asse in posizione**. Il parametro è espresso in impulsi.

Default = 0 (min. = 0, max. = 100000).

03 Tempo asse in tolleranza

[Unsigned32, rw]

Rappresenta il tempo di assestamento dell'asse all'interno della finestra di tolleranza definita in **02 Tolleranza di posizione** dopo il quale viene dichiarata la condizione di **Asse in posizione** mediante il relativo bit. Il parametro è espresso in millisecondi.

Default = 0 (min. = 0, max. = 10000).

04 Max errore di inseguimento

[Unsigned32, rw]

Questo parametro definisce la differenza massima ammissibile tra la posizione reale del dispositivo e quella teorica calcolata istante per istante. Se il dispositivo rileva un valore superiore a questo parametro viene segnalato l'allarme **Errore di inseguimento** e il dispositivo blocca il proprio movimento. Il parametro è espresso in impulsi.

Default = 1024 (min. = 0, max. = 100000).

05 Kp anello di posizione

[Unsigned32, rw]

Questo parametro contiene il guadagno proporzionale usato dal controllore PI relativo all'anello di posizione. Tale valore è già ottimizzato da Lika Electronic in relazione alle caratteristiche del dispositivo.

Default = 400 (min. = 0, max. = 1000).

06 Ki anello di posizione

[Unsigned32, rw]

Questo parametro contiene il guadagno integrale usato dal controllore PI relativo all'anello di posizione. Tale valore è già ottimizzato da Lika Electronic in relazione alle caratteristiche del dispositivo.

Default = 100 (min. = 0, max. = 1000).

07 Accelerazione

[Unsigned32, rw]

Questo parametro definisce il valore di accelerazione usato dal dispositivo. Il parametro è espresso in [impulsi/sec²].

Default = 5000 (min. = 100, max. = 10000) per modello RD1xA-...-T12-...

Default = 2500 (min. = 100, max. = 10000) per modello RD1xA-...-T24-...

Default = 1000 (min. = 100, max. = 10000) per modello RD1xA-...-T48-...

Default = 500 (min. = 100, max. = 10000) per modello RD1xA-...-T92-...

08 Decelerazione

[Unsigned32, rw]

Questo parametro definisce il valore di decelerazione usato dal dispositivo. Il parametro è espresso in [impulsi/sec²].

Default = 5000 (min. = 100, max. = 10000) per modello RD1xA-...-T12-...

Default = 2500 (min. = 100, max. = 10000) per modello RD1xA-...-T24-...

Default = 1000 (min. = 100, max. = 10000) per modello RD1xA-...-T48-...

Default = 500 (min. = 100, max. = 10000) per modello RD1xA-...-T92-...

09 Delta positivo

[Unsigned32, rw]

È il valore utilizzato per calcolare il massimo spostamento in avanti (positivo) rispetto al Preset. Se si raggiunge il massimo spostamento in avanti viene attivata la segnalazione sul bit di stato **Finecorsa SW +**. Il parametro è espresso in impulsi encoder.

Finecorsa SW + = 28 Valore di preset + 09 Delta positivo.

Default = 523263 (min. = 0, max. = 523263).



ATTENZIONE

Si badi che il valore massimo di volta in volta ammesso in questo parametro tiene conto dello scaling impostato.



ESEMPIO

Con **01 Distanza_giro** = 1024 e **28 Valore di preset** = 0, il valore massimo ammesso in **09 Delta positivo** sarà:

$(1024 \text{ imp./giro} * 511 \text{ giri}) - 1 = 523263$

Con **01 Distanza_giro** = 256 e **28 Valore di preset** = 0, il valore massimo ammesso in **09 Delta positivo** sarà:

$(256 \text{ imp./giro} * 511 \text{ giri}) - 1 = 130815$

Si vedano anche gli ulteriori esempi al paragrafo "6.4 Distanza_giro, Velocità Jog, Velocità di lavoro, Valore di preset e valori di finecorsa" a pagina 36.



ATTENZIONE

I valori di finecorsa devono essere verificati ogniqualvolta vengono modificati i parametri **01 Distanza_giro** e **28 Valore di preset**. Dopo ogni modifica del parametro **01 Distanza_giro** si deve reimpostare anche **28 Valore di preset** in modo da definire lo zero asse in quanto il sistema di riferimento è variato. Dopo la modifica del parametro **28 Valore di preset** non occorre invece reimpostare il valore dei finecorsa in quanto la funzione di Preset provvede a ricalcolarli automaticamente reinizializzando i limiti positivo e negativo sulla base dei valori di **09 Delta positivo** e **0A Delta negativo** impostati. Per una descrizione dettagliata si veda a pagina 36.

0A Delta negativo

[Unsigned32, rw]

E' il valore utilizzato per calcolare il massimo spostamento all'indietro (negativo) rispetto al Preset. Se si raggiunge il massimo spostamento all'indietro viene attivata la segnalazione sul bit di stato **Fincorsa SW -**. Il parametro è espresso in impulsi encoder.

Fincorsa SW - = 28 Valore di preset - 0A Delta negativo.

Default = 523263 (min. = 0, max. = 523263).



ATTENZIONE

Si badi che il valore massimo di volta in volta ammesso in questo parametro tiene conto dello scaling impostato.



ESEMPIO

Con **01 Distanza_giro** = 1024 e **28 Valore di preset** = 0, il valore massimo ammesso in **0A Delta negativo** sarà:

$$(1024 \text{ imp./giro} * 511 \text{ giri}) - 1 = 523263$$

Con **01 Distanza_giro** = 256 e **28 Valore di preset** = 0, il valore massimo ammesso in **0A Delta negativo** sarà:

$$(256 \text{ imp./giro} * 511 \text{ giri}) - 1 = 130815$$

Si vedano anche gli ulteriori esempi al paragrafo "6.4 Distanza_giro, Velocità Jog, Velocità di lavoro, Valore di preset e valori di finecorsa" a pagina 36.



ATTENZIONE

I valori di finecorsa devono essere verificati ogniqualvolta vengono modificati i parametri **01 Distanza_giro** e **28 Valore di preset**. Dopo ogni modifica del parametro **01 Distanza_giro** si deve reimpostare anche **28 Valore di preset** in modo da definire lo zero asse in quanto il sistema di riferimento è variato. Dopo la modifica del parametro **28 Valore di preset** non occorre invece reimpostare il valore dei finecorsa in quanto la funzione di Preset provvede a ricalcolarli automaticamente reinizializzando i limiti positivo e negativo sulla base dei valori di **09 Delta positivo** e **0A Delta negativo** impostati. Per una descrizione dettagliata si veda a pagina 36.

0B Velocità Jog

[Unsigned32, rw]

Questo parametro definisce la velocità massima del dispositivo nell'utilizzo con le funzioni **Jog +** e **Jog -**. Il parametro è espresso in impulsi/secondo.

Default = 4266 (min. = 1, max. = 4266) per modello RD1xA-...-T12-...

Default = 2133 (min. = 1, max. = 2133) per modello RD1xA-...-T24-...

Default = 1066 (min. = 1, max. = 1066) per modello RD1xA-...-T48-...

Default = 556 (min. = 1, max. = 556) per modello RD1xA-...-T92-...

0C Velocità di lavoro

[Unsigned32, rw]

Questo parametro definisce la velocità massima del dispositivo nell'utilizzo in modalità automatica (movimenti controllati con **Start** e **Target position (Byte 4 ... 7)**). Il parametro è espresso in impulsi/secondo.

Default = 4266 (min. = 1, max. = 4266) per modello RD1xA-...-T12-...

Default = 2133 (min. = 1, max. = 2133) per modello RD1xA-...-T24-...

Default = 1066 (min. = 1, max. = 1066) per modello RD1xA-...-T48-...

Default = 556 (min. = 1, max. = 556) per modello RD1xA-...-T92-...



ATTENZIONE

I parametri **0B Velocità Jog** e **0C Velocità di lavoro** devono essere reimpostati ogniqualvolta il parametro **01 Distanza_giro** viene modificato.

Rispettare sempre la seguente relazione:

$$\frac{vel_{min} * Distanza / giro}{1024} \leq Velocità \leq \frac{vel_{max} * Distanza / giro}{1024}$$

Per una spiegazione dettagliata si veda a pagina 36.

0D Durata corrente di stacco

[Unsigned32, rw]

Questo parametro definisce il tempo massimo per il quale il motore è controllato con la corrente di stacco durante la partenza (si veda il parametro **12 Corrente di stacco**). Il parametro è espresso in millisecondi.

Default = 2000 (min.=0, max.=3000).

0E Direzione conteggio

[Boolean, rw]

Direzione di rotazione del motore per avere un incremento positivo della posizione. Il senso è stabilito guardando il dispositivo dall'estremità dell'asse.

0 = rotazione oraria (default)

1 = rotazione antioraria



ATTENZIONE

La modifica di questo parametro influenza la posizione calcolata dal controllore. Si deve quindi reimpostare il parametro **28 Valore di preset** e verificare i parametri **09 Delta positivo** e **0A Delta negativo**.

0F Kp anello di corrente

[Unsigned32, rw]

Questo parametro contiene il guadagno proporzionale usato dal controllore PI relativo all'anello di corrente. Tale valore è già ottimizzato da Lika Electronic in base alle caratteristiche del dispositivo.

Default = 200 (min. = 0, max. = 1000).

10 Ki anello di corrente

[Unsigned32, rw]

Questo parametro contiene il guadagno integrale usato dal controllore PI relativo all'anello di corrente. Tale valore è già ottimizzato da Lika Electronic in base alle caratteristiche del dispositivo.

Default = 30 (min. = 0, max. = 1000).

11 Corrente massima

[Unsigned32, rw]

Questo parametro definisce la corrente massima erogata dall'elettronica di potenza per il controllo del motore. Il parametro è espresso in mA (milliampere). Il valore in questo parametro non può essere superiore al valore impostato in **12 Corrente di stacco**.

Default = 2000 (min. = 10, max. = 2000).

12 Corrente di stacco

[Unsigned32, rw]

Questo parametro definisce la corrente massima erogata solo alla partenza e per un tempo massimo definito al parametro **0D Durata corrente di stacco**. Il parametro è espresso in mA (milliampere).

Default = 4000 (min. = 10, max. = 4000).

13 Rapporto di riduzione

[Unsigned32, ro]

È il rapporto di riduzione degli ingranaggi interni interposti tra il motore e l'asse del dispositivo. Questo parametro è disponibile in sola lettura.

Default = 12 per modello RD1xA-...-T12-...

Default = 24 per modello RD1xA-...-T24-...

Default = 48 per modello RD1xA-...-T48-...

Default = 92 per modello RD1xA-...-T92-...

14 Ampiezza passo jog

[Unsigned32, rw]

Se è abilitato il controllo del jog a passo (bit 4 **Jog incrementale** in **Control Word (Byte 0 e 1)** = 1), l'attivazione dei bit **Jog +** e **Jog -** procura in corrispondenza del fronte di salita l'esecuzione di un singolo passo in direzione positiva o negativa la cui ampiezza, espressa in impulsi, è impostata in questo parametro; quindi lo slave si arresta in attesa di un nuovo comando.

Default = 100 (min. = 1, max. = 10000).

Parametri dati operativi

28 Valore di preset

[Integer32, rw]

Usare questo parametro per assegnare un valore di Preset. La funzione di Preset è utilizzata per assegnare un determinato valore a una posizione fisica nella rotazione dell'albero encoder. La posizione fisica prescelta avrà perciò il valore assegnato in questo parametro e tutte le altre posizioni assumeranno un valore conseguente. Il valore di preset sarà assegnato alla posizione dell'asse al momento della scrittura del parametro. La scrittura del preset è eseguita in corrispondenza del fronte di salita del bit 7 del byte 8 **Numero parametro (Byte 8)** del messaggio Data_Exchange inviato dal Master allo Slave; in altri termini la scrittura del preset è richiesta alla variazione del bit 7 del byte 8 **Numero parametro (Byte 8)** dal livello logico basso (0) al livello logico alto (1). Il valore che si vuole impostare è specificato nei successivi quattro byte 9...12 **Valore parametro (Byte 9 ... 12)** del messaggio Data_Exchange. All'atto della impostazione del preset, il sistema esegue contestualmente anche un salvataggio automatico di tutti i parametri nella memoria interna.

Default = 0 (min. = -1048576, max. = +1048576).



ESEMPIO

Per scrivere un valore nel parametro **28 Valore di preset** (indice 28h) impostare 0xA8 = 1010 1000 nel byte 8 **Numero parametro (Byte 8)** del messaggio Data_Exchange inviato dal Master allo Slave:

	R/W	-	Indice parametro					
bit	7	6	5	4	3	2	1	0
binario	1	0	1	0	1	0	0	0

dove:

bit 7 = R/W = 1, cioè "scrittura parametro"

bit 6 = bit sempre impostato a 0

bit 5 ... 0 = indice parametro = 101000 binario = 28h = **28 Valore di preset**



ATTENZIONE

Il parametro **28 Valore di preset** deve essere reimpostato ogniqualvolta il parametro **01 Distanza_giro** viene modificato. Dopo la variazione del parametro **28 Valore di preset** non occorre reimpostare il valore dei finecorsa in quanto la funzione di Preset provvede a ricalcolarli automaticamente reinizializzando i limiti positivo e negativo sulla base dei valori di **09 Delta positivo** e **0A Delta negativo** impostati. Per una descrizione dettagliata si veda a pagina 36.

29 Velocità attuale

[Integer32, ro]

Questo parametro contiene il valore della velocità calcolato ogni secondo. Il parametro è espresso in impulsi/secondo.

2A Temperatura

[Integer32, ro]

Questo parametro contiene il valore della temperatura interna al dispositivo rilevata per mezzo di una sonda. Il parametro è espresso in °C (gradi Celsius). La temperatura minima rilevabile è di -20°C.

2B Corrente attuale

[Integer32, ro]

Questo parametro rappresenta il valore della corrente assorbita dal motore. Il parametro è espresso in mA (milliampere).

2C Errore di posizione

[Integer32, ro]

Questo parametro contiene la differenza tra la posizione richiesta e la posizione attuale istante per istante. Se questo valore supera il dato macchina **04 Max errore di inseguimento** il dispositivo genera l'allarme **Errore di inseguimento** e il dispositivo blocca il proprio movimento.

2D Versione software

[Unsigned16, ro]

Visualizza la versione software del dispositivo.

Il significato dei 16 bit che compongono l'indice 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 258 decimale, che corrisponde alla rappresentazione binaria 00000001 00000010, deve perciò essere interpretato come: 01 02 hex, vale a dire: versione 1.2.

2E Versione hardware

[Unsigned16, ro]

Visualizza la versione hardware del dispositivo.

Il significato dei 16 bit che compongono l'indice è il seguente:

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Modello ROTADRIVE				Interfaccia				Freno	-			Versione hardware			

dove:

00 ... 03	= versione hardware
04 ... 06	= bit non utilizzati
07	= freno (0 = senza freno; 1 = con freno)
08 .. 11	= interfaccia (00 = Modbus; 01 = Profibus; 02 = CANopen; 03 ... 0F = non utilizzati)
12 ... 15	= modello ROTADRIVE (00 = RD4; 01 = RD1xA; 02 = RD5; 03 ... 0F = non utilizzati)

Il valore 4481 decimale, che corrisponde alla rappresentazione binaria 00010001 10000001, deve perciò essere interpretato come: versione hardware 1 (bit 0 = 1); dispositivo con freno (bit 7 = 1); interfaccia Profibus (bit 8 = 1); modello RD1xA (bit 12 = 1).

2F Valore di offset

[Integer32, ro]

Questo parametro definisce la differenza tra la posizione trasmessa dal dispositivo e la posizione reale: posizione reale – preset. Il valore è espresso in impulsi.

30 Finecorsa assoluto positivo

[Unsigned32, ro]

E' il valore di limite massimo positivo calcolato in base ai valori dei parametri **28 Valore di preset** e **09 Delta positivo**. Se si raggiunge il massimo spostamento in avanti viene attivata la segnalazione sul bit di stato **Finecorsa SW +**.

Finecorsa SW + = 28 Valore di preset + 09 Delta positivo.

Il valore è espresso in impulsi encoder.

31 Finecorsa assoluto negativo

[Unsigned32, ro]

È il valore di limite massimo negativo calcolato in base ai valori dei parametri **28 Valore di preset** e **0A Delta negativo**. Se si raggiunge il massimo spostamento all'indietro viene attivata la segnalazione sul bit di stato **Finecorsa SW -**.

Finecorsa SW - = 28 Valore di preset - 0A Delta negativo.

Il valore è espresso in impulsi encoder.

32 Lista parametri errati

[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 quali parametri contengono valori errati, secondo la lista riportata nella seguente tabella.

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



NOTA

L'indice **32 Lista parametri errati** è visibile dal PLC solo se l'unità RD è entrata in linea. L'unità ROTADRIVE entra in linea solo se il PLC invia parametri corretti. Se, ad esempio, si modifica l'indice **01 Distanza_giro** direttamente nel file GSD senza modificare in modo opportuno anche i valori delle velocità, l'unità RD non può entrare in linea e di conseguenza non è possibile visualizzare questo indice e trarne informazioni. L'indice **32 Lista parametri errati** è visibile nel PLC solo se si eseguono modifiche ad esempio dalla tabella delle variabili. Quando si modifica un parametro tramite i campi **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)**, l'unità ROTADRIVE si trova necessariamente in modalità **Data_Exchange**; quindi lo scambio dati tra Master e Slave è attivo e di conseguenza è possibile leggere l'indice **32 Lista parametri errati**. All'accensione invece l'unità ROTADRIVE entra in modalità **Data_Exchange** solo dopo aver superato la fase SET_PRM inviata dal file GSD, che non deve contenere incongruenze.

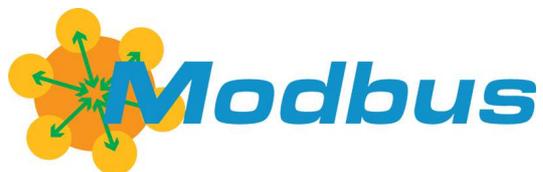
Bit	Nome parametro
1	01 Distanza_giro
2	02 Tolleranza di posizione
3	03 Tempo asse in tolleranza
4	04 Max errore di inseguimento

5	05 Kp anello di posizione
6	06 Ki anello di posizione
7	07 Accelerazione
8	08 Decelerazione
9	09 Delta positivo
10	0A Delta negativo
11	0B Velocità Jog
12	0C Velocità di lavoro
13	0D Durata corrente di stacco
14	0E Direzione conteggio
15	0F Kp anello di corrente
16	10 Ki anello di corrente
17	11 Corrente massima
18	12 Corrente di stacco
19	13 Rapporto di riduzione
20	14 Ampiezza passo jog
26	28 Valore di preset

Interfaccia Modbus®

RD1A

RD12A



Versione RS-232



Smart encoders & actuators

1 Utilizzo con seriale di servizio

1.1 Configurazione mediante software di Lika Electronic

Le unità RD1A / RD12A possono comunicare in bus di campo mediante le interfacce Profibus-DP, CANopen DS 301 e Modbus RTU. Nelle versioni Profibus e CANopen la parametrizzazione e la messa in funzione possono essere effettuate anche tramite un'interfaccia seriale di servizio RS-232 in protocollo Modbus. A questo scopo sono fornite 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 del dispositivo. 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 **SW_RDX_MODBUS.EXE** è contenuto nel supporto informatico allegato oppure disponibile all'indirizzo **www.lika.it > ATTUATORI ROTATIVI > CAMBIAFORMATI (DRIVECOD) > RD1A / RD12A**. Il programma non richiede installazione; per lanciarlo è sufficiente fare un doppio click sull'icona del file. Per chiudere poi il programma, premere il pulsante **DISCONNETTI** nella pagina **Configurazione seriale**, quindi il pulsante **CHIUDI** nella barra del titolo.



ATTENZIONE

Si badi di rispettare sempre le seguenti compatibilità tra la versione del file GSD e la versione software dell'eseguibile Modbus.

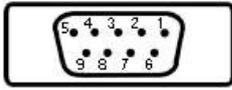
Compatibilità	File GSD	EXE Modbus
	V1	fino a 2.1
	V2	fino a 2.4
	V3-V4-V5	da 2.5 a ...



NOTA

Prima di connettersi al dispositivo è necessario collegarlo serialmente al personal computer. L'interfaccia seriale dell'unità ROTADRIVE per collegamento in rete Modbus RTU è del tipo RS-232. Nel caso in cui il personal computer non sia provvisto di porta seriale, sarà necessario installare un convertitore USB / RS-232, facilmente reperibile in commercio.

Sul lato ROTADRIVE, il cavo seriale deve essere collegato al connettore M12 8 pin maschio (INGRESSI / USCITE + RS-232 Modbus). Si veda anche il capitolo "Connessioni elettriche" a pagina 20.

			
Funzione	Ingressi / uscite + RS-232 connettore M12 maschio 8 pin	Connettore vaschetta 9 pin femmina	Funzione
TD	6	2	RD
RD	7	3	TD
0 VDC	8	5	0 VDC

Assicurarsi che RD dell'unità ROTADRIVE sia connesso con TD del PC e che TD del PC sia connesso con RD dell'unità ROTADRIVE.

Il cavo precablato con connettori M12F8 / D9F è disponibile su richiesta contattando il servizio di assistenza tecnica di Lika Electronic s.r.l. e citando il seguente codice: **EXC-M12F8-LK-xx-D9F-S51**.

Si badi che i parametri di configurazione della porta seriale di servizio Modbus sono fissi e perciò non modificabili.

Essi sono:

Impostazioni porta seriale di servizio Modbus RS-232

	Valore di default
Baud rate	9600
Byte size	8
Parity	Even
Stop bits	1



ATTENZIONE

Utilizzando l'interfaccia seriale RS-232 di servizio Modbus è possibile impostare e salvare i parametri di modo che, anche a seguito di uno spegnimento, i valori impostati siano mantenuti. Questo però vale solamente fino a che sia mantenuta attiva l'interfaccia seriale: infatti nel momento in cui il dispositivo sia collegato alla rete Profibus, saranno caricati i dati memorizzati nel PLC e di conseguenza i dati impostati tramite la seriale Modbus saranno sovrascritti.

In Profibus utilizzando i moduli **Lika RD1xA-T12**, **Lika RD1xA-T24**, **Lika RD1xA-T48** e **Lika RD1xA-T92** il valore di tutti i parametri è caricato all'accensione prendendo i dati memorizzati sul PLC a partire dal file GSD. Ogni modifica locale tramite **Numero parametro (Byte 8)** e **Valore parametro**

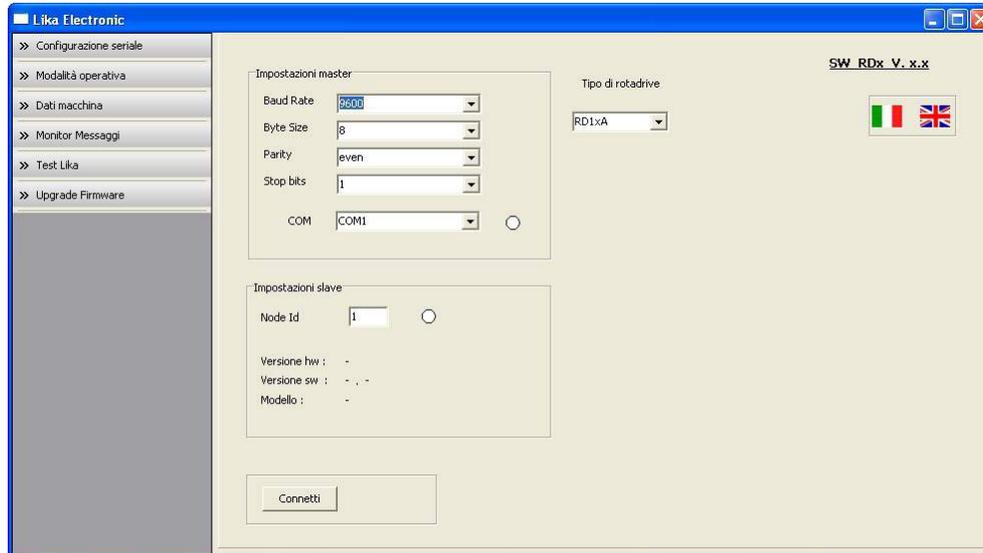
(Byte 9 ... 12) è perciò temporanea: allo spegnimento del dispositivo ogni valore impostato viene perso (a eccezione del preset, unico parametro non compreso nel file GSD; e, per tutti i parametri, nel caso in cui si esegua precedentemente un'impostazione del preset, si veda alla pagina 83) e alla successiva riaccensione viene caricato il valore presente nel PLC (per cui anche i parametri salvati a seguito di un preset sono poi comunque sovrascritti).

Questo dispositivo prevede altresì la possibilità di installazione del modulo **Lika RD1xA-no param** (disponibile a partire dalla versione H3S3, file GSD V5). Con questo modulo è possibile bypassare il trasferimento dei parametri dal PLC all'accensione (i parametri infatti vengono letti dalla memoria flash) e memorizzare in flash (tramite il bit 9 **Salva parametri** in **Control Word (Byte 0 e 1)**) ogni modifica locale effettuata mediante **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)**. Con il modulo **Lika RD1xA-no param** NON è possibile leggere e modificare i parametri tramite la pagina **Parametrizza** della finestra **Proprietà slave DP** di STEP7 (si veda il paragrafo "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). La modifica dei parametri è quindi possibile tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** oppure tramite l'interfaccia Modbus. Il modulo **Lika RD1xA-no param** è indipendente dal rapporto di riduzione e può essere installato quale che sia quello del dispositivo che si va a programmare. Naturalmente i parametri da impostare dovranno tenere conto delle caratteristiche meccaniche ed elettriche del dispositivo.

Utilizzando STEP7 di Siemens invece è possibile memorizzare in maniera permanente ogni modifica al valore di un parametro (eccetto che nel modulo -no param). Per fare questo modificare il valore dei parametri nella pagina **Proprietà slave DP** di STEP7 (si veda alla sezione "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). Si badi che, al contrario, le modifiche realizzate mediante la tabella delle variabili (si veda alla sezione "1.3.1 Impostazione parametro 28 Valore di preset" a pagina 49) sono temporanee.

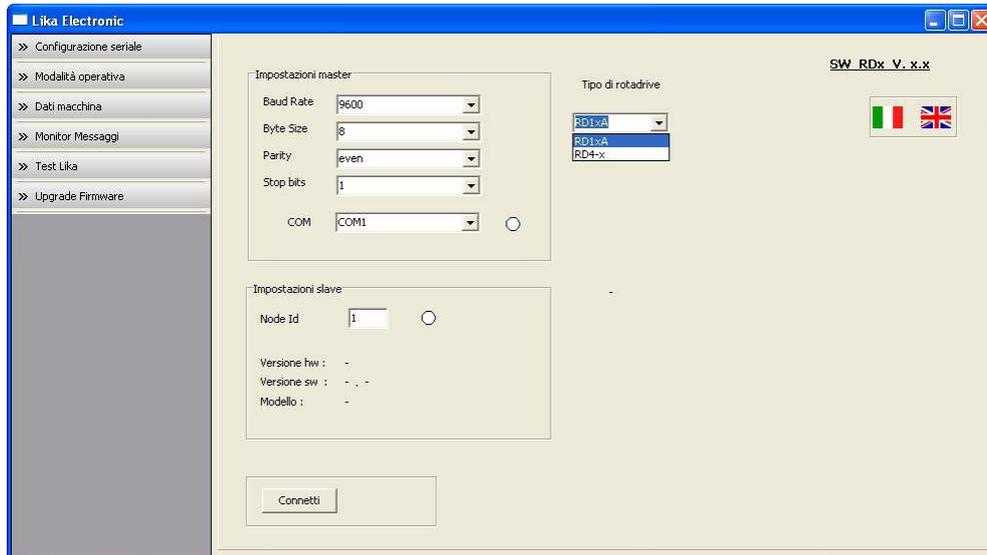
1.2 Pagina "Configurazione seriale"

All'avvio del programma, si visualizza la pagina **Configurazione seriale**.

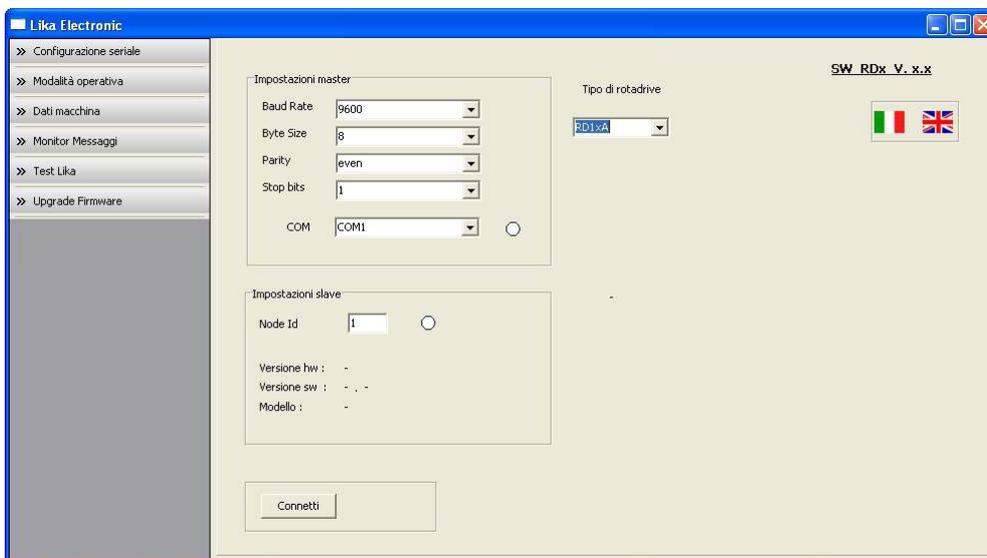


Questa pagina permette anzitutto la scelta della lingua di visualizzazione dei testi. Cliccare sul pulsante  **Bandiera italiana** per scegliere la lingua di visualizzazione italiana; cliccare sul pulsante  **Bandiera inglese** per scegliere la lingua di visualizzazione inglese.

Prima di accedere alle pagine successive attraverso i pulsanti del menu sul lato sinistro, è necessario attivare la connessione seriale con il dispositivo. Per fare questo aprire il menu a tendina **Tipo di Rotadrive** e selezionare il modello **RD1xA**.



Sul lato sinistro del menu a tendina si attiverà il box **Impostazioni master** che permette di scegliere la porta seriale del personal computer cui il dispositivo è connesso (menu a tendina **COM**). I parametri della porta seriale del PC dovranno necessariamente corrispondere ai parametri della porta seriale del dispositivo collegato. Si badi che i parametri di configurazione della porta seriale di servizio Modbus sono fissi e perciò non modificabili.



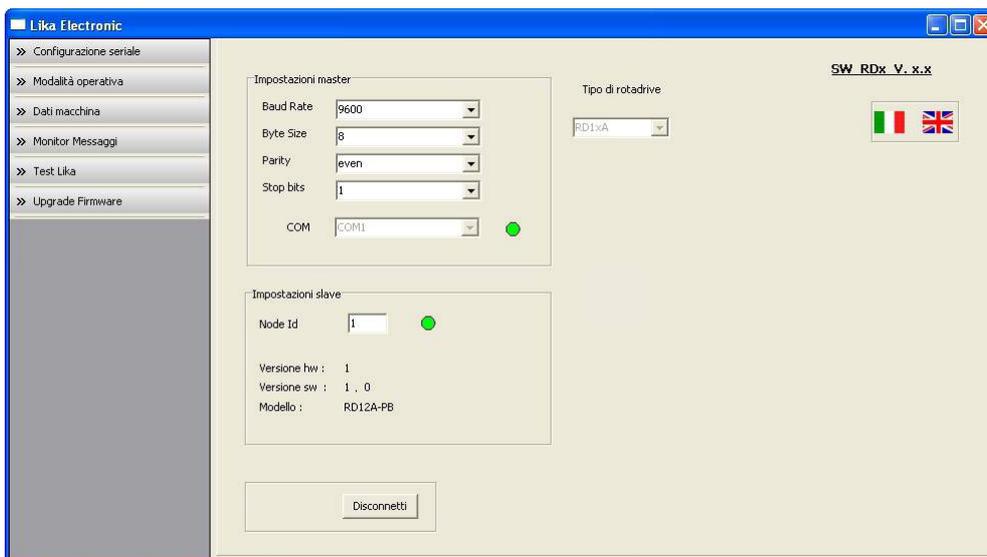
Essi sono:

Impostazioni porta seriale di servizio Modbus RS-232

	Valore di default
Baud rate	9600
Byte size	8
Parity	Even
Stop bits	1

Impostare quindi l'indirizzo del nodo cui ci si vuole connettere nel box **Impostazioni slave** (valore di default dei dispositivi RD1xA = 1). Per impostare l'indirizzo del nodo riferirsi al paragrafo "4.4.1 Indirizzo nodo: Node ID (Figura 5)" a pagina 28.

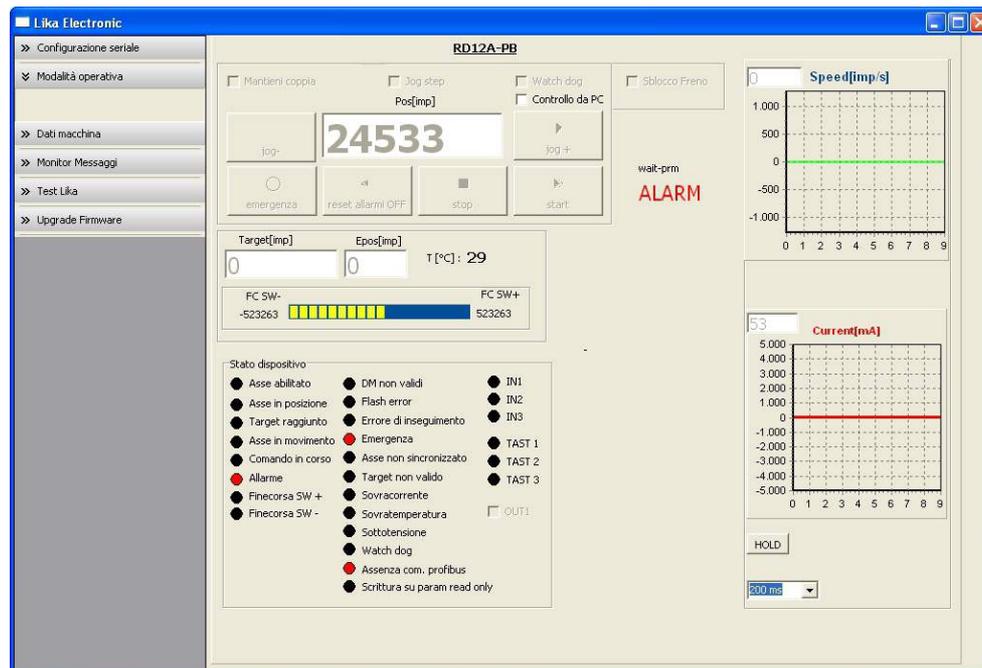
E' ora possibile attivare la connessione premendo il pulsante **CONNETTI** in basso nella pagina.



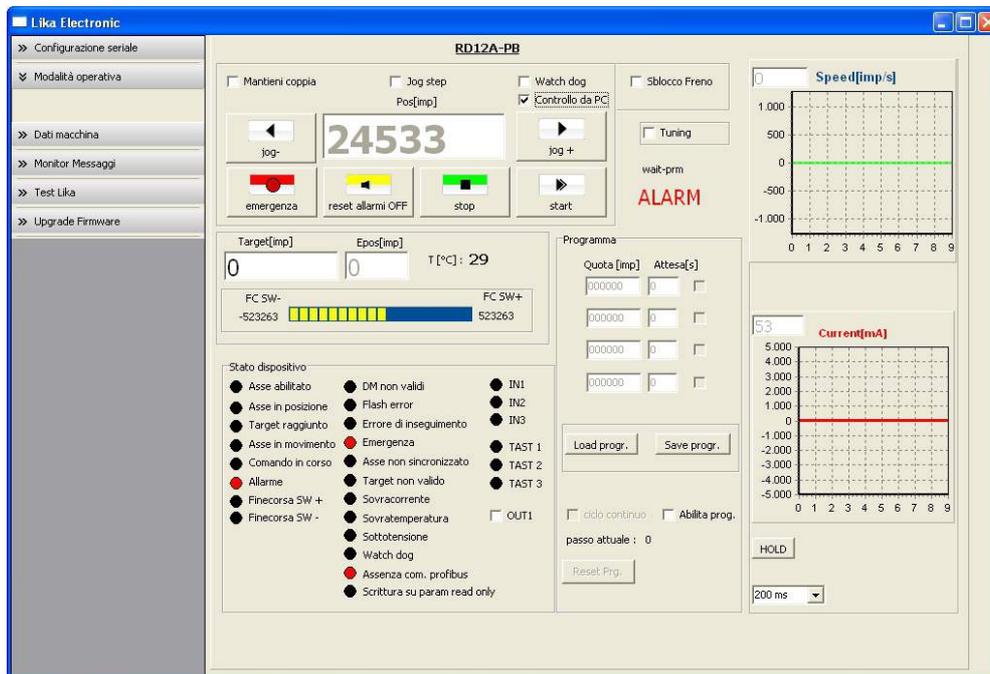
Se la connessione ha esito positivo, le due segnalazioni in corrispondenza dei campi di impostazione della **porta seriale** e dell'**indirizzo del nodo** si illuminano colorandosi di verde, mentre il pulsante **CONNETTI** sparisce sostituito dal pulsante **DISCONNETTI**. Compaiono inoltre la versione hardware e quella software, nonché il modello del dispositivo nel box **Impostazioni slave**.

1.3 Pagina "Modalità operativa"

Premere ora il pulsante **MODALITA' OPERATIVA** nel menu sul lato sinistro. Compare la seguente schermata.



All'apertura della pagina i comandi disponibili sono disabilitati in quanto il dispositivo si trova ancora sotto il controllo della rete Profibus (assieme al messaggio **ALARM** infatti lampeggia il messaggio di stato **WAIT_PRM**: esso indica che il dispositivo è attivo in rete Profibus e si trova in attesa di ricevere i parametri dal Master Profibus-DP). Per avviare l'attività di programmazione, gestione manuale e monitoraggio del dispositivo attraverso la seriale di servizio Modbus è necessario abilitare i comandi acquisendo prima il controllo del dispositivo in rete Modbus mediante il PC. Per fare questo selezionare il check box **CONTROLLO DA PC**, l'unico comando attivo nella pagina (si veda il [Registro comandi Extra \[0x29\]](#) a pagina 133).



Tutti i comandi diventano immediatamente disponibili all'uso.

All'attivazione della modalità operativa il dispositivo si trova necessariamente in condizione di emergenza: il pulsante **EMERGENZA** è evidenziato in rosso e la segnalazione **ALARM** lampeggia sulla destra, mentre nel box **Stato dispositivo** in basso a sinistra sono attive le due segnalazioni **Allarme** e **Emergenza**; inoltre il LED 2 rosso del dispositivo è acceso fisso. Per ripristinare lo stato **Idle** del dispositivo, premere il pulsante **EMERGENZA** prima e il pulsante **RESET ALLARMI** poi. Le segnalazioni di allarme scompaiono e il LED 3 verde del dispositivo inizia a lampeggiare.

Nel box **RD1xA-PB** in alto a sinistra sono disponibili le seguenti funzioni.

Mantieni coppia

Si veda **Asse in coppia** a pagina 136. Disponibile solo per la versione RD1A, nella versione RD12A il check box non è visualizzato.

Jog step

Si veda **Jog incrementale** a pagina 135.

Watch dog

Si veda **Abilitazione Watch dog** a pagina 136.

Jog -

Si veda **Jog -** a pagina 134.

Pos [imp]

Si veda **Posizione corrente [0x02-0x03]** a pagina 143.

Jog +

Si veda **Jog +** a pagina 134.

Emergenza

Se è presente una condizione di emergenza, il pulsante **EMERGENZA** è evidenziato in rosso; premere il pulsante per ripristinare lo stato normale di lavoro. Con dispositivo in movimento, comanda un arresto di emergenza. Si veda **Emergenza** a pagina 135.

Reset allarmi

Se è presente un allarme, il pulsante **RESET ALLARMI** è evidenziato in giallo; premere il pulsante per resettare l'allarme. Si veda **Reset allarmi** a pagina 135.

Stop

Comanda l'arresto normale del dispositivo. Si veda **Stop** a pagina 134.

Start

Comanda il movimento del dispositivo per il raggiungimento della posizione impostata in **Target [imp]**; sul lato destro lampeggia l'avvertenza **MOVING**. Al raggiungimento della quota impostata il dispositivo si arresta attivando i bit di stato **Asse in posizione** e **Target raggiunto**. Per un arresto normale del dispositivo premere **STOP**; per un arresto di emergenza premere **EMERGENZA**. Si veda **Start** a pagina 135.

Nel box sottostante sono disponibili le seguenti funzioni:

Target [imp]

Si veda **Target position [0x2B-0x2C]** a pagina 137. Impostare il valore desiderato e confermare con il tasto **INVIO** della tastiera. Alla pressione del pulsante **START** il dispositivo avvia il posizionamento alla quota impostata in questo campo **Target [imp]**, quindi si arresta attivando i bit di stato **Asse in posizione** e **Target raggiunto**.

E pos [imp]

Si veda **Errore di inseguimento [0x05-0x06]** a pagina 144.

T [°C]

Si veda **Temperatura [0x08]** a pagina 144.

FC SW - / FC SW +

Visualizzazione grafica dei valori di finecorsa positivo e negativo. Si vedano **Delta spazio positivo [0x08-0x09]** a pagina 128 e **Delta spazio negativo [0x0A-0x0B]** a pagina 129.

Nel box **Stato dispositivo** in basso nella pagina è visualizzato l'elenco degli stati e degli allarmi disponibili per il dispositivo. Gli stati attivi sono evidenziati in verde; gli allarmi attivi sono evidenziati in rosso. Per una descrizione completa degli stati si veda **Status word [0x01]** a pagina 141; per una descrizione completa degli allarmi si veda **Registro allarmi [0x00]** a pagina 139. Il check box **OUT1** attiva / disattiva l'uscita digitale 1 del dispositivo. Per ogni informazione si veda a pagina 137.

Le funzioni disponibili nel box **Programma** permettono la creazione e il salvataggio di un programma di lavoro del dispositivo. Le funzioni del box **Programma** sono per default disabilitate; per abilitarle selezionare il check box **ABILITA PROG**.

Nei campi **QUOTA [IMP]** si impostano le quote che il dispositivo deve raggiungere (posizione target); si possono impostare fino a quattro quote successive. Nei campi **ATTESA [S]** si imposta la pausa tra un posizionamento e il successivo. I valori impostati devono essere confermati premendo il tasto **INVIO** della tastiera. Ogni campo deve essere preventivamente abilitato selezionando il check box a lato.

Il check box **CICLO CONTINUO** in basso permette di attivare l'esecuzione continua del ciclo di posizionamento impostato.

Se **CICLO CONTINUO** è selezionato, alla pressione del pulsante **START**, il dispositivo avvia il posizionamento alla prima quota impostata; i campi **QUOTA [IMP]** e **ATTESA [S]** si evidenziano in giallo; al raggiungimento della quota impostata, il dispositivo si arresta per la pausa impostata e il campo **QUOTA [IMP]** si evidenzia in verde; allo scadere della pausa impostata (si visualizza un contatore a ritroso) anche il campo **ATTESA [S]** si evidenzia in verde. Allo scadere della pausa, immediatamente il dispositivo avvia il posizionamento alla

seconda quota impostata; e così di seguito dalla prima alla quarta quota (se abilitate) e di nuovo dalla prima alla quarta senza soluzione di continuità fino alla pressione del pulsante **STOP**.

Se invece **CICLO CONTINUO** non è selezionato, alla pressione del pulsante **START**, il dispositivo avvia il posizionamento alla prima quota impostata e al raggiungimento della quota si arresta e attiva il contatore per l'esecuzione della pausa; quindi, allo scadere della pausa, rimane in attesa di un nuovo comando **START**; e così di seguito.

E' possibile salvare il programma di lavoro creato. Per fare questo premere il pulsante **SAVE PROGR**. Alla pressione del pulsante si visualizza il dialog box **Salva con nome**: l'operatore deve digitare il nome del file .prg dove i dati saranno salvati e indicarne il repository. Alla conferma mediante il pulsante **SALVA**, il box si chiude.

Per caricare un programma precedentemente salvato, premere il pulsante **LOAD PROGR**. Alla pressione del pulsante si visualizza il dialog box **Apri**: l'operatore deve selezionare il file .prg precedentemente salvato e confermare la scelta mediante il pulsante **APRI**, il box si chiude.

Il pulsante **RESET PRG** resetta l'esecuzione del programma in corso: alla successiva pressione del pulsante **START** il dispositivo avvierà il posizionamento alla prima quota impostata, quale che fosse la quota raggiunta al precedente arresto.

Per disattivare l'esecuzione di un programma deselezionare il check box **ABILITA PROG**.

Sul lato destro della pagina sono visualizzati graficamente la velocità del dispositivo (espressa in impulsi/secondo) e il valore della corrente assorbita dal motore (espresso in milliampere). Premere il pulsante **HOLD** per disattivare la visualizzazione; premere lo stesso pulsante (che presenta poi l'etichetta **GO**) per riattivarla. Il menu a tendina in basso permette di scegliere la scala di visualizzazione temporale.

In alto, a destra del box **RD1xA-PB** è presente il check box **TUNING**. Alla selezione del check box, i diagrammi della velocità e della corrente assorbita presenti sulla destra della pagina scompaiono sostituiti da quattro slider per l'impostazione dei valori di guadagno proporzionale e integrale relativi all'anello di posizione e all'anello di corrente. Per ogni informazione riferirsi alle rispettive variabili in questo manuale (sezione "3.1.1 Parametri Dati macchina" a pagina 124).

Quando ci si collega a un dispositivo modello RD12A, appena sopra il check box **TUNING** compare un ulteriore check box: **SBLOCCO FRENO**. Il modello RD12A si differenzia dal modello RD1A per la presenza di un freno che, all'arresto del dispositivo, si attiva inibendo la possibilità di qualsiasi movimento dell'asse. Selezionando questo check box, il freno è momentaneamente disattivato, per cui, per esempio, è possibile muovere manualmente l'asse dell'unità ROTADRIVE. Il freno non rimane disattivato per un tempo indefinito, il sistema prevede un time-out di 20 secondi allo scadere del quale il freno si riattiva automaticamente.

1.4 Pagina "Dati macchina"

Premendo il pulsante **DATI MACCHINA** nel menu sul lato sinistro si accede alla pagina **Dati Macchina**.

The screenshot shows the 'Lika Electronic' software window with the 'Dati macchina' (Machine Data) configuration page. The interface is divided into a sidebar menu on the left and a main parameter configuration area on the right. The sidebar menu includes options like 'Configurazione seriale', 'Modalità operativa', 'Dati macchina', 'Monitor Messaggi', 'Test Lika', and 'Upgrade Firmware'. The main area displays various parameters with input fields and min/max values.

Parameter	Value	Min	Max
Rapporto di riduzione	12		
Distanza/giro [imp/giro]	1024	1	1024
Delta spazio pos[imp]	523263	0	523263
Delta spazio neg[imp]	523263	0	523263
Preset [imp]	0	-1048576	1048576
Offset [imp]	0		
Velocità di jog [imp/s]	4266	1	4266
Velocità di lavoro [imp/s]	4266	1	4266
Accelerazione [imp/s²]	5000	100	10000
Decelerazione [imp/s²]	5000	100	10000
Direzione conteggio	0 (CW)		
Amplezza passo jog [imp]	100	1	10000
Max err di ins.[imp]	1024	0	65535
Toll. di posizione [imp]	0	0	65535
Tempo asse in toll.[ms]	0	0	10000
Kp (anello di posizione)	500	0	1000
Ki (anello di posizione)	30	0	1000
Kp (anello di corrente)	200	0	1000
Ki (anello di corrente)	15	0	1000
Corrente massima [mA]	2000	0	2000
Corrente di stacco [mA]	4000	0	4000
Durata coppia stacco [ms]	2000	0	3000

Buttons: Salva Parametri, Carica parametri di default

In questa pagina è disponibile l'elenco dei parametri di impostazione del dispositivo (dati macchina). A sinistra di ciascun campo di impostazione è visualizzato il valore attualmente caricato sul dispositivo; a destra invece sono indicati i valori minimo e massimo ammessi. Per ogni informazione sulla funzione e la programmazione di ciascun parametro riferirsi alla sezione "3.1.1 Parametri Dati macchina" a pagina 124.

Per impostare un nuovo valore in un parametro digitare il valore desiderato, quindi premere il tasto **INVIO** della tastiera. Se si imposta un valore non conforme, alla conferma il campo di impostazione si colora di rosso e il dispositivo è forzato in condizione di allarme (attivazione del bit di stato **Allarme** e visualizzazione dei messaggi di errore **Dati macchina non validi** e/o **Emergenza**). Per ripristinare la normale modalità operativa del dispositivo, impostare un valore valido, quindi premere il pulsante **RESET ALLARMI** nella pagina **Modalità operativa**.

Per salvare i parametri modificati nella memoria non volatile del dispositivo premere il pulsante **SALVA PARAMETRI**. Nel caso di spegnimento del dispositivo i dati non salvati andranno persi! Per ogni informazione sul salvataggio dei parametri riferirsi alla variabile **Salva parametri** a pagina 136.

Per caricare i parametri di default (parametri impostati durante la messa a punto a vuoto del dispositivo in azienda che permettono un funzionamento standard e sicuro del dispositivo) premere il pulsante **CARICA PARAMETRI DI DEFAULT**. Per ogni informazione sull'impostazione dei parametri di default riferirsi alla variabile **Carica parametri di default** a pagina 136. A pagina 40 è disponibile l'elenco dei dati macchina e il rispettivo valore di default preimpostato da Lika Electronic.



ATTENZIONE

Utilizzando l'interfaccia seriale RS-232 di servizio Modbus è possibile impostare e salvare i parametri di modo che, anche a seguito di uno spegnimento, i valori impostati siano mantenuti. Questo però vale solamente fino a che sia mantenuta attiva l'interfaccia seriale: infatti nel momento in cui il dispositivo sia collegato alla rete Profibus, saranno caricati i dati memorizzati nel PLC e di conseguenza i dati impostati tramite la seriale Modbus saranno sovrascritti.

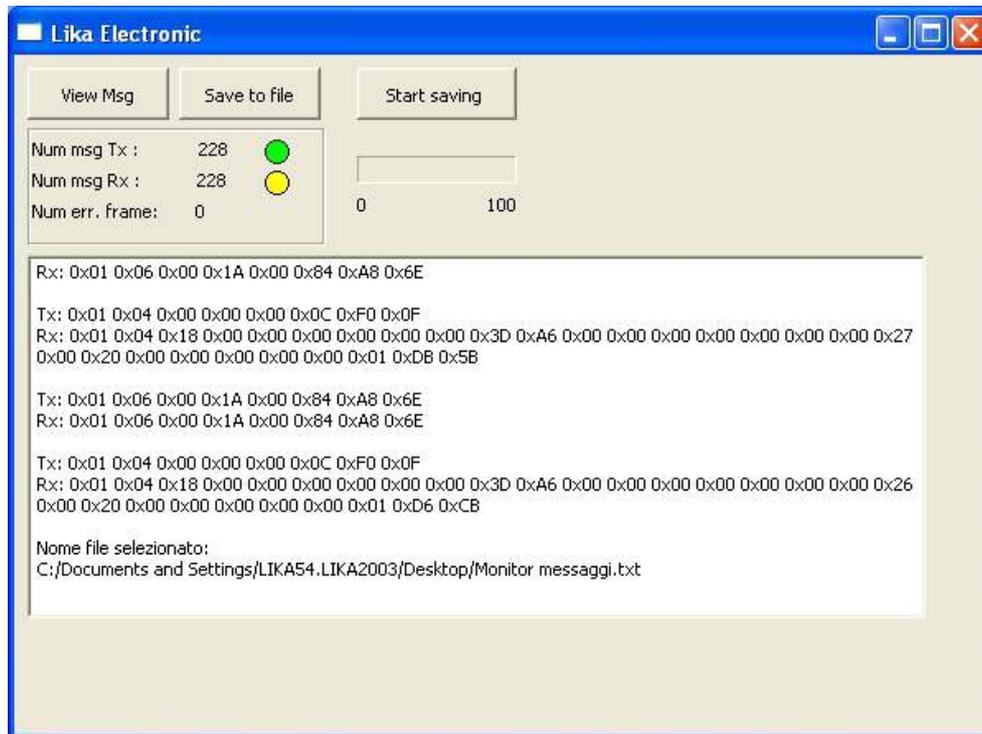
In Profibus utilizzando i moduli **Lika RD1xA-T12**, **Lika RD1xA-T24**, **Lika RD1xA-T48** e **Lika RD1xA-T92** il valore di tutti i parametri è caricato all'accensione prendendo i dati memorizzati sul PLC a partire dal file GSD. Ogni modifica locale tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** è perciò temporanea: allo spegnimento del dispositivo ogni valore impostato viene perso (a eccezione del preset, unico parametro non compreso nel file GSD; e, per tutti i parametri, nel caso in cui si esegua precedentemente un'impostazione del preset, si veda alla pagina 83) e alla successiva riaccensione viene caricato il valore presente nel PLC (per cui anche i parametri salvati a seguito di un preset sono poi comunque sovrascritti).

Questo dispositivo prevede altresì la possibilità di installazione del modulo **Lika RD1xA-no param** (disponibile a partire dalla versione H3S3, file GSD V5). Con questo modulo è possibile bypassare il trasferimento dei parametri dal PLC all'accensione (i parametri infatti vengono letti dalla memoria flash) e memorizzare in flash (tramite il bit 9 **Salva parametri in Control Word (Byte 0 e 1)**) ogni modifica locale effettuata mediante **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)**. Con il modulo **Lika RD1xA-no param** NON è possibile leggere e modificare i parametri tramite la pagina **Parametrizza** della finestra **Proprietà slave DP** di STEP7 (si veda il paragrafo "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). La modifica dei parametri è quindi possibile tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** oppure tramite l'interfaccia Modbus. Il modulo **Lika RD1xA-no param** è indipendente dal rapporto di riduzione e può essere installato quale che sia quello del dispositivo che si va a programmare. Naturalmente i parametri da impostare dovranno tenere conto delle caratteristiche meccaniche ed elettriche del dispositivo.

Utilizzando STEP7 di Siemens invece è possibile memorizzare in maniera permanente ogni modifica al valore di un parametro (eccetto che nel modulo -no param). Per fare questo modificare il valore dei parametri nella pagina **Proprietà slave DP** di STEP7 (si veda alla sezione "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). Si badi che, al contrario, le modifiche realizzate mediante la tabella delle variabili (si veda alla sezione "1.3.1 Impostazione parametro 28 Valore di preset" a pagina 49) sono temporanee.

1.5 Pagina "Monitor messaggi"

Alla pressione del pulsante **MONITOR MESSAGGI** nel menu sul lato sinistro si accede alla pagina **Monitor Messaggi**.



Tramite questa pagina l'operatore può monitorare la comunicazione tra Master e Slave, visualizzando i messaggi Request PDU (Tx) e Response PDU (Rx). Alla prima apertura della pagina, il campo di visualizzazione dei messaggi è vuoto. Nel box posto immediatamente sotto i pulsanti è visualizzato il numero di messaggi trasmessi (Num msg TX) e il numero di messaggi ricevuti (Num msg Rx).

Per visualizzare i messaggi è necessario premere il pulsante **VIEW MSG**. Non appena il pulsante viene premuto la pagina inizia a visualizzare il flusso di trasmissione tra Master e Slave. I messaggi sono visualizzati in formato esadecimale. Dopo la pressione del pulsante **VIEW MSG**, la label dello stesso pulsante cambia in **HOLD MSG**. Premere dunque il pulsante **HOLD MSG** per arrestare la visualizzazione dei messaggi.

E' possibile memorizzare i messaggi in un file di testo. Per fare questo premere il pulsante **SAVE TO FILE**. Alla pressione del pulsante si apre il dialog box **Apertura file di log**: l'operatore deve digitare il nome del file .txt dove i dati saranno salvati e indicarne il repository. Alla conferma mediante il pulsante **APRI**, il box si chiude e nel campo di visualizzazione della pagina **Monitor**

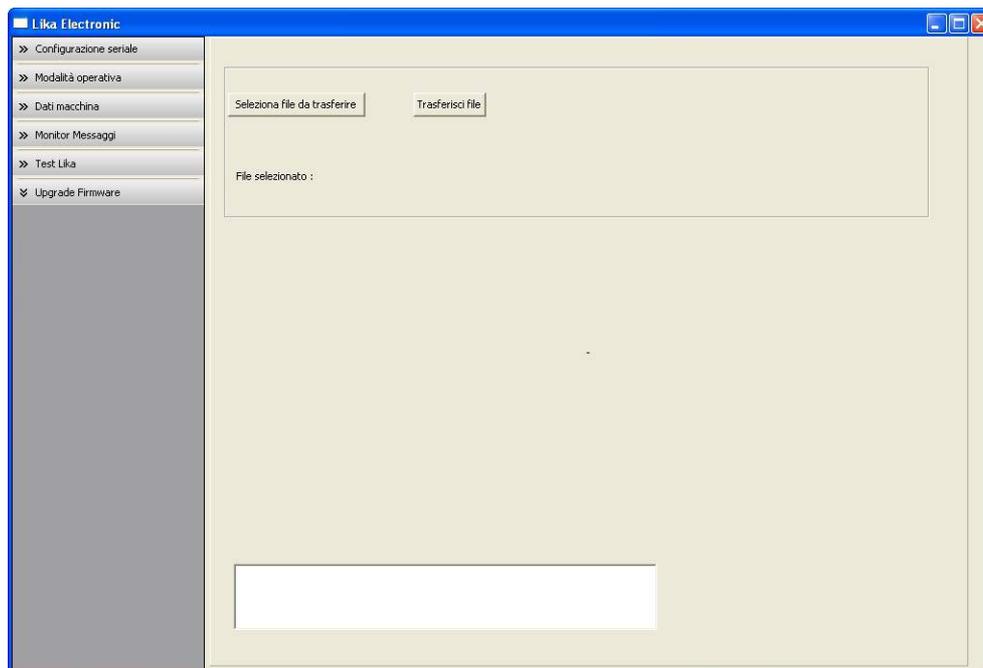
Messaggi compare il path completo del file indicato. Premere quindi il pulsante **START SAVING** per iniziare il salvataggio dei messaggi, nel campo di visualizzazione della pagina **Monitor Messaggi** compare il messaggio "File aperto correttamente". Dopo la pressione del pulsante **START SAVING**, la label dello stesso pulsante cambia in **STOP SAVING**. Premere dunque il pulsante **STOP SAVING** per arrestare la memorizzazione dei messaggi.

1.6 Pagina "Test Lika"

La pagina **Test Lika** è riservata all'utilizzo da parte dei tecnici di Lika Electronic e non è accessibile all'utente.

1.7 Pagina "Upgrade Firmware"

Alla pressione del pulsante **UPGRADE FIRMWARE** nel menu sul lato sinistro si accede alla pagina **Upgrade Firmware**.



Le funzioni disponibili in questa pagina permettono l'aggiornamento del firmware dell'unità ROTADRIVE mediante il trasferimento dei dati di upgrade alla memoria flash.

Il firmware è un programma software che permette la gestione e il controllo del funzionamento di un dispositivo; il programma firmware, qui chiamato anche "user program" o "programma utente", è memorizzato nella memoria flash integrata all'interno dell'unità ROTADRIVE. I dispositivi ROTADRIVE sono progettati in modo che il firmware possa essere aggiornato agevolmente e direttamente dall'utente finale. Questo permette 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 o talora aggiungere nuove funzionalità al dispositivo.

L'aggiornamento firmware consiste in un file con estensione .BIN fornito direttamente dal Servizio di Assistenza Tecnica di Lika Electronic.



ATTENZIONE

Il processo di aggiornamento del firmware di un dispositivo ROTADRIVE deve essere eseguito da personale esperto e competente. L'applicazione di un aggiornamento errato o incompatibile pregiudica il funzionamento del dispositivo stesso.

Se la versione firmware più recente è già installata nel dispositivo ROTADRIVE, non è necessario procedere con l'installazione di alcun aggiornamento. La versione firmware correntemente installata può essere verificata alla voce **VERSIONE SW** nel box **Impostazioni slave** della pagina **Configurazione seriale** dopo la corretta connessione al dispositivo (si veda a pagina 94).

La funzione di aggiornamento firmware è implementata:

- a partire dalla versione firmware 1.0 nelle unità ROTADRIVE con interfaccia CANopen e Modbus;
- a partire dalla versione firmware 2.0 nelle unità ROTADRIVE con interfaccia Profibus.

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



NOTA

La funzione di aggiornamento del firmware è implementata a partire dalla versione 2.0 del software di programmazione via seriale in rete Modbus **SW_RDX_MODBUS_2.0.EXE**. Prima di procedere all'aggiornamento del firmware assicurarsi di possedere la versione citata o successiva del programma oppure scaricarla all'indirizzo **www.lika.it > ATTUATORI ROTATIVI > CAMBIAFORMATI (DRIVECOD) > RD1A / RD12A**.

Per l'installazione dell'aggiornamento firmware procedere come segue:

1. assicurarsi che l'unità ROTADRIVE da aggiornare sia l'unico nodo collegato al personal computer;
2. assicurarsi di avviare la versione 2.0 o successiva del software di programmazione via seriale in rete Modbus SW_RDX_MODBUS_X.EXE;
3. collegarsi al dispositivo e accedere alla pagina **Upgrade Firmware**;
4. se, all'accensione, i LED 2 e 3 lampeggiassero rossi a una frequenza di 5 Hz (assenza dello user programma nella memoria flash), non sarà possibile collegarsi al dispositivo mediante la pagina **Configurazione seriale**; in questo caso accedere direttamente alla pagina **Upgrade Firmware**; accertarsi che nella pagina **Configurazione seriale** sia selezionata la corretta porta seriale del personal computer collegata all'unità ROTADRIVE; per maggiori informazioni si veda qui sotto alla sezione "1.7.1 In caso di errore";
5. premere il pulsante **SELEZIONA FILE DA TRASFERIRE** e, mediante la finestra **APRI**, individuare il repository dove è contenuto il file di aggiornamento firmware .BIN fornito da Lika Electronic;



ATTENZIONE

Si badi che, per ogni modello di unità ROTADRIVE con diversa interfaccia bus, è disponibile uno specifico firmware. Assicurarsi di possedere l'aggiornamento appropriato per il proprio modello. Il file .BIN rilasciato da Lika Electronic presenta un nome che deve essere interpretato nel modo seguente.

Per esempio: RD1xA_PB_H1S2.0.BIN, dove:

- RD1xA = modello unità ROTADRIVE;
- PB = interfaccia bus unità ROTADRIVE (MB = Modbus; CB = CANopen; PB = Profibus);
- H1 = versione hardware;
- S2.0 = versione firmware.

6. selezionare il file .BIN fornito da Lika Electronic e confermare la scelta mediante la pressione del pulsante **APRI** in basso nella finestra;
7. in corrispondenza dell'etichetta **FILE SELEZIONATO** compare il percorso completo relativo al file appena confermato;
8. premere ora il pulsante **TRASFERISCI FILE** per procedere al download dell'aggiornamento firmware;
9. al centro della pagina è visualizzato lo stato di avanzamento del processo;
10. durante l'aggiornamento della memoria flash i LED 2 e 3 lampeggiano verdi a una frequenza di 5 Hz;



ATTENZIONE

Non uscire dalla pagina **Upgrade Firmware** durante l'installazione, altrimenti il processo sarà interrotto!

11. al completamento dell'operazione, se l'esito è positivo, viene visualizzato il messaggio **TRASFERIMENTO COMPLETATO CON SUCCESSO**;
12. l'unità ROTADRIVE si trova ora in condizione di emergenza;
13. chiudere il programma SW_RDX_MODBUS_X.EXE e riavviarlo; riconnettersi all'unità ROTADRIVE e, accedendo alla pagina **Modalità operativa**, ripristinare la normale condizione di lavoro.

1.7.1 In caso di errore

Durante il download dell'aggiornamento firmware 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 2 e 3 (pagina 24), come esplicitato qui di seguito.

LED 2 E 3 LAMPEGGIANTI ROSSI A 5 Hz

Durante il trasferimento dei dati alla memoria flash, se si verifica, per esempio, una caduta di tensione e lo spegnimento dell'unità, sarà impossibile ripristinare il firmware che si è proceduto a cancellare dalla memoria flash all'inizio del processo. Nel box in basso nella pagina **Upgrade Firmware** compare il messaggio: **ERRORE DI COMUNICAZIONE. AGGIORNAMENTO FLASH ANNULLATO**. L'assenza dello user program all'accensione procura quindi l'attivazione dei due LED che lampeggiano rossi a una frequenza di 5 Hz. Occorre chiudere il programma SW_RDX_MODBUS_X.EXE e poi riavviarlo. Non è possibile collegarsi al dispositivo mediante la pagina **Configurazione seriale**, ma si deve accedere direttamente alla pagina **Upgrade Firmware**; iniziare quindi la procedura di ripristino a partire dal punto 5. Accertarsi sempre che nella pagina **Configurazione seriale** sia selezionata la corretta porta seriale del personal computer collegata all'unità ROTADRIVE.

LED 2 E 3 FISSI ROSSI

Durante il trasferimento dei dati alla memoria flash, 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 LED 2 e 3 rossi fissi. In questo caso sarà impossibile ripristinare il firmware che si è proceduto a cancellare dalla memoria flash all'inizio del processo. Nel box in basso nella pagina **Upgrade Firmware** compare il messaggio: **ERRORE DI COMUNICAZIONE. AGGIORNAMENTO FLASH ANNULLATO**. E' necessario spegnere e riaccendere l'unità ROTADRIVE. L'assenza dello user program all'accensione procura quindi l'attivazione dei due LED che ora lampeggiano rossi a una frequenza di 5 Hz. Occorre inoltre chiudere il programma SW_RDX_MODBUS_X.EXE e poi riavviarlo. Non è possibile collegarsi al dispositivo mediante la pagina **Configurazione seriale**, ma si deve accedere

direttamente alla pagina **Upgrade Firmware**; iniziare quindi la procedura di ripristino a partire dal punto 5. Accertarsi sempre che nella pagina **Configurazione seriale** sia selezionata la corretta porta seriale del personal computer collegata all'unità ROTADRIVE.

2 Interfaccia Modbus®

Le unità ROTADRIVE 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.

2.1 Principi guida del protocollo Modbus Master / Slave

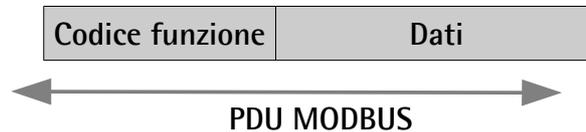
Il protocollo seriale Modbus è un protocollo Master – Slave. Un solo Master può essere connesso contemporaneamente alla rete Modbus; mentre il numero di Slave connessi allo stesso bus seriale può essere compreso tra 1 e 247. Una comunicazione Modbus è sempre iniziata dal Master. I nodi Slave non sono 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:

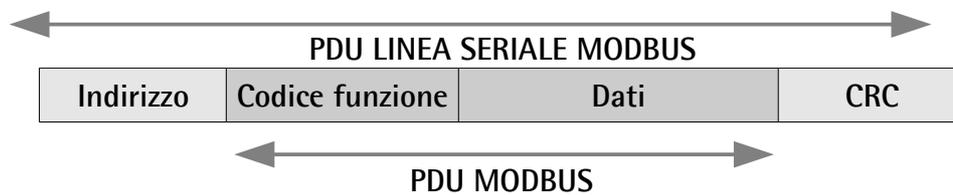
- **modalità UNICAST:** il Master invia la richiesta a un singolo Slave. Dopo aver ricevuto e processato la richiesta, lo Slave invia una risposta al Master. In questa modalità, la transazione Modbus consiste di due messaggi: una richiesta da parte del Master e una risposta da parte dello Slave. Ogni 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".

2.2 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 "4.4.1 Indirizzo nodo: Node ID (Figura 5)" 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 "2.4 Codici funzione" a pagina 114.
- **DATI:** byte dedicati alle informazioni aggiuntive 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 "2.4 Codici funzione" a pagina 114).
- **CRC (Cyclical Redundancy Checking, 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 trasmettente 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 + 0x80, 1 byte;
exception_code = Modbus Exception code, riferirsi alla tabella "Modbus Exception Codes" nel documento "Modbus Application Protocol Specification V1.1b".

2.3 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.

2.3.1 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 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
-------	---	---	---	---	---	---	---	---	---------	------

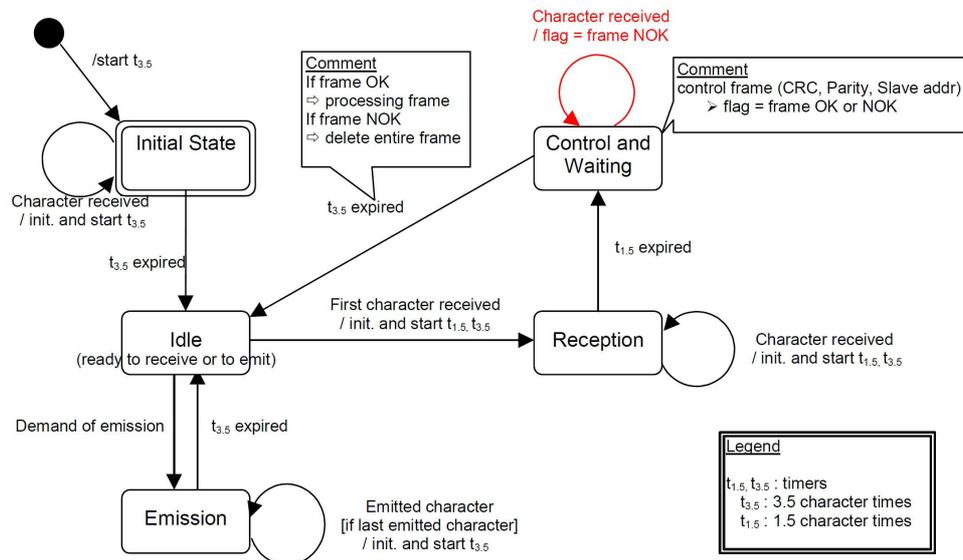
* Con "No parity", il bit di parità è sostituito da un bit di stop.

Come default bisogna impostare il parity bit = 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.

2.4 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 funzione 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.

2.4.1 Codici funzione implementati

I dispositivi Lika RD1xA-Modbus implementano esclusivamente i codici funzione pubblici descritti qui di seguito.

03 Read Holding Registers

FC = 03 (Hex = 0x03) 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 "3.1.1 Parametri Dati macchina" a pagina 124.

Request PDU

Codice funzione	1 byte	0x03
Indirizzo iniziale	2 byte	da 0x0000 a 0xFFFF
Numero di registri	2 byte	da 1 a 125 (0x7D)

Response PDU

Codice funzione	1 byte	0x03
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	0x83 (=0x03 + 0x80)
Codice eccezione	1 byte	01 o 02 o 03 o 04



Esempio di richiesta di lettura dei parametri **Accelerazione [0x06]** (registro 7) e **Decelerazione [0x07]** (registro 8).

Request		Response	
Nome campo	(Hex)	Nome campo	(Hex)
Codice funzione	03	Codice funzione	03
Indirizzo iniziale Hi	00	Numero byte	04
Indirizzo iniziale Lo	06	Valore registro 7 Hi	03
Numero di registri Hi	00	Valore registro 7 Lo	E8
Numero di registri Lo	02	Valore registro 8 Hi	05
		Valore registro 8 Lo	DC

Come si evince dalla tabella, il parametro **Accelerazione [0x06]** (registro 7) ha valore 03 E8 hex, cioè 1000 in notazione decimale; il parametro **Decelerazione [0x07]** (registro 8) ha valore 05 DC hex, cioè 1500 in notazione decimale.

Il frame completo di richiesta di lettura dei parametri **Accelerazione [0x06]** (registro 7) e **Decelerazione [0x07]** (registro 8) allo Slave con indirizzo 1 è il seguente:

Request PDU (in formato esadecimale)

[01][03][00][06][00][02][24][0A]

dove:

[01] = indirizzo dello Slave

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

[00][06] = indirizzo iniziale (parametro **Accelerazione [0x06]**, registro 7)

[00][02] = numero di registri richiesti

[24][0A] = CRC

Il frame completo di invio dei valori nei parametri **Accelerazione [0x06]** (registro 7) e **Decelerazione [0x07]** (registro 8) da parte dello Slave con indirizzo 1 è il seguente:

Response PDU (in formato esadecimale)

[01][03][04][03][E8][05][DC][78][8A]

dove:

[01] = indirizzo dello Slave

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

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

[03][E8] = valore del registro 7 **Accelerazione [0x06]**, 03 E8 hex = 1000 dec

[05][DC] = valore del registro 8 **Decelerazione [0x07]**, 05 DC hex = 1500 dec

[78][8A] = CRC

04 Read Input Register

FC = 04 (Hex = 0x04)

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 "3.1.2 Parametri Input Register" a pagina 139.

Request PDU

Codice funzione	1 byte	0x04
Indirizzo iniziale	2 byte	da 0x0000 a 0xFFFF
Numero di registri	2 byte	da 0x0000 a 0x007D

Response PDU

Codice funzione	1 byte	0x04
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	0x84 (=0x04 + 0x80)
Codice eccezione	1 byte	01 o 02 o 03 o 04



Esempio di richiesta di lettura del parametro **Posizione corrente [0x02-0x03]** (registri 3 e 4).

Request		Response	
Nome campo	(Hex)	Nome campo	(Hex)
Codice funzione	04	Codice funzione	04
Indirizzo iniziale Hi	00	Numero byte	04
Indirizzo iniziale Lo	02	Valore registro 3 Hi	00
Numero di registri Hi	00	Valore registro 3 Lo	00
Numero di registri Lo	02	Valore registro 4 Hi	2F
		Valore registro 4 Lo	F0

Come si evince dalla tabella, il parametro **Posizione corrente [0x02-0x03]** (registri 3 e 4) ha valore 00 00 2F F0 hex, cioè 12272 in notazione decimale.

Il frame completo di richiesta di lettura del parametro **Posizione corrente [0x02-0x03]** (registri 3 e 4) allo Slave con indirizzo 1 è il seguente:

Request PDU (in formato esadecimale)

[01][04][00][02][00][02][D0][0B]

dove:

[01] = indirizzo dello Slave

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

[00][02] = indirizzo iniziale (parametro **Posizione corrente [0x02-0x03]**, registro 3)

[00][02] = numero di registri richiesti

[D0][0B] = CRC

Il frame completo di invio dei valori del parametro **Posizione corrente [0x02-0x03]** (registri 3 e 4) 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 3 **Posizione corrente [0x02-0x03]**, 00 00 hex = 0 dec

[2F][F0] = valore del registro 4 **Posizione corrente [0x02-0x03]**, 2F F0 hex = 12272 dec

[E7][F0] = CRC

06 Write Single Register

FC = 06 (Hex = 0x06)

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 "3.1.1 Parametri Dati macchina" a pagina 124.

Request PDU

Codice funzione	1 byte	0x06
Indirizzo del registro	2 byte	da 0x0000 a 0xFFFF
Valore del registro	2 byte	da 0x0000 a 0xFFFF

Response PDU

Codice funzione	1 byte	0x06
Indirizzo del registro	2 byte	da 0x0000 a 0xFFFF
Valore del registro	2 byte	da 0x0000 a 0xFFFF

Exception Response PDU

Codice errore	1 byte	0x86 (=0x06 + 0x80)
Codice eccezione	1 byte	01 o 02 o 03 o 04



Esempio di scrittura del valore 05 DC hex (= 1500 dec) nel parametro **Accelerazione [0x06]** (registro 7).

Request		Response	
Nome campo	(Hex)	Nome campo	(Hex)
Codice funzione	06	Codice funzione	06
Indirizzo registro Hi	00	Indirizzo registro Hi	00
Indirizzo registro Lo	06	Indirizzo registro Lo	06
Valore del registro Hi	05	Valore del registro Hi	05
Valore del registro Lo	DC	Valore del registro Lo	DC

Come si evince dalla tabella, nel parametro **Accelerazione [0x06]** (registro 7) si imposta il valore 05 DC hex, cioè 1500 in notazione decimale.

Il frame completo di richiesta di scrittura del valore 05 DC hex (= 1500 dec) nel parametro **Accelerazione [0x06]** (registro 7) dello Slave con indirizzo 1 è il seguente:

Request PDU (in formato esadecimale)

[01][06][00][06][05][DC][6B][02]

dove:

[01] = indirizzo dello Slave

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

[00][06] = indirizzo del registro (parametro **Accelerazione [0x06]**, registro 7)

[05][DC] = valore da impostare nel registro

[6B][02] = CRC

Il frame completo di risposta alla richiesta di scrittura nel parametro **Accelerazione [0x06]** (registro 7) da parte dello Slave con indirizzo 1 è il seguente:

Response PDU (in formato esadecimale)

[01][06][00][06][05][DC][6B][02]

dove:

[01] = indirizzo dello Slave

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

[00][06] = indirizzo del registro (parametro **Accelerazione [0x06]**, registro 7)

[05][DC] = valore impostato nel registro

[6B][02] = CRC

16 Write Multiple Registers

FC = 16 (Hex = 0x10)

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 "3.1.1 Parametri Dati macchina" a pagina 124.

Request PDU

Codice funzione	1 byte	0x10
Indirizzo iniziale	2 byte	da 0x0000 a 0xFFFF
Numero di registri	2 byte	da 0x0001 a 0x007B
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	0x10
Indirizzo iniziale	2 byte	da 0x0000 a 0xFFFF
Numero di registri	2 byte	da 1 a 123 (0x7B)

Exception Response PDU

Codice errore	1 byte	0x90 (= 0x10 + 0x80)
Codice eccezione	1 byte	01 o 02 o 03 o 04



Esempio di richiesta di scrittura dei valori 1500 e 1000 nei parametri **Accelerazione [0x06]** (registro 7) e **Decelerazione [0x07]** (registro 8).

Request		Response	
Nome campo	(Hex)	Nome campo	(Hex)
Codice funzione	10	Codice funzione	10
Indirizzo iniziale Hi	00	Indirizzo iniziale Hi	00
Indirizzo iniziale Lo	06	Indirizzo iniziale Lo	06
Numero di registri Hi	00	Numero di registri Hi	00
Numero di registri Lo	02	Numero di registri Lo	02
Numero byte	04		
Valore registro 7 Hi	05		

Valore registro 7 Lo	DC
Valore registro 8 Hi	03
Valore registro 8 Lo	E8

Come si evince dalla tabella, nel parametro **Accelerazione [0x06]** (registro 7) si imposta il valore 05 DC hex, cioè 1500 in notazione decimale; nel parametro **Decelerazione [0x07]** (registro 8) si imposta il valore 03 E8 hex, cioè 1000 in notazione decimale.

Il frame completo di richiesta di scrittura dei valori 1500 e 1000 nei parametri **Accelerazione [0x06]** (registro 7) e **Decelerazione [0x07]** (registro 8) dello Slave con indirizzo 1 è il seguente:

Request PDU (in formato esadecimale)

[01][10][00][06][00][02][04][05][DC][03][E8][B2][0D]

dove:

[01] = indirizzo dello Slave

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

[00][06] = indirizzo iniziale (parametro **Accelerazione [0x06]**, registro 7)

[00][02] = numero di registri richiesti

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

[05][DC] = valore da impostare nel registro 7 **Accelerazione [0x06]**, 05 DC hex = 1500 dec

[03][E8] = valore da impostare nel registro 8 **Decelerazione [0x07]**, 03 E8 hex = 1000 dec

[B2][0D] = CRC

Il frame completo di risposta alla richiesta di impostazione dei valori 1000 e 1500 nei parametri **Accelerazione [0x06]** (registro 7) e **Decelerazione [0x07]** (registro 8) da parte dello Slave con indirizzo 1 è il seguente:

Response PDU (in formato esadecimale)

[01][10][00][06][00][02][A1][C9]

dove:

[01] = indirizzo dello Slave

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

[00][06] = indirizzo iniziale (parametro **Accelerazione [0x06]**, registro 7)

[00][02] = numero di registri su cui si è scritto

[A1][C9] = CRC



NOTA

Per ulteriori esempi si veda anche alla sezione "Esempi di programmazione Modbus®" a pagina 149.



IMPORTANTE

Per motivi di sicurezza, durante il movimento dell'unità ROTADRIVE 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 Watch dog. Il Watch dog 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 -a esempio un comando di jog- il Watch dog interverrebbe comandando lo stop in sicurezza del dispositivo e l'attivazione di un allarme. Per abilitare il Watch dog impostare a "=1" il bit **Abilitazione Watch dog** in **Control Word [0x2A]**. Se impostato a "=0" il Watch dog non è attivo; se impostato a "=1" il Watch dog è attivo. Con Watch dog attivo, se il dispositivo non riceve un messaggio dal Server entro 1 secondo, il sistema forza una condizione di allarme (visualizzazione dell'allarme **Watch dog** al ripristino della comunicazione con la rete Modbus).

3 Parametri di programmazione Modbus®

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

3.1.1 Parametri Dati macchina

I parametri **Dati macchina** sono accessibili sia in lettura che 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 "2.4.1 Codici funzione implementati" a pagina 114.



ATTENZIONE

Utilizzando l'interfaccia seriale RS-232 di servizio Modbus è possibile impostare e salvare i parametri di modo che, anche a seguito di uno spegnimento, i valori impostati siano mantenuti. Questo però vale solamente fino a che sia mantenuta attiva l'interfaccia seriale: infatti nel momento in cui il dispositivo sia collegato alla rete Profibus, saranno caricati i dati memorizzati nel PLC e di conseguenza i dati impostati tramite la seriale Modbus saranno sovrascritti.

In Profibus utilizzando i moduli **Lika RD1xA-T12**, **Lika RD1xA-T24**, **Lika RD1xA-T48** e **Lika RD1xA-T92** il valore di tutti i parametri è caricato all'accensione prendendo i dati memorizzati sul PLC a partire dal file GSD. Ogni modifica locale tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** è perciò temporanea: allo spegnimento del dispositivo ogni valore impostato viene perso (a eccezione del preset, unico parametro non

compreso nel file GSD; e, per tutti i parametri, nel caso in cui si esegua precedentemente un'impostazione del preset, si veda alla pagina 83) e alla successiva riaccensione viene caricato il valore presente nel PLC (per cui anche i parametri salvati a seguito di un preset sono poi comunque sovrascritti).

Questo dispositivo prevede altresì la possibilità di installazione del modulo **Lika RD1xA-no param** (disponibile a partire dalla versione H3S3, file GSD V5). Con questo modulo è possibile bypassare il trasferimento dei parametri dal PLC all'accensione (i parametri infatti vengono letti dalla memoria flash) e memorizzare in flash (tramite il bit 9 **Salva parametri** in **Control Word (Byte 0 e 1)**) ogni modifica locale effettuata mediante **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)**. Con il modulo **Lika RD1xA-no param** NON è possibile leggere e modificare i parametri tramite la pagina **Parametrizza** della finestra **Proprietà slave DP** di STEP7 (si veda il paragrafo "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). La modifica dei parametri è quindi possibile tramite **Numero parametro (Byte 8)** e **Valore parametro (Byte 9 ... 12)** oppure tramite l'interfaccia Modbus. Il modulo **Lika RD1xA-no param** è indipendente dal rapporto di riduzione e può essere installato quale che sia quello del dispositivo che si va a programmare. Naturalmente i parametri da impostare dovranno tenere conto delle caratteristiche meccaniche ed elettriche del dispositivo.

Utilizzando STEP7 di Siemens invece è possibile memorizzare in maniera permanente ogni modifica al valore di un parametro (eccetto che nel modulo -no param). Per fare questo modificare il valore dei parametri nella pagina **Proprietà slave DP** di STEP7 (si veda alla sezione "1.1.3 Configurazione dei parametri "Dati macchina"" a pagina 45). Si badi che, al contrario, le modifiche realizzate mediante la tabella delle variabili (si veda alla sezione "1.3.1 Impostazione parametro 28 Valore di preset" a pagina 49) sono temporanee.

Distanza_giro [0x00]

[Registro 1, Unsigned16, rw]

Questo parametro definisce il numero di impulsi per ogni giro completo dell'albero. Si rivela utile per relazionare un giro dell'asse con una grandezza lineare; per esempio, se il dispositivo è montato su una vite senza fine con passo 5 mm, impostando **Distanza_giro [0x00] = 500** si ottiene che a ogni giro dell'asse il sistema trasla di 5 mm con una risoluzione al centesimo di millimetro. Default = 1024 (min. = 1, max. = 1024)



ATTENZIONE

Dopo la modifica di questo parametro è necessario reimpostare i parametri **Velocità Jog [0x0C]**, **Velocità di lavoro [0x0D]** e **Valore di preset [0x16-0x17]**. Per maggiori informazioni riferirsi a pagina 36 e ai rispettivi parametri. Si badi inoltre che i parametri di seguito elencati sono tutti espressi in relazione al parametro **Distanza_giro [0x00]**; di conseguenza la modifica del valore nel parametro **Distanza_giro [0x00]** comporta necessariamente una ridefinizione dei valori da essi espressi. I parametri sono: **Tolleranza di posizione [0x01]**, **Max errore di inseguimento [0x03]**, **Accelerazione [0x06]**, **Decelerazione [0x07]**, **Delta spazio positivo [0x08-0x09]**, **Delta spazio negativo [0x0A-0x0B]**, **Target position [0x2B-0x2C]**, **Posizione corrente [0x02-0x03]**, **Velocità corrente [0x04]** e **Errore di inseguimento [0x05-0x06]**. Si veda per esempio la relazione che intercorre tra **Distanza_giro [0x00]** e i valori di velocità, illustrata alla pagina 130.



NOTA

Se **Distanza_giro [0x00]** non è una potenza di due (2, ..., 512, 1024), durante il controllo di posizione potrebbe verificarsi un errore di posizionamento pari a un impulso.

Tolleranza di posizione [0x01]

[Registro 2, Unsigned16, rw]

Questo parametro definisce la finestra di tolleranza da applicare al valore di **Target position [0x2B-0x2C]**. Se l'asse è all'interno di questa finestra per il tempo impostato nel parametro **Tempo asse in tolleranza [0x02]**, allora è segnalata la condizione nel bit di stato **Asse in posizione**. Il parametro è espresso in impulsi.

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

Tempo asse in tolleranza [0x02]

[Registro 3, Unsigned16, rw]

Rappresenta il tempo di assestamento dell'asse all'interno della finestra di tolleranza definita in **Tolleranza di posizione [0x01]** dopo il quale viene dichiarata la condizione nel bit di stato **Asse in posizione**. Il parametro è espresso in millisecondi.

Default = 0 (min. = 0, max. = 10000)

Max errore di inseguimento [0x03]

[Registro 4, Unsigned16, rw]

Questo parametro definisce la differenza massima ammissibile tra la posizione reale del dispositivo e quella teorica. Se il dispositivo rileva un valore superiore a quello impostato in questo parametro viene segnalato l'allarme **Errore di inseguimento** e il dispositivo arresta il proprio movimento. Il parametro è espresso in impulsi.

Default = 1024 (min. = 0, max. = 65535)

Kp anello di posizione [0x04]

[Registro 5, Unsigned16, rw]

Questo parametro contiene il guadagno proporzionale usato dal controllore PI relativo all'anello di posizione. Tale valore è già ottimizzato da Lika Electronic in relazione alle caratteristiche del dispositivo.

Default = 400 (min. = 0, max. = 1000)

Ki anello di posizione [0x05]

[Registro 6, Unsigned16, rw]

Questo parametro contiene il guadagno integrale usato dal controllore PI relativo all'anello di posizione. Tale valore è già ottimizzato da Lika Electronic in relazione alle caratteristiche del dispositivo.

Default = 100 (min. = 0, max. = 1000)

Accelerazione [0x06]

[Registro 7, Unsigned16, rw]

Questo parametro definisce il valore di accelerazione usato dal dispositivo. Il parametro è espresso in [impulsi/sec²].

Default = 5000 per RD1xA-...-T12-... (min. = 100, max. = 10000)

Default = 2500 per RD1xA-...-T24-... (min. = 100, max. = 10000)

Default = 1000 per RD1xA-...-T48-... (min. = 100, max. = 10000)

Default = 500 per RD1xA-...-T92-... (min. = 100, max. = 10000)

Decelerazione [0x07]

[Registro 8, Unsigned16, rw]

Questo parametro definisce il valore di decelerazione usato dal dispositivo. Il parametro è espresso in [impulsi/sec²].

Default = 5000 per RD1xA-...-T12-... (min. = 100, max. = 10000)

Default = 2500 per RD1xA-...-T24-... (min. = 100, max. = 10000)

Default = 1000 per RD1xA-...-T48-... (min. = 100, max. = 10000)

Default = 500 per RD1xA-...-T92-... (min. = 100, max. = 10000)

Delta spazio positivo [0x08-0x09]

[Registri 9-10, Unsigned32, rw]

E' il valore utilizzato per calcolare il massimo spostamento in avanti (positivo) rispetto al Preset. Se si raggiunge il massimo spostamento in avanti viene attivata la segnalazione sul bit di stato **Finecorsa SW +**. Il parametro è espresso in impulsi encoder.

Finecorsa SW + = Valore di preset [0x16-0x17] + Delta spazio positivo [0x08-0x09].



ATTENZIONE

Si badi che il valore massimo di volta in volta ammesso in questo parametro tiene conto dello scaling impostato.



ESEMPIO

Con **Distanza_giro [0x00]** = 1024 e **Valore di preset [0x16-0x17]** = 0, il valore massimo ammesso in **Delta spazio positivo [0x08-0x09]** sarà:

$(1024 \text{ imp./giro} * 511 \text{ giri}) - 1 = 523263$

Con **Distanza_giro [0x00]** = 256 e **Valore di preset [0x16-0x17]** = 0, il valore massimo ammesso in **Delta spazio positivo [0x08-0x09]** sarà:

$(256 \text{ imp./giro} * 511 \text{ giri}) - 1 = 130815$

Si vedano anche gli ulteriori esempi al paragrafo "6.4 Distanza_giro, Velocità Jog, Velocità di lavoro, Valore di preset e valori di finecorsa" a pagina 36.



ATTENZIONE

I valori di finecorsa devono essere verificati ogniqualvolta vengono modificati i parametri **Distanza_giro [0x00]** e **Valore di preset [0x16-0x17]**. Dopo ogni modifica del parametro **Distanza_giro [0x00]** si deve reimpostare anche **Valore di preset [0x16-0x17]** in modo da definire lo zero asse in quanto il sistema di riferimento è variato. Dopo la modifica del parametro **Valore di preset [0x16-0x17]** non occorre invece reimpostare il valore dei finecorsa in quanto la funzione di Preset provvede a ricalcolarli automaticamente reinizializzando i limiti positivo e negativo sulla base dei valori di **Delta spazio positivo [0x08-0x09]** e **Delta spazio negativo [0x0A-0x0B]** impostati. Per una descrizione dettagliata si veda a pagina 36.

Delta spazio negativo [0x0A-0x0B]

[Registri 11-12, Unsigned32, rw]

E' il valore utilizzato per calcolare il massimo spostamento all'indietro (negativo) rispetto al Preset. Se si raggiunge il massimo spostamento all'indietro viene attivata la segnalazione sul bit di stato **Finecorsa SW -**. Il parametro è espresso in impulsi encoder.

Finecorsa SW - = Valore di preset [0x16-0x17] - Delta spazio negativo [0x0A-0x0B].

Default = 523263 (min. = 0, max. = 523263)



ATTENZIONE

Si badi che il valore massimo di volta in volta ammesso in questo parametro tiene conto dello scaling impostato.



ESEMPIO

Con **Distanza_giro [0x00]** = 1024 e **Valore di preset [0x16-0x17]** = 0, il valore massimo ammesso in **Delta spazio negativo [0x0A-0x0B]** sarà:
 $(1024 \text{ imp./giro} * 511 \text{ giri}) - 1 = 523263$

Con **Distanza_giro [0x00]** = 256 e **Valore di preset [0x16-0x17]** = 0, il valore massimo ammesso in **Delta spazio negativo [0x0A-0x0B]** sarà:
 $(256 \text{ imp./giro} * 511 \text{ giri}) - 1 = 130815$

Si vedano anche gli ulteriori esempi al paragrafo "6.4 Distanza_giro, Velocità Jog, Velocità di lavoro, Valore di preset e valori di finecorsa" a pagina 36.



ATTENZIONE

I valori di finecorsa devono essere verificati ogniqualvolta vengono modificati i parametri **Distanza_giro [0x00]** e **Valore di preset [0x16-0x17]**. Dopo ogni modifica del parametro **Distanza_giro [0x00]** si deve reimpostare anche **Valore di preset [0x16-0x17]** in modo da definire lo zero asse in quanto il sistema di riferimento è variato. Dopo la modifica del parametro **Valore di preset [0x16-0x17]** non occorre invece reimpostare il valore dei finecorsa in quanto la funzione di Preset provvede a ricalcolarli automaticamente reinizializzando i limiti positivo e negativo sulla base dei valori di **Delta spazio positivo [0x08-0x09]** e **Delta spazio negativo [0x0A-0x0B]** impostati. Per una descrizione dettagliata si veda a pagina 36.

Velocità Jog [0x0C]

[Registro 13, Unsigned16, rw]

Questo parametro definisce la velocità massima del dispositivo nell'utilizzo delle funzioni **Jog +** e **Jog -**. Il parametro è espresso in impulsi/secondo.

Default = 4266 per RD1xA-...-T12-... (min. = 1, max. = 4266)

Default = 2133 per RD1xA-...-T24-... (min. = 1, max. = 2133)

Default = 1066 per RD1xA-...-T48-... (min. = 1, max. = 1066)

Default = 556 per RD1xA-...-T92-... (min. = 1, max. = 556)

Velocità di lavoro [0x0D]

[Registro 14, Unsigned16, rw]

Questo parametro definisce la velocità massima del dispositivo utilizzato in modalità automatica (movimenti controllati mediante il bit di comando **Start** per il raggiungimento della quota impostata in **Target position [0x2B-0x2C]**). Il parametro è espresso in impulsi/secondo.

Default = 4266 per RD1xA-...-T12-... (min. = 1, max. = 4266)

Default = 2133 per RD1xA-...-T24-... (min. = 1, max. = 2133)

Default = 1066 per RD1xA-...-T48-... (min. = 1, max. = 1066)

Default = 556 per RD1xA-...-T92-... (min. = 1, max. = 556)



ATTENZIONE

A ogni modifica del parametro **Distanza_giro [0x00]** si devono poi reimpostare **Velocità Jog [0x0C]** e **Velocità di lavoro [0x0D]** in quanto le velocità sono espresse in impulsi al secondo. Nel calcolo delle velocità si deve sempre rispettare la seguente relazione:

$$\frac{vel_{min} * Distanza / giro}{1024} \leq Velocità \leq \frac{vel_{max} * Distanza / giro}{1024}$$

Per una spiegazione dettagliata si veda a pagina 36.

Durata corrente di stacco [0x0E]

[Registro 15, Unsigned16, rw]

Questo parametro definisce il tempo massimo per il quale il motore è controllato con la corrente di stacco durante la partenza (si veda al parametro **Corrente di stacco [0x13]**). Il parametro è espresso in millisecondi.

Default = 2000 (min. = 0, max. = 3000)

Direzione conteggio [0x0F]

[Registro 16, Unsigned16, rw]

Direzione di rotazione del motore per ottenere un incremento positivo della posizione. La direzione di rotazione è stabilita guardando il dispositivo dall'estremità dell'asse.

0 = rotazione oraria (default)

1 = rotazione antioraria



ATTENZIONE

La modifica di questo parametro influenza la posizione calcolata dal controllore. Si deve quindi reimpostare il parametro **Valore di preset [0x16-0x17]** e verificare i valori nei parametri **Delta spazio positivo [0x08-0x09]** e **Delta spazio negativo [0x0A-0x0B]**.

Kp anello di corrente [0x10]

[Registro 17, Unsigned16, rw]

Questo parametro definisce il guadagno proporzionale usato dal controllore PI relativo all'anello di corrente. Tale valore è già ottimizzato da Lika Electronic in base alle caratteristiche del dispositivo.

Default = 200 (min. = 0, max. = 1000)

Ki anello di corrente [0x11]

[Registro 18, Unsigned16, rw]

Questo parametro definisce il guadagno integrale usato dal controllore PI relativo all'anello di corrente. Tale valore è già ottimizzato da Lika Electronic in base alle caratteristiche del dispositivo.

Default = 30 (min. = 0, max. = 1000)

Corrente massima [0x12]

[Registro 19, Unsigned16, rw]

Questo parametro definisce la corrente massima erogata dall'elettronica di potenza per il controllo del motore. Il parametro è espresso in mA (milliampere). Il valore in questo parametro non può essere superiore al valore impostato in

Corrente di stacco [0x13].

Default = 2000 (min. = 10, max. = 2000)

Corrente di stacco [0x13]

[Registro 20, Unsigned16, rw]

Questo parametro definisce la corrente massima erogata solo alla partenza e per un tempo massimo definito alla variabile **Durata corrente di stacco [0x0E]**. Il parametro è espresso in mA (milliampere).

Default = 4000 (min. = 10, max. = 4000)

Valore di offset [0x14-0x15]

[Registri 21-22, Integer32, ro]

Questo parametro definisce la differenza tra la posizione trasmessa dal dispositivo e la posizione reale: posizione reale – preset. Il valore è espresso in impulsi.

Default = 0

Valore di preset [0x16-0x17]

[Registri 23-24, Integer32, rw]

Usare questo parametro per assegnare un valore di Preset. La funzione di Preset è utilizzata per assegnare un determinato valore a una posizione fisica dell'asse. La posizione fisica prescelta avrà perciò il valore assegnato in questo parametro e tutte le altre posizioni assumeranno un valore conseguente. Il valore di preset sarà assegnato alla posizione dell'asse al momento dell'impostazione. L'impostazione del preset è eseguita in corrispondenza del fronte di salita del bit 11 **Esegui preset conteggio** in **Control Word [0x2A]**, cioè alla variazione del bit dal livello logico basso (0) al livello logico alto (1). All'atto della impostazione del preset, il sistema esegue contestualmente anche un salvataggio automatico di tutti i parametri nella memoria interna. Per ulteriori informazioni riferirsi a pagina 136.

Default = 0 (min. = -1048576, max. = +1048576)



ATTENZIONE

Il parametro **Valore di preset [0x16-0x17]** deve essere reimpostato ogniqualvolta viene modificato il parametro **Distanza_giro [0x00]**. Dopo la modifica del parametro **Valore di preset [0x16-0x17]** non occorre invece reimpostare il valore dei finecorsa in quanto la funzione di Preset provvede a ricalcolarli automaticamente reinizializzando i limiti positivo e negativo sulla base dei valori di **Delta spazio positivo [0x08-0x09]** e **Delta spazio negativo [0x0A-0x0B]** impostati. Per una descrizione dettagliata si veda a pagina 36.

Rapporto di riduzione [0x18]

[Registro 25, Unsigned16, ro]

E' il rapporto di riduzione degli ingranaggi interni interposti tra il motore e l'asse del dispositivo. E' un valore di sola lettura.

Default = 12 per RD1xA-...-T12-...

Default = 24 per RD1xA-...-T24-...

Default = 48 per RD1xA-...-T48-...

Default = 92 per RD1xA-...-T92-...

Ampiezza passo jog [0x19]

[Registro 26, Unsigned16, rw]

Se è abilitato il controllo del jog a passo (bit 4 **Jog incrementale** in **Control Word [0x2A]** = 1), l'attivazione dei bit **Jog +** e **Jog -** procura in corrispondenza del fronte di salita l'esecuzione di un singolo passo in direzione positiva o negativa la cui ampiezza, espressa in impulsi, è impostata in questo parametro; quindi lo slave si arresta in attesa di un nuovo comando.

Default = 100 (min. = 1, max. = 10000).

Registro comandi Extra [0x29]

[Registro 42, Unsigned16, rw]

Struttura byte del registro **Registro comandi Extra [0x29]**:

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

Byte 0

Lettura assoluta

bit 0 Questa funzione è riservata all'utilizzo dei soli tecnici di Lika Electronic.

Controllo da PC

bit 1 Se impostato "=0" il dispositivo comunica in rete Profibus; se impostato a "=1" il dispositivo comunica in rete Modbus tramite l'interfaccia seriale di servizio RS-232 (si veda alla sezione "Interfaccia Modbus®" a pagina 88).

bit 2 ... 7 Non utilizzati.

Byte 1 Non utilizzato.

Control Word [0x2A]

[Registro 43, Unsigned16, rw]

Contiene i comandi da inviare in tempo reale al dispositivo per controllarlo.

Struttura byte del registro **Control Word [0x2A]**:

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

Byte 0

Jog +

bit 0

Se il bit 4 **Jog incrementale** = 0, lo Slave si muove in direzione positiva per tutto il tempo in cui **Jog +** = 1; se invece il bit 4 **Jog incrementale** = 1 (abilitazione jog a passo), lo slave esegue un singolo passo in direzione positiva in corrispondenza del fronte di salita di **Jog +** la cui ampiezza, espressa in impulsi, è impostata al parametro **Ampiezza passo jog [0x19]**; poi si arresta in attesa di un nuovo comando. La velocità, l'accelerazione e la decelerazione sono definiti dai dati macchina **Velocità Jog [0x0C]**, **Accelerazione [0x06]** e **Decelerazione [0x07]**. Per una descrizione più dettagliata del controllo jog si veda a pagina 34.

Jog -

bit 1

Se il bit 4 **Jog incrementale** = 0, lo Slave si muove in direzione negativa per tutto il tempo in cui **Jog -** = 1; se invece il bit 4 **Jog incrementale** = 1 (abilitazione jog a passo), quando **Jog -** = 1 lo slave esegue un singolo passo in direzione negativa in corrispondenza del fronte di salita di **Jog -** la cui ampiezza, espressa in impulsi, è impostata al parametro **Ampiezza passo jog [0x19]**; poi si arresta in attesa di un nuovo comando. La velocità, l'accelerazione e la decelerazione sono definiti dai dati macchina **Velocità Jog [0x0C]**, **Accelerazione [0x06]** e **Decelerazione [0x07]**. Per una descrizione più dettagliata del controllo jog si veda a pagina 34.

Stop

bit 2

Se impostato a "1" lo Slave è libero di eseguire i comandi di movimento ricevuti. Se durante il movimento questo bit diventa "0" allora lo Slave si ferma seguendo la decelerazione prevista nel dato macchina **Decelerazione**

[0x07]. Per un arresto immediato del movimento, utilizzare il bit 7 **Emergenza**.

Reset allarmi

bit 3



Questo bit è normalmente impostato a "0". Lo stato normale del dispositivo è ristabilito nel cambio di stato da "0" a "1" di questo bit. Questo comando toglie lo Slave dalla condizione di allarme solo se non sono più presenti le condizioni che hanno causato l'errore.

Si badi che, se l'allarme è relativo a dati macchina non validi (si veda **Dati macchina non validi** e **Elenco DM errati [0x09-0x0A]**), si può tornare allo stato normale solo impostando dati macchina validi. L'allarme **Errore memoria flash** non è ripristinabile.

Jog incrementale

bit 4



Se il bit 4 "= 0", l'attivazione dei bit **Jog +** e **Jog -** procura il movimento manuale dello slave per tutto il tempo in cui **Jog + / Jog - = 1**. Impostando questo bit a 1 si abilita la funzione di jog a passo. L'attivazione dei bit **Jog +** e **Jog -** procura in corrispondenza del fronte di salita l'esecuzione di un singolo passo in direzione positiva o negativa la cui ampiezza, espressa in impulsi, è impostata al parametro **Ampiezza passo jog [0x19]**; quindi lo slave si arresta in attesa di un nuovo comando.

Si badi che nell'utilizzo dei pulsanti manuali di jog (si veda "4.4.3 Pulsanti JOG + e JOG - (Figura 5)" a pagina 29) la funzione di jog incrementale è disabilitata; non è cioè possibile eseguire passi jog utilizzando i pulsanti manuali.

bit 5

Non utilizzato.

Start

bit 6

Se impostato a "=1" il dispositivo si muove allo scopo di raggiungere la posizione di target specificata (vedi **Target position [0x2B-0x2C]** a pagina 137). Per la descrizione del controllo di posizione si veda a pagina 33.

Emergenza

bit 7

Questo bit deve essere normalmente alto ("=1") altrimenti il dispositivo bloccherà istantaneamente ogni suo movimento. Per procurare un arresto non immediato, che utilizzi la decelerazione programmata, usare il bit 2 **Stop**.

Byte 1

Abilitazione Watch dog

bit 8 Se impostato a "0" il Watch dog non è attivo; se impostato a "1" il Watch dog è attivo. Con Watch dog attivo, se il dispositivo non riceve un messaggio dal Server entro 1 secondo, il sistema forza una condizione di allarme (visualizzazione dell'allarme **Watch dog** al ripristino della comunicazione con la rete Modbus). Il Watch dog 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 -a esempio un comando di jog- il Watch dog interverrebbe comandando lo stop in sicurezza del dispositivo e 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 valori 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 a vuoto e sicuro del dispositivo) 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 40 è disponibile l'elenco dei dati macchina e il rispettivo valore di default preimpostato da Lika Electronic. All'atto del caricamento dei valori di default, il sistema esegue contestualmente anche un salvataggio automatico di tali valori nella memoria flash.

Esegui preset conteggio

bit 11 Il conteggio assume il valore impostato nella variabile **Valore di preset [0x16-0x17]**. 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). All'atto della impostazione del preset, il sistema esegue contestualmente anche un salvataggio automatico di tutti i parametri nella memoria interna.

Asse in coppia

bit 12 Mantenimento asse in coppia a fine posizionamento. Funzione disponibile solo per la versione RD1A (versione

senza freno), nella versione RD12A (versione con freno) il bit 12 non è gestito.

Se impostato "=0", con asse in posizione il PWM è disattivato.

Se impostato a "=1", con asse in posizione il PWM rimane attivo.

OUT 1

bit 13

Attiva / disattiva l'uscita digitale 1 del dispositivo. Il significato delle uscite è esplicitato nel capitolo "Parametri di programmazione Modbus®" a pagina 124.

OUT 1 = 0 uscita 1 bassa (non attiva)

OUT 1 = 1 uscita 1 alta (attiva)

Sblocco freno

bit 14

Disponibile solo per la versione RD12A (versione con freno), nella versione RD1A (versione senza freno) il bit 14 non è gestito. Il modello RD12A è provvisto di un freno che, all'arresto del dispositivo, si attiva inibendo la possibilità di qualsiasi movimento dell'asse. Se si imposta questo bit "=1", il freno è disattivato; se si imposta "=0", il freno è gestito in modo automatico dal sistema.



Si badi che lo sblocco del freno è ammesso solo in assenza di allarmi.

bit 15

Non utilizzato.

Target position [0x2B-0x2C]

[Registri 44-45, Integer32, rw]

Questo parametro definisce la posizione di arrivo programmata. Il dispositivo si muove allo scopo di raggiungere questa posizione quando viene inviato il comando **Start**, i bit di **Stop** e **Emergenza** sono "=1" e il dispositivo non è in stato di allarme.



Funzione override di posizione

Durante il posizionamento è possibile modificare la posizione target; per fare ciò è sufficiente inviare nuovamente il comando **Start** con il nuovo valore in **Target position [0x2B-0x2C]**.



NOTA

Non è possibile abilitare le funzioni **Jog +**, **Jog -** e **Start** contemporaneamente. Per esempio: se viene inviato allo Slave il comando **Jog +** durante il movimento verso la posizione target, il comando di jog sarà ignorato; se si inviano i comandi **Jog +** e **Jog -** contemporaneamente il dispositivo non si muove o, se è già in movimento, arresta la sua corsa.

Solo con watch dog attivo (**Abilitazione Watch dog** in **Control Word [0x2A]** impostato a "=1"), se durante un movimento la comunicazione nella rete Modbus dovesse interrompersi (per esempio a causa della rottura del cavo), il dispositivo arresta ogni suo movimento e attiva il bit di allarme **Watch dog** (visualizzazione dell'allarme al ripristino della comunicazione con la rete Modbus).



NOTA

Per salvare i parametri modificati eseguire **Salva parametri**.
Nel caso di spegnimento del dispositivo i dati non salvati andranno persi!

3.1.2 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 "2.4.1 Codici funzione implementati" a pagina 114.

Registro allarmi [0x00]

[Registro 1, Unsigned16, ro]

Informa sugli allarmi presenti nel dispositivo.

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. Controllare la lista dei parametri errati in **Elenco DM errati [0x09-0x0A]**.

Errore memoria flash

bit 1 Errore interno non ripristinabile.

bit 2 Non utilizzato.

Errore di inseguimento

bit 3 La differenza tra la posizione reale e quella teorica è superiore al valore del parametro **Max errore di inseguimento [0x03]**; si consiglia di ridurre la velocità di lavoro.

Asse non sincronizzato

bit 4 Errore interno non ripristinabile.

Target non valido

bit 5 Posizione comandata oltre i finecorsa.

Emergenza

bit 6 Il bit 7 **Emergenza** in **Control Word [0x2A]** è stato forzato a 0; oppure sono attivi allarmi nel dispositivo.

Sovracorrente

bit 7 Il valore della corrente di alimentazione è superiore al massimo consentito.

Byte 1

Sovratemperatura

bit 8 Superamento della temperatura interna accettabile rilevata per mezzo di una sonda (si veda il parametro **Temperatura [0x08]**).

bit 9 Non utilizzato.

Sottotensione

bit 10 Il valore della tensione di alimentazione è inferiore al minimo consentito.

Watch dog

bit 11 Con Watch dog attivo (**Abilitazione Watch dog** in **Control Word [0x2A]** 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 **Watch dog**). L'allarme è segnalato al ripristino della comunicazione con la rete Modbus. Il Watch dog 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 -a esempio un comando di jog- il Watch dog interverrebbe comandando lo stop in sicurezza del dispositivo e l'attivazione dell'allarme.

bit 12 ... 15 Non utilizzati.

Per ripristinare una condizione di errore utilizzare il bit **Reset allarmi**. Questo bit è normalmente impostato a "0". Lo stato normale del dispositivo è ristabilito nel cambio di stato da "0" a "1" di questo bit. Questo comando toglie lo Slave dalla condizione di allarme solo se non sono più presenti le condizioni che hanno causato l'errore.



Si badi che, se l'allarme è relativo a dati macchina non validi (si veda **Dati macchina non validi** e **Elenco DM errati [0x09-0x0A]**), si può tornare allo

stato normale solo impostando dati macchina validi. L'allarme **Errore memoria flash** non è ripristinabile.

Status word [0x01]

[Registro 2, Unsigned16, ro]

Questo registro contiene le informazioni relative allo stato del dispositivo.

Struttura byte del registro **Status word [0x01]**:

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

Byte 0

Asse in posizione

bit 0

Se "=1" il dispositivo ha raggiunto la posizione programmata per il tempo definito al parametro **Tempo asse in tolleranza [0x02]**. Rimane attivo fino a quando l'errore di posizione è inferiore al valore impostato in **Tolleranza di posizione [0x01]**.

bit 1

Non utilizzato.

Asse abilitato

bit 2

Riporta lo stato di abilitazione del motore. Il bit è "=1" quando il motore è abilitato, cioè il PWM è attivo e l'asse in controllo ad anello chiuso (per esempio, durante un posizionamento o un jog). E' "=0" quando il motore è disabilitato, vale a dire quando l'azionamento che controlla il motore viene spento al termine di un posizionamento o di un jog o a seguito di un allarme.

Finecorsa SW +

bit 3

Se "=1" il dispositivo ha raggiunto la posizione programmata come finecorsa positiva. Si veda il dato macchina **Delta spazio positivo [0x08-0x09]**.

Finecorsa SW -

bit 4

Se "=1" il dispositivo ha raggiunto la posizione programmata come finecorsa negativa. Si veda il dato macchina **Delta spazio negativo [0x0A-0x0B]**.

Allarme

bit 5

Se "=1" si è verificato un allarme, si vedano i dettagli in **Registro allarmi [0x00]** a pagina 139.

Asse in movimento

bit 6

Se "=0" il dispositivo è fermo.

Se "=1" il dispositivo è in movimento.

Comando in corso

bit 7

Se "=0" il controllore non sta eseguendo nessun comando.

Se "=1" il controllore sta eseguendo un comando.

Byte 1

Target raggiunto

bit 8

Se "=1" il dispositivo ha raggiunto la posizione programmata in **Target position [0x2B-0x2C]**. Rimane attivo fino ai successivi comandi **Target position [0x2B-0x2C]** o **Reset allarmi**.

Pulsante 1 Jog +

bit 9

L'unità RD1xA prevede l'installazione di tre pulsanti alloggiati all'interno del corpo del dispositivo e accessibili mediante la rimozione di un coperchio PG. Quando si preme il pulsante 1 JOG + il bit 9 è forzato alto "=1"; se il pulsante 1 non è attivo il bit è "=0". Per maggiori informazioni si veda la sezione "4.4 Selettori e pulsanti (Figura 5)" a pagina 27.

Pulsante 2 Jog -

bit 10

L'unità RD1xA prevede l'installazione di tre pulsanti alloggiati all'interno del corpo del dispositivo e accessibili mediante la rimozione di un coperchio PG. Quando si preme il pulsante 2 JOG - il bit 10 è forzato alto "=1"; se il pulsante 2 non è attivo il bit è "=0". Per maggiori informazioni si veda la sezione "4.4 Selettori e pulsanti (Figura 5)" a pagina 27.

Pulsante 3 Preset

bit 11

L'unità RD1xA prevede l'installazione di tre pulsanti alloggiati all'interno del corpo del dispositivo e accessibili mediante la rimozione di un coperchio PG. Quando si preme il pulsante 3 PRESET il bit 11 è forzato alto "=1"; se il pulsante 3 non è attivo il bit è "=0". Per maggiori informazioni si veda la sezione "4.4 Selettori e pulsanti (Figura 5)" a pagina 27.

Saturazione DAC

bit 12

La corrente erogata dall'elettronica di potenza per il controllo del motore ha raggiunto il livello massimo e non può essere ulteriormente aumentata.

IN 1

bit 13

Stato dell'ingresso digitale 1 del dispositivo. Il significato degli ingressi è esplicitato nel capitolo "Parametri di programmazione Modbus®" a pagina 124.

IN 1 = 0 ingresso 1 basso (non attivo)

IN 1 = 1 ingresso 1 alto (attivo)

IN 2

bit 14

Stato dell'ingresso digitale 2 del dispositivo. Il significato degli ingressi è esplicitato nel capitolo "Parametri di programmazione Modbus®" a pagina 124.

IN 2 = 0 ingresso 2 basso (non attivo)

IN 2 = 1 ingresso 2 alto (attivo)

IN 3

bit 15

Stato dell'ingresso digitale 3 del dispositivo. Il significato degli ingressi è esplicitato nel capitolo "Parametri di programmazione Modbus®" a pagina 124.

IN 3 = 0 ingresso 3 basso (non attivo)

IN 3 = 1 ingresso 3 alto (attivo)

Posizione corrente [0x02-0x03]

[Registri 3-4, Integer32, ro]

Posizione del dispositivo al momento dell'invio della richiesta. Il valore è espresso in impulsi.

Velocità corrente [0x04]

[Registro 5, Integer16, ro]

Velocità del dispositivo espressa in impulsi/secondo [imp/s], aggiornata ogni secondo.

Errore di inseguimento [0x05-0x06]

[Registri 6-7, Integer32, ro]

Questa variabile contiene la differenza tra la posizione richiesta e la posizione attuale istante per istante. Se questo valore supera il dato macchina **Max errore di inseguimento [0x03]** il dispositivo genera l'allarme **Errore di inseguimento** e il dispositivo interrompe il proprio movimento. Il valore è espresso in impulsi.

Corrente attuale [0x07]

[Registro 8, Integer16, ro]

Questa variabile rappresenta il valore della corrente assorbita dal motore. Il parametro è espresso in mA (milliampere).

Temperatura [0x08]

[Registro 9, Integer16, ro]

Questo oggetto contiene il valore della temperatura interna al dispositivo rilevata per mezzo di una sonda. Il parametro è espresso in °C (gradi Celsius). La temperatura minima rilevabile è di -20°C.

Elenco DM errati [0x09-0x0A]

[Registri 10-11, 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 quali parametri contengono valori errati, secondo la lista riportata nella seguente tabella.

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

Bit	Nome parametro
1	Distanza_giro [0x00]
2	Tolleranza di posizione [0x01]
3	Tempo asse in tolleranza [0x02]
4	Max errore di inseguimento [0x03]
5	Kp anello di posizione [0x04]
6	Ki anello di posizione [0x05]
7	Accelerazione [0x06]
8	Decelerazione [0x07]
9	Delta spazio positivo [0x08-0x09]

10	Delta spazio negativo [0x0A-0x0B]
11	Velocità Jog [0x0C]
12	Velocità di lavoro [0x0D]
13	Durata corrente di stacco [0x0E]
14	Direzione conteggio [0x0F]
15	Kp anello di corrente [0x10]
16	Ki anello di corrente [0x11]
17	Corrente massima [0x12]
18	Corrente di stacco [0x13]
19	Rapporto di riduzione [0x18]
20	Ampiezza passo jog [0x19]
26	Valore di preset [0x16-0x17]

I2t [0x0B]

[Registro 12, Unsigned16, ro]

Immagine termica o segnale proporzionale alla corrente (solo per monitoraggio).

Versione software [0x0E]

[Registro 15, Unsigned16, ro]

Visualizza la versione software del dispositivo.

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.

Versione hardware [0x0F]

[Registro 16, Unsigned16, ro]

Visualizza la versione hardware del dispositivo.

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
Modello ROTADRIVE				Interfaccia			Freno	-			Versione hardware				

dove:

00 ... 03	= versione hardware
04 ... 06	= bit non utilizzati
07	= freno (0 = senza freno; 1 = con freno)
08 .. 11	= interfaccia (00 = Modbus; 01 = Profibus; 02 = CANopen; 03 ... 0F = non utilizzati)
12 ... 15	= modello ROTADRIVE (00 = RD4; 01 = RD1xA; 02 = RD5; 03 ... 0F = non utilizzati)

Il valore 11 81 hex, che corrisponde alla rappresentazione binaria 00010001 10000001, deve perciò essere interpretato come: versione hardware 1 (bit 0 = 1); dispositivo con freno (bit 7 = 1); interfaccia Profibus (bit 8 = 1); modello RD1xA (bit 12 = 1).



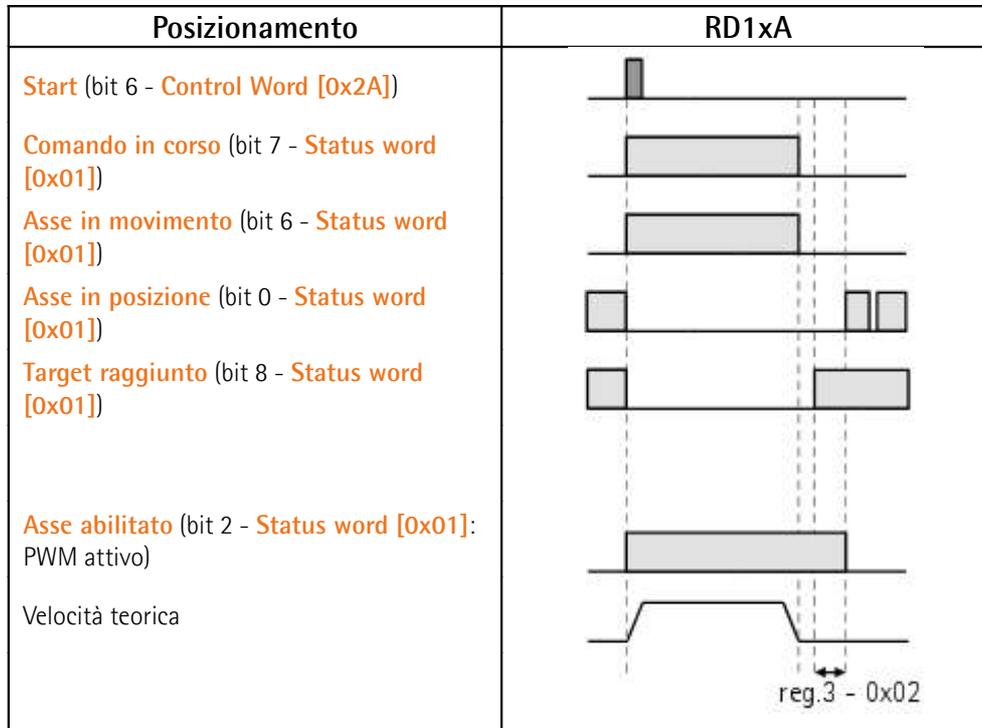
NOTA

Per salvare i parametri modificati eseguire **Salva parametri**.

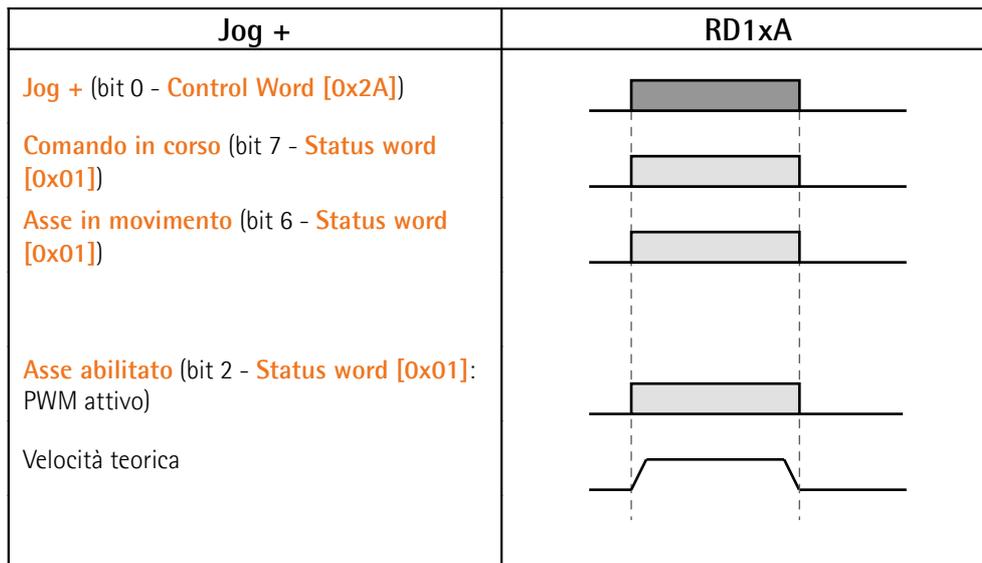
Nel caso di spegnimento del dispositivo i dati non salvati andranno persi!



Esempio 1



Esempio 2



3.2 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.

Per ogni informazione sul significato dei codici eccezione (exception code) fare riferimento alla sezione "MODBUS Exception Responses" alla pagina 48 del documento "MODBUS Application Protocol Specification V1.1b".

4 Esempi di programmazione Modbus®

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

4.1 Utilizzo del codice funzione 03 Read Holding Registers



Esempio 1

Richiesta di lettura dei parametri **Accelerazione [0x06]** (registro 7) e **Decelerazione [0x07]** (registro 8) allo Slave con indirizzo 1.

Request PDU

[01][03][00][06][00][02][24][0A]

dove:

[01] = indirizzo dello Slave

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

[00][06] = indirizzo iniziale (parametro **Accelerazione [0x06]**, registro 7)

[00][02] = numero di registri richiesti

[24][0A] = CRC

Response PDU

[01][03][04][03][E8][05][DC][78][8A]

dove:

[01] = indirizzo dello Slave

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

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

[03][E8] = valore del registro 7 **Accelerazione [0x06]**, 03 E8 hex = 1000 dec

[05][DC] = valore del registro 8 **Decelerazione [0x07]**, 05 DC hex = 1500 dec

[78][8A] = CRC

Il parametro **Accelerazione [0x06]** (registro 7) ha valore 03 E8 hex, cioè 1000 in notazione decimale; il parametro **Decelerazione [0x07]** (registro 8) ha valore 05 DC hex, cioè 1500 in notazione decimale.

4.2 Utilizzo del codice funzione **04 Read Input Register**



Esempio 1

Richiesta di lettura del parametro **Posizione corrente [0x02-0x03]** (registri 3 e 4) allo Slave con indirizzo 1.

Request PDU

[01][04][00][02][00][02][D0][0B]

dove:

[01] = indirizzo dello Slave

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

[00][02] = indirizzo iniziale (parametro **Posizione corrente [0x02-0x03]**, registro 3)

[00][02] = numero di registri richiesti

[D0][0B] = CRC

Response PDU

[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 3 **Posizione corrente [0x02-0x03]**, 00 00 hex = 0 dec

[2F][F0] = valore del registro 4 **Posizione corrente [0x02-0x03]**, 2F F0 hex = 12272 dec

[E7][F0] = CRC

Il parametro **Posizione corrente [0x02-0x03]** (registri 3 e 4) ha valore 00 00 2F F0 hex, cioè 12272 in notazione decimale.



Esempio 2

Richiesta di lettura della variabile **Registro allarmi [0x00]** (registro 1) allo Slave con indirizzo 1.

Request PDU

[01][04][00][00][00][01][31][CA]

dove:

[01] = indirizzo dello Slave

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

[00][00] = indirizzo iniziale (variabile **Registro allarmi [0x00]**, registro 1)

[00][01] = numero di registri richiesti

[31][CA] = CRC

Response PDU

[01][04][02][00][81][79][50]

dove:

[01] = indirizzo dello Slave

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

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

[00][81] = valore del registro 1 **Registro allarmi [0x00]**, 00 81 hex = 0000 0000
1000 0001 bin

[79][50] = CRC

Ne consegue che nella variabile **Registro allarmi [0x00]** (registro 1) sono attivi i bit 0 e 7, vale a dire (vedi a pagina 139): **Dati macchina non validi e Emergenza.**

4.3 Utilizzo del codice funzione **06 Write Single Register**



Esempio 1

Richiesta di scrittura del valore 05 DC hex (= 1500 dec) nel parametro **Accelerazione [0x06]** (registro 7) dello Slave con indirizzo 1.

Request PDU

[01][06][00][06][05][DC][6B][02]

dove:

[01] = indirizzo dello Slave

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

[00][06] = indirizzo del registro (parametro **Accelerazione [0x06]**, registro 7)

[05][DC] = valore da impostare nel registro

[6B][02] = CRC

Response PDU

[01][06][00][06][05][DC][6B][02]

dove:

[01] = indirizzo dello Slave

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

[00][06] = indirizzo del registro (parametro **Accelerazione [0x06]**, registro 7)

[05][DC] = valore impostato nel registro

[6B][02] = CRC

Nel parametro **Accelerazione [0x06]** (registro 7) viene impostato il valore 05 DC hex, cioè 1500 in notazione decimale.



Esempio 2

Richiesta di scrittura del valore 00 84 hex nella variabile **Control Word [0x2A]** (registro 43) dello Slave con indirizzo 1.

Request PDU

[01][06][00][2A][00][84][A8][61]

dove:

[01] = indirizzo dello Slave

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

[00][2A] = indirizzo del registro (variabile **Control Word [0x2A]**, registro 43)

[00][84] = valore da impostare nel registro

[A8][61] = CRC

Response PDU

[01][06][00][2A][00][84][A8][61]

dove:

[01] = indirizzo dello Slave

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

[00][2A] = indirizzo del registro (variabile **Control Word [0x2A]**, registro 43)

[00][84] = valore impostato nel registro

[A8][61] = CRC

Nella variabile **Control Word [0x2A]** (registro 43) viene impostato il valore 00 84 hex = 0000 0000 1000 0100 bin. In altri termini, sono forzati a 1 il bit di **Stop** (bit 2 = 1) e il bit di **Emergenza** (bit 7 = 1): il dispositivo è pronto per eseguire il comando di movimento richiesto.



Esempio 3

Richiesta di scrittura del valore 0A 80 hex nella variabile **Control Word [0x2A]** (registro 43) dello Slave con indirizzo 1.

Request PDU

[01][06][00][2A][0A][80][AF][02]

dove:

[01] = indirizzo dello Slave

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

[00][2A] = indirizzo del registro (variabile **Control Word [0x2A]**, registro 43)

[0A][80] = valore da impostare nel registro

[AF][02] = CRC

Response PDU

[01][06][00][2A][0A][80][AF][02]

dove:

[01] = indirizzo dello Slave

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

[00][2A] = indirizzo del registro (variabile **Control Word [0x2A]**, registro 43)

[0A][80] = valore impostato nel registro

[AF][02] = CRC

Nella variabile **Control Word [0x2A]** (registro 43) viene impostato il valore 0A 80 hex = 0000 0010 1000 0000 bin. In altri termini, il dispositivo è forzato in stop (bit di **Stop** 2 = 0), ma non in emergenza (bit di **Emergenza** 7 = 1) ed è richiesto il salvataggio dei parametri (bit **Salva parametri** 9 = 1).

4.4 Utilizzo del codice funzione **16 Write Multiple Registers**



Esempio 1

Richiesta di scrittura dei valori 1500 e 1000 nei parametri **Accelerazione [0x06]** (registro 7) e **Decelerazione [0x07]** (registro 8) dello Slave con indirizzo 1.

Request PDU

[01][10][00][06][00][02][04][05][DC][03][E8][B2][0D]

dove:

[01] = indirizzo dello Slave

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

[00][06] = indirizzo iniziale (parametro **Accelerazione [0x06]**, registro 7)

[00][02] = numero di registri richiesti

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

[05][DC] = valore da impostare nel registro 7 **Accelerazione [0x06]**, 05 DC hex = 1000 dec

[03][E8] = valore da impostare nel registro 8 **Decelerazione [0x07]**, 03 E8 hex = 1500 dec

[B2][0D] = CRC

Response PDU

[01][10][00][06][00][02][A1][C9]

dove:

[01] = indirizzo dello Slave

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

[00][06] = indirizzo iniziale (parametro **Accelerazione [0x06]**, registro 7)

[00][02] = numero di registri su cui si è scritto

[A1][C9] = CRC

Ne consegue che nel parametro **Accelerazione [0x06]** (registro 7) si imposta il valore 05 DC hex, cioè 1500 in notazione decimale; nel parametro **Decelerazione [0x07]** (registro 8) si imposta il valore 03 E8 hex, cioè 1000 in notazione decimale.

Pagina lasciata intenzionalmente bianca



Versione HW-SW	Versione documento	Descrizione
1-1	1.0	Prima stampa
1-2	1.1	Aggiornamento software, modifica sezione "1.7 Pagina "Upgrade Firmware"", modifica sezione "4.3 Indicatori LED (Figura 4)"
1-3 1-4 1-5 1-6	1.2	Aggiunto "Jog step" in Control Word e parametro "Ampiezza passo jog". Aggiornate definizioni jog. Aggiornata descrizione pulsanti jog. Aggiornata definizione "Asse abilitato". Aggiornamento file GSD (V2) e file Modbus.exe (V2.2/V2.3/V2.4).
2-7	1.3	Aggiornamento informazioni freno, aggiornamento file GSD (V3) e Modbus.exe (V2.5)
3-0 3-1	1.4	Aggiornamento sezione "Connessioni elettriche". Aggiornamento "KP anello di corrente". Aggiornamento file GSD.
3-2	1.5	Introdotta versione -T92. Aggiornamento file GSD (V4) e file Modbus.exe (V2.7).
3-3	1.6	Introdotta versione con modulo "no param", salvataggio parametri e caricamento default gestiti anche via Profibus. Aggiornamento file GSD (V5). Aggiornamento definizioni preset.
3-3	1.7	Aggiornamento informazione LED
3-3	1.8	Avvertenza per tensione controelettromotrice, aggiornamento registri Modbus Tolleranza di posizione [0x01], Max errore di inseguimento [0x03], Ampiezza passo jog [0x19]



Smaltire separatamente

LIKA Electronic

Via S. Lorenzo, 25
36010 Carré (VI) • Italy
Tel. +39 0445 806600
Fax +39 0445 806699



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