Manuale d'uso

RD1A

RD12A

CANopen

Modbus

Versione RS-232

lika

Smart encoders & actuators

Questa pubblicazione è edita da Lika Electronic s.r.l. 2015. All rights reserved. Tutti i diritti riservati. Alle Rechte vorbehalten. Todos los derechos reservados. Tous droits réservés.

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

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

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

Indice generale

Manuale d'uso 1
Indice generale 3
Indice analitico 9
Convenzioni grafiche e iconografiche 11
Informazioni preliminari 12
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 Velocità di trasmissione dei dati: Baud rate (Figura 5) 29
4.4.3 Resistenza di terminazione: RT (Figura 5) 30
4.4.4 Pulsanti JOG + e JOG - (Figura 5) 30
4.4.5 Pulsante PRESET (Figura 5) 31
5 Quick reference 32
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 CANopen® 41
1 Interfaccia CANopen® 42
1.1 File EDS 42
1.2 Funzionamento a stati 43
1.2.1 Initialization 43
1.2.2 Pre-operational 44
1.2.3 Operational 44
1.2.4 Stopped 44
1.3 Tipi di messaggi 45
1.3.1 COB-ID previsti o riconosciuti 46
1.4 Messaggi NMT 46
1.5 Messaggi di Boot-up 47
1.6 Messaggi PDO 47
1.6.1 Messaggi "PDO1 ricevuti" inviati dal Master allo Slave 48
Control Word 48
Jog + ... 48

Jog - 48
Stop 49
Reset allarmi 49
Jog incrementale 49
Start 49
Emergenza 50
Asse in coppia 50
OUT 1 50
Sblocco freno 50
Target position 51
1.6.2 Messaggi "PDO1 trasmessi" inviati dallo Slave al Master 52
Status word 52
Asse in posizione 52
Asse abilitato 52
Finecorsa SW + 52
Finecorsa SW - 52
Allarme 52
Asse in movimento 52
Comando in corso 53
Target raggiunto 53
Pulsante 1 Jog + 53
Pulsante 2 Jog - 53
Pulsante 3 Preset 53
Saturazione DAC 53
IN 1 54
IN 2 54
IN 3 54
Velocità corrente 54
Posizione corrente 54
1.7 Messaggi SDO 56
1.7.1 Command 56
2 Parametri di programmazione CANopen® 57
2.1 Dizionario oggetti 57
2.1.1 Oggetti standard (DS 301) 57
1000-00 Tipo di dispositivo 57
1001-00 Registro errori 57
1003 Campo errori predefinito 58
1005-00 COB-ID messaggi Sync 58
1008-00 Nome del dispositivo 58
1009-00 Versione hardware 58
100A-00 Versione software 58
100C-00 Guard Time 58
100D-00 Life Time Factor 59
1010-01 Salvataggio parametri 59
1011-01 Parametri di default 59
1014-00 COB-ID EMCY 60
1015-00 Inhibit time EMCY 60
1018 Informazioni di identificazione 60
1400 Parametri PDO1 ricevuti 61
1600 Mappatura PDO1 ricevuti 62
1800 Parametri PDO1 inviati 62

1A00 Mappatura PDO1 inviati....64
2.1.2 Oggetti specifici del costruttore....65
3000-00 Velocità trasmissione dati....65
3001-00 Node ID....65
3108-00 Accelerazione....66
3109-00 Decelerazione....66
310C-00 Max errore di inseguimento....66
310D-00 Tolleranza di posizione....66
310E-00 Tempo asse in tolleranza....67
310F Delta spazio....67
3111-00 Kp anello di posizione....68
3112-00 Ki anello di posizione....68
3114-00 Velocità Jog....68
3115-00 Velocità di lavoro....69
3116-00 Velocità massima....69
3117-00 Ampiezza passo jog....69
3118-00 Durata corrente di stacco....70
3120-00 Distanza_giro....70
3300-00 Valore di preset....71
3301-00 Valore di offset....71
3302-00 Direzione conteggio....71
3330-00 Kp anello di corrente....72
3331-00 Ki anello di corrente....72
3340-00 Corrente massima....72
3341-00 Corrente di stacco....72
3342-00 Rapporto di riduzione....72
3343-00 Finecorsa assoluto positivo....73
3344-00 Finecorsa assoluto negativo....73
3005-00 Corrente attuale....74
3006-00 Temperatura....74
3100-00 Control word....74
3101-00 Status word....74
3102-00 Posizione richiesta....74
3103-00 Posizione attuale....74
3105-00 Velocità attuale....75
3106-00 Target position....75
3107-00 Velocità target....75
310B-00 Errore di posizione....75
3110-00 Cyclic Time....75
3200-00 Allarmi_scheda....76
Dati macchina non validi....76
Errore memoria flash....76
Errore di inseguimento....76
Asse non sincronizzato....76
Target non valido....76
Emergenza....76
Sovracorrente....76
Sovratemperatura....77
Sottotensione....77
CAN Life guard error....77
2.2 Messaggi di avvertenza (Warning)....78

2.3 Messaggi di emergenza (Emergency) ... 78
Nessun errore ... 78
Errore generico ... 78
Sovracorrente ... 78
Sovratensione ... 78
Sottotensione ... 78
Sovratemperatura ... 78
Memoria flash ... 78
Life Guard ... 78
Inseguimento ... 78
2.4 Node guarding protocol ... 79
3 Esempi di programmazione CANopen® ... 80
Interfaccia Modbus® ... 81
1 Utilizzo con seriale di servizio ... 82
1.1 Configurazione mediante software di Lika Electronic ... 82
1.2 Pagina "Configurazione seriale" ... 84
1.3 Pagina "Modalità operativa" ... 87
1.4 Pagina "Dati macchina" ... 92
1.5 Pagina "Monitor messaggi" ... 94
1.6 Pagina "Test Lika" ... 95
1.7 Pagina "Upgrade Firmware" ... 95
1.7.1 In caso di errore ... 98
2 Interfaccia Modbus® ... 100
2.1 Principi guida del protocollo Modbus Master / Slave ... 100
2.2 Frame Modbus ... 101
2.3 Modalità di trasmissione ... 102
2.3.1 Modalità di trasmissione RTU ... 102
2.4 Codici funzione ... 105
2.4.1 Codici funzione implementati ... 105
03 Read Holding Registers ... 105
04 Read Input Register ... 107
06 Write Single Register ... 109
16 Write Multiple Registers ... 111
3 Parametri di programmazione Modbus® ... 115
3.1 Parametri disponibili ... 115
3.1.1 Parametri Dati macchina ... 115
Distanza_giro [0x00] ... 115
Tolleranza di posizione [0x01] ... 116
Tempo asse in tolleranza [0x02] ... 116
Max errore di inseguimento [0x03] ... 117
Kp anello di posizione [0x04] ... 117
Ki anello di posizione [0x05] ... 117
Accelerazione [0x06] ... 117
Decelerazione [0x07] ... 117
Delta spazio positivo [0x08-0x09] ... 118
Delta spazio negativo [0x0A-0x0B] ... 119
Velocità Jog [0x0C] ... 120
Velocità di lavoro [0x0D] ... 120
Durata corrente di stacco [0x0E] ... 120
Direzione conteggio [0x0F] ... 121
Kp anello di corrente [0x10] ... 121

Ki anello di corrente [0x11] 121
Corrente massima [0x12] 121
Corrente di stacco [0x13] 122
Valore di offset [0x14-0x15] 122
Valore di preset [0x16-0x17] 122
Rapporto di riduzione [0x18] 123
Ampiezza passo jog [0x19] 123
Registro comandi Extra [0x29] 123
Lettura assoluta 123
Controllo da PC 123
Control Word [0x2A] 124
Jog + 124
Jog - 124
Stop 124
Reset allarmi 125
Jog incrementale 125
Start 125
Emergenza 125
Abilitazione Watch dog 126
Salva parametri 126
Carica parametri di default 126
Esegui preset conteggio 126
Asse in coppia 126
OUT 1 127
Sblocco freno 127
Target position [0x2B-0x2C] 127
3.1.2 Parametri Input Register 129
Registro allarmi [0x00] 129
Dati macchina non validi 129
Errore memoria flash 129
Errore di inseguimento 129
Asse non sincronizzato 129
Target non valido 129
Emergenza 130
Sovracorrente 130
Sovratemperatura 130
Sottotensione 130
Watch dog 130
Status word [0x01] 131
Asse in posizione 131
Asse abilitato 131
Finecorsa SW + 131
Finecorsa SW - 131
Allarme 132
Asse in movimento 132
Comando in corso 132
Target raggiunto 132
Pulsante 1 Jog + 132
Pulsante 2 Jog - 132
Pulsante 3 Preset 132
Saturazione DAC 133

IN 1 133
IN 2 133
IN 3 133
Posizione corrente [0x02-0x03] 133
Velocità corrente [0x04] 133
Errore di inseguimento [0x05-0x06] 134
Corrente attuale [0x07] 134
Temperatura [0x08] 134
Elenco DM errati [0x09-0x0A] 134
I2t [0x0B] 135
Dip-switch baud rate [0x0C] 135
Dip-switch indirizzo [0x0D] 135
Versione software [0x0E] 135
Versione hardware [0x0F] 136
3.2 Codici eccezione 138
4 Esempi di programmazione Modbus® 139
4.1 Utilizzo del codice funzione 03 Read Holding Registers 139
4.2 Utilizzo del codice funzione 04 Read Input Register 140
4.3 Utilizzo del codice funzione 06 Write Single Register 142
4.4 Utilizzo del codice funzione 16 Write Multiple Registers 144

Indice analitico

1

1000-00 Tipo di dispositivo....57
1001-00 Registro errori....57
1003 Campo errori predefinito....58
1005-00 COB-ID messaggi Sync....58
1008-00 Nome del dispositivo....58
1009-00 Versione hardware....58
100A-00 Versione software....58
100C-00 Guard Time....58
100D-00 Life Time Factor....59
1010-01 Salvataggio parametri....59
1011-01 Parametri di default....59
1014-00 COB-ID EMCY....60
1015-00 Inhibit time EMCY....60
1018 Informazioni di identificazione....60
1400 Parametri PDO1 ricevuti....61
1600 Mappatura PDO1 ricevuti....62
1800 Parametri PDO1 inviati....62
1A00 Mappatura PDO1 inviati....64

3

3000-00 Velocità trasmissione dati....65
3001-00 Node ID....65
3005-00 Corrente attuale....74
3006-00 Temperatura....74
3100-00 Control word....74
3101-00 Status word....74
3102-00 Posizione richiesta....74
3103-00 Posizione attuale....74
3105-00 Velocità attuale....75
3106-00 Target position....75
3107-00 Velocità target....75
3108-00 Accelerazione....66
3109-00 Decelerazione....66
310B-00 Errore di posizione....75
310C-00 Max errore di inseguimento....66
310D-00 Tolleranza di posizione....66
310E-00 Tempo asse in tolleranza....67
310F Delta spazio....67
3110-00 Cyclic Time....75
3111-00 Kp anello di posizione....68
3112-00 Ki anello di posizione....68
3114-00 Velocità Jog....68
3115-00 Velocità di lavoro....69
3116-00 Velocità massima....69
3117-00 Ampiezza passo jog....69
3118-00 Durata corrente di stacco....70
3120-00 Distanza_giro....70
3200-00 Allarmi_scheda....76
3300-00 Valore di preset....71
3301-00 Valore di offset....71
3302-00 Direzione conteggio....71
3330-00 Kp anello di corrente....72
3331-00 Ki anello di corrente....72
3340-00 Corrente massima....72
3341-00 Corrente di stacco....72
3342-00 Rapporto di riduzione....72
3343-00 Finecorsa assoluto positivo....73
3344-00 Finecorsa assoluto negativo....73

A

Abilitazione Watch dog....126
Accelerazione [0x06]....117
Allarme....52, 132
Ampiezza passo jog [0x19]....123
Asse abilitato....52, 131
Asse in coppia....50, 126
Asse in movimento....52, 132
Asse in posizione....52, 131
Asse non sincronizzato....76, 129

C

CAN Life guard error....77
Carica parametri di default....126
Comando in corso....53, 132
Control Word....48
Control Word [0x2A]....124
Controllo da PC....123
Corrente attuale [0x07]....134
Corrente di stacco [0x13]....122
Corrente massima [0x12]....121

D

Dati macchina non validi....76, 129
Decelerazione [0x07]....117
Delta spazio negativo [0x0A-0x0B]....119
Delta spazio positivo [0x08-0x09]....118
Dip-switch baud rate [0x0C]....135
Dip-switch indirizzo [0x0D]....135
Direzione conteggio [0x0F]....121
Distanza_giro [0x00]....115
Durata corrente di stacco [0x0E]....120

E

Elenco DM errati [0x09-0x0A]....134
Emergenza....50, 76, 125, 130
Errore di inseguimento....76, 129

Errore di inseguimento [0x05-0x06]....134
Errore generico....78
Errore memoria flash....76, 129
Esegui preset conteggio....126
F
Finecorsa SW -....52, 131
Finecorsa SW +....52, 131
I
I2t [0x0B]....135
IN 1....54, 133
IN 2....54, 133
IN 3....54, 133
Inseguimento....78
J
Jog -....48, 124
Jog +....48, 124
Jog incrementale....49, 125
K
Ki anello di corrente [0x11]....121
Ki anello di posizione [0x05]....117
Kp anello di corrente [0x10]....121
Kp anello di posizione [0x04]....117
L
Lettura assoluta....123
Life Guard....78
M
Max errore di inseguimento [0x03]....117
Memoria flash....78
N
Nessun errore....78
O
OUT 1....50, 127
P
Posizione corrente....54
Posizione corrente [0x02-0x03]....133
Pulsante 1 Jog +....53, 132
Pulsante 2 Jog -....53, 132
Pulsante 3 Preset....53, 132
R
Rapporto di riduzione [0x18]....123
Registro allarmi [0x00]....129
Registro comandi Extra [0x29]....123
Reset allarmi....49, 125
S
Salva parametri....126
Saturazione DAC....53, 133
Sblocco freno....50, 127
Sottotensione....77 e seg., 130
Sovracorrente....76, 78, 130
Sovratemperatura....77 e seg., 130
Sovratensione....78
Start....49, 125
Status word....52
Status word [0x01]....131
Stop....49, 124
T
Target non valido....76, 129
Target position....51
Target position [0x2B-0x2C]....127
Target raggiunto....53, 132
Temperatura [0x08]....134
Tempo asse in tolleranza [0x02]....116
Tolleranza di posizione [0x01]....116
V
Valore di offset [0x14-0x15]....122
Valore di preset [0x16-0x17]....122
Velocità corrente....54
Velocità corrente [0x04]....133
Velocità di lavoro [0x0D]....120
Velocità Jog [0x0C]....120
Versione hardware [0x0F]....136
Versione software [0x0E]....135
W
Watch dog....130

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 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 CANopen 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 CANopen**, sono fornite tutte le informazioni sia generali che specifiche relative all'interfaccia CANopen. In questa sezione sono descritte le caratteristiche dell'interfaccia e gli oggetti CANopen 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.

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 standard 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.rl. 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.

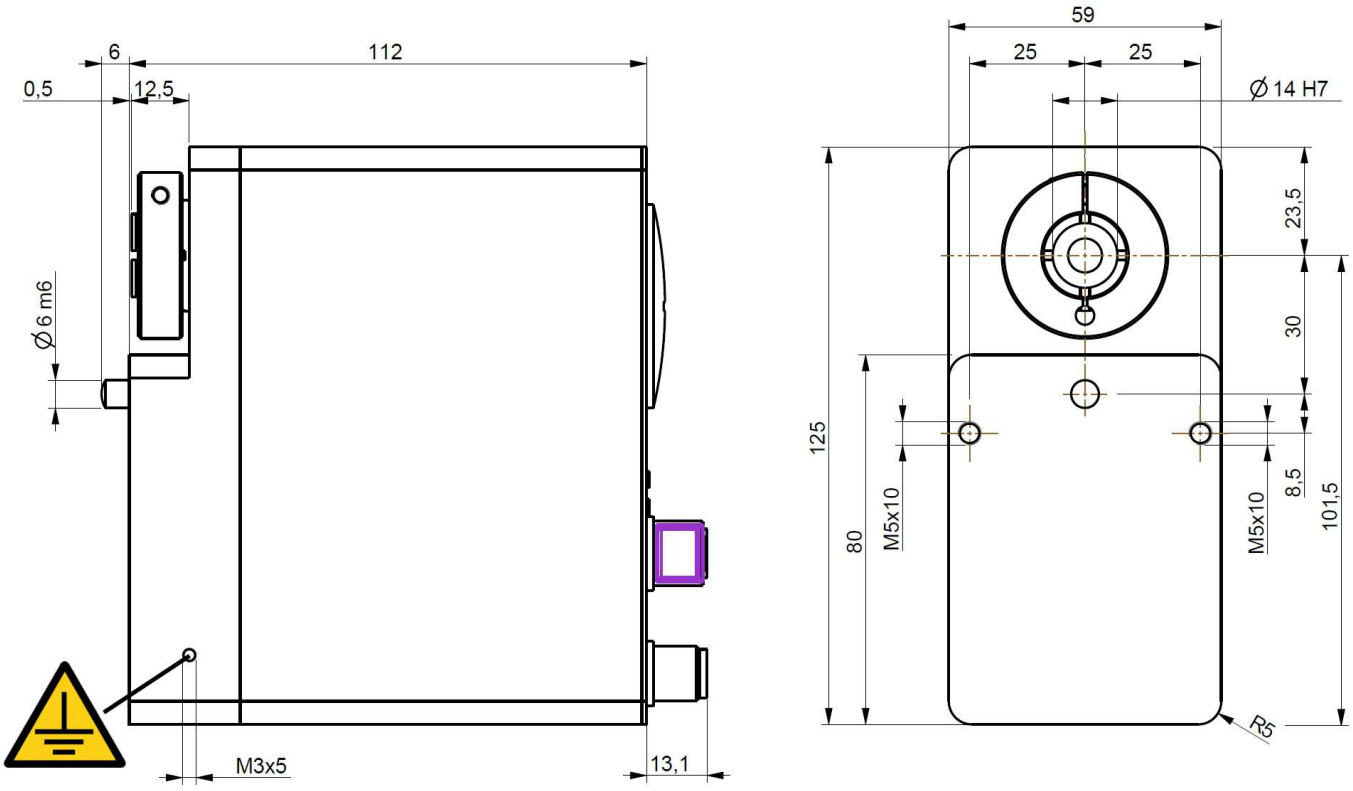


Figura 1 - Lay-out unità RD1A

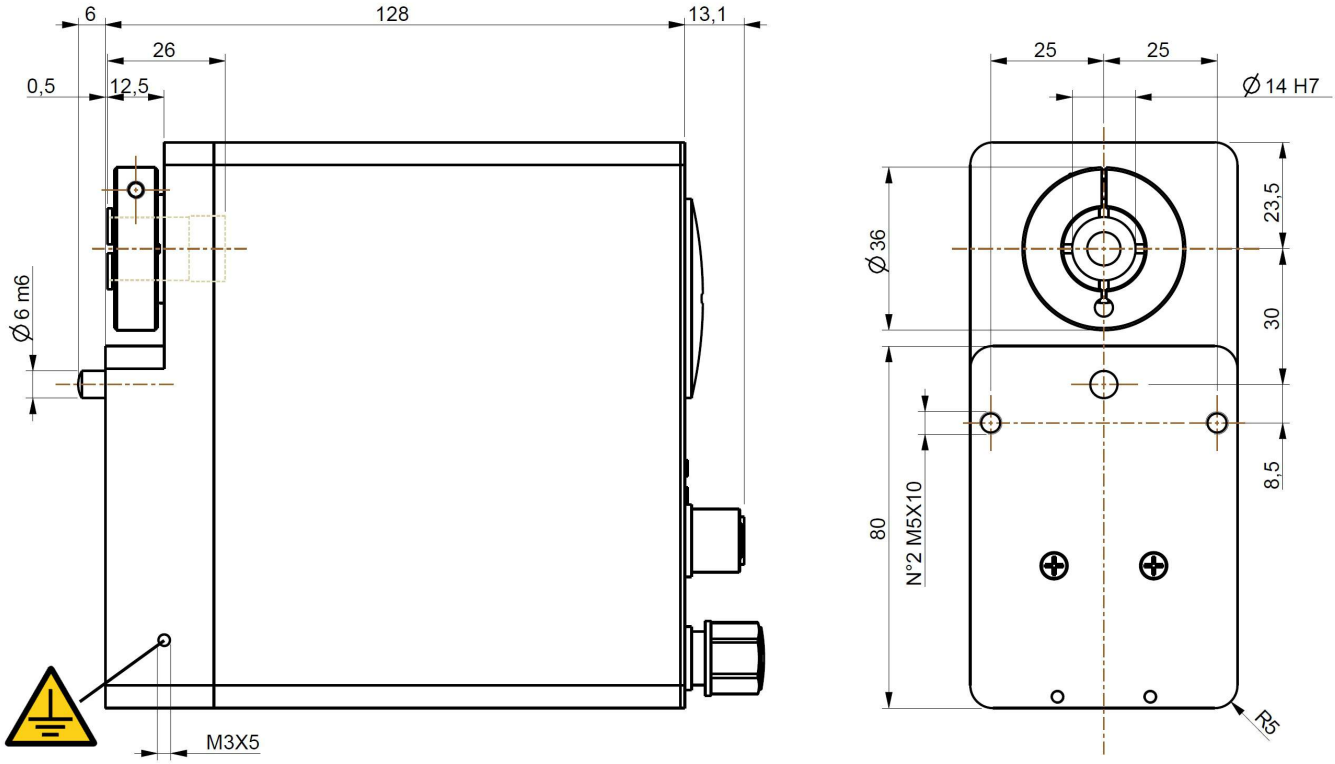


Figura 2 - Lay-out unità RD12A

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!

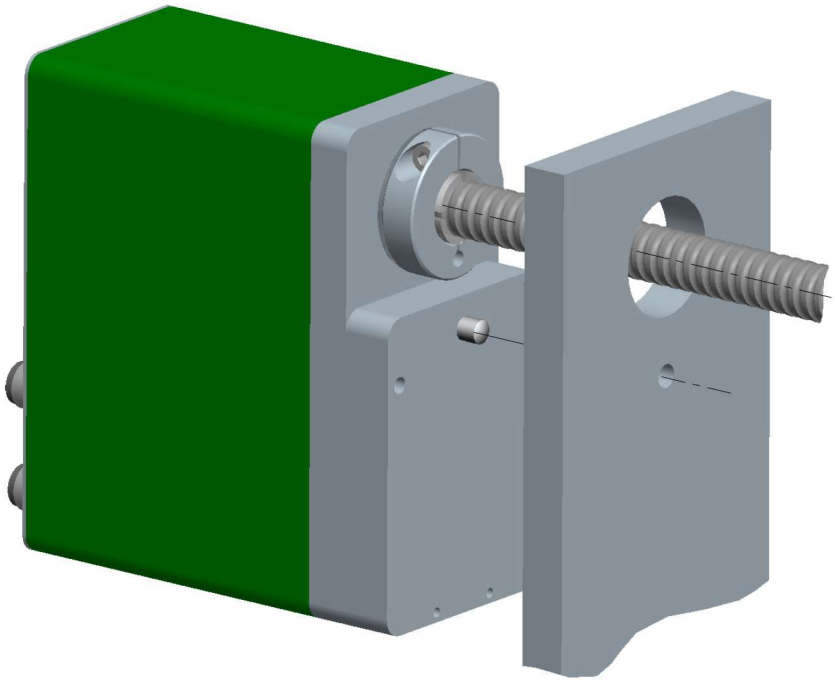
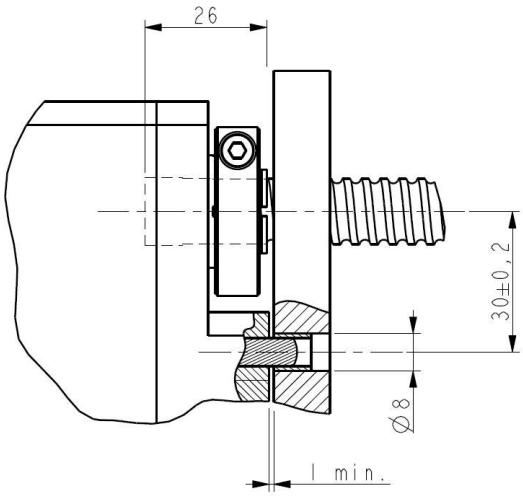


Figura 3 - Esempio di installazione di unità RD1xA su vite senza fine

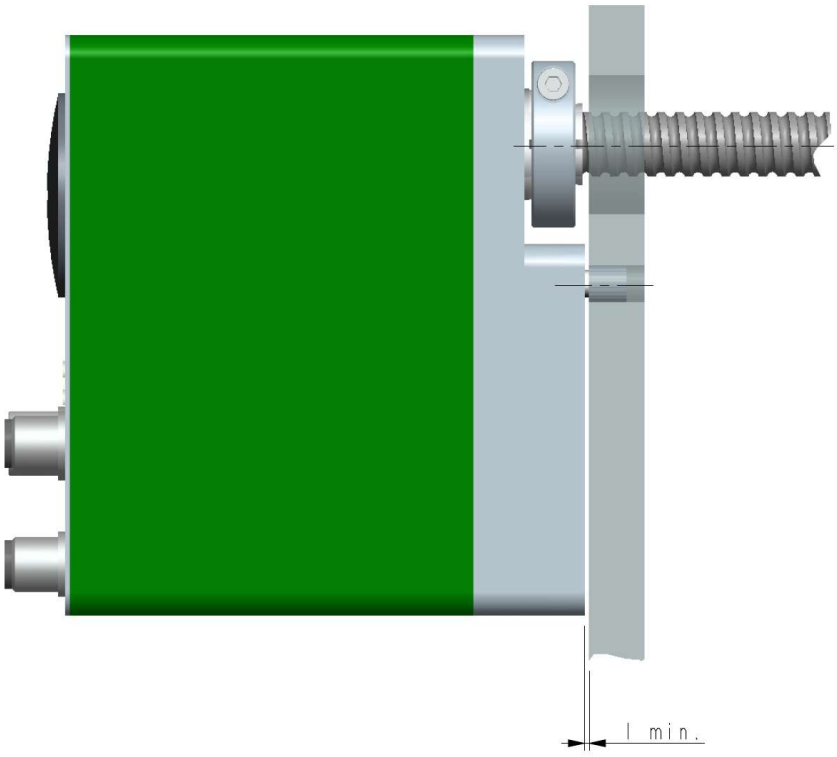
Per installare correttamente l'unità ROTADRIVE seguire scrupolosamente le istruzioni riportate qui di seguito, tenendo conto a ogni modo che le possibilità di installazione possono essere molteplici e dipendenti dalla specifica applicazione.

- Eseguire un foro Ø 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 ± 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
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. I cavi intestati EC- di Lika Electronic prevedono il collegamento della calza alla ghiera del connettore per la messa a terra attraverso il corpo del dispositivo. I connettori volanti E- di Lika Electronic utilizzano un connettore plastico; pertanto non è possibile la raccolta calza. Nel caso in cui si utilizzi un connettore metallico collegare opportunamente la calza del cavo attenendosi alle istruzioni del costruttore. 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.

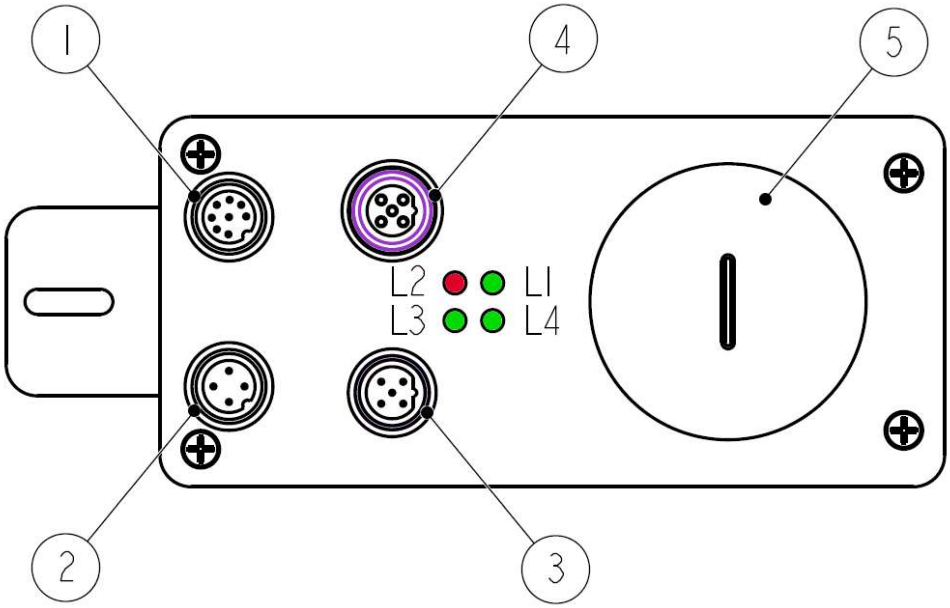


Figura 4: Connessioni elettriche

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 CANopen
4	Connettore M12 femmina 5 pin BUS OUT CANopen
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

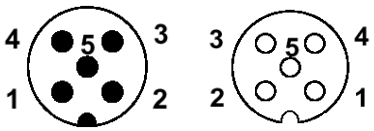
4.2 Connettori (Figura 4)

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 CANopen
connettore M12 5 pin
codifica A
(vista lato contatti)



maschio (BUS IN) femmina (BUS OUT)

Pin	Funzione
1	n.c.
2	n.c.
3	CAN GND[1]
4	CAN High
5	CAN Low
Custodia	Calza[2]

[1] Riferimento di tensione 0VDC del segnale CAN. Non è collegato a 0VDC dell'alimentazione.
[2] Solo con cavi intestati EC- Lika Electronic.

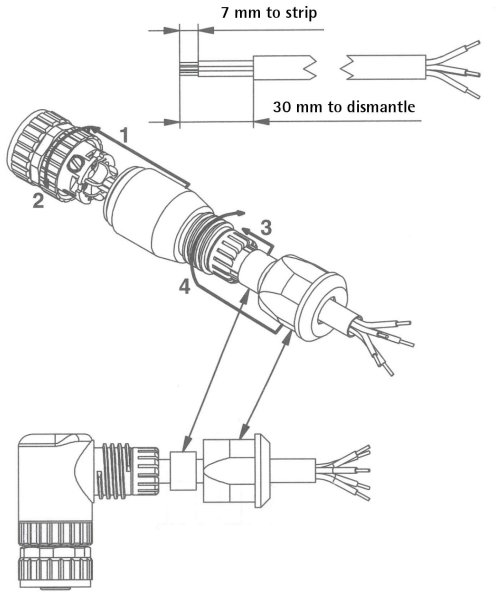
n.c. = non connesso

Per il collegamento del bus si raccomanda di utilizzare cavi certificati CANopen.

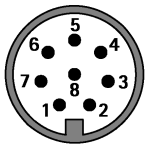
NOTA

Si consiglia di prevedere sempre il collegamento della calza del cavo a un buon punto di terra sul lato utilizzatore.

I cavi intestati EC- di Lika Electronic prevedono il collegamento della calza alla ghiera del connettore per la messa a terra attraverso il corpo del dispositivo. I connettori volanti E- di Lika Electronic utilizzano un connettore plastico; pertanto non è possibile la raccolta calza (Figura sotto). Nel caso in cui si utilizzi un connettore metallico collegare opportunamente la calza del cavo attenendosi alle istruzioni del costruttore.



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 81.

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 e del dispositivo, come esplicitato nelle seguenti tabelle.
Si badi che i led hanno significato diverso a seconda che sia attiva la rete CANopen oppure la rete Modbus.

CANopen

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

LED 2 ROSSO ERR	Descrizione
ON	Bus off
Doppio flash	Node guarding error (si veda a pagina 79)
Singolo flash	Massimo numero di warning raggiunto
Lampeggiante	Errore generico o **Errore memoria flash**
OFF	Nessun errore presente

LED 3 VERDE RUN	Descrizione
ON	Dispositivo in stato **Operational**
Singolo flash	Dispositivo in stato **Stopped**
Lampeggiante	Dispositivo in stato **Pre-Operational**

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 95), 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 95), 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 95.
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 95), 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 95.

LED 4 VERDE	Descrizione
ON	Motore alimentato, presenza di tensione
OFF	Motore non alimentato, assenza di tensione

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	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 95), 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 95), 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 95.
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 95), 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 95.

LED 4 VERDE	Descrizione
ON	Motore alimentato, presenza di tensione
OFF	Motore non alimentato, assenza di tensione

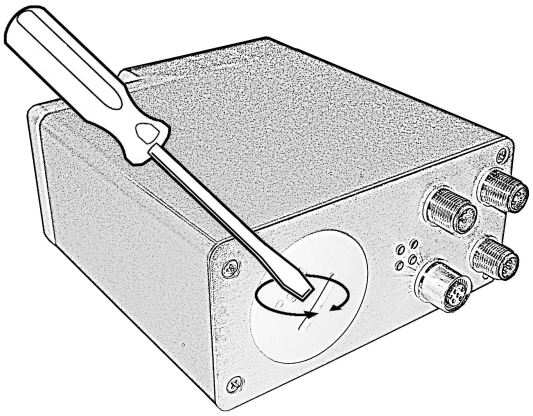
Durante l'inizializzazione del dispositivo, il sistema esegue un controllo sul corretto funzionamento degli indicatori led; pertanto gli indicatori led lampeggiano per un istante.

4.4 Selettori e pulsanti (Figura 5)

ATTENZIONE
Questa operazione deve essere eseguita con dispositivo non alimentato!

NOTA
Eseguire 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.

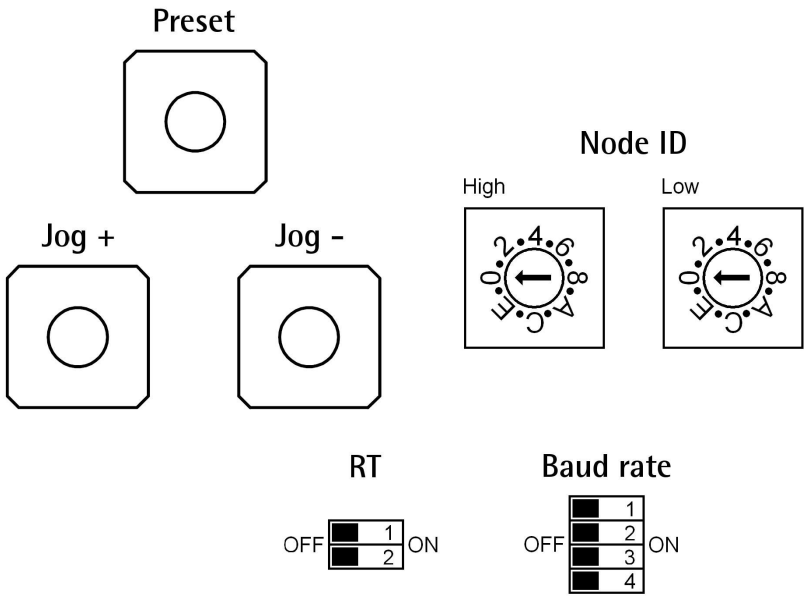


Figura 5: Selettori e pulsanti

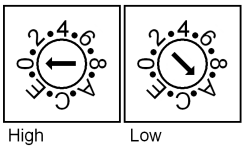
4.4.1 Indirizzo nodo: Node ID (Figura 5)

ATTENZIONE
Questa operazione deve essere eseguita con dispositivo non alimentato!

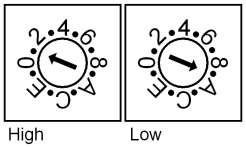
Impostare il valore esadecimale dell'indirizzo del nodo.
L'indirizzo del nodo deve avere un valore compreso tra 1 e 127 (127 = 7F hex).

Esempio

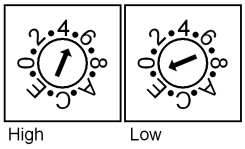
Indirizzo 10 = 0A hex:



Indirizzo 25 = 19 hex:



Indirizzo 95 = 5F hex:



NOTA
L'indirizzo assegnato tramite questo selettore è valido per entrambi i bus di campo CANopen 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 127, il dispositivo utilizzerà automaticamente l'indirizzo 127.

4.4.2 Velocità di trasmissione dei dati: Baud rate (Figura 5)

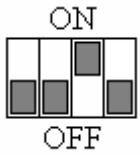
ATTENZIONE
Questa operazione deve essere eseguita con dispositivo non alimentato!

Impostare il valore binario della velocità di trasmissione dei dati secondo la seguente tabella, considerando che: ON = 1; OFF = 0.

Switch 1... 4	Baud rate
0000	20 Kbit/s
1000	50 Kbit/s
0100	100 Kbit/s
1100	125 Kbit/s
0010	250 Kbit/s
1010 (default)	500 Kbit/s
0110	800 Kbit/s
1110	1000 Kbit/s

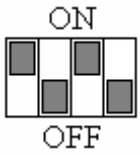
Esempio
Impostare il baud rate a 250 Kbit/s:

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



Impostare il baud rate a 500 Kbit/s:

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



NOTA
Il baud rate impostato tramite questo selettore si riferisce alla sola configurazione in rete CANopen.
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

Per ogni informazione sulla configurazione e l'utilizzo della seriale di servizio RS-232 riferirsi alla sezione "Interfaccia Modbus®" a pagina 81.

4.4.3 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

4.4.4 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 48 (interfaccia CANopen) o a pagina 124 (interfaccia Modbus).

NOTA

Si badi che nell'utilizzo dei pulsanti manuali la funzione di jog incrementale (si veda **Jog incrementale** in **Control Word** a pagina 49 -versione CANopen; **Jog incrementale** in **Control Word [0x2A]** a pagina 125 -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.

4.4.5 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 71 (interfaccia CANopen) o a pagina 122 (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 la velocità di trasmissione dei dati (il baud rate impostato di default è 500 Kbit/s, si veda a pagina 29);
- 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 70 -interfaccia CANopen- o a pagina 115 - interfaccia Modbus);
- impostare il valore in **Velocità Jog** (si veda alla pagina 68 - interfaccia CANopen- o a pagina 120 - interfaccia Modbus);
- impostare il valore in **Velocità di lavoro** (si veda alla pagina 69 - interfaccia CANopen- o a pagina 120 - interfaccia Modbus);
- impostare il valore in **Valore di preset** (si veda a pagina 71 - interfaccia CANopen- o a pagina 122 - interfaccia Modbus);
- impostare i valori dei finecorsa in **310F Delta spazio** (oggetti **310F Delta spazio** sub 1 **Delta positivo** e **310F Delta spazio** sub 2 **Delta negativo**; si veda a pagina 67 - interfaccia CANopen) oppure in **Delta spazio positivo [0x08-0x09]** e **Delta spazio negativo [0x0A-0x0B]** a pagina 118 - interfaccia Modbus;
- procedere al salvataggio dei nuovi valori tramite l'oggetto **1010-01 Salvataggio parametri** 1010h sub 1 (si veda a pagina 59 - interfaccia CANopen) oppure il bit 9 **Salva parametri** del registro **Control Word [0x2A]** (si veda a pagina 126 - interfaccia Modbus).

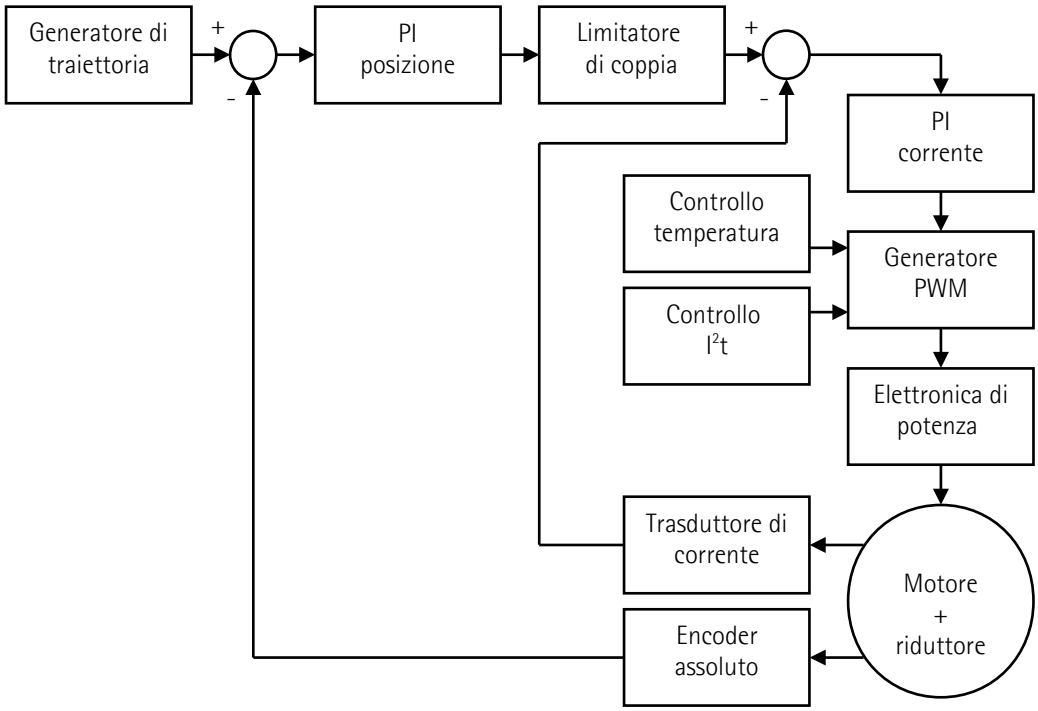
NOTA

I dati macchina **Distanza_giro**, **Velocità Jog**, **Velocità di lavoro** e **Valore di preset**, nonché i valori dei finecorsa (**310F Delta spazio** sub 1 **Delta positivo** e **310F Delta spazio** sub 2 **Delta negativo** - interfaccia CANopen - 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:



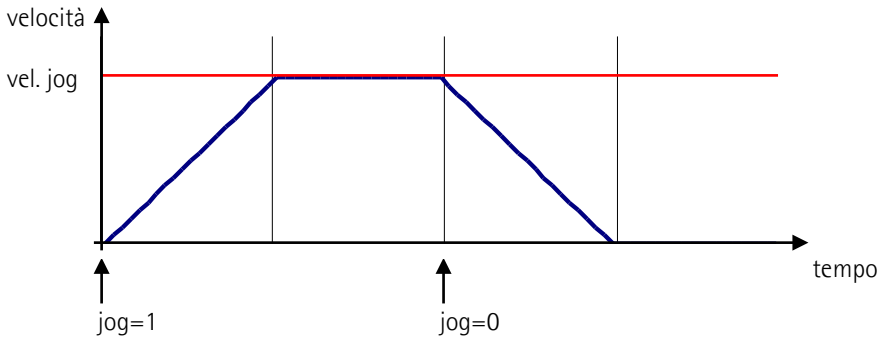
6.2 Tipi di movimento: jog e posizione

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

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

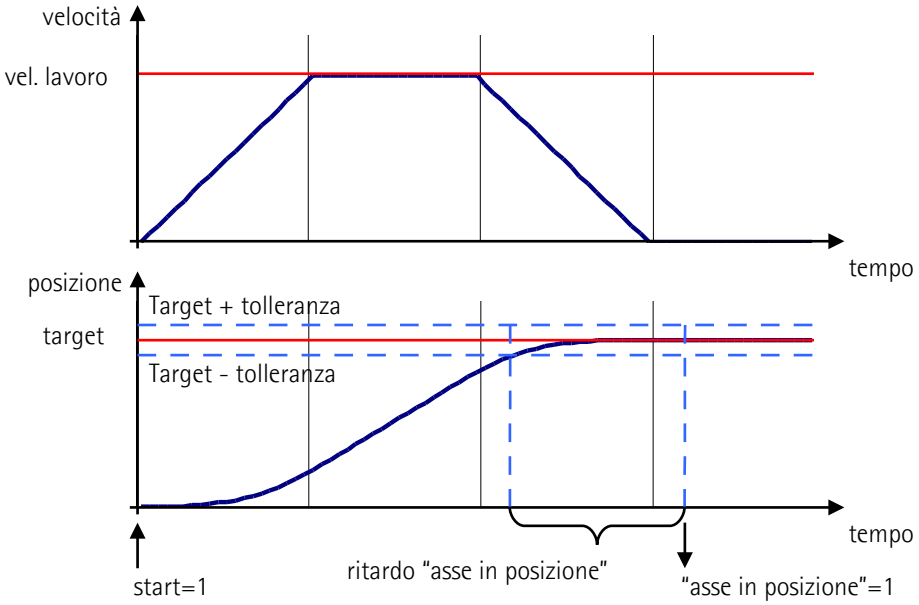
Jog: controllo di velocità

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 68 -interfaccia CANopen-, pagina 120 -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 69 -interfaccia CANopen-, pagina 120 -interfaccia Modbus) che verrà raggiunta solo se lo spazio da percorrere è sufficientemente grande.



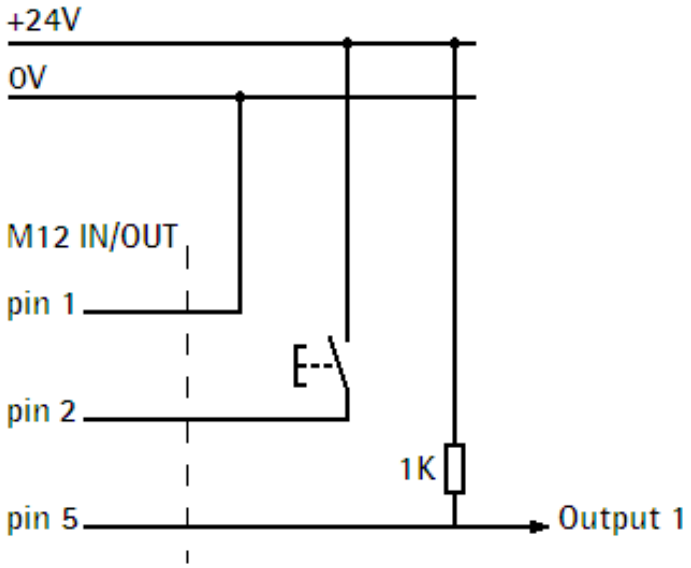
6.3 Ingressi e uscite digitali

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 54 -interfaccia CANopen- o a pagina 133 -interfaccia Modbus) quando il dispositivo si trova nello stato **Operational** -interfaccia CANopen- / **Idle** -interfaccia Modbus.
Il valore logico "Alto" è letto con tensione pari a +24VDC ±10%.

L'uscita è comandata dal Master allo Slave attraverso **Control Word** (bit 13, si veda alla pagina 50 -interfaccia CANopen- o a pagina 127 -interfaccia Modbus) quando il dispositivo si trova nello stato **Operational** -interfaccia CANopen- / **Idle** -interfaccia Modbus.
E' un'uscita open collector con Imax = 150mA.

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 (**310F Delta spazio** sub 1 **Delta positivo** e **310F Delta spazio** sub 2 **Delta negativo** - interfaccia CANopen - 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 70 -interfaccia CANopen- o a pagina 115 - interfaccia Modbus);
- impostazione del valore di **Velocità Jog** (si veda alla pagina 68 - interfaccia CANopen- o a pagina 120 - interfaccia Modbus);
- impostazione del valore di **Velocità di lavoro** (si veda alla pagina 69 - interfaccia CANopen- o a pagina 120 - interfaccia Modbus);
- impostazione di **Valore di preset** (si veda a pagina 71 - interfaccia CANopen- o a pagina 122 - interfaccia Modbus);
- verifica della correttezza del valore di finecorsa positivo (**310F Delta spazio** sub 1 **Delta positivo**, si veda alla pagina 67 -interfaccia CANopen; **Delta spazio positivo [0x08-0x09]**, si veda alla pagina 118 -interfaccia Modbus);
- verifica della correttezza del valore di finecorsa negativo (**310F Delta spazio** sub 2 **Delta negativo**, si veda alla pagina 67 -interfaccia CANopen; **Delta spazio negativo [0x0A-0x0B]**, si veda alla pagina 119 -interfaccia Modbus);
- salvataggio dei nuovi valori tramite l'oggetto **1010-01 Salvataggio parametri** 1010h sub 1 (si veda a pagina 59 - interfaccia CANopen) oppure il bit 9 **Salva parametri** del registro **Control Word [0x2A]** (si veda a pagina 126 - interfaccia Modbus).

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:

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

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 310F Delta spazio sub 1 Delta positivo e 310F Delta spazio sub 2 Delta negativo (interfaccia CANopen) 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 310F Delta spazio sub 1 Delta positivo -interfaccia CANopen; 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 310F Delta spazio sub 2 Delta negativo -interfaccia CANopen; 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).

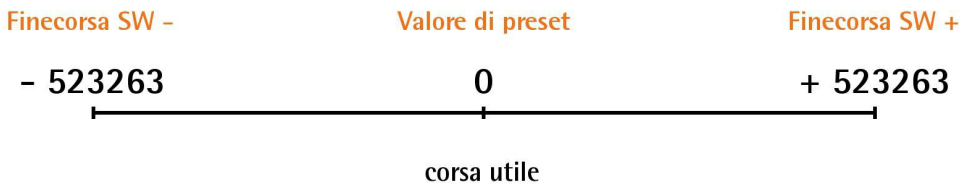
ATTENZIONE
Si badi inoltre che i parametri di seguito elencati sono tutti espressi in relazione al parametro Distanza_giro; di conseguenza la modifica del valore nel parametro Distanza_giro comporta necessariamente una ridefinizione dei valori da essi espressi. I parametri sono: Accelerazione, Decelerazione, Max errore di inseguimento, Tolleranza di posizione, 310F Delta spazio, (interfaccia CANopen) e Delta spazio positivo [0x08-0x09] / Delta spazio negativo [0x0A-0x0B] (interfaccia Modbus), Velocità massima, 3343-00 Finecorsa assoluto positivo (solo interfaccia CANopen), 3344-00 Finecorsa assoluto negativo (solo interfaccia CANopen), Posizione attuale, Velocità

attuale, Target position e Velocità target. 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*1024/1024 = 4266)
- = 2133 impulsi/secondo per RD1xA-...T24-... (2133*1024/1024 = 2133)
- = 1066 impulsi/secondo per RD1xA-...T48-... (1066*1024/1024 = 1066)
- = 556 impulsi/secondo per RD1xA-...T92-... (556*1024/1024 = 556)

Valore di preset = 0

Finecorsa positivo (310F Delta spazio sub 1 Delta positivo -interfaccia CANopen; Delta spazio positivo [0x08-0x09] -interfaccia Modbus) e finecorsa negativo (310F Delta spazio sub 2 Delta negativo -interfaccia CANopen; 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*100/1024 = 417, arrotondato al numero intero)

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

Finecorsa positivo (**310F Delta spazio** sub 1 **Delta positivo** -interfaccia CANopen; **Delta spazio positivo [0x08-0x09]** -interfaccia Modbus) e finecorsa negativo (**310F Delta spazio** sub 2 **Delta negativo** -interfaccia CANopen; **Delta spazio negativo [0x0A-0x0B]**-interfaccia Modbus) massimi = 100 impulsi/giro x 511 giri = 51100 impulsi

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

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

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

7 Tabella parametri di default

Lista parametri	Oggetto CANopen	Indirizzo registro Modbus	Valore di default	
Distanza_giro imp/giro	3120-00	0x00	1024	
Tolleranza di posizione imp	310D-00	0x01	0	
Tempo asse in tolleranza ms	310E-00	0x02	0	
Max errore di inseguimento imp	310C-00	0x03	1024	
Kp anello di posizione	3111-00	0x04	400	
Ki anello di posizione	3112-00	0x05	100	
Accelerazione imp/s^2	3108-00	0x06	500 (RD1xA-...T92-...) 1000 (RD1xA-...T48-...) 2500 (RD1xA-...T24-...) 5000 (RD1xA-...T12-...)	
Decelerazione imp/s^2	3109-00	0x07	500 (RD1xA-...T92-...) 1000 (RD1xA-...T48-...) 2500 (RD1xA-...T24-...) 5000 (RD1xA-...T12-...)	
Delta spazio positivo imp	310F-01	0x08-0x09	523263	
Delta spazio negativo imp	310F-02	0x0A-0x0B	523263	
Velocità jog imp/s	3114-00	0x0C	556 (RD1xA-...T92-...) 1066 (RD1xA-...T48-...) 2133 (RD1xA-...T24-...) 4266 (RD1xA-...T12-...)	
Velocità di lavoro imp/s	3115-00	0x0D	556 (RD1xA-...T92-...) 1066 (RD1xA-...T48-...) 2133 (RD1xA-...T24-...) 4266 (RD1xA-...T12-...)	
Durata corrente di stacco ms	3118-00	0x0E	2000	
Direzione conteggio	3302-00	0x0F	0	
Kp anello di corrente	3330-00	0x10	200	
Ki anello di corrente	3331-00	0x11	30	
Corrente massima mA	3340-00	0x12	2000	
Corrente di stacco mA	3341-00	0x13	4000	
Valore di preset imp	3300-00	0x16-0x17	0	
Cyclic Time ms	3110-00	-	0	
Rapporto di riduzione	3342-00	0x18	92 (RD1xA-...T92-...) 48 (RD1xA-...T48-...) 24 (RD1xA-...T24-...) 12 (RD1xA-...T12-...)	
Ampiezza passo jog imp	3117-00	0x19	100	

Interfaccia CANopen®

RD1A

RD12A

CANopen

Smart encoders & actuators

1 Interfaccia CANopen®

Le unità ROTADRIVE Lika sono dispositivi Slave e implementano l' "Application Layer and Communication Profile DS 301".
Per ogni informazione e specifica omessa fare riferimento al documento "CiA Draft Standard 301" disponibili sul sito www.can-cia.org.

1.1 File EDS

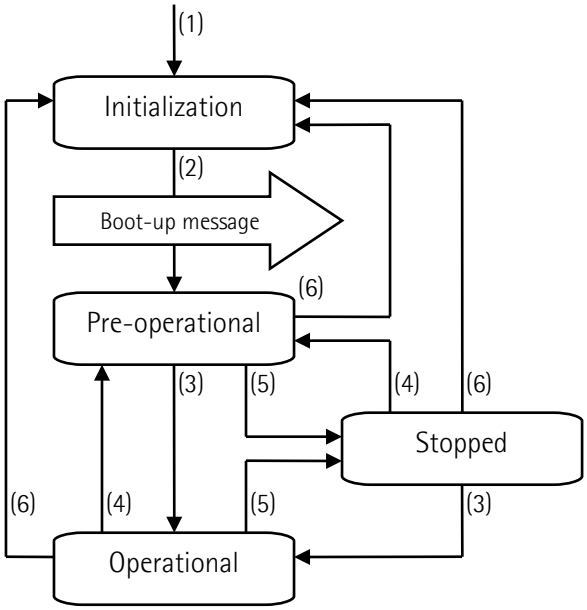
I dispositivi con interfaccia CANopen® sono forniti di un loro proprio file EDS **LIKA_RD12A_Tx_CB_Vx.EDS** (si veda il supporto informatico allegato oppure all'indirizzo **www.lika.it > ATTUATORI ROTATIVI > ATTUATORI ROTATIVI (DRIVECOD) > RD1A / RD12A**). Il file EDS deve essere installato sul dispositivo Master CANopen®.
Installare il file **LIKA_RD12A_T12_CB_Vx.EDS** per le unità con rapporto di riduzione T12.
Installare il file **LIKA_RD12A_T24_CB_Vx.EDS** per le unità con rapporto di riduzione T24.
Installare il file **LIKA_RD12A_T48_CB_Vx.EDS** per le unità con rapporto di riduzione T48.
Installare il file **LIKA_RD12A_T92_CB_Vx.EDS** per le unità con rapporto di riduzione T92.
I file sono disponibili sia in versione inglese (_en) che in versione italiana (_it).

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

Compatibilità	File EDS	EXE Modbus
	V1	fino a 2.1
	V2	fino a 2.4
	V3	da 2.5 a ...

1.2 Funzionamento a stati

I dispositivi CANopen® prevedono un funzionamento a stati, il passaggio da uno stato all'altro si effettua tramite l'invio al dispositivo di specifici messaggi NMT. Il diagramma degli stati è il seguente:



(1)	Accensione dispositivo
(2)	Inizializzazione dispositivo conclusa, invio automatico del messaggio di boot-up
(3)	Messaggio NMT: **Start remote node**
(4)	Messaggio NMT: **Enter pre-operational**
(5)	Messaggio NMT: **Stop remote node**
(6)	Messaggio NMT: **Reset node** o **Reset communication**

1.2.1 Initialization

E' il primo stato dopo l'accensione del dispositivo o dopo la ricezione di un comando **Reset node**. In questa fase il dispositivo è inizializzato e sono caricati i parametri salvati in EPROM. Alla fine dell'inizializzazione il dispositivo invia un messaggio di "boot-up" e passa automaticamente nello stato **Pre-operational**.

1.2.2 Pre-operational

In questo stato è possibile comunicare con lo Slave attraverso messaggi SDO, grazie ai quali si possono impostare i parametri di funzionamento del dispositivo. Il nodo Slave non è in grado di effettuare comunicazioni con messaggi PDO. Per portare il dispositivo nello stato **Operational** il Master deve inviare un messaggio di **Start remote node** attraverso un messaggio NMT.

1.2.3 Operational

In questo stato lo Slave è operativo, utilizza i valori dei parametri impostati nel "Dizionario oggetti" e può inviare i valori di processo tramite messaggi PDO. E' possibile comunicare attraverso messaggi SDO. Per portare il dispositivo nello stato **Pre-Operational** il Master deve inviare un messaggio di **Enter pre-operational** attraverso un messaggio NMT.

ATTENZIONE
Per motivi di sicurezza, in modalità **Operational** il Master dovrà controllare lo Slave in modo adeguato. Si veda al paragrafo "1.6 Messaggi PDO" a pagina 47 per la descrizione della corretta procedura.

1.2.4 Stopped

In questo stato lo Slave è forzato a interrompere la comunicazione con il Master (a eccezione del "Node guarding", se attivo). Non è possibile comunicare attraverso messaggi PDO e SDO. Per portare il dispositivo nello stato **Pre-Operational** o **Operational** il Master deve inviare il messaggio NMT specifico: **Enter pre-operational** o **Start remote node**.

1.3 Tipi di messaggi

Il modello CANopen prevede quattro tipi di messaggi di comunicazione tra i dispositivi:

- Messaggi amministrativi **NMT**: usati dal Master per la gestione dei nodi e della rete; si veda al paragrafo "1.4 Messaggi NMT" a pagina 46.
- Messaggi **Process Data Object PDO**: usati per trasmettere valori di processo in "real time"; si veda al paragrafo "1.6 Messaggi PDO" a pagina 47.
- Messaggi **Service Data Object SDO**: usati per accedere al "Dizionario oggetti" (pagina 57) di un dispositivo, per leggere o modificare i parametri in esso contenuti; si veda al paragrafo "1.7 Messaggi SDO" a pagina 56.
- Messaggi **Special Function Object**:
  - **SYNC**: messaggio di sincronismo usato dal Master per abilitare gli Slave a trasmettere i valori di processo (quota e velocità; si veda a pagina 58);
  - **Emergency**: messaggi di notifica errori, si veda a pagina 78;
  - **Nodeguard**: usato per conoscere lo stato di un dispositivo, si veda a pagina 79.

Uso dei messaggi per ogni stato di funzionamento:

	Initialization	Pre-operat.	Operational	Stopped
NMT		X	X	X
PDO			X	
SDO		X	X	
Sync.			X	
Emerg.		X	X	
Boot-up	X			
Nodeg.		X	X	X

1.3.1 COB-ID previsti o riconosciuti

Trasmissione Master → Slave		
Tipo di COB (Object)	Codice funzione (binario)	COB-ID (hex)
NMT	0000	000
SYNC	0001	080

Trasmissione peer-to-peer tra nodi		
EMERGENCY	0001	081 - 0FF
PDO 1 (tx)	0011	181 - 1FF
PDO 1 (rx)	0100	200 - 27F
SDO (tx)	1011	581 - 5FF
SDO (rx)	1100	601 - 67F
Nodeguard	1110	701 - 77F
Boot-up	1110	701 - 77F

Il "tipo di COB" è considerato trasmesso (tx) o ricevuto (rx) dal punto di vista del nodo Slave.

1.4 Messaggi NMT

Struttura messaggi NMT:

COB-ID (11 bit)	
Cod.Funz.	Node ID
0000	0

2 CAN Data Bytes	
Command	Slave ID
Funzione NMT	Slave ID

Per spedire un messaggio NMT a tutti gli Slave impostare: Slave ID = 00h.

Command	Funzione NMT	Stato del nodo
01 hex	Start remote node	Operational
02 hex	Stop remote node	Stopped
80 hex	Enter pre-operational	Pre-operational
81 hex	Reset node	Pre-operational
82 hex	Reset communication	Pre-operational

1.5 Messaggi di Boot-up

Struttura messaggi Boot-up:

COB-ID(hex)	1 CAN Data Byte
700+Node ID	00

1.6 Messaggi PDO

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.

Uno dei possibili metodi per monitorare la presenza di comunicazione consiste nell'implementazione del Node guarding protocol (secondo le specifiche DS301, si veda a pagina 79).

L'unità ROTADRIVE invia al Master i PDO secondo la modalità ciclica o sincrona impostata (si veda all'oggetto 1800h: **1800 Parametri PDO1 inviati**).

I messaggi PDO sono sempre composti da 8 CAN Data Bytes. Si badi però che la struttura dei messaggi ricevuti e inviati è diversa.

Struttura messaggi **PDO1 ricevuti** dal dispositivo (inviati dal Master):

IDENTIFIER		8 CAN data byte							
COB-ID(hex)		0	1	2	3	4	5	6	7
F.C.	Node-ID	Control Word				Target position			
200+Node ID		Low	...	...	High	Low	...	...	High

Struttura messaggi **PDO1 trasmessi** dal dispositivo (ricevuti dal Master):

IDENTIFIER		8 CAN data byte							
COB-ID(hex)		0	1	2	3	4	5	6	7
F.C.	Node-ID	**Status word**		**Velocità corrente**		**Posizione corrente**			
180+Node ID		Low	High	Low	High	Low	...	...	High

Struttura dei byte:

bit	7	6	5	4	3	2	1	0
	M.S.bit							L.S.bit

1.6.1 Messaggi "PDO1 ricevuti" inviati dal Master allo Slave

Control Word

Indice 0x3100-00. 32 bit. Contiene i comandi da inviare in tempo reale allo Slave per controllarlo. Si veda anche l'indice **3100-00 Control word** a pagina 74.

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 **3117-00 Ampiezza passo jog**; poi si arresta in attesa di un nuovo comando. La velocità, l'accelerazione e la decelerazione sono definiti dai dati macchina **3114-00 Velocità Jog**, **3108-00 Accelerazione** e **3109-00 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 **3117-00 Ampiezza passo jog**; poi si arresta in attesa di un

nuovo comando. La velocità, l'accelerazione e la decelerazione sono definiti dai dati macchina **3114-00 Velocità Jog**, **3108-00 Accelerazione** e **3109-00 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 **3109-00 Decelerazione**. Per un arresto immediato del movimento, utilizzare il bit 7 **Emergenza**.

Reset allarmi

bit 3

Impostando questo bit a "=1" si ristabilisce lo stato normale del dispositivo. Questo comando toglie lo Slave dalla condizione di allarme solo se non sono più presenti le condizioni che hanno causato l'errore. Utilizzando i messaggi SDO si può leggere il dettaglio allarme all'indice **1003 Campo errori predefinito**.

Si badi che, se l'allarme è relativo a dati macchina non validi (**Dati macchina non validi**), 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 **3117-00 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 "4.4.4 Pulsanti JOG + e JOG - (Figura 5)" a pagina 30) 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. Per la

descrizione del controllo di posizione si veda a pagina 33. Per ogni informazione sulla posizione di target si veda **Target position** a pagina 51.

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

bit 8 ... 11 — Non utilizzati.

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 CANopen®" a pagina 57.

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.

Byte 4 ... 7

Target position

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**.

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.

1.6.2 Messaggi "PDO1 trasmessi" inviati dallo Slave al Master

Status word

Indice 0x3101-00. 16 bit. In questi byte è indicato lo stato del controllore PI nella modalità **Operational**. Si veda anche l'indice **3101-00 Status word** a pagina 74.

Byte 0

Asse in posizione

bit 0 — Se "=1" il dispositivo ha raggiunto la posizione programmata per il tempo definito all'oggetto **310E-00 Tempo asse in tolleranza**. Rimane attivo fino a quando l'errore di posizione è inferiore a **310D-00 Tolleranza di posizione**.

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 **310F Delta spazio** 310Fhex sub 1 **Delta positivo**.

Finecorsa SW -

bit 4 — Se "=1" il dispositivo ha raggiunto la posizione programmata come finecorsa negativa. Si veda il dato macchina **310F Delta spazio** 310Fhex sub 2 **Delta negativo**.

Allarme

bit 5 — Se "=1" si è verificato un allarme, si vedano i dettagli all'indice **1003 Campo errori predefinito** e **3200-00 Allarmi_scheda**.

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**. Rimane attivo fino ai successivi comandi **Target position** 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 CANopen®" a pagina 57.

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 CANopen®" a pagina 57.

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 CANopen®" a pagina 57.

IN 3 = 0 ingresso 3 basso (non attivo)

IN 3 = 1 ingresso 3 alto (attivo)

Byte 2 e 3

Velocità corrente

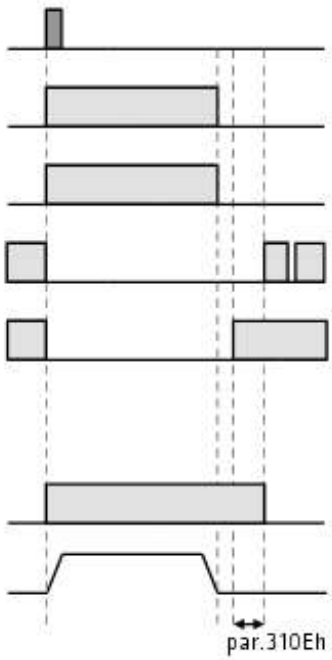
Velocità del dispositivo espressa in impulsi/secondo [imp/s], aggiornata ogni secondo.

Byte 4 ... 7

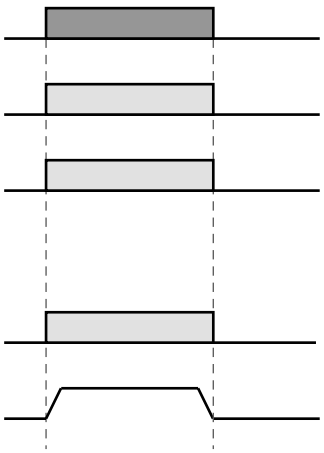
Posizione corrente

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

Esempio 1

Posizionamento	RD1xA
Start (bit 6 - Control Word) Comando in corso (bit 7 - Status word) Asse in movimento (bit 6 - Status word) Asse in posizione (bit 0 - Status word) Target raggiunto (bit 8 - Status word) Asse abilitato (bit 2 - Status word: PWM attivo) Velocità teorica	

Esempio 2

Jog +	RD1xA
Jog + (bit 0 - Control Word) Comando in corso (bit 7 - Status word) Asse in movimento (bit 6 - Status word) Asse abilitato (bit 2 - Status word: PWM attivo) Velocità teorica	

1.7 Messaggi SDO

I messaggi SDO sono utilizzati per conoscere o modificare i parametri del dispositivo; tali parametri sono contenuti nel "Dizionario oggetti" (pagina 57). Il numero di byte utilizzato per i dati è di 4 al massimo, altri 4 byte sono utilizzati per Command, Index e Subindex. Gli SDO sono sempre seguiti da conferma: ciò significa che per qualsiasi SDO inviato dal Master allo Slave, quest'ultimo risponde con un SDO adeguato.

Struttura messaggio SDO:

IDENTIFIER		da 4 a 8 CAN data byte							
COB-ID(hex)		0	1	2	3	4	5	6	7
F.C.	Node-ID	Com	Index		Sub	Data			
		1 byte	LSB	MSB	1 byte	LSB	...	...	MSB

Com command: comando
Index indice del parametro
Sub sub-index: secondo indice del parametro
Data valore letto o scritto del parametro

1.7.1 Command

Il Command byte specifica il tipo di COB inviato nella rete CAN.
I principali COB sono:

- **Set**: usato per inviare dati di configurazione;
- **Req**: usato per richiedere dati di configurazione;
- **Warning**: usato dallo Slave per notificare anomalie negli SDO inviati dal Master (es. "Indice inesistente", "Valore process data non valido", ...).

Command	COB	Tipo COB	Nr. byte Data
22h	Set	richiesta M → S	non spec.
23h	Set	richiesta M → S	4 byte
2Bh	Set	richiesta M → S	2 byte
2Fh	Set	richiesta M → S	1 byte
60h	Set	conferma S → M	0 byte
40h	Req	richiesta M → S	0 byte
42h	Req	risposta S → M	non spec.
43h	Req	risposta S → M	4 byte
4Bh	Req	risposta S → M	2 byte
4Fh	Req	risposta S → M	1 byte
41h	Req	risposta S → M	SDO concatenati
80h	Warning	risposta S → M	4 byte

2 Parametri di programmazione CANopen®

2.1 Dizionario oggetti

Di seguito sono riportati gli oggetti implementati nel dispositivo, per ognuno è indicato:

Index-subindex Nome oggetto
[tipo var, attributo]

- Index e subindex sono espressi in notazione esadecimale.
- Attributo:
  ro = oggetto accessibile in sola lettura
  rw = oggetto accessibile in lettura e scrittura
  const = ro + valore costante

Struttura oggetti Unsigned16:

Data byte	
byte 4	byte 5
LSByte	MSByte

Struttura oggetti Unsigned32:

Data byte			
byte 4	byte 5	byte 6	byte 7
LSByte	...	...	MSByte

2.1.1 Oggetti standard (DS 301)

1000-00 Tipo di dispositivo
[Unsigned32, const]
Default = 0000012Dh

1001-00 Registro errori
[Unsigned8, ro]
Per ogni situazione di errore: bit 0 = "1"
Default = 00h

1003 Campo errori predefinito

Mantiene memoria degli ultimi cinque errori che hanno generato un messaggio Emergency (EMCY).

- **00** Numero errori presenti [Unsigned8, rw]
  Per cancellare la cronologia degli errori scrivere 00h.
- **01** Ultimo errore verificatosi [Unsigned32, ro]
- **02...05** Errori precedenti [Unsigned32, ro]

1005-00 COB-ID messaggi Sync

[Unsigned32, rw]
Default = 0000 0080h

1008-00 Nome del dispositivo

[String, const]
Contiene il nome del dispositivo.
Default = "RD1A" o "RD12A"

1009-00 Versione hardware

[String, const]
Contiene la versione hardware del dispositivo.

100A-00 Versione software

[String, const]
Contiene la versione software del dispositivo.

100C-00 Guard Time

[Unsigned16, rw]
Contiene il valore di "node guard time" espresso in msec (millisecondi). Il Master interroga ogni Slave a intervalli di tempo regolari attraverso messaggi RTR. Questo intervallo di tempo è chiamato "node guard time" e può essere differenziato per ogni Slave. Per maggiori dettagli si veda il paragrafo "2.4 Node guarding protocol" a pagina 79.
Default = 0

100D-00 Life Time Factor

[Unsigned8, rw]
I parametri 100C-00 Guard Time e 100D-00 Life Time Factor sono utilizzati nel "Node guarding protocol" gestito dal Master. Per maggiori dettagli si veda il paragrafo "2.4 Node guarding protocol" a pagina 79.
Default = 0

1010-01 Salvataggio parametri

[Unsigned32, rw]
Oggetto utilizzato per eseguire il salvataggio di tutti i parametri nella memoria non volatile. Scrivere **save** (in codifica ASCII esadecimale) nei data byte:

Master → Slave

COB-ID	Cmd	Index		Sub	Data bytes			
600+ID	23	10	10	01	73	61	76	65

Slave → Master (conferma)

COB-ID	Cmd	Index		Sub	Data bytes			
580+ID	60	10	10	01	00	00	00	00

1011-01 Parametri di default

[Unsigned32, rw]
Oggetto utilizzato per caricare tutti i parametri di default.
Scrivere **load** (in codifica ASCII esadecimale) nei data byte e successivamente eseguire un comando Reset node:

Master → Slave

COB-ID	Cmd	Index		Sub	Data bytes			
600+ID	23	11	10	01	6C	6F	61	64

Slave → Master (conferma)

COB-ID	Cmd	Index		Sub	Data bytes			
580+ID	60	11	10	01	00	00	00	00

Master → Slave (reset node)

COB-ID	Cmd	Slave ID
000	81	ID

Slave → Master (Boot-up)

COB-ID	Cmd
700+ID	00

NOTA
Per conservare i parametri di default impostati, eseguire **1010-01 Salvataggio parametri**.

1014-00 COB-ID EMCY

[Unsigned32, rw]
Questo oggetto definisce il COB-ID usato dal dispositivo per i messaggi Emergency (EMCY).
All'accensione questo oggetto assume sempre il valore di default.
Per una lista dei messaggi di emergenza riferirsi al paragrafo "2.3 Messaggi di emergenza (Emergency)" a pagina 78.
Default = 80h+NodeID

1015-00 Inhibit time EMCY

[Unsigned16, rw]
Inhibit time dei messaggi Emergency (EMCY), cioè intervallo minimo tra i messaggi Emergency espresso in multipli di 100 µs.
Default = 0

1018 Informazioni di identificazione

- **01** Identificativo del costruttore [Unsigned32, ro]
- **02** Codice prodotto [Unsigned32, ro]
- **03** Numero revisione [Unsigned32, ro]
- **04** Numero di serie [Unsigned32, ro]

1400 Parametri PDO1 ricevuti

Questi oggetti contengono i parametri di comunicazione dei PDO ricevuti.

- **01** COB-ID usato da PDO1 [Unsigned32, rw]

Numero bit	Valore	Significato
31 (MSB)	0	PDO esistente / valido
	1	PDO non esistente / non valido
30	0	RTR permesso nel PDO **(non implementato)**
	1	nessun RTR permesso nel PDO
29	0	ID a 11 bit (CAN 2.0A)
	1	ID a 29 bit (CAN 2.0B)
28 ... 11	0	se il bit 29 = 0
	X	Se il bit 29 = 1: bit 28-11 del COB-ID a 29 bit
10 ... 0 (LSB)	X	bit 10-0 del COB-ID

Default = 4000 0200h+NodeID (no RTR, COB-ID)

ATTENZIONE
Bisogna sempre impostare il bit 30 del COB-ID a 1 (il valore 0 non è ammesso). Questo significa che "nessun RTR è permesso nel PDO".
All'accensione questo oggetto assume sempre il valore di default.

- **02** Tipo di trasmissione [Unsigned8, rw]

Tipo di trasmissione	Trasmissione PDO	
00h (0)	Aciclica, sincrona	**non implementata**
01h ... F0h (1 ... 240)	Ciclica, sincrona	**implementata**
F1h ... FBh (241 ... 251)	**non implementata - riservata**	
FCh (252)	Sincrona, solo RTR	**non implementata**
FDh (253)	Asincrona, solo RTR	**non implementata**
FEh (254)	Asincrona, specifica del costruttore	**implementata**
FFh (255)	Asincrona, specifica del device profile	**non implementata**

Default = FEh (event-driven, asincrona, specifica del costruttore)

ATTENZIONE

Nel caso in cui si imposti un **Tipo di trasmissione** = 0 il messaggio PDO non viene inviato; negli altri casi di **Tipo di trasmissione** non implementato, viene generato un messaggio di errore (codice errore = 0609 0030h: **Range di valori del parametro superato**).

NOTA

- Prima di procedere alla modifica del valore del **COB-ID** è necessario disattivare preventivamente la ricezione del PDO1, quindi inserire il nuovo valore, infine attivare nuovamente la ricezione del PDO1.
- La ricezione del PDO1 può essere attivata (disattivata) impostando a "0" ("1") il bit più significativo dell'oggetto **1400 Parametri PDO1 ricevuti** sub 1 **COB-ID**.

1600 Mappatura PDO1 ricevuti

In questo oggetto sono indicati quali parametri sono inseriti nei PDO ricevuti.

- **01** **3100-00 Control word** [Unsigned32, ro] Default = 3100 0020h.
- **02** **3106-00 Target position** [Unsigned32, ro] Default = 3106 0020h.

1800 Parametri PDO1 inviati

Questi oggetti contengono i parametri di comunicazione dei PDO inviati.

- **01** **COB-ID** usato da PDO1 [Unsigned32, rw]

Numero bit	Valore	Significato
31 (MSB)	0	PDO esistente / valido
	1	PDO non esistente / non valido
30	0	RTR permesso nel PDO **(non implementato)**
	1	nessun RTR permesso nel PDO
29	0	ID a 11 bit (CAN 2.0A)
	1	ID a 29 bit (CAN 2.0B)
28 ... 11	0	se il bit 29 = 0
	X	Se il bit 29 = 1: bit 28-11 del COB-ID a 29 bit
10 ... 0 (LSB)	X	bit 10-0 del COB-ID

Default = 4000 0180h+NodeID (no RTR, COB-ID)

ATTENZIONE
Bisogna sempre impostare il bit 30 del COB-ID a 1 (il valore 0 non è ammesso). Questo significa che "nessun RTR è permesso nel PDO".
All'accensione questo oggetto assume sempre il valore di default.

- **02** Tipo di trasmissione [Unsigned8, rw]

Tipo di trasmissione	Trasmissione PDO	
00h (0)	Aciclica, sincrona	**non implementata**
01h ... F0h (1 ... 240)	Ciclica, sincrona	**implementata**
F1h ... FBh (241 ... 251)	**non implementata - riservata**	
FCh (252)	Sincrona, solo RTR	**non implementata**
FDh (253)	Asincrona, solo RTR	**non implementata**
FEh (254)	Asincrona, specifica del costruttore	**implementata**
FFh (255)	Asincrona, specifica del device profile	**non implementata**

Default = FEh (trasmissione "al cambio di stato, posizione, velocità" o trasmissione ciclica, vedi qui di seguito e oggetto 3110-00 Cyclic Time)

ATTENZIONE
Nel caso in cui si imposti un Tipo di trasmissione = 0 il messaggio PDO non viene inviato; negli altri casi di Tipo di trasmissione non implementato, viene generato un messaggio di errore (codice errore = 0609 0030h: Range di valori del parametro superato).
Se il valore dell'oggetto 3110-00 Cyclic Time ≠ 0, il messaggio PDO è trasmesso ciclicamente con intervallo impostato nell'oggetto 3110-00 Cyclic Time; se invece il valore dell'oggetto 3110-00 Cyclic Time = 0, il messaggio PDO è inviato solamente a una variazione di stato (oggetto 3101-00 Status word) e/o di posizione (oggetto 3103-00 Posizione attuale) e/o di velocità (oggetto 3105-00 Velocità attuale); in questo caso il tempo di pausa minimo tra due trasmissioni di PDO è impostato al successivo parametro Inhibit time oggetto 1800h sub 3.

- **03** Inhibit time [Unsigned16, rw]
  Tempo minimo tra due trasmissioni di PDO nella modalità "Cambio di stato, posizione, velocità" (vedi il parametro Tipo di trasmissione

oggetto 1800h sub 2 e la NOTA qui a seguire), è espresso in multipli di 100 µs.
Default = 0190h (40ms)

NOTA

- Prima di procedere alla modifica del valore del **COB-ID** è necessario disattivare preventivamente la trasmissione del PDO1, quindi inserire il nuovo valore, infine attivare nuovamente la trasmissione del PDO1.
- La trasmissione del PDO1 può essere attivata (disattivata) impostando a "0" ("1") il bit più significativo dell'oggetto **1800 Parametri PDO1 inviati** sub 1 **COB-ID**.
- La modalità di trasmissione ciclica o sincrona può essere modificata impostando opportunamente il valore dell'oggetto **1800 Parametri PDO1 inviati** sub 2 **Tipo di trasmissione**. Per ottenere la trasmissione del PDO1 ogni "n" SYNC, impostare il valore "n" nell'oggetto 1800h, sub 2.
  01h = trasmissione sincrona ogni SYNC
  02h = trasmissione sincrona ogni 2 SYNC
  ...
  FEh = "trasmissione ciclica" o "trasmissione al cambio di stato, posizione, velocità":
  se **3110-00 Cyclic Time** ≠ 0 → "trasmissione ciclica": il tempo di ciclo è specificato nell'oggetto 3110h;
  se **3110-00 Cyclic Time** = 0 → "trasmissione al cambio di stato, posizione, velocità": il dispositivo invia un PDO a ogni variazione dei parametri mappati nel PDO (si veda all'oggetto **1A00 Mappatura PDO1 inviati**) con periodo non inferiore a **1800 Parametri PDO1 inviati** sub 3 **Inhibit time**.

1A00 Mappatura PDO1 inviati

In questo oggetto sono indicati quali parametri sono inseriti nei PDO inviati.

- **01** **3101-00 Status word** [Unsigned32, ro] Default = 3101 0010h.
- **02** **3105-00 Velocità attuale** [Unsigned32, ro] Default = 3105 0010h.
- **03** **3103-00 Posizione attuale** [Unsigned32, ro] Default = 3103 0020h.

NOTA

Per salvare i parametri modificati eseguire **1010-01 Salvataggio parametri**.
Nel caso di spegnimento del dispositivo o di invio dei comandi **Reset node** o **Reset communication** i dati non salvati andranno persi!

2.1.2 Oggetti specifici del costruttore

Parametri di comunicazione

3000-00 Velocità trasmissione dati

[Unsigned8, ro]

Questo oggetto visualizza il valore della velocità di trasmissione dei dati impostata tramite il selettore dedicato, secondo la seguente tabella; per ogni informazione sull'impostazione del baud rate del dispositivo riferirsi al paragrafo "4.4.2 Velocità di trasmissione dei dati: Baud rate (Figura 5)" a pagina 29.

Switch 1... 4	Baud rate
0000	20 Kbit/s
1000	50 Kbit/s
0100	100 Kbit/s
1100	125 Kbit/s
0010	250 Kbit/s
1010 (default)	500 Kbit/s
0110	800 Kbit/s
1110	1000 Kbit/s

3001-00 Node ID

[Unsigned8, ro]

Questo oggetto visualizza il numero di nodo del dispositivo impostato tramite il selettore dedicato; per ogni informazione sull'impostazione dell'indirizzo del nodo riferirsi al paragrafo "4.4.1 Indirizzo nodo: Node ID (Figura 5)" a pagina 28. L'indirizzo di default è 1.

Parametri Dati macchina

3108-00 Accelerazione

[Unsigned32, rw]
Questo oggetto definisce il valore di accelerazione usato dal dispositivo. Il parametro è espresso in [impulsi/sec^2].
Default = 5000 per RD1xA-...-T12-...
Default = 2500 per RD1xA-...-T24-...
Default = 1000 per RD1xA-...-T48-...
Default = 500 per RD1xA-...-T92-...

3109-00 Decelerazione

[Unsigned32, rw]
Questo oggetto definisce il valore di decelerazione usato dal dispositivo. Il parametro è espresso in [impulsi/sec^2].
Default = 5000 per RD1xA-...-T12-...
Default = 2500 per RD1xA-...-T24-...
Default = 1000 per RD1xA-...-T48-...
Default = 500 per RD1xA-...-T92-...

310C-00 Max errore di inseguimento

[Unsigned32, rw]
Questo oggetto 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 blocca il proprio movimento. Il parametro è espresso in impulsi.
Default = 1024

310D-00 Tolleranza di posizione

[Unsigned16, rw]
Questo oggetto definisce la finestra di tolleranza da applicare al valore di **3106-00 Target position**. Se l'asse è all'interno di questa finestra per il tempo impostato nell'oggetto **310E-00 Tempo asse in tolleranza**, allora è segnalato lo stato mediante il bit **Asse in posizione**. Il parametro è espresso in impulsi.
Default = 0

310E-00 Tempo asse in tolleranza

[Unsigned16, rw]
Rappresenta il tempo di assestamento dell'asse all'interno della finestra di tolleranza definita in **310D-00 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

310F Delta spazio

- **01 Delta positivo** [Integer32, 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 + = **3300-00 Valore di preset** + **310F Delta spazio** sub 1 **Delta positivo**.
  Default = 523263
- **02 Delta negativo** [Integer32, 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 - = **3300-00 Valore di preset** - **310F Delta spazio** sub 2 **Delta negativo**.
  Default = 523263

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

ESEMPIO
Con **3120-00 Distanza_giro** = 1024 e **3300-00 Valore di preset** = 0, il valore massimo ammesso in **310F Delta spazio** (sub 1 **Delta positivo** e sub 2 **Delta negativo**) sarà:
(1024 imp./giro * 511 giri) -1 = 523263
Con **3120-00 Distanza_giro** = 256 e **3300-00 Valore di preset** = 0, il valore massimo ammesso in **310F Delta spazio** (sub 1 **Delta positivo** e sub 2 **Delta negativo**) sarà:
(256 imp./giro * 511 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 **3120-00 Distanza_giro** e **3300-00 Valore di preset**. Dopo ogni modifica del parametro **3120-00 Distanza_giro** si deve reimpostare anche **3300-00 Valore di preset** in modo da definire lo zero asse in quanto il sistema di riferimento è variato. Dopo la modifica del parametro **3300-00 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 **310F Delta spazio** sub 1 **Delta positivo** e **310F Delta spazio** sub 2 **Delta negativo** impostati. Per una descrizione dettagliata si veda a pagina 36.

3111-00 Kp anello di posizione
[Unsigned32, rw]
Questo oggetto 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

3112-00 Ki anello di posizione
[Unsigned32, rw]
Questo oggetto 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

3114-00 Velocità Jog
[Unsigned32, rw]
Questo oggetto definisce la velocità massima del dispositivo nell'utilizzo con le funzioni **Jog +** e **Jog -**. Il parametro è espresso in impulsi/secondo.
Default = 4266 per RD1xA-...-T12-...
Default = 2133 per RD1xA-...-T24-...
Default = 1066 per RD1xA-...-T48-...
Default = 556 per RD1xA-...-T92-...

3115-00 Velocità di lavoro

[Unsigned32, rw]
Questo oggetto definisce la velocità massima del dispositivo nell'utilizzo in modalità automatica (movimenti controllati con **Start** e **Target position**). Il parametro è espresso in impulsi/secondo.
Default = 4266 per RD1xA-...-T12-...
Default = 2133 per RD1xA-...-T24-...
Default = 1066 per RD1xA-...-T48-...
Default = 556 per RD1xA-...-T92-...

ATTENZIONE
A ogni modifica del parametro **3120-00 Distanza_giro** si devono poi reimpostare **3114-00 Velocità Jog** e **3115-00 Velocità di lavoro** in quanto le velocità sono espresse in impulsi al secondo. Nel calcolo delle velocità si deve sempre rispettare la seguente relazione:

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

Per una spiegazione dettagliata si veda a pagina 36.

3116-00 Velocità massima

[Unsigned32, ro]
Questo oggetto visualizza la velocità massima impostabile negli oggetti **3114-00 Velocità Jog** e **3115-00 Velocità di lavoro**. Il parametro è espresso in impulsi/secondo.
Default = 4266 per RD1xA-...-T12-...
Default = 2133 per RD1xA-...-T24-...
Default = 1066 per RD1xA-...-T48-...
Default = 556 per RD1xA-...-T92-...

3117-00 Ampiezza passo jog

[Unsigned32, rw]
Se è abilitato il controllo del jog a passo (bit 4 **Jog incrementale** in **Control Word** = 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

3118-00 Durata corrente di stacco

[Unsigned32, rw]
Questo oggetto definisce il tempo massimo per il quale il motore è controllato con la corrente di stacco durante la partenza (si veda all'oggetto **3341-00 Corrente di stacco**). Il parametro è espresso in millisecondi. Il valore massimo consentito è di 5 secondi.
Default = 2000

3120-00 Distanza_giro

[Unsigned32, rw]
Questo oggetto 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 con passo 5 mm, impostando **3120-00 Distanza_giro** = 500 si ottiene che a ogni giro dell'asse il sistema trasla di 5 mm con una risoluzione al centesimo di millimetro. Il valore massimo è 1024.
Default = 1024

ATTENZIONE
Dopo la modifica di questo parametro è necessario reimpostare i parametri **3114-00 Velocità Jog**, **3115-00 Velocità di lavoro** e **3300-00 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 **3120-00 Distanza_giro**; di conseguenza la modifica del valore nel parametro **3120-00 Distanza_giro** comporta necessariamente una ridefinizione dei valori da essi espressi. I parametri sono: **3108-00 Accelerazione**, **3109-00 Decelerazione**, **310C-00 Max errore di inseguimento**, **310D-00 Tolleranza di posizione**, **310F Delta spazio**, **3116-00 Velocità massima**, **3343-00 Finecorsa assoluto positivo**, **3344-00 Finecorsa assoluto negativo**, **3103-00 Posizione attuale**, **3105-00 Velocità attuale**, **3106-00 Target position** e **3107-00 Velocità target**. Si veda per esempio la relazione che intercorre tra **3120-00 Distanza_giro** e i valori di velocità, illustrata a pagina 69.

NOTA
Se **3120-00 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.

3300-00 Valore di preset

[Integer32, rw]
Usare questo oggetto per assegnare e impostare 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'invio del parametro.
Default = 0

ATTENZIONE
Il parametro **3300-00 Valore di preset** deve essere reimpostato ogniqualvolta viene modificato il parametro **3120-00 Distanza_giro**. Dopo la modifica del parametro **3300-00 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 **310F Delta spazio** sub 1 **Delta positivo** e **310F Delta spazio** sub 2 **Delta negativo** impostati. Per una descrizione dettagliata si veda a pagina 36.

3301-00 Valore di offset

[Integer32, ro]
Questo oggetto definisce la differenza tra la posizione trasmessa dal dispositivo e la posizione reale: posizione reale – preset. Il valore è espresso in impulsi.

3302-00 Direzione conteggio

[Boolean, 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 **3300-00 Valore di preset** e verificare i valori nel parametro **310F Delta spazio**.

3330-00 Kp anello di corrente

[Unsigned32, rw]
Questo oggetto 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

3331-00 Ki anello di corrente

[Unsigned32, rw]
Questo oggetto 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

3340-00 Corrente massima

[Unsigned32, rw]
Questo oggetto 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 oggetto non può essere superiore al valore impostato in **3341-00 Corrente di stacco**). Il valore massimo è 2000.
Default = 2000

3341-00 Corrente di stacco

[Unsigned32, rw]
Questo oggetto definisce la corrente massima erogata solo alla partenza e per un tempo massimo definito all'oggetto **3118-00 Durata corrente di stacco**. Il parametro è espresso in mA (milliampere). Il valore massimo è 4000.
Default = 4000

3342-00 Rapporto di riduzione

[Unsigned32, ro]
E' 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 RD1xA-...-T12-...
Default = 24 per RD1xA-...-T24-...
Default = 48 per RD1xA-...-T48-...
Default = 92 per RD1xA-...-T92-...

3343-00 Finecorsa assoluto positivo

[Integer32, ro]
E' il valore di limite massimo positivo calcolato in base ai valori dei parametri **3300-00 Valore di preset** e **310F Delta spazio** sub 1 **Delta positivo**. Se si raggiunge il massimo spostamento in avanti viene attivata la segnalazione sul bit di stato **Finecorsa SW +**.
Finecorsa SW + = **3300-00 Valore di preset** + **310F Delta spazio** sub 1 **Delta positivo**.
Il valore è espresso in impulsi encoder.
Default = 523263

3344-00 Finecorsa assoluto negativo

[Integer32, ro]
E' il valore di limite massimo negativo calcolato in base ai valori dei parametri **3300-00 Valore di preset** e **310F Delta spazio** sub 2 **Delta negativo**. Se si raggiunge il massimo spostamento all'indietro viene attivata la segnalazione sul bit di stato **Finecorsa SW -**.
Finecorsa SW - = **3300-00 Valore di preset** - **310F Delta spazio** sub 2 **Delta negativo**.
Il valore è espresso in impulsi encoder.
Default = -523263

NOTA
Per salvare i parametri modificati eseguire **1010-01 Salvataggio parametri**.
Nel caso di spegnimento del dispositivo o di invio dei comandi **Reset node** o **Reset communication** i dati non salvati andranno persi!

Parametri dati operativi

3005-00 Corrente attuale

[Integer16, ro]
Questo oggetto rappresenta il valore della corrente assorbita dal motore. Il parametro è espresso in mA (milliampere).

3006-00 Temperatura

[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.

3100-00 Control word

[Unsigned32, rw]
Questo oggetto contiene i comandi da inviare in tempo reale allo Slave per controllarlo. Il parametro **3100-00 Control word** è utilizzato nella composizione dei messaggi PDO ricevuti dallo Slave (si vedano i dettagli al paragrafo "1.6 Messaggi PDO" a pagina 47).

3101-00 Status word

[Unsigned16, ro]
Questo oggetto contiene le informazioni relative allo stato del dispositivo. Il parametro **3101-00 Status word** è parte dei messaggi PDO trasmessi dallo Slave (si vedano i dettagli al paragrafo "1.6 Messaggi PDO" a pagina 47).

3102-00 Posizione richiesta

[Integer32, ro]
Questo oggetto contiene il valore della posizione teorica calcolata dal dispositivo durante il movimento. Il valore di questo parametro è utilizzato dal PI nel controllo motore.

3103-00 Posizione attuale

[Integer32, ro]
Questo oggetto contiene il valore attuale della posizione. Il parametro **3103-00 Posizione attuale** è parte dei messaggi PDO trasmessi dallo Slave (si vedano i dettagli al paragrafo "1.6 Messaggi PDO" a pagina 47).

3105-00 Velocità attuale

[Integer16, ro]

Questo oggetto contiene il valore attuale della velocità. Il parametro **3105-00 Velocità attuale** è parte dei messaggi PDO trasmessi dallo Slave (si vedano i dettagli al paragrafo "1.6 Messaggi PDO" a pagina 47). Il parametro è espresso in impulsi/secondo.

3106-00 Target position

[Integer32, rw]

Questo oggetto definisce la posizione di arrivo programmata. Il parametro **3106-00 Target position** è usato nella composizione dei messaggi PDO ricevuti dallo Slave (si vedano i dettagli al paragrafo "1.6 Messaggi PDO" a pagina 47).

3107-00 Velocità target

[Integer32, rw]

Questo oggetto contiene il valore della velocità teorica utilizzata dal dispositivo per la generazione della traiettoria di posizione. Il parametro è espresso in impulsi/secondo.

310B-00 Errore di posizione

[Integer32, ro]

Questo oggetto contiene la differenza tra la posizione richiesta e la posizione attuale istante per istante. Se questo valore supera il dato macchina **310C-00 Max errore di inseguimento**, il dispositivo genera l'allarme **Errore di inseguimento** e interrompe il proprio movimento.

3110-00 Cyclic Time

[Unsigned16, rw]

3110-00 Cyclic Time è usato in modalità operativa asincrona e specifica il periodo che intercorre tra una trasmissione di PDO e quella successiva.

Se il valore in questo oggetto ≠ 0, il messaggio PDO è trasmesso ciclicamente con l'intervallo impostato; se invece il valore in questo oggetto = 0, il messaggio PDO è inviato solamente a una variazione di stato (oggetto **3101-00 Status word**) e/o di posizione (oggetto **3103-00 Posizione attuale**) e/o di velocità (oggetto **3105-00 Velocità attuale**); in questo caso il tempo di pausa minimo tra due trasmissioni di PDO è impostato al parametro **Inhibit time** oggetto 1800h sub 3.

Questo parametro è relativo alla trasmissione di PDO dello Slave e non del Master. Il parametro è espresso in millisecondi. Si veda anche all'oggetto **1800 Parametri PDO1 inviati** a pagina 62.
Default = 0

3200-00 Allarmi_scheda

[Unsigned16, ro]
Informa sugli allarmi presenti nel dispositivo.
Struttura byte allarmi:

byte	Data byte 4			Data byte 5		
bit	7	...	0	15	...	8
			L.S.bit	M.S.bit		

Codice errore allarme previsti:

bit 0 : 0001h	**Dati macchina non validi**	Uno o più parametri non sono validi, impostare valori corretti per ristabilire la normale condizione di lavoro.
bit 1: 0002h	**Errore memoria flash**	Errore interno non ripristinabile.
bit 2	Non utilizzato	
bit 3: 0004h	**Errore di inseguimento**	La differenza tra la posizione reale e quella teorica è superiore al valore del parametro **310C-00 Max errore di inseguimento**; si consiglia di ridurre la velocità di lavoro.
bit 4: 0008h	**Asse non sincronizzato**	Errore interno non ripristinabile.
bit 5: 0010h	**Target non valido**	Posizione comandata oltre i finecorsa.
bit 6: 0020h	**Emergenza**	Il bit 7 **Emergenza** in **Control Word** è stato forzato a 0; oppure sono attivi allarmi nel dispositivo.
bit 7: 0040h	**Sovracorrente**	Il valore della corrente di alimentazione è superiore al massimo consentito.

bit 8: 0080h	**Sovratemperatura**	Superamento della temperatura interna accettabile rilevata per mezzo di una sonda (si veda l'oggetto **3006-00 Temperatura**).
bit 9	Non utilizzato	
bit 10: 0400h	**Sottotensione**	Il valore della tensione di alimentazione è inferiore al minimo consentito.
bit 11: 0800h	**CAN Life guard error**	Errore nel "Node guarding protocol". Si veda la sezione "2.4 Node guarding protocol" a pagina 79.
bit 12 ... 15	Non utilizzati	

Per ripristinare una condizione di errore utilizzare il bit **Reset allarmi**. Impostando questo bit a "1" si ristabilisce lo stato normale del dispositivo. Questo comando toglie lo Slave dalla condizione di allarme solo se non sono più presenti le condizioni che hanno causato l'errore. Utilizzando i messaggi SDO si può leggere il dettaglio allarme all'indice **1003 Campo errori predefinito**.

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

NOTA

Per salvare i parametri modificati eseguire **1010-01 Salvataggio parametri**.

Nel caso di spegnimento del dispositivo o di invio dei comandi **Reset node** o **Reset communication** i dati non salvati andranno persi!

2.2 Messaggi di avvertenza (Warning)

Per ogni informazione sul significato dei messaggi di avvertenza (warning) fare riferimento al documento "CiA Draft Standard 301" alla sezione "SDO abort codes" disponibile sul sito www.can-cia.org.

2.3 Messaggi di emergenza (Emergency)

I messaggi di emergenza (emergency EMCY) sono trasmessi dal dispositivo per segnalare situazioni di errore interno.

Struttura messaggio EMCY:

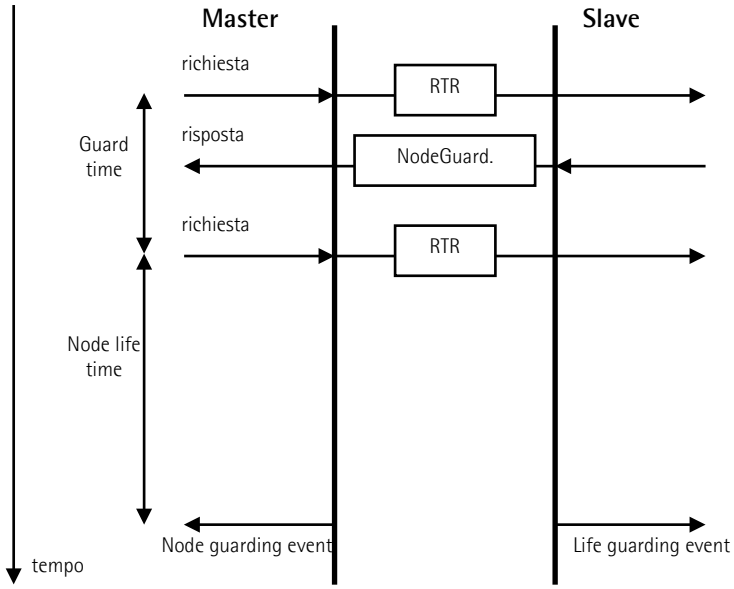
IDENTIFIER	CAN Data Byte			
COB-ID(hex)	0	1	2	3...7
vedi oggetto 1014h 1014-00 COB-ID EMCY	Codice errore		Sub registro errori	Codici specifici
	LSB	MSB	01	00...00

Codici errore emergenza previsti:

0000h	Nessun errore
1000h	Errore generico
2220h	Sovracorrente
3110h	Sovratensione
3120h	Sottotensione
4310h	Sovratemperatura
5530h	Memoria flash
8130h	Life Guard
8611h	Inseguimento

2.4 Node guarding protocol

All'accensione il "Node guarding protocol" non è attivo; tale protocollo si attiva automaticamente alla prima richiesta RTR (Remote Transmit Request) da parte del dispositivo Master.



100C-00 Guard Time: tempo previsto tra due RTR (si veda alla pagina 58).
Node life time: tempo entro il quale lo Slave deve ricevere un ulteriore RTR dal Master.

"**Node life time**" = "100C-00 Guard Time" * "100D-00 Life Time Factor".

Il "Node guarding" è abilitato solo se "Node life time" ≠ 0.
Se lo Slave non riceve un messaggio RTR entro il tempo "Node life time" viene segnalato un "Life Guarding Event" con relativa segnalazione LED "Node guarding error" (si veda a pagina 24), aggiornamento degli oggetti 1001-00 Registro errori e 1003 Campo errori predefinito e invio del messaggio di errore.
Per resettare l'errore è sufficiente eseguire un Reset node.

3 Esempi di programmazione CANopen®

Di seguito sono riportati esempi di lettura e impostazione di alcuni parametri. Negli esempi si considera l'indirizzo dell'encoder pari a "ID", il Master ha sempre indirizzo 0.
Tutti i valori sono espressi in notazione esadecimale.

Impostazione stati Operational, Pre-operational

Messaggio NMT — Master → Slave

	COB-ID	Cmd	Nodo
Operational	000	01	ID
Pre-operational	000	80	ID

Impostazione di 3300-00 Valore di preset (preset = 1000 = 3E8h)

Master → Slave (richiesta Set)

COB-ID	Cmd	Index		Sub	Process data			
600+ID	23	00	33	00	E8	03	00	00

Slave → Master (conferma Set)

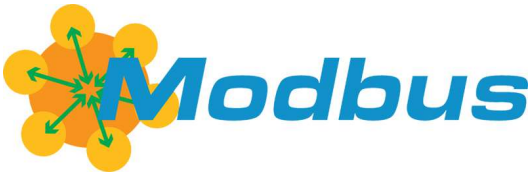
COB-ID	Cmd	Index		Sub	Process data			
580+ID	60	00	33	00	00	00	00	00

NOTA
Per salvare i parametri modificati eseguire **1010-01 Salvataggio parametri**.
Nel caso di spegnimento del dispositivo o di invio dei comandi **Reset node** o **Reset communication** i dati non salvati andranno persi!

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 > ATTUATORI ROTATIVI (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

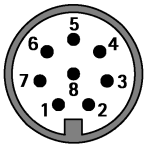
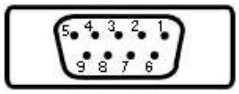
Si badi di rispettare sempre le seguenti compatibilità tra la versione del file EDS e la versione software dell'eseguibile Modbus.

Compatibilità	File EDS	EXE Modbus
	V1	fino a 2.1
	V2	fino a 2.4
	V3	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**.

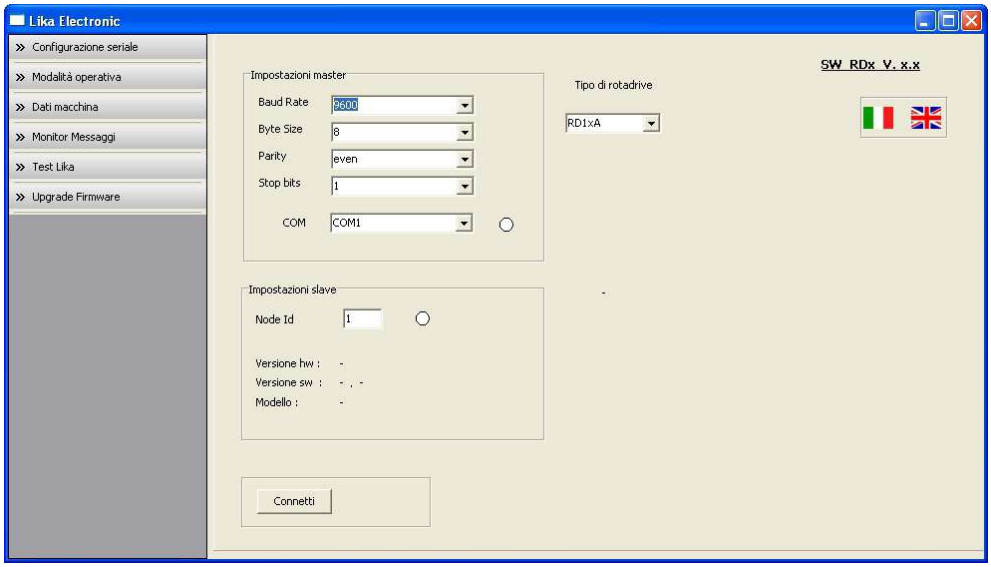
Si badi che i parametri di configurazione della porta seriale di servizio Modbus sono fissi e perciò non modificabili.
Essi sono:

Impostazioni porta seriale di servizio Modbus RS-232	**Valore di default**
Baud rate	9600
Byte size	8
Parity	Even
Stop bits	1

La modifica del valore di baud rate mediante il selettore installato nella parte interna del dispositivo -si veda la sezione "4.4.2 Velocità di trasmissione dei dati: Baud rate (Figura 5)" a pagina 29- non ha alcun effetto sui parametri di configurazione della porta seriale di servizio Modbus.

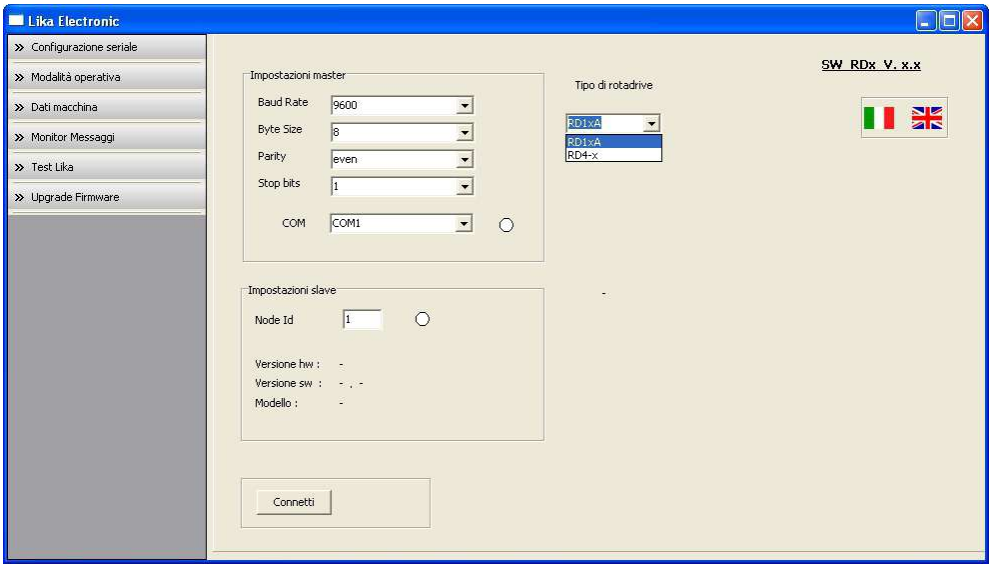
1.2 Pagina "Configurazione seriale"

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

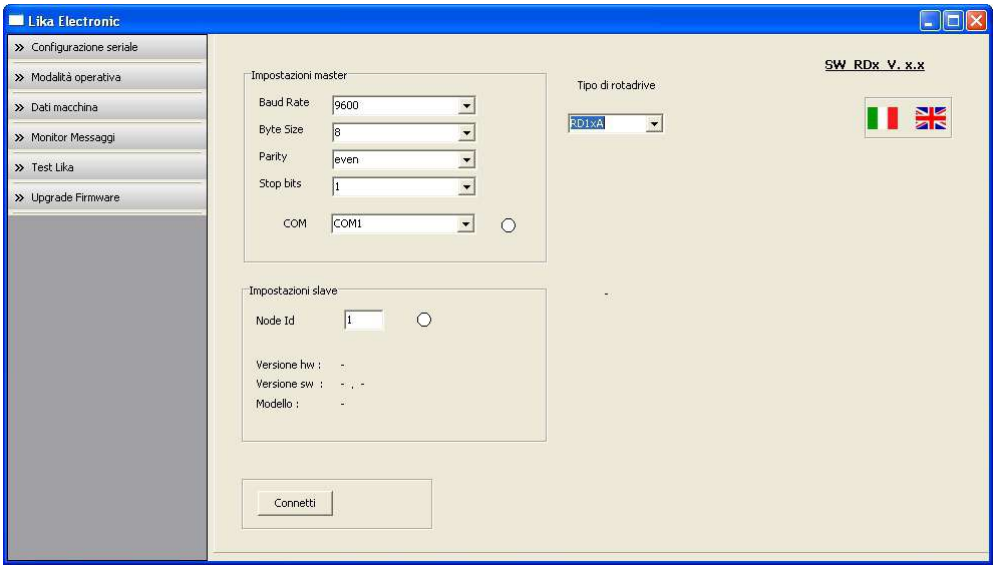


Questa pagina permette anzitutto la scelta della lingua di visualizzazione dei testi. Cliccare sul pulsante **Bandiera italiana** per scegliere la lingua di visualizzazione italiana; cliccare sul pulsante **Bandiera inglese** per scegliere la lingua di visualizzazione inglese.

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.

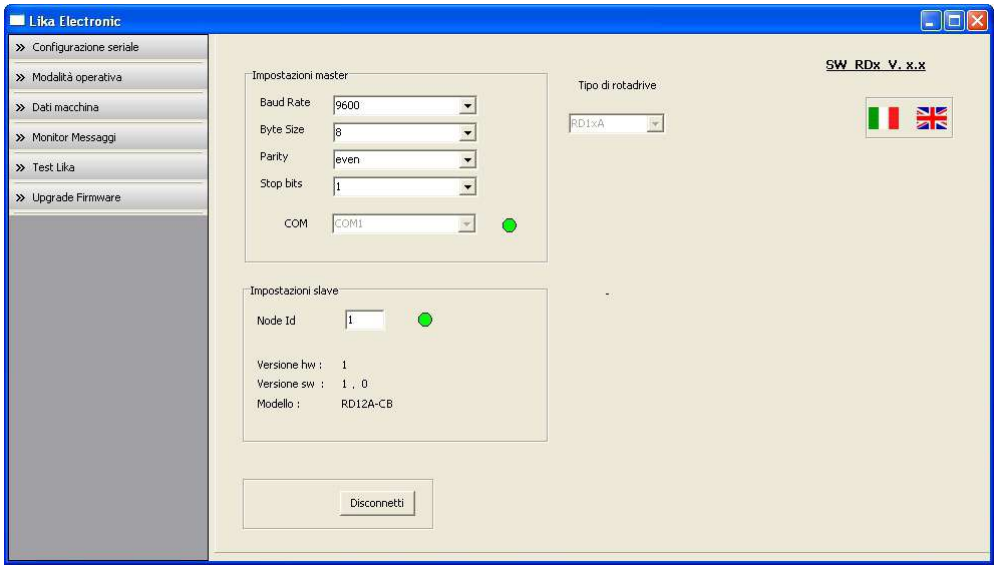


Essi sono:

Impostazioni porta seriale di servizio Modbus RS-232	Valore di default
Baud rate	9600
Byte size	8
Parity	Even
Stop bits	1

Si consideri inoltre che la modifica del valore di baud rate mediante il selettore installato nella parte interna del dispositivo (si veda la sezione "4.4.2 Velocità di trasmissione dei dati: Baud rate (Figura 5)" a pagina 29) non ha alcun effetto sui parametri di configurazione della porta seriale di servizio Modbus.

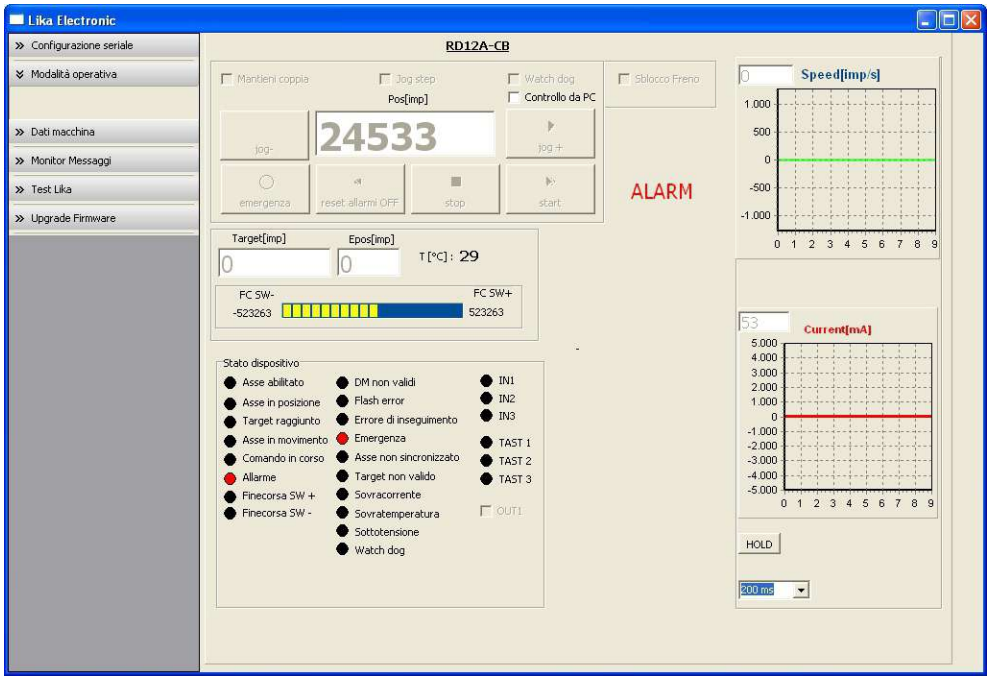
Impostare quindi l'indirizzo del nodo cui ci si vuole connettere nel box **Impostazioni slave** (valore di default dei dispositivi RD1xA = 1). Per impostare l'indirizzo del nodo riferirsi al paragrafo "4.4.1 Indirizzo nodo: Node ID (Figura 5)" a pagina 28.
E' ora possibile attivare la connessione premendo il pulsante **CONNETTI** in basso nella pagina.



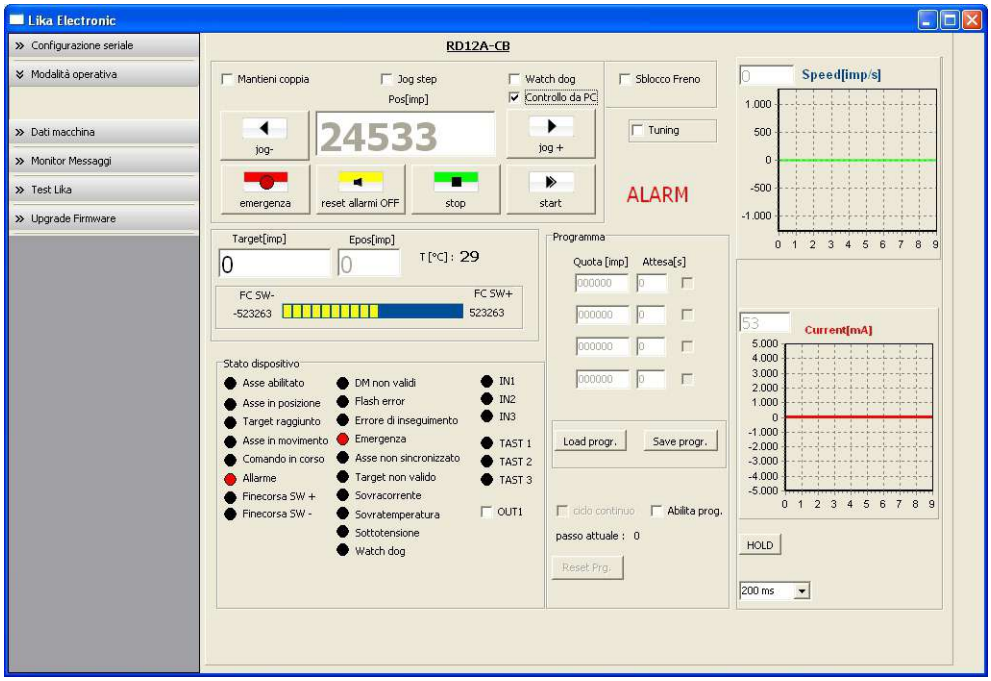
Se la connessione ha esito positivo, le due segnalazioni in corrispondenza dei campi di impostazione della **porta seriale** e dell'**indirizzo del nodo** si illuminano colorandosi di verde, mentre il pulsante **CONNETTI** sparisce sostituito dal pulsante **DISCONNETTI**. Compaiono inoltre la versione 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 CANopen. 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 123).



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-CB** in alto a sinistra sono disponibili le seguenti funzioni.

Mantieni coppia

Si veda Asse in coppia a pagina 126. Disponibile solo per la versione RD1A, nella versione RD12A il check box non è visualizzato.

Jog step

Si veda Jog incrementale a pagina 125.

Watch dog
Si veda Abilitazione Watch dog a pagina 126.

Jog -
Si veda Jog - a pagina 124.

Pos [imp]
Si veda Posizione corrente [0x02-0x03] a pagina 133.

Jog +
Si veda Jog + a pagina 124.

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 125.

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 125.

Stop
Comanda l'arresto normale del dispositivo. Si veda Stop a pagina 124.

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 125.

Nel box sottostante sono disponibili le seguenti funzioni.

Target [imp]
Si veda Target position [0x2B-0x2C] a pagina 127. 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 134.

T [°C]
Si veda Temperatura [0x08] a pagina 134.

FC SW - / FC SW +
Visualizzazione grafica dei valori di finecorsa positivo e negativo. Si vedano Delta spazio positivo [0x08-0x09] a pagina 118 e Delta spazio negativo [0x0A-0x0B] a pagina 119.

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 131; per una descrizione completa degli allarmi si veda Registro allarmi [0x00] a pagina 129. Il check box **OUT1** attiva / disattiva l'uscita digitale 1 del dispositivo. Per ogni informazione si veda a pagina 127.

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 repositorio. 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**.

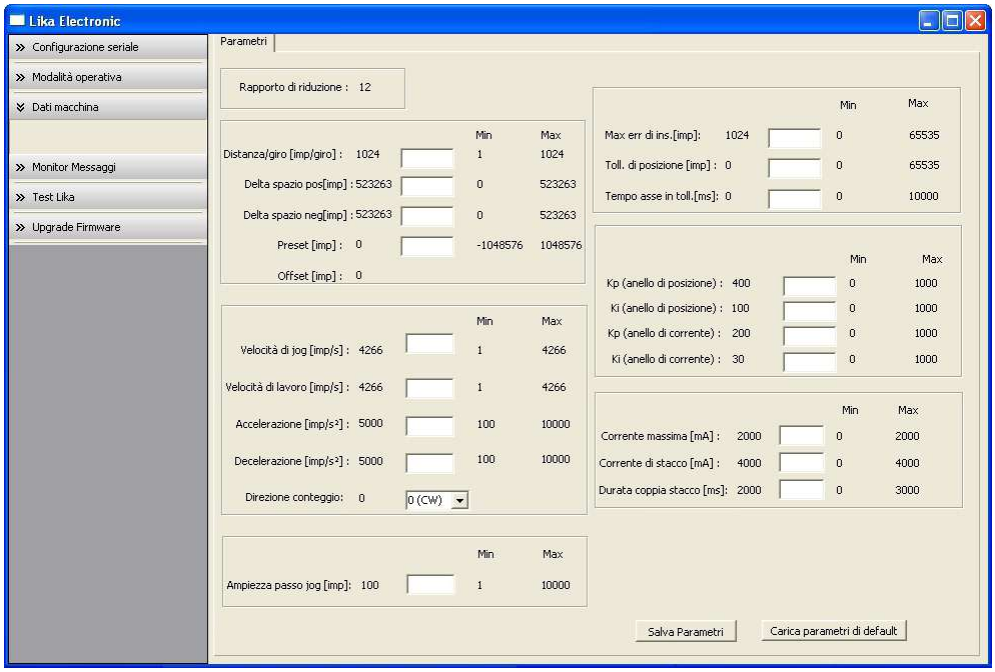
Sul lato destro della pagina sono visualizzati graficamente la velocità del dispositivo (espressa in impulsi/secondo) e il valore della corrente assorbita dal motore (espresso in milliampere). Premere il pulsante **HOLD** per disattivare la visualizzazione; premere lo stesso pulsante (che presenta poi 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-CB** 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 115).
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**.



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

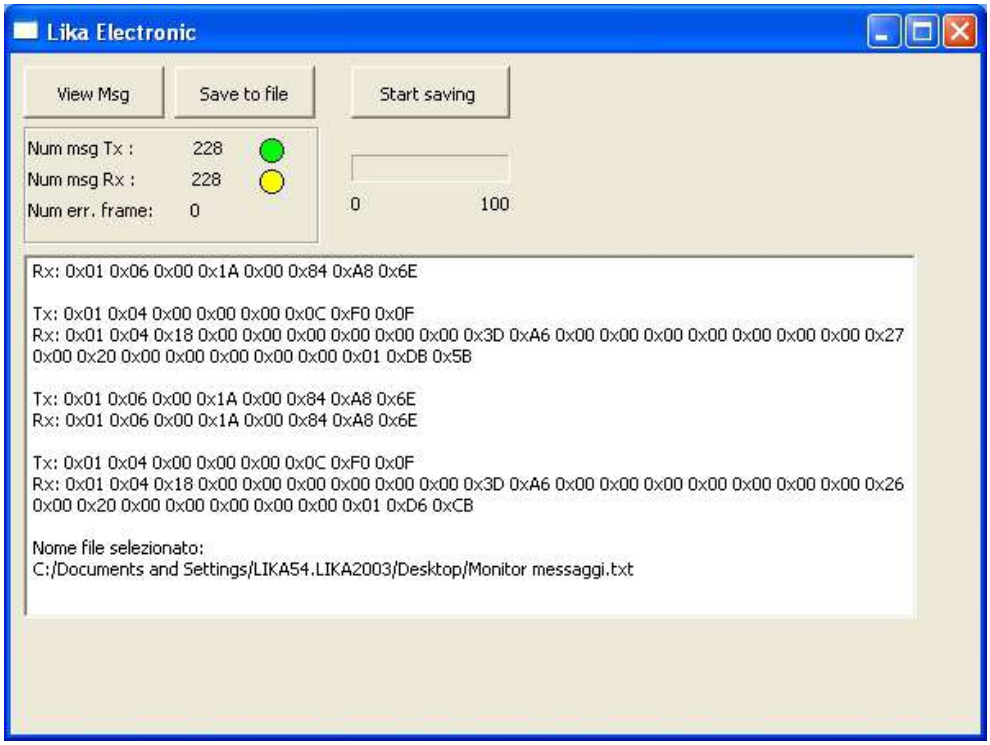
funzione e la programmazione di ciascun parametro riferirsi alla sezione "3.1.1 Parametri Dati macchina" a pagina 115.
Per impostare un nuovo valore in un parametro digitare il valore desiderato, quindi premere il tasto **INVIO** della tastiera. Se si imposta un valore non conforme, alla conferma il 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 126.

Per caricare i parametri di default (parametri impostati durante la messa a punto in azienda del dispositivo 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 126. A pagina 40 è disponibile l'elenco dei dati macchina e il rispettivo valore di default preimpostato da Lika Electronic.

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 repositorio. Alla conferma mediante il pulsante **APRI**, il box si chiude e nel campo di visualizzazione della pagina **Monitor**

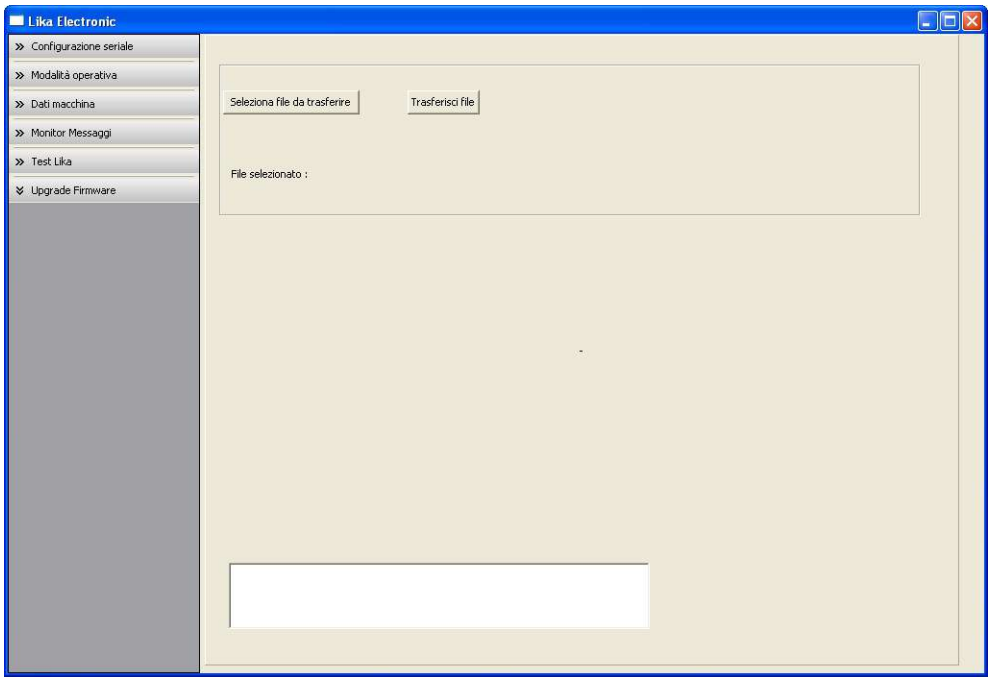
Messaggi compare il path completo del file indicato. Premere quindi il pulsante **START SAVING** per iniziare il salvataggio dei messaggi, nel campo di visualizzazione della pagina **Monitor Messaggi** compare il messaggio "File aperto correttamente". Dopo la pressione del pulsante **START SAVING**, 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 86).
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 del programma oppure scaricarla all'indirizzo **www.lika.it > ATTUATORI ROTATIVI > ATTUATORI ROTATIVI (DRIVECOD) > RD1A / RD12A**.

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

1. assicurasi 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 repositorio 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

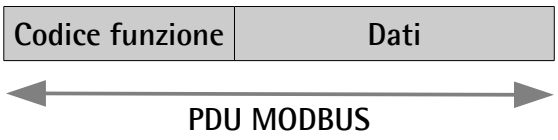
Il protocollo seriale Modbus è un protocollo Master – Slave. Un solo Master può essere connesso contemporaneamente alla rete Modbus; mentre il numero di Slave connessi allo stesso bus seriale può essere compreso tra 1 e 247. Una comunicazione Modbus è sempre iniziata dal Master. I nodi Slave non sono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

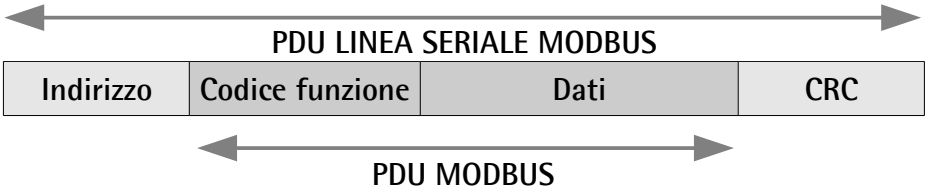
- **modalità UNICAST**: il Master invia la richiesta a un singolo Slave. Dopo aver ricevuto e processato la richiesta, lo Slave invia una risposta al Master. In questa modalità, la transazione Modbus consiste di due messaggi: una richiesta da parte del Master e una risposta da parte dello Slave. Ogni 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 105.
- **DATI**: byte dedicati alle informazioni addizionali e alla trasmissione dei dati, il numero di byte e la struttura dipendono da ciascun codice funzione. Il campo **DATI** include valori come per esempio indirizzi di registro, numero di registri da processare, numero di data byte presenti nel campo, ecc. (si veda alla sezione "2.4 Codici funzione" a pagina 105).
- **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 trasmittente che lo allega al messaggio. Il

dispositivo che riceve il messaggio ricalcola il valore del CRC alla ricezione e lo confronta con quello ricevuto. Se i due valori non sono uguali, il dispositivo attiva un allarme.

Il protocollo Modbus definisce tre PDU. Esse sono:

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

La **Modbus Request PDU** consiste di {function_code, request_data}, dove:
function_code = codice funzione Modbus, 1 byte;
request_data = questo campo dipende dal codice funzione utilizzato e solitamente contiene informazioni quali coordinate di variabili, valori di variabili, offset dati, codici di sottofunzioni, ecc., n byte.

La **Modbus Response PDU** consiste di {function_code, response_data}, dove:
function_code = codice funzione Modbus, 1 byte;
response_data = questo campo dipende dal codice funzione utilizzato e solitamente contiene informazioni quali coordinate di variabili, valori di variabili, offset dati, codici di sottofunzioni, ecc., n byte.

La **Modbus Exception Response PDU** consiste di {exception-function_code, exception_code}, dove:
exception-function_code = codice funzione Modbus + 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

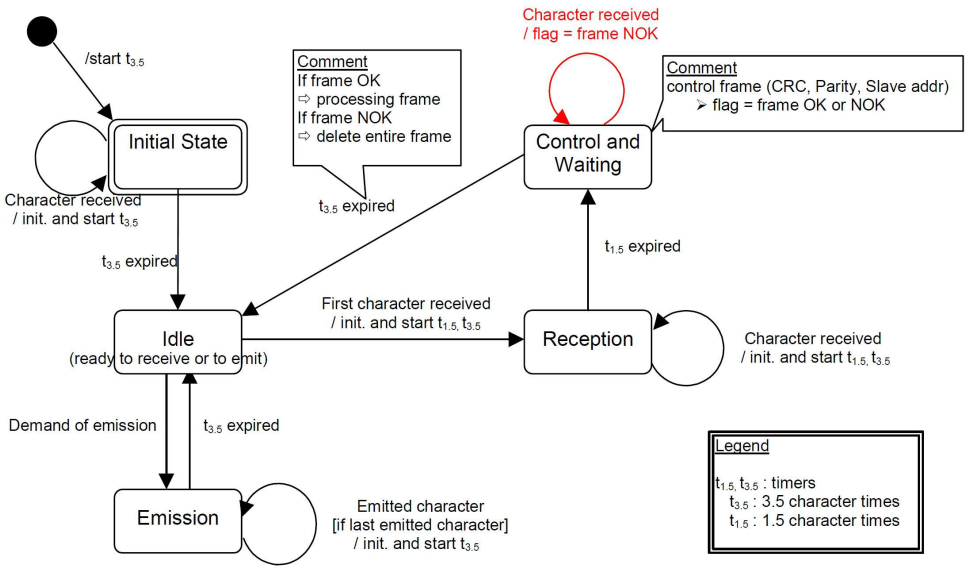
* Con "No parity", il bit di parità è sostituito da un bit di stop.

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 115.

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	
Nome campo	(Hex)
Codice funzione	**03**
Indirizzo iniziale Hi	**00**
Indirizzo iniziale Lo	**06**
Numero di registri Hi	**00**
Numero di registri Lo	**02**

Response	
Nome campo	(Hex)
Codice funzione	**03**
Numero byte	**04**
Valore registro 7 Hi	**03**
Valore registro 7 Lo	**E8**
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 129.

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

Response	
Nome campo	(Hex)
Codice funzione	**04**
Numero byte	**04**
Valore registro 3 Hi	**00**
Valore registro 3 Lo	**00**
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 115.

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

Response	
Nome campo	(Hex)
Codice funzione	**06**
Indirizzo registro Hi	**00**
Indirizzo registro Lo	**06**
Valore del registro Hi	**05**
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 115.

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

Response	
Nome campo	(Hex)
Codice funzione	**10**
Indirizzo iniziale Hi	**00**
Indirizzo iniziale Lo	**06**
Numero di registri Hi	**00**
Numero di registri Lo	**02**

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 = 1000 dec
[03][E8] = valore da impostare nel registro 8 **Decelerazione [0x07]**, 03 E8 hex = 1500 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 139.

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 105.

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 120.

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/sec^2].
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/sec^2].
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]**.
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 positivo [0x08-0x09]** sarà:
(1024 imp./giro * 511 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 imp./giro * 511 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 imp./giro * 511 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 imp./giro * 511 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:

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

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.

Valore di preset [0x16-0x17]

[Registri 23-24, Integer32, rw]
Usare questo parametro per assegnare e impostare 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'invio del parametro.
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 CANopen; 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 81).

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 (si veda "4.4.4 Pulsanti JOG + e JOG - (Figura 5)" a pagina 30) 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 127). 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 standard 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.

Esegui preset conteggio

bit 11

Il conteggio assume il valore scritto 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).

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 115.

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 105.

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 129.

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 115.

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 115.

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 115.

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]
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]
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).

Dip-switch baud rate [0x0C]

[Registro 13, Unsigned16, ro]
Visualizza il valore della velocità di trasmissione dei dati (baud rate) della porta seriale del dispositivo impostato mediante il dip-switch previsto (rete CANopen). Per ogni informazione sull'impostazione del baud rate riferirsi alla sezione "4.4.2 Velocità di trasmissione dei dati: Baud rate (Figura 5)" a pagina 29.

Dip-switch indirizzo [0x0D]

[Registro 14, Unsigned16, ro]
Visualizza l'indirizzo del nodo del dispositivo impostato mediante il dip-switch previsto. Per ogni informazione sull'impostazione dell'indirizzo del nodo riferirsi alla sezione "4.4.1 Indirizzo nodo: Node ID (Figura 5)" a pagina 28.

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			

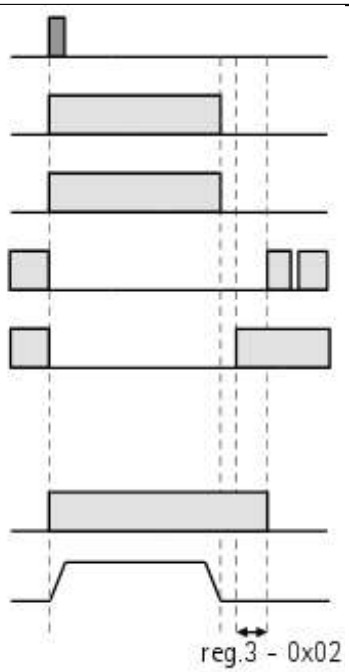
dove:

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)

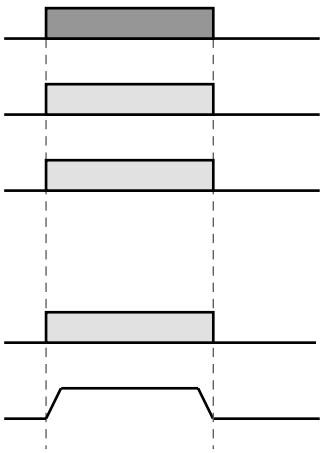
Il valore 11 81 hex, che corrisponde alla rappresentazione binaria 00010001 10000001, deve perciò essere interpretato come: versione hardware 1 (bit 0 = 1); dispositivo con freno (bit 7 = 1); interfaccia Profibus (bit 8 = 1); modello RD1xA (bit 12 = 1).

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

Esempio 1

Posizionamento	RD1xA
Start (bit 6 - Control Word [0x2A]) Comando in corso (bit 7 - Status word [0x01]) Asse in movimento (bit 6 - Status word [0x01]) Asse in posizione (bit 0 - Status word [0x01]) Target raggiunto (bit 8 - Status word [0x01]) Asse abilitato (bit 2 - Status word [0x01]: PWM attivo) Velocità teorica	

Esempio 2

Jog +	RD1xA
Jog + (bit 0 - Control Word [0x2A]) Comando in corso (bit 7 - Status word [0x01]) Asse in movimento (bit 6 - Status word [0x01]) Asse abilitato (bit 2 - Status word [0x01]: PWM attivo) Velocità teorica	

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 129): **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.

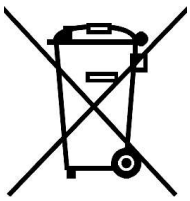
Pagina lasciata intenzionalmente bianca

Pagina lasciata intenzionalmente bianca

Pagina lasciata intenzionalmente bianca

Versione HW-SW	Versione documento	Descrizione
1-1	1.0	Prima stampa
1-2 1-3 1-4	1.1	Aggiunto "Jog step" in Control Word e parametro "Ampiezza passo jog". Aggiornate definizioni jog. Aggiornata descrizione pulsanti jog. Aggiornata definizione "Asse abilitato". Aggiornamento file EDS (V2) e file Modbus.exe (V2.2/V2.3/V2.4).
2-5	1.2	Aggiornamento informazioni freno, aggiornamento file EDS (V3) e Modbus.exe (V2.5)
3-0	1.3	Aggiornamento sezione "Connessioni elettriche". Aggiornamento "KP anello di corrente". Aggiornamento file EDS.
3-1 3-2	1.4	Introduzione modello con rapporto di riduzione T92, file EDS per modello T92, aggiornamento file Modbus.exe (V2.7)
3-2	1.5	Aggiornamento informazioni oggetti 1400 Parametri PDO1 ricevuti, 1800 Parametri PDO1 inviati, 3110-00 Cyclic Time
3-2	1.6	Aggiornamento informazione LED
3-2	1.7	Avvertenza per tensione controelettromotrice, aggiornamento registri Tolleranza di posizione [0x01], Max errore di inseguimento [0x03], Ampiezza passo jog [0x19] Modbus

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